



**Diseño, desarrollo e implementación de un producto software basado en el enfoque STEAM para el aprendizaje a través de video juegos en niños y jóvenes aprendientes**

Morales Arroyo, Dudley Santiago

Departamento De Ciencias De La Computación

Carrera de Ingeniería de Sistemas e Informática

Trabajo de titulación, previo a la obtención del título de Ingeniero en Sistemas e Informática

Ing. Zambrano Rivera, Margarita Elizabeth

9 de septiembre del 2020

URKUND

**Document Information**

Analyzed document Proyecto\_Titulacion\_Santiago\_Morales.docx (D79950536)  
Submitted 9/26/2020 1:14:00 AM  
Submitted by  
Submitter email mezambrano@espe.edu.ec  
Similarity 2%  
Analysis address mezambrano.espe@analysis.arkund.com

**Sources included in the report**

**SA** **Universidad de las Fuerzas Armadas ESPE / tesis 21 marzo 2017.docx**  
Document tesis 21 marzo 2017.docx (D26672243)  
Submitted by: mezambrano@espe.edu.ec  
Receiver: mezambrano.espe@analysis.arkund.com

6



Zambrano Rivera, Margarita Elizabeth

**DIRECTORA**



**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**  
**CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**CERTIFICACIÓN**

Certifico que el trabajo de titulación, “**Diseño, desarrollo e implementación de un producto software basado en el enfoque STEAM para el aprendizaje a través de video juegos en niños y jóvenes aprendientes**” fue realizado por el señor **Morales Arroyo, Dudley Santiago** el cual ha sido revisado y analizado en su totalidad por la herramienta de verificación de similitud de contenido; por lo tanto cumple con los requisitos legales, teóricos, científicos, técnicos y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, razón por la cual me permito acreditar y autorizar para que lo sustente públicamente.

Sangolquí, 24 de agosto de 2020.

Firma:

**Ing. Zambrano Rivera, Margarita Elizabeth**

C. C. 1714024666



**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**  
**CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**RESPONSABILIDAD DE AUTORÍA**

Yo, **Morales Arroyo, Dudley Santiago**, con cédula de ciudadanía N° 1721144416, declaro que el contenido, ideas y criterios del trabajo de titulación: “**Diseño, desarrollo e implementación de un producto software basado en el enfoque STEAM para el aprendizaje a través de video juegos en niños y jóvenes aprendientes**” es de mi autoría y responsabilidad, cumpliendo con los requisitos legales, teóricos, científicos, técnicos, y metodológicos establecidos por la Universidad de las Fuerzas Armadas ESPE, respetando los derechos intelectuales de terceros y referenciando las citas bibliográficas.

**Sangolquí, 24 de agosto de 2020**

Firma

**Morales Arroyo, Dudley Santiago**

C.C. 1721144416



**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN  
CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA**

**AUTORIZACIÓN DE PUBLICACIÓN**

Yo **Morales Arroyo, Dudley Santiago**, con cédula de ciudadanía n° 1721144416, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar el trabajo de titulación: **“Diseño, desarrollo e implementación de un producto software basado en el enfoque STEAM para el aprendizaje a través de video juegos en niños y jóvenes aprendientes”** en el Repositorio Institucional, cuyo contenido, ideas y criterios son de mi responsabilidad.

**Sangolquí, 24 de agosto de 2020**

Firma

**Morales Arroyo, Dudley Santiago**

C.C. 1721144416

## Dedicatoria

El siguiente trabajo de titulación se lo dedico principalmente a mis padres Dudley y Judy, por ser mi principal apoyo y ejemplo a seguir, cada momento en mi vida se ha visto superado gracias a su apoyo incondicional y sus consejos que siempre me guían por el buen camino, lleno de humildad con la bendición de Dios.

A mis abuelitos Segundo Arroyo, María Hidalgo, Carlos Morales y Nidia Mora, por todo su cariño y comprensión, por siempre ser un ejemplo de firmeza y constancia ante las adversidades y formar así excelentes seres humanos.

A mis hermanos Daniel Morales, Daniel Jaramillo, Nicolás Díaz y Dylan Ortiz por su constante apoyo, quienes se han encargado de mostrarme que la paciencia es una virtud de sabios y que a pesar de que fallemos muchas veces, está únicamente en nosotros volver a levantarnos y continuar nuestro camino.

A Nathaly Gualotuña por ser una gran persona que siempre me apoyo para continuar y culminar mis estudios y mostrarme muchas de las alegrías que da la vida universitaria y su cariño incondicional.

A mis queridos amigos y amigas, Antonella Martínez, Cristina Ulloa, Dayana Almeida, Estefanía Tipán, Edgar Bustamante y Juan Ávila por su amistad sincera durante todos estos años y quienes con su granito de arena me han ayudado a culminar este proyecto; gracias.

Dudley S. Morales

## **Agradecimiento**

Agradezco a Dios por sus infinitas bendiciones, por darme la fuerza y permitirme disfrutar de este momento de realización profesional, fruto de la dedicación y esfuerzo en cada actividad académica realizada dentro de la institución.

A mis padres Dudley Morales y Judy Arroyo por continuar apoyándome y seguir creyendo en mí a pesar de todas las dificultades que se presentaron en el camino.

A mis hermanos Daniel Morales y Daniel Jaramillo por su apoyo incondicional y las gratas experiencias que hemos sembrado a lo largo de este camino.

Agradezco a la Universidad de las Fuerzas Armadas ESPE, por fomentar en mí el deseo de ser un excelente profesional que con humildad dejen en alto su nombre.

A mi directora de proyecto de titulación, Ing. Margarita Zambrano, por compartir sus conocimientos y experiencias, las cuales forman parte de este proyecto de titulación.

Por último, deseo expresar mi más sincero agradecimiento a todas las personas que me apoyaron de una u otra manera en este largo camino hacia el éxito profesional, de todo corazón; gracias.

Dudley S. Morales

## Índice

Certificado del director .....	3
Autoría de responsabilidad .....	4
Autorización.....	5
Dedicatoria .....	6
Agradecimiento .....	7
Índice .....	8
Índice de tablas .....	11
Índice de figuras.....	12
Resumen .....	13
Abstract .....	14
Capítulo I .....	15
Introducción.....	15
Antecedentes .....	16
Planteamiento del problema .....	17
Justificación.....	18
Objetivos.....	19
Objetivo General .....	19
Objetivos Específicos.....	19
Alcance.....	20
Capítulo II .....	22
Marco referencial .....	22
El proceso de desarrollo de software .....	22
Modelo en espiral.....	23
Modelo Oohdm para el diseño y desarrollo de aplicaciones multimedia.....	25
Ventajas y desventajas .....	26
Metodología STEAM una nueva tendencia.....	27
Metodología tradicional vs metodología STEAM .....	31
Herramientas de desarrollo de aplicación multimedia en educación STEAM ..	32
Sprite Lab .....	33
Artista .....	33
Laboratorio de Juegos .....	34
Aplicación de videos en el proceso de enseñanza y aprendizaje .....	35
El uso de videos en STEAM .....	35
Aplicación de educación STEAM en videojuegos .....	36
Pensamiento computacional.....	37
Habilidades.....	39
Juguetes STEAM .....	41
Metodología CROA para el diseño de objetos de aprendizaje .....	42
Análisis .....	43
Diseño .....	44
Desarrollo .....	44
Publicación .....	45
Evaluación.....	45



Capítulo III .....	46
Análisis y diseño de la solución tecnológica de software .....	46
Especificación de requerimientos .....	46
Propósito .....	46
Ámbito del sistema .....	46
Definiciones y acrónimos .....	47
Funciones del producto.....	48
Características de los usuarios.....	48
Restricciones.....	49
Requisitos futuros .....	49
Requisitos funcionales .....	49
Requisitos no funcionales .....	50
Diseño de la aplicación multimedia aplicando la metodología OOHDM .....	51
Diseño conceptual.....	51
Diseño navegacional.....	52
Interfaz abstracta .....	53
Análisis y diseño de los objetos de aprendizaje aplicando la metodología CROA .....	56
Análisis .....	56
¿Qué es lo que los destinatarios necesitan aprender? .....	56
¿Por qué se cree que es necesario utilizar un OA y no otro tipo de material educativo? .....	56
¿A qué nivel educativo se orienta?.....	57
¿Cuál es el tema que el OA abordará? .....	57
¿Qué conocimientos previos debería tener el alumno para utilizar al OA? ....	57
¿Con qué otros conocimientos se relacionan el OA que pueden ser adquiridos en forma posterior a trabajar con lo que se está diseñando? .....	57
Diseño .....	57
Diseño instruccional del OA.....	58
¿Qué objetivo de aprendizaje específico se propone para el OA? .....	58
¿Qué información proveerá el OA acerca del tema a tratar, como se organizará y presentará esa información? .....	58
¿Qué actividades se le propondrán al alumno? .....	58
¿En qué consistirá la autoevaluación? .....	58
¿De qué manera se le dará feedback al estudiante sobre su desempeño en esta?.....	59
Estructura del OA.....	59
Diseño multimedia basado en escenarios.....	59
Capítulo IV.....	61
Desarrollo e implementación .....	61
Desarrollo de la aplicación multimedia aplicando la metodología OOHDM ...	61
Implementación del menú lateral.....	61
Implementación de footer y links .....	62
Implementación del frame central y la función buscar .....	63
Desarrollo de los objetos de aprendizaje aplicando la metodología CROA ....	67

Selección de recursos.....	68
Desarrollo de contenido basado en plantillas.....	68
Ingreso de metadatos del OA .....	71
Evaluación.....	71
Capítulo V.....	75
Conclusiones y recomendaciones .....	75
Conclusiones.....	75
Recomendaciones .....	76
Trabajos Futuros .....	76
Referencias bibliográficas.....	77
Anexos.....	79

### Índice de tablas

Tabla 1 Comparación entre educación tradicional y educación STEAM .....	31
Tabla 2 Cuadro comparativo de las metodologías de diseño de OA .....	42
Tabla 3 Requisitos funcionales.....	50
Tabla 4 Requisitos no funcionales .....	50
Tabla 5 Matriz de Evaluación CODA.....	72

## Índice de figuras

Figura 1 Modelo de espiral .....	23
Figura 2 Etapas del modelo OOHDM.....	26
Figura 3 Etapas de la educación STEAM .....	29
Figura 4 Pantalla de Sprite Lab .....	33
Figura 5 Pantalla de Artista .....	34
Figura 6 Pantalla de Laboratorio de Juegos .....	34
Figura 7 Diagrama de casos de uso .....	51
Figura 8 Diagrama conceptual .....	52
Figura 9 Diagrama navegacional.....	52
Figura 10 Diseño de la interfaz abstracta 1 .....	53
Figura 11 Diseño de la interfaz abstracta 2 .....	54
Figura 12 Diseño de la interfaz abstracta 3 .....	54
Figura 13 Diseño de la interfaz abstracta 4 .....	55
Figura 14 Diseño de la interfaz abstracta 5 .....	55
Figura 15 Diseño de la interfaz abstracta 6 .....	56
Figura 16 Estructura Lineal.....	59
Figura 17 Menú Lateral.....	61
Figura 18 Menú Lateral Oculto .....	62
Figura 19 Pantalla del footer .....	62
Figura 20 Botones del footer .....	63
Figura 21 Pantalla de inicio .....	64
Figura 22 Pantalla de glosario.....	64
Figura 23 Pantalla de ejercicios propuestos .....	65
Figura 24 Pantalla de Ayuda .....	65
Figura 25 Pantalla de créditos.....	66
Figura 26 Pantalla de créditos.....	66
Figura 27 Pantalla del Buscador .....	67
Figura 28 Objetos de aprendizaje en ExELearning .....	68
Figura 29 Plantilla de autoevaluación .....	69
Figura 30 Sopa de letras en Ardora .....	69
Figura 31 Crucigrama en Ardora.....	70
Figura 32 Juego del ahorcado en Ardora.....	70
Figura 33 Metadatos en estándar LOM.....	71
Figura 34 Resultados de la evaluación .....	74

## Resumen

El presente proyecto de titulación presenta el proceso de conceptualización, análisis y desarrollo del curso de pensamiento computacional en combinación con microsítios web.

Los principales objetivos del curso son instruir y aplicar el pensamiento computacional mediante la enseñanza del desarrollo de videojuegos en la niñez ecuatoriana de entre 11 a 12 años de edad.

Los microsítios web son diseñados y estructurados mediante el modelo OOHDm para su funcionalidad, mientras que el curso de pensamiento computacional y sus objetos de aprendizaje son diseñados y desarrollados mediante la metodología CROA, parte de estos objetos son las actividades de aprendizaje y las autoevaluaciones, las cuales se desarrollan mediante herramientas de autor como ExElearning y Ardora.

Entre otros elementos se tiene un buscador para palabras y letras que permite navegar a determinadas páginas relacionadas con la búsqueda realizada, también se tiene un video guía que tiene como funcionalidad, enseñar a navegar y conocer las funcionalidades de los microsítios y un glosario de los términos con las palabras más utilizadas en el curso de pensamiento computacional.

Finalmente, se evaluaron los objetos de aprendizaje con la matriz de evaluación de CODA por sus múltiples parámetros de evaluación y sus resultados utilizados para una futura mejora de la funcionalidad

Palabras clave:

- **PENSAMIENTO COMPUTACIONAL**
- **MICROSITIOS.**
- **ARDORA**
- **EXELEARNING**
- **SCRATCH**

## Abstract

This degree project presents the process of conceptualization, analysis and development of the course of computational thinking in combination with microsities.

The main objectives of the course are to instruct and apply computational thinking by teaching video game development in Ecuadorian children between 11 and 12 years old.

The microsities are designed and structured using the OOHDM model for their functionality, while the course of computational thinking and its learning objects are designed and developed using the CROA methodology. Part of these objects are the learning activities and self-assessments, which are developed using authoring tools such as ExELearning and Ardora.

Among other elements, there is a search engine for words and letters that allows you to navigate to certain pages related to the search performed, there is also a video guide that has the functionality to teach how to navigate and learn about the functionalities of the microsities and a glossary of terms with the most used words in the course of computer thinking.

Finally, the learning objects were evaluated with the CODA evaluation matrix for their multiple evaluation parameters and their results used for future improvement of functionality.

Keywords:

- **COMPUTER THINKING,**
- **MICROSITES.**
- **ARDORA**
- **EXELEARNING**
- **SCRATCH**

## **Capítulo I**

### **Introducción**

En el presente proyecto, se enmarca como punto relevante la educación mediante la metodología STEAM a través de un curso de pensamiento computacional, para lo cual se estructura de manera sistemática el desarrollo del proyecto de investigación, que engloba todas las partes importantes organizadas de la siguiente manera:

En el capítulo uno se refleja la problemática a resolver, se exponen los objetivos generales y específicos, además del alcance que tendrá el desarrollo de los micrositos y el curso de pensamiento computacional.

En el capítulo dos se muestra la investigación referencial y el estado del arte que permitirán enmarcar el camino adecuado del desarrollo de los micrositos y el curso de pensamiento computación para resolver el problema de la educación ecuatoriana.

El capítulo tres y el capítulo cuatro se enfocan respectivamente en el desarrollo de los micrositos y el curso de pensamiento computacional, culminando con la realización de pruebas y análisis de resultados para la aceptación o rechazo de los objetivos propuestos anteriormente.

Finalmente, el capítulo cinco se refiere a las conclusiones, recomendaciones y posibles trabajos futuros relacionados con el presente proyecto.

## **Antecedentes**

El método de educación tradicional se enfoca en instruir a los estudiantes en diferentes asignaturas para que tengan un conocimiento base de diversas áreas como las matemáticas, las ciencias sociales, las ciencias naturales entre otros. Este tipo de educación es la más utilizada en los países de Sudamérica para los niveles básico e intermedio (Greenberg, D.,2008); además, el uso de herramientas y materiales como pizarra, cuaderno y lápiz hacen de esta educación un proceso monótono, aburrido y en muchos casos, poco efectivo para que el alumno desarrolle sus conocimientos y habilidades; la situación se agrava en los sectores rurales de las grandes ciudades como Quito, donde la población campesina sufre muchas veces la falta de infraestructura, equipos, maestros y materiales, lo que hace de este sector el más vulnerable en cuanto a desarrollo del conocimiento de los alumnos. (Ministerio de Educación del Ecuador, 2008)

El ámbito del desarrollo de software a edades tempranas solo se ha probado y utilizado en países de primer mundo como Inglaterra, Estados Unidos y Canadá, pero en países de Sudamérica como Ecuador, Perú y Colombia esto apenas se ha explorado, esto debido a que principalmente se utiliza el método tradicional y las herramientas no son las más adecuadas para la enseñanza en cuestión. Las soluciones de STEAM (SCIENCE, TECHONOLGY, ENGINEERING, ARTS AND MATHEMATICS) están enfocadas a despertar el pensamiento científico sobre todo en los niños a través de la inclusión de actividades de Ciencia, Tecnología, Ingeniería, Matemáticas y Arte en cada una de las asignaturas centrales de educación básica. (Yakman, G., 2017).



El pensamiento computacional es el proceso de pensamiento involucrado en la formulación de un problema y la expresión de su(s) solución(es) de tal manera que una computadora -humano o máquina - puede llevar a cabo efectivamente. (Wing, J., 2014).

El departamento de pediatría de la Clínica Universidad de Navarra señala los efectos positivos que brindan los videojuegos en niños y adolescentes, tales como: refuerzo social, esfuerzo, potenciamiento del razonamiento lógico, memoria y habilidades, siendo esta última muy notoria al observar una mejoría en la coordinación mano-ojo.

Este tipo de actividades involucra la espontaneidad y la originalidad; sin embargo, no es solo el motor de la creatividad sino también del encuentro consigo mismo. La creatividad en el individuo se conecta con su persona y despliega sus aptitudes (Vergara, E., 2013)

### **Planteamiento del problema**

En el Ecuador existen pocos proyectos sobre el enfoque STEAM en la educación, siendo uno de estos pocos la denominada “Coalición STEM Ecuador”, (SENECYT, 2018); además, no existen herramientas o software que puedan ser utilizados para la enseñanza y exploración de nuevas áreas de conocimiento tales como el desarrollo de video juegos, para los niños de entre 11 y 12 años, evitando así que expandan sus mentes hacia nuevos paradigmas como el pensamiento computacional y las metodologías constructivistas.

La ausencia de estas herramientas o software hacen que la educación primaria solo sea impartida a los alumnos mediante la metodología tradicional, enfocada en conocimientos y herramientas básicas, de esta manera no se les permite a los niños explotar al máximo sus cualidades individuales tales como la creatividad, autocrítica, confianza, la paciencia, entre otros, tampoco se les permite explorar nuevos ámbitos educativos, haciendo así que la niñez ecuatoriana quede rezagada en comparación a la niñez de otros países como Estados Unidos, España y el Reino Unido, en los cuales ya se muestran casos de aplicación exitosos de este enfoque en la educación.

### **Justificación**

El presente proyecto plantea una nueva alternativa a los métodos tradicionales de enseñanza y educación que se imparten en los centros educativos ecuatorianos.

Aporta a la educación ecuatoriana con nuevas técnicas y materiales para hacer de la educación un proceso diferente, creativo y divertido, ayudando a despertar el pensamiento computacional y el interés por el desarrollo de video juegos en los niños de 11 y 12 años, dejando un precedente para que futuras generaciones de educadores puedan impartir otras asignaturas de diferentes formas.

En España se desarrolla el proyecto "Programamos" el cual consiste en modificar la forma en la que los estudiantes se relacionan con la tecnología, pasando de ser consumidores a creadores tecnológicos a través de la programación de videojuegos y el desarrollo de aplicaciones para dispositivos móviles. (El proyecto – Programamos, 2019)

La idea fundamental, es que deben ser capaces de concienciar a los estudiantes de que los ordenadores no son máquinas que vienen ya hechas y simplemente podemos usarlas, sino que se pueden tocar y alterar para modificar su comportamiento de manera que realicen las tareas que se quieren en la forma que se desea. (Fernández, A., 2012).

## **Objetivos**

### **Objetivo General**

Diseñar, desarrollar e implementar un curso de pensamiento computacional, para el aprendizaje mediante videojuegos en niños y jóvenes aprendientes, con un enfoque de educación STEAM.

### **Objetivos Específicos**

- Investigar y desarrollar el marco teórico sobre la educación STEAM y el pensamiento computacional.
- Investigar y desarrollar el marco teórico sobre la metodología CROA y OOHDM.
- Aplicar la metodología CROA para el diseño de los objetos de aprendizaje a ser utilizados en el producto software.
- Aplicar la metodología OOHDM para el diseño de los micrositos y los videojuegos que forman parte del producto software.
- Implementar test para la autoevaluación del aprendizaje de los niños de 11 a 12 años a través de los videojuegos.

## Alcance

El alcance del presente proyecto será diseñar, desarrollar e implementar un curso enfocado en la educación STEAM, el mismo que permitirá a los niños y jóvenes aprendientes explorar, conocer y aprender a través de videojuegos, así como a autoevaluarse su conocimiento adquirido.

El proyecto consta de los siguientes componentes:

- Módulo de aprendizaje básico con SCRATCH
  - Lección 1: Bloques de Lápiz y Movimiento
  - Lección 2: Bloques de Apariencia y Sonido
  - Lección 3: Bloques de Eventos
  - Lección 4: Bloques de Sensores y Controles
  - Cada una de las lecciones presenta su actividad de aprendizaje y autoevaluación
  
- Módulo de aprendizaje básico con RPG Maker.
  - Lección 1: Escenarios
  - Lección 2: Personajes
  - Lección 3: Ciudades
  - Lección 4: Misiones y Nuevos Jugadores
  - Cada una de las lecciones presenta su actividad de aprendizaje y autoevaluación

- Módulo de aprendizaje básico con App Inventor.
  - Lección 1: Bola Mágica
  - Lección 2: Ping Pong
  - Lección 3: Tres en Raya
  - Cada una de las lecciones presenta su actividad de aprendizaje y autoevaluación
- Adicionalmente el curso estará implementado en la plataforma SELI para su futura evaluación.

## **Capítulo II**

### **Marco referencial**

En el presente capítulo se describe y se menciona el adecuado uso de las herramientas, métodos y técnicas para poder alcanzar los resultados esperados en el producto, en el tiempo correcto.

El proceso de desarrollo de software

El proceso de desarrollo de software se le conoce también como Ciclo de vida del software, consiste en desarrollar procesos mediante etapas o actividades previamente planificadas o preestablecidas para desarrollar el producto software.

El proceso de desarrollo de software varía de acuerdo con los objetivos de cada empresa u organización, es decir que no hay un único proceso para el desarrollo de software, la ISO 12207 se encarga de regular los métodos de selección, implementación y monitoreo del ciclo de vida del software. (Fischer, G., 2001)

El proceso de desarrollo de software generalmente consta de cinco principales actividades, que son Requisitos, Diseño, Implementación, Verificación y Mantenimiento, estas actividades en algunos casos se acoplan o se modifican en otras sub actividades acorde a la finalidad del software que se va a desarrollar, estos modelos poseen tanto sus ventajas como desventajas las cuales son consideradas al momento de realizar el proyecto y su factibilidad, los principales métodos son en Cascada, Espiral, Ágil y Reutilización.

## Modelo en espiral

Creado por Barry Boehm en 1988 este modelo combina aspectos del método en cascada y el desarrollo ágil de aplicaciones, se caracteriza por su gestión de riesgos de forma periódica en la etapa de desarrollo, permitiendo así que sistemas sencillos evolucionen a sistemas más complejos, esto gracias a la programación orientada a objetos.

José Drake (2008) sostiene que, el modelo entrega prototipos con pocos objetos y funcionalidad limitada para tener un elemento base del cual continuar el desarrollo de los demás objetivos de funcionalidad hasta llegar a un software completamente terminado, el cual será sometido a comprobaciones y verificaciones para la inclusión de nuevos requisitos en futuras etapas, como se puede ver en la Figura 1.

**Figura 1**  
*Modelo de espiral*



*Nota:* El gráfico representa los procesos y las etapas del modelo espiral.

Tomado de *Metodología de desarrollo de software*, por AspGems, 2019.

## **Planificación**

Es la primera etapa y la más importante ya que en esta se definen aspectos como la descripción de los requisitos acordados, la funcionalidad del software, sus componentes, el tiempo estimado, roles de usuario y los costos, luego de una primera versión del software, se puede retornar a esta etapa para aumentar o disminuir requisitos y la funcionalidad cubierta.

## **Análisis de riesgo**

Es la segunda etapa, en la que se evalúa todo aquello que pueda afectar al proyecto de acuerdo al estado en que se encuentre y su grado de avance, para lo cual se diseñan prototipos que deberán ser validados durante el ciclo.

## **Implementación**

Es la tercera etapa, aquí se combina los requisitos acordados con la infraestructura adecuada, se elaboran los algoritmos y el código que controle el comportamiento del software, puede ser uno o varios lenguajes de programación los que se utilice, esto por los beneficios que conlleva cada tipo de lenguaje y en qué proceso o componente de la infraestructura del proyecto se va a utilizar.

## **Evaluación**

Es la cuarta etapa, se enfoca en comprobar que todos los componentes y procesos de la etapa anterior estén en completa funcionalidad acorde a los requisitos acordados, se realizan pruebas con el objetivo de conocer si los riesgos detectados anteriormente ya han sido solucionados, lo cual determinará el avance del proyecto y hacia a donde debe enfocarse las iteraciones futuras.



## **Modelo Oohdm para el diseño y desarrollo de aplicaciones multimedia**

El modelo Object Oriented Hyperedia Design Methodology conocido por sus siglas como OOHDM fue diseñado por Gustavo Rossi, Daniel Schwabe y Simone Barbosa en 1996 como una extensión del modelo HDM, posee cuatro etapas principales:

- Diseño Conceptual
- Diseño Navegacional
- Diseño Abstracto de Interfaz
- Implementación

En la primera etapa se construye un esquema conceptual representado por clases y las relaciones entre dichos objetos, el modelo OOHDM propone un esquema conceptual basado en clases, relaciones y subsistemas.

En la segunda etapa se define clases navegacionales tales como nodos, enlaces y estructuras de acceso, los enlaces se derivan de las relaciones y los nodos representan ventanas lógicas sobre las clases conceptuales.

En la tercera etapa se define las especificaciones de la interfaz abstracta, así se define la forma en que deben aparecer los contextos navegacionales, aquí se incluye el modo en que dichos objetos de la interfaz activaran la navegación y el resto de funcionalidades de la aplicación.

En la cuarta etapa es donde se hacen corresponder los objetos de interfaz con los objetos de implementación. (Lapiente, M., 2013), en la Figura 2 se puede observar más a detalle las etapas del modelo OOHDM.

**Figura 2**  
*Etapas del modelo OOHDM*

<b>Etapas</b>	<b>Productos</b>	<b>Formalismos</b>	<b>Mecanismos</b>	<b>Descripción</b>
<b>Obtención de Requerimientos</b>	Casos de Uso (actores, escenarios)	Plantillas del formato del documento, Diagramas de Interacción de Usuario (UIDs)	Técnicas de Observación, Entrevistas	<b>Se crea un documento que describe actividades y requerimientos de los usuarios</b>
<b>Diseño Conceptual</b>	Clases, subsistemas, relaciones, atributos	Modelos Orientados a Objetos	Clasificación, agregación, generalización y especialización	<b>Se modela la semántica del dominio de la aplicación</b>
<b>Diseño Navegacional</b>	Nodos, enlaces, estructuras de acceso, contextos navegacionales, transformaciones de navegación	Vistas Orientadas a Objetos, Cartas de navegación orientadas a objetos, Clases de Contexto	Clasificación, agregación, generalización y especialización	<b>Se tiene en cuenta el perfil del usuario y las tareas. Se enfatiza en los aspectos cognitivos. Se crea la estructura de navegación de la aplicación</b>
<b>Diseño de Interfaz Abstracta</b>	Objetos de la interfaz abstracta, respuestas a eventos externos, transformaciones de la interfaz	Vistas Abstractas de Datos (ADV), Diagramas de Configuración, Cartas de navegación de los ADVs	Mapeado entre la navegación y los objetos visibles	<b>Se modelan los objetos visibles. Se describe la interfaz para los objetos de navegación. Se define el aspecto de los objetos de la interfaz</b>
<b>Implementación</b>	<b>Aplicación en funcionamiento</b>	<b>Los soportados por el entorno</b>	<b>Los que provea el entorno</b>	<b>Se realiza la puesta en producción del sistema</b>

*Nota:* El gráfico muestra las diferentes etapas y procesos de la metodología OOHDM. Tomado de *Construyendo aplicaciones web con una metodología de diseño orientada a objetos* (p.21), por D. Silva, 2002.

#### Ventajas y desventajas

- Al generar una cantidad considerable de documentación a través de sus distintas etapas, permite llevar un control del desarrollo de estas y tener la posibilidad de realizar una rápida detección, corrección de errores y mantenimiento. (Lapuente, M., 2013)

- En la etapa de diseño navegacional se crean enlaces entre cualquier nodo de manera que exista una verdadera interoperabilidad entre estos.
- El uso de UIDs nos ayuda de forma clara, rápida y precisa a representar los casos de uso obtenidos.
- El diseño navegacional tiene una gran cantidad de diagramas que muchas veces entregan información similar a los UIDs.
- El diseño pierde continuidad del modelo navegacional al diseño de interfaz.

### **Metodología STEAM una nueva tendencia**

Las siglas S.T.E.A.M hacen referencia a las palabras en inglés Science, Technology, Engineering, Art and Mathematics, en español, Ciencia, Tecnología, Ingeniería, Arte y Matemáticas, es un nuevo modelo de aprendizaje que busca integrar estas disciplinas de manera exitosa en la educación.

Originalmente se utilizó el término STEM, sin incluir la disciplina artística, su origen corresponde al aporte de la National Science Foundation (NSF) que buscaba que los estudiantes sean más receptivos y se interesen más por las ciencias y la investigación científica, con esta nueva metodología se logra que los estudiantes trabajen en equipo para resolver problemas reales, haciendo uso del pensamiento crítico y fortaleciendo las relaciones interpersonales. (Mitra, S.,2007)

Esta necesidad de hacer un cambio en la educación obedecía a la escasez de estudiantes y trabajadores en campos relacionados a tecnología. Para las autoridades estadounidenses era preocupante porque durante muchos años fueron considerados líderes en innovación y desarrollo y deseaban asegurar esta posición. (Innobasque, 2016).

En el año 2006 se incorporó las artes al concepto, convirtiéndose así en STEAM. En este nuevo enfoque más completo, las prácticas artísticas facilitan el desarrollo de nuevos conceptos y conocimientos logrando nuevos hallazgos en investigación y desarrollo. (Cilleruelo, L., Zubiaga, A., 2014).

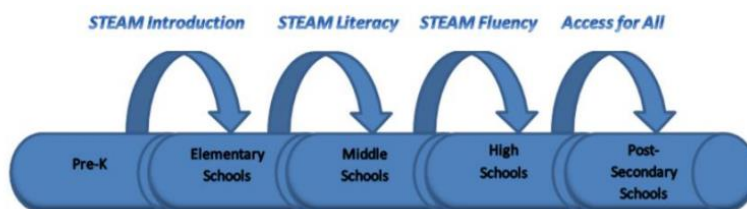
Para explicar en qué consiste la metodología STEAM y dar un ejemplo de su aplicación se ha tomado el White Paper presentado a la Junta Escolar de Escuelas Públicas del Condado de Chesterfield, de fecha 27 de agosto de 2013, donde mencionan que la educación basada en la metodología STEAM se ha desarrollado para impartirse a niños que van desde la etapa de pre kínder, es decir 4 años, hasta adolescentes de 16 años en promedio que están terminando el colegio, con el fin de potencializar sus habilidades integrando las cinco áreas en las que se trabaja.

Para lograr aquello se utiliza el aprendizaje basado en proyectos con el método científico como principal motor, la tecnología como un conjunto de herramientas, y los estudiantes como conductores. (Bushley, M., Gettings, M., Hancock, B., Fiege, W., White, J., 2013).

Mientras los estudiantes continúan avanzando a través de STEAM desde prekínder hasta el post educación secundaria, en los primeros años se realiza una Introducción al STEAM, luego se Alfabetiza en STEAM, posteriormente se alcanza una Fluidez STEAM como se puede apreciar en la Figura 3.

### Figura 3

*Etapas de la educación STEAM*



*Nota:* El gráfico muestra las diferentes etapas de la metodología STEAM para la educación. Tomado de *STEAM White Paper: Planning for the Plan* (p.3), por B. Melody, G. Michael, H. Barb, F. William, W. James, 2013.

Las características claves de una educación basada en STEAM se han clasificado en tres categorías, las cuales son: resultados de aprendizaje, alcance y pedagogía. (Sanders, M., 2012)

- Resultado de aprendizaje, acá los estudiantes podrán demostrar el conocimiento y las prácticas de STEAM, como las actitudes y disposiciones relacionadas. También usan los conceptos y prácticas de STEAM más convenientes de acuerdo con el grado del estudiante.
- La educación STEAM se encuentra disponible para todos, es implementada en uno o más salones y por uno o más profesores. Se puede implementar durante y después del horario normal de clases y se puede llevar a cabo en múltiples grados.

- Pedagogía, utiliza conceptos y metodologías actuales y aceptadas en instrucción y aprendizaje. También utiliza enfoques multidisciplinarios e involucra a los estudiantes de manera intencional en un pensamiento integrador.

Una de las principales diferencias que se puede identificar entre la metodología tradicional y STEAM es la caracterización del aprendizaje, siendo la primera de carácter vertical, donde el docente planifica, imparte y evalúa el aprendizaje, mientras que la metodología STEAM tiene un enfoque horizontal entre el docente y el aprendiz, que se traduce en la elaboración conjunta de los conocimientos, generando un ambiente interactivo en el cual la interrogante a resolver sea "qué pensar" en lugar de "cómo pensar", para de esta manera, ampliar las soluciones posibles por parte del alumno a un mismo problema.

(Cilleruelo, L.,2014)

Un estudio aplicativo de la metodología STEAM en Colombia determinó que "el uso de tecnología facilita la programación de los temas de las ciencias básicas en robótica, haciendo de esto una experiencia sencilla y agradable, reduciendo el temor a las matemáticas y generando espacios de diversión para los estudiantes, quienes descubren que las matemáticas se pueden aplicar para resolver problemas cotidianos, esto a través de la planificación de estrategias pedagógicas fundamentadas en la metodología STEAM.

## Metodología tradicional vs metodología STEAM

La fundamentación de la metodología tradicional fue escolástica, es decir, método y orden, el principal elemento es el profesor, siendo este el motor y la principal fuente de conocimiento, mismo que decide cómo organizar el conocimiento, que se debe aislar para enseñar al alumno acorde a su edad y que debe dejarse a un lado para una futura continuación del aprendizaje.

(Greenberg, D.,2008).

En la Tabla 1, se puede observar una comparación entre la educación tradicional y la educación con STEAM.

**Tabla 1**

*Comparación entre educación tradicional y educación STEAM*

Educación STEAM	Educación Tradicional
Los métodos no tradicionales de enseñanza se enfocan en la perseverancia.	Los exámenes suelen depender más de la memorización que del entendimiento.
El profesor actúa como un mentor y facilitador para organizar la información que brinda a los estudiantes en una forma fácil y entendible.	Los profesores son los que transmiten los conocimientos y los niños escriben y reciben la información.

Educación STEAM	Educación Tradicional
Los alumnos se sienten comprometidos a participar, ansiosos de aprender y listos para ser la siguiente generación de empresarios.	No siempre hay una conexión entre lo aprendido y la vida real. Es decir, que los niños no pueden aplicar lo aprendido en su vida diaria.
Los temas de enseñanza están interconectados	La educación tradicional enseña cursos que no tienen relación entre sí.
El proyecto es hecho paso a paso, de manera grupal, dentro de tiempos determinados.	No existe la retroalimentación ni oportunidad de aporte de parte de los estudiantes.

*Nota:* Recuperado de Mujer Peruana. Copyright 2019. Reprinted with permission

### **Herramientas de desarrollo de aplicación multimedia en educación STEAM**

En diferentes países se desarrollan proyectos basados en la metodología STEAM, estos proyectos están enfocados principalmente en fomentar e impulsar el desarrollo de software en la niñez, uno de los más importantes es el proyecto estadounidense “Code” fundado por Hadi y Ali Partovi (Partovi, H., 2013).

Surgió a raíz de una investigación en la cual los resultados mostraron que en el 90% de las escuelas no se impartía informática o programación, su principal objetivo es incluir la informática como materia obligatoria en la educación secundaria estadounidense y la inclusión de las mujeres en las ciencias computacionales.

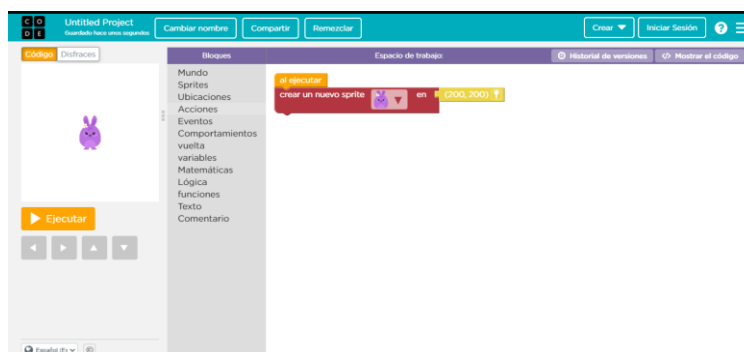


Este proyecto consta con herramientas importantes para desarrollar sus proyectos por ejemplo Sprite Lab, Artista, Laboratorio de Aplicaciones, Laboratorio de Juego.

## Sprite Lab

Sprite Lab es un entorno de programación basado en bloques, como se puede apreciar en la Figura 4, donde se puede hacer animaciones y juegos simples con objetos y personajes que interactúan entre sí, los estudiantes practican la codificación con algoritmos, bucles, condicionales, antes de pasar a las funciones. (Code.org, 2013).

**Figura 4**  
*Pantalla de Sprite Lab*

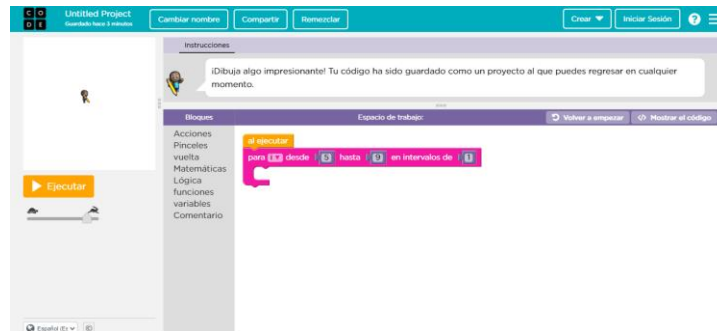


*Nota:* El gráfico muestra la pantalla de la herramienta Sprite Lab. Tomado de *Code.Org, 2013.*

## Artista

Es una herramienta que permite dibujar y pintar paisajes de igual manera que Sprite Lab, es decir, mediante bloques de pseudocódigo, como se puede apreciar en la Figura 5, la diferencia es que esta herramienta se enfoca únicamente en el dibujo y sus funciones.

**Figura 5**  
*Pantalla de Artista*

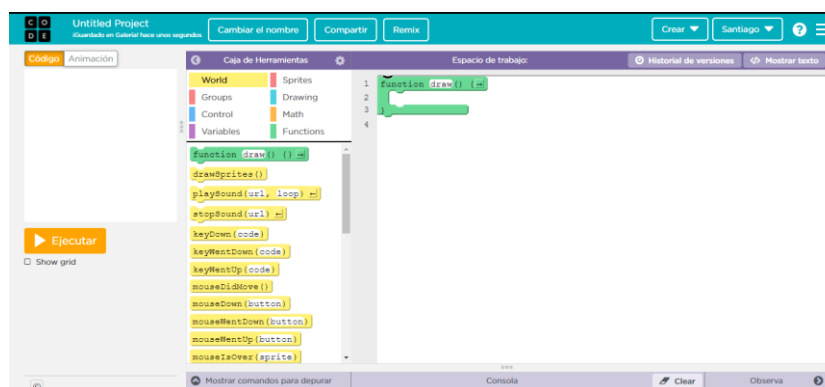


*Nota:* El gráfico muestra la pantalla de la herramienta Artista. Tomado de *Coge.Org, 2013.*

## Laboratorio de Juegos

Permite realizar de juegos, paisajes y escenarios, como se puede apreciar en la Figura 6, de igual manera permite emplear lenguajes de programación para desarrollar la funcionalidad del juego, de preferencia este lenguaje es JavaScript. (Code Org, 2013)

**Figura 6**  
*Pantalla de Laboratorio de Juegos*



*Nota:* El gráfico muestra la pantalla de Laboratorio de Juegos. Tomado de *Coge.Org, 2013.*

## **Aplicación de videos en el proceso de enseñanza y aprendizaje**

En la mayoría de las universidades más eficaces de los países desarrollados existen rutas de circuitos para televisiones donde se visualiza videos didácticos y en diferentes instituciones concurren varios medios que pueden suministrar estos materiales a quienes les interese, ya que el video está muy ligado a la educación y a distintos programas de salud, política y economía (Monteagudo, P., Sánchez, A., 2017). El uso del video en la enseñanza es estimado de manera positiva ya que es un refuerzo para el aprendizaje, teniendo muchas ventajas en el proceso de instrucción donde ayuda en los estudios a los alumnos y motiva en la asimilación de contenidos de las diferentes áreas impartidas (Sánchez, A., 2017).

## **El uso de videos en STEAM**

Dentro de la rama de tecnología en el ámbito STEAM, se ha contemplado el uso de videos tutoriales y videos interactivos para la enseñanza de temas tales como, la programación de software, el ensamble de robots, etc., siendo esta herramienta la más adecuada ya que se adapta al ritmo de aprendizaje del alumno y no el alumno debe adaptarse al ritmo de la herramienta. Los videos tutoriales y videos interactivos cuentan con un contenido elaborado más detalladamente por el docente, el procedimiento es más explicativo gracias a que se pueden apreciar diferentes puntos de vista del mismo, gracias al uso de cámaras en diferentes ángulos y se puede repetir ya sea todo el procedimiento o segmentos específicos que el alumno haya pasado por alto.

## **Aplicación de educación STEAM en videojuegos**

Durante más de 60 años, las investigaciones y actividades educativas de la N.S.F. en ciencia, tecnología, ingeniería y matemáticas (STEM) han ayudado a sostener el liderazgo científico internacional de los Estados Unidos y han cultivado una fuerza laboral STEM diversa y creativa (Bruer, J.,2010).

Es por ello que hoy en día universidades de diferentes países como Argentina y España emprenden proyectos los cuales tienen como principal objetivo fomentar la educación STEAM con los videojuegos como principal herramienta, tal es el caso de “Programamos” en España, cuyo objetivo es modificar la forma en que el alumno se relaciona con la tecnología, pasando de ser consumidores a creadores tecnológicos mediante la programación de videojuegos. Es así que aplicaciones como Scratch y App Inventor desarrolladas e impulsadas por el Instituto Tecnológico de Massachusetts, son las más adecuadas para guiar este desarrollo ya que permiten desarrollar aplicaciones de niveles intermedio a complejo, pero también permite realizar programas más vistosos, interactivos y simples, propios de un estudiante de nivel básico.

En el caso de “Programamos” podemos ver que el conocimiento se ha dividido en niveles, básico para niños de 3 a 5 años con Scratch Jr., Scratch para chicos de 6 a 10 años y para niveles superiores Lego Wedo, Makey Makey, App Inventor. La vinculación de STEAM en video juegos se puede entender de la siguiente manera, explicando cómo cada sigla es cubierta para la enseñanza de video juegos:

- Science, se enfoca en fomentar a los niños al conocimiento de las ciencias computacionales, principalmente la programación, para desarrollar el pensamiento computacional a temprana edad, y motivar el desarrollo de software que cubra necesidades futuras.
- Technology, es cubierta por el uso de equipos computacionales y software orientado a la enseñanza del desarrollo de software, mediante bloques de código o pseudocódigo.
- Engineering, es cubierta mediante el uso herramientas educativas como Beebots, Lego Wedo, etc.
- Art, es cubierta mediante la fase de la educación en la que el alumno puede diseñar a su gusto escenarios y personajes, mismos que se utilizaran en la programación del juego.
- Mathematics, es cubierta mediante cálculos de distancia, espacio y estadísticas que los alumnos deben realizar para la programación de videojuegos.

### **Pensamiento computacional**

Es una habilidad fundamental que implica resolver problemas, diseñar sistemas y comprender el comportamiento humano basándose en conceptos fundamentales de la informática, de acuerdo con (Wing, J., 2006) en su publicación "Pensamiento Computacional", sostiene que el pensamiento computacional es el uso de la abstracción y la descomposición al atacar una tarea muy compleja o al diseñar un gran sistema complejo.

Mediante el pensamiento computacional, un estudiante puede reformular un problema aparentemente difícil para convertirlo en uno que se sepa ya como resolver, utilizando elementos como la reducción, incrustación, transformación o simulación, se puede llegar a dicho replanteamiento desde otra perspectiva, es por ello por lo que se dice que el pensamiento computacional es pensar recursivamente y que es un pensamiento paralelo, ya que se está interpretando el código como datos y los datos como código.

Como principales pilares el pensamiento computacional tiene:

- Descomposición de un problema en fases más pequeñas, cada pequeño problema al solucionarse abre paso a una posible solución del siguiente problema pequeño, no necesariamente la misma solución, pero si una iniciativa.
- Reconocimiento de patrones repetitivos, encontrar estas semejanzas en los pequeños problemas, ayuda a resolver el problema principal de forma más eficiente.
- Abstracción de información irrelevante al problema propuesto, se trata principalmente de concentrarse en información importante, dejando de lado detalles innecesarios e irrelevantes.
- Algoritmos escritos presentados para la resolución del problema, consiste en plantear las reglas a seguir para resolver cada uno de los pequeños problemas mediante programación de computadoras.

## Habilidades

Las habilidades que se desarrollan con el pensamiento computacional van más allá de codificar programas, ya que este proceso permite identificar y entender un problema, considerar, reflexionar y definir las soluciones efectivas, aplicar una metodología y ejecutarla. (Pensamiento computacional, 2019). Para desarrollar el pensamiento computacional entre los jóvenes estudiantes, es necesario fomentar programas educativos como el proyecto de la Universidad de Deusto llamado “El Club Trastea”. (Guenaga, M., 2015)

El Club Trastea comenzó en enero de 2014 con el objetivo de aumentar el interés de los jóvenes estudiantes por la tecnología, promover las vocaciones tecnológicas y de ingeniería e investigar cómo los estudiantes aprenden sobre tecnología y aprenden mediante la tecnología.

Entre las herramientas desarrolladas por nosotros mismos tenemos “Make World”, una plataforma en línea que combina el aprendizaje de los conceptos de STEM con la programación, desafíos que deben resolverse mediante la programación en bloque.

“Kodetu”, es un conjunto de desafíos de pensamiento computacional de dificultad creciente con la recopilación de datos de interacción, con características orientadas a la investigación para análisis de gamificación, generalización, etc. (Guenaga, M., 2015). Para el proyecto del Club Trastea se consideran siete ideas principales para la ciencia de la computación, las cuales son, algoritmos, modularización, estructuras de control, representación, diseño de procesos, depuración, hardware y software.

- Algoritmos, su objetivo es hacer que los estudiantes construyan algo mediante instrucciones, pasos finitos y concretos que los lleven a la solución de un problema.
- Modularización, los estudiantes deben descomponer el problema principal en subproblemas más sencillos que permitan mediante la combinación de sus soluciones llegar a la solución del problema general, al realizar el código, la modularización se implementa a través de procedimientos.
- Estructuras de control, son todas las herramientas que se pueden utilizar para el desarrollo del código tal como Scratch, AppInventor, RPG Maker, Kodetu y Make World.
- Representación, software como Scratch y AppInventor presentan formas de abstraer y representar datos utilizados algoritmos para obtener objetivos específicos como caminar una distancia determinada, solicitar una información al usuario.
- Diseño de procesos, para fomentar esta etapa, los alumnos deben averiguar cuál es el objetivo del desafío al que enfrentan, cuál es la necesidad que se desea satisfacer, para luego imaginar, plantear y diseñar la solución, después se desarrolla y se prueba, si es necesario se puede mejorar aún más.
- Depuración, los niños utilizan principalmente la forma de prueba y error, esto debido a que ellos no temen cometer errores, y si es el caso de que su solución no funcione, revisan el comportamiento para contrastarlo con lo que debería ser la funcionalidad correcta, Scratch y AppInventor poseen pistas visuales sobre la ejecución del código para que puedan comprender en donde se altera el comportamiento del programa.



- Hardware y Software, son utilizados de manera conjunta para desarrollar el pensamiento computacional, tanto aplicaciones con conexión a internet como aplicaciones que no lo necesitan, a las que se accede a través de equipos computacionales además de incluir también robots, bloques de construcción y tarjetas.

### **Juguetes STEAM**

Otras aplicaciones del pensamiento computacional en la educación son juegos como:

- Robot Turtles que se encarga de enseñar fundamentos de programación en niños.
- Bee-Bot, es una herramienta perfecta para enseñar secuencias, estimación y resolución de problemas.
- Dash and Dot, es un robot que mediante aplicaciones permite a los niños programarlos para que realicen varios movimientos.

El fomentar el pensamiento computacional en niños trae beneficios tales como, desarrollar habilidades lingüísticas y numéricas, estimular su creatividad, trabajar en soluciones alternativas y fomentar el aprendizaje colaborativo.

Para los jóvenes, adolescentes y adultos del siglo 21 se ha vuelto crucial desarrollarlo, ya que trae ventajas como:

- Mejorar las competencias laborales que son beneficios para las carreras de la mayoría de los sectores.
- Ayuda a los trabajadores de las empresas a tener un papel más activo al proponer soluciones

- Utilizando técnicas de descomposición y la construcción de secuencias de pasos, se puede resolver problemas usando el pensamiento algorítmico.
- Permite entender cuáles aspectos del problema son susceptibles a ser resueltos mediante la programación y la computación.

### **Metodología CROA para el diseño de objetos de aprendizaje**

De acuerdo con (Maldonado, J., Carvallo, J., Siguencia, J., 2017) en su artículo, “Metodologías y Propuestas Metodológicas para el Diseño de Objetos de Aprendizaje: Un Estado del Arte en Iberoamérica”, muestra la identificación y comparación de 19 metodologías y diseños metodológicos para los Objetos de Aprendizaje en Iberoamérica, también podemos observar las principales diferencias, ventajas y desventajas de cada uno, las cuales se presentan en la Tabla 2.

**Tabla 2**

*Cuadro comparativo de las metodologías de diseño de OA*

Metodología	País	Interoperable	Diseño Educativo	Aborda Ensamblaje de OA	Componentes del OA
<b>MEDEOVA</b>	Colombia	No	Si	No	Objetivo Contenido jerárquico Actividades de Aprendizaje Evaluación
<b>MEDOA</b>	México	Si	No	No	Elementos Introdutorios
<b>CROA</b>	Argentina	Si	No	Si	Objetivos Simples

Metodología	País	Interoperable	Diseño Educativo	Aborda Ensamblaje de OA	Componentes del OA
<b>MESOVA</b>	Colombia	Si	No	No	Contenido Actividades

*Nota.* Recuperado de Metodologías y Propuestas Metodológicas para el Diseño de Objetos de Aprendizaje: Un Estado del Arte en Iberoamérica. Copyright 2017. Reprinted with permission.

El presente proyecto utiliza la metodología CROA de origen argentino, ya que sus características permiten considerarla como la más adecuada, gracias a sus objetivos simples, actividades propuestas, la autoevaluación, adecuada para docentes con nociones básicas de informática.

La Metodología CROA presenta 5 etapas para la creación de Objetos de Aprendizaje. (Sanz, C., Moralejo, L., Barranquero, F.,2014).

### **Análisis**

En esta etapa se debe plantear cuál es la necesidad educativa que se da origen al Objeto de Aprendizaje, para este planteamiento se deben responder preguntas como:

- ¿Qué es lo que los destinatarios necesitan aprender?
- ¿Por qué se cree que es necesario utilizar un OA y no otro tipo de material educativo?
- ¿A qué nivel educativo se orienta?
- ¿Qué conocimientos previos debería tener el alumno para utilizar al OA?

## **Diseño**

Esta etapa se trabaja sobre aspectos de diseño institucional, aspectos de la estructura que tendrá el OA, y se analiza el diseño multimedial, a partir de una serie de escenarios que se proponen por defecto, a manera de guía para los autores del OA.

## **Desarrollo**

En esta etapa se debe abordar la selección de recursos (imágenes, videos, documentos de texto, presentaciones) para incorporar en el OA.

Luego, se deben utilizar las plantillas elegidas en la etapa anterior, desarrollarlas en la/s herramienta/s de autor correspondientes, y finalmente integrarlas para conformar el OA, y empaquetarlo. (Sanz, C., Moralejo, L., Barranquero, F.,2014).

- Selección de recursos, en función del diseño instruccional planteado, se deberán buscar los recursos necesarios.
- Desarrollo de contenidos basado en plantillas, a partir del escenario seleccionado en la etapa anterior, se deberá trabajar con las herramientas de autor correspondiente y desarrollar cada uno de los contenidos planificados
- Ingreso de Metadatos del OA, se recomienda trabajar aquí con el estándar de metadatos de LOM. Se sugiere trabajar con ExELearning para editar los metadatos, también, se recomienda, luego de realizar el paquete SCORM en la siguiente subetapa.

- Integración y empaquetamiento, se recomienda en esta etapa trabajar con la herramienta de autor ExeLearning o la herramienta Reload.

### **Publicación**

En esta etapa se propone la publicación del OA en un repositorio de OA o en un Entorno Virtual de Enseñanza y Aprendizaje. La publicación del OA puede realizarse en diferentes entornos tecnológicos que respeten el estándar de empaquetamiento utilizado en la etapa anterior, esta etapa consiste en incorporar el paquete a través de las funcionalidades específicas de la plataforma en la que se publicará. (Sanz, C., Moralejo, L., Barranquero, F.,2014).

### **Evaluación**

En esta etapa se analizarán las posibilidades y barreras encontradas en el uso del OA en contextos específicos. Existen diferentes métodos de evaluación de OA, incluso previo a la publicación pueden realizarse evaluaciones de calidad del OA que complementarán a las que se realicen en esta etapa.

Desde la metodología CROA se consideran algunos puntos para el análisis de la calidad del OA, tales como, la capacidad de reutilización del OA en diferentes contextos, la relación entre el objetivo del OA y su real cumplimiento a partir de lo ocurrido en el proceso de aprendizaje del alumno, los usos didácticos que el docente ha realizado del OA. (Sanz, C., Moralejo, L., Barranquero, F.,2014).

## Capítulo III

### **Análisis y diseño de la solución tecnológica de software**

En este capítulo se va a mencionar el análisis y diseño de la solución tecnológica y el curso de pensamiento computacional.

#### **Especificación de requerimientos**

##### Propósito

La especificación de requisitos de la solución se basó en el estándar IEEE/ANSI 830-1998, el cual se tomará como referencia, más no para seguirlo al pie de la letra, el propósito de la especificación de requisitos es determinar la estructura y diseño de los microsítios que tienen como finalidad presentar y organizar el curso de pensamiento computacional con las siguientes secciones:

- Glosario
- Ejercicios
- Buscador
- Ayuda
- Créditos

##### Ámbito del sistema

El micrositio ahora conocido como “Programando” cumple con el propósito de ser un organizador de contenido, este micrositio permitirá mostrar el contenido del curso de pensamiento computacional mediante un frame central que despliega el contenido generado por la herramienta de autor ExE Learning.

Además, permitirá desplegar las actividades de aprendizaje y las autoevaluaciones, los cursos estarán divididos en varias lecciones y todos con la misma organización.

Las funcionalidades que no cumple el micrositio son:

- No se solicitará registro de alumnos o docentes.
- No se solicitará registro de calificaciones o promedios
- No se permite retroalimentar los contenidos.
- No contiene chats ni foros de discusión.
- No permite crear contenidos, videos ni autoevaluaciones.

El micrositio puede ser levantado en la mayoría de plataformas educativas como Moodle y SELI.

## **Definiciones y acrónimos**

### Definiciones

- **Lección:** Contenido, con una introducción, objetivo y demostración sobre un tema en específico.
- **Actividad:** Tarea con el objetivo de reforzar los conceptos y conocimientos adquiridos.
- **Autoevaluación:** Son preguntas para comprobar cuánto conocimiento ha adquirido el estudiante.
- **Metadatos:** Es un grupo de datos que describen el contenido informativo de un objeto al que se denomina recurso.

- Objeto de Aprendizaje: Es una colección de elementos de contenido, práctica y valoración que se combinan en función de un solo objetivo de aprendizaje.
- Glosario: Es un catálogo alfabético de las palabras y expresiones de uno o varios textos que son difíciles de comprender, junto con su significado o algún comentario.

#### Acrónimos

- IR: Identificación de requerimiento.
- RE: Requerimiento específico.
- NR: Nombre del requerimiento.
- ERS: Especificación de Requerimientos de Software.

#### **Funciones del producto**

Las principales funcionalidades del micrositio son:

- Desplegar información acerca del contenido del curso de pensamiento computacional
- Organizar la información que se ingrese de tal manera que los temas a tratarse son fácilmente asociados.
- Buscar información mediante palabras que pueden asociarse a diferentes páginas web o en su defecto a definiciones en el glosario de términos.

#### **Características de los usuarios**

El micrositio está orientado a profesionales con sólidos conocimientos en HTML, JavaScript, Css y Bootstrap para su manipulación y mejoras.



## **Restricciones**

Como restricciones para el micrositio tenemos que:

- No es un asistente para desarrollar cursos.
- No es una herramienta de autor que genere contenidos web.
- No cuenta con un sistema de calificaciones.
- No cuenta con sistemas de comunicación.
- La seguridad no es imprescindible en los micrositios.

## **Requisitos futuros**

Los requisitos futuros para el micrositio son:

- El buscador de palabras y páginas puede ser implementado con nuevas herramientas que permitan vincular los principales buscadores de internet.
- Un sistema de calificaciones que permita al docente evaluar los conocimientos.
- Incorporar un gestor de contenidos, actividades y evaluaciones, de tal manera que se convierta en una herramienta de autor.
- El micrositio pueda generar un paquete SCORM para ser levantado y utilizado en cualquier plataforma educativa.

## **Requisitos funcionales**

En la Tabla 3 se puede observar los requisitos funcionales que se han establecido para el micrositio web.

**Tabla 3**  
*Requisitos funcionales*

<b>RF</b>	<b>NR</b>	<b>Características</b>
RF01	Interfaz para organizar las lecciones.	Los educadores podrán organizar el contenido de aprendizaje de acuerdo a lecciones y las páginas relacionadas.
RF02	Interfaz de despliegue de información de las lecciones.	Los micrositos muestran los objetos de aprendizaje.
RF03	Buscador de información	Se permitirá a los usuarios buscar palabras o páginas relacionadas dentro del contenido y las páginas relacionadas.
RF04	Compatibilidad con los navegadores	Los micrositos deben ser compatibles con los navegadores de internet actuales.

*Nota:* Esta tabla muestra los requisitos funcionales del proyecto.

### **Requisitos no funcionales**

En la Tabla 4 se puede observar los requisitos no funcionales que se han establecido para el micrositio web.

**Tabla 4**  
*Requisitos no funcionales*

<b>RNF</b>	<b>NR</b>	<b>Características</b>
RNF01	Rendimiento	Los micrositos no requieren muchas características de procesamiento y almacenamiento dependiendo del navegador en el que se ejecute.
RNF02	Fiabilidad	Los micros sitios constan de páginas debidamente organizadas y estructuradas lo que garantiza que la información se despliegue siempre.

RNF	NR	Características
RNF03	Disponibilidad	Garantiza la organización del contenido del curso mientras que las actividades y autoevaluación pueden ser incorporadas mediante código embebido.
RNF04	Operatividad	El micrositio es intuitivo y fácil de utilizar para los usuarios que no tienen conocimiento en manejo de páginas web.

*Nota:* Esta tabla muestra los requisitos NO funcionales del proyecto.

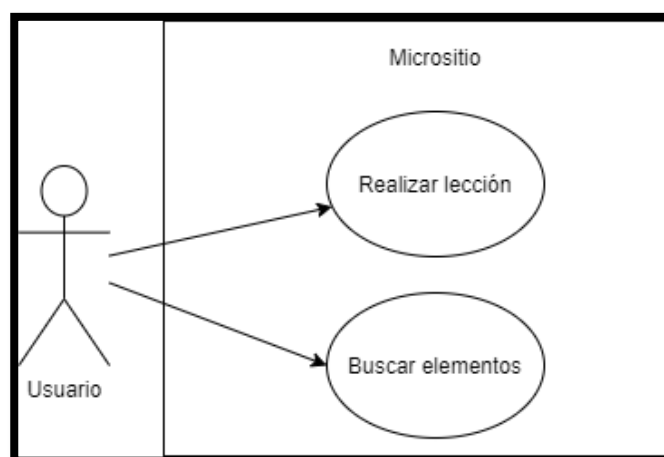
Diseño de la aplicación multimedia aplicando la metodología OOHDM

### Diseño conceptual

A continuación, en las figuras 7 y 8, se puede observar el diseño conceptual y el diagrama de casos de uso del micrositio representado por las clases, subsistemas, relaciones y atributos.

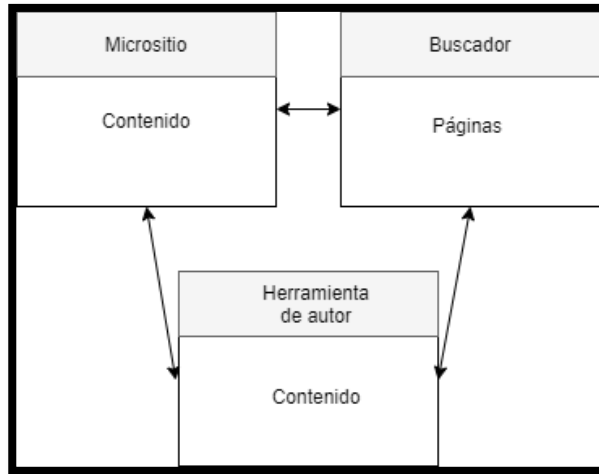
#### Figura 7

*Diagrama de casos de uso*



*Nota:* El gráfico muestra el diagrama de casos de uso del proyecto.

**Figura 8**  
*Diagrama conceptual*

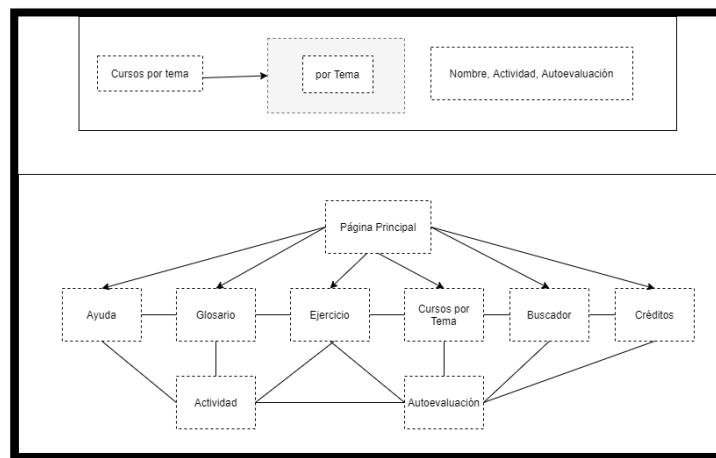


*Nota:* El gráfico muestra el diagrama conceptual del proyecto.

### Diseño navegacional

Este diseño provee una vista subjetiva del diseño conceptual sobre el micrositio, el curso de pensamiento computacional, los enlaces, nodos y estructuras de acceso, como se puede observar en la Figura 9.

**Figura 9**  
*Diagrama navegacional*



*Nota:* El gráfico muestra el diagrama navegacional del proyecto.

## Interfaz abstracta

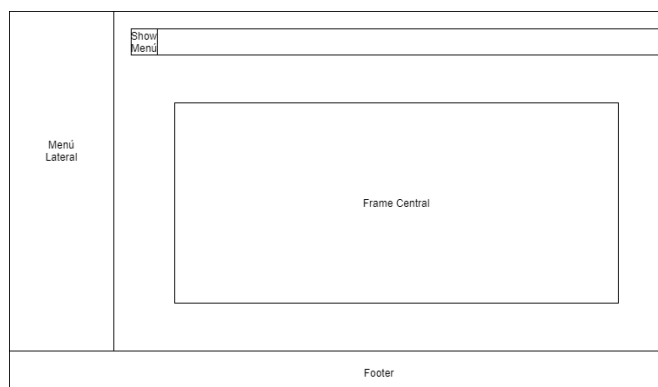
Para el micrositio, hemos definido que la interfaz estará compuesta de la siguiente manera:

- Un frame central el cual permitirá desplegar los objetos de aprendizaje mediante la vinculación con los archivos HTML que generan ExE Learning y Ardora respectivamente.
- En el costado izquierdo constará de una barra de navegación con los temas a tratarse generada con Bootstrap.
- Un footer con links a paginas especificas tales como el buscador, glosario de términos, ayuda, créditos y un ejercicio propuesto,
- En el header de la página se encuentra un botón para poder desplegar y ocultar el menú navegador lateral.

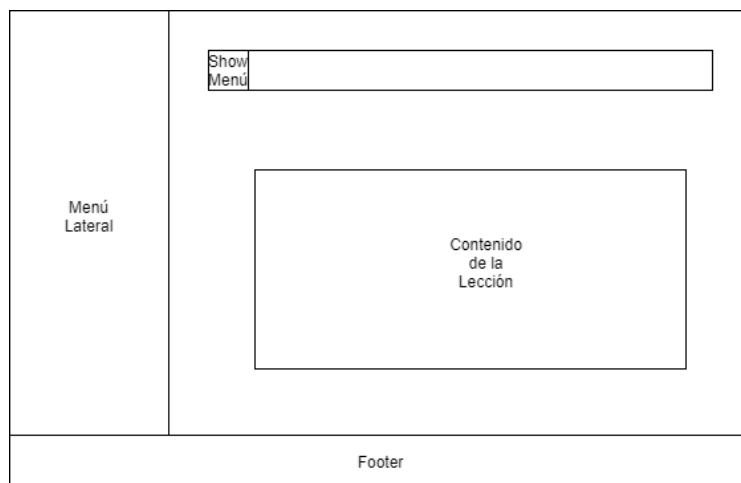
En las figuras 10, 11, 12, 13, 14, 15 se puede observar los diseños para las diferentes páginas del micrositio.

### Figura 10

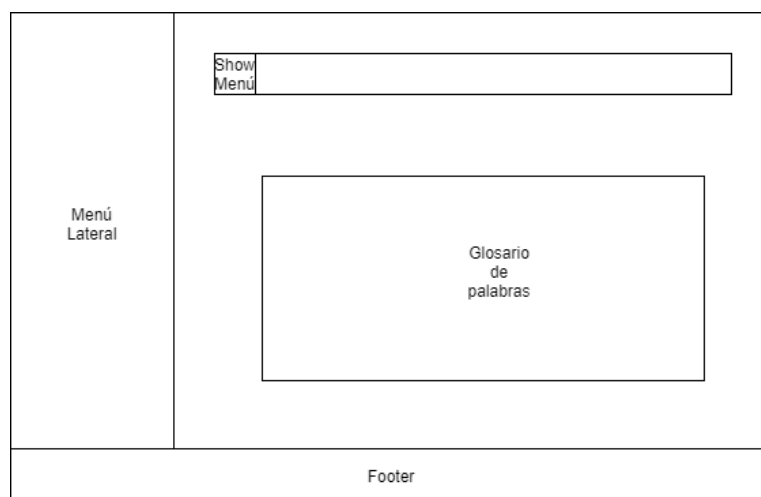
*Diseño de la interfaz abstracta 1*



*Nota:* El gráfico muestra una interfaz abstracta del proyecto.

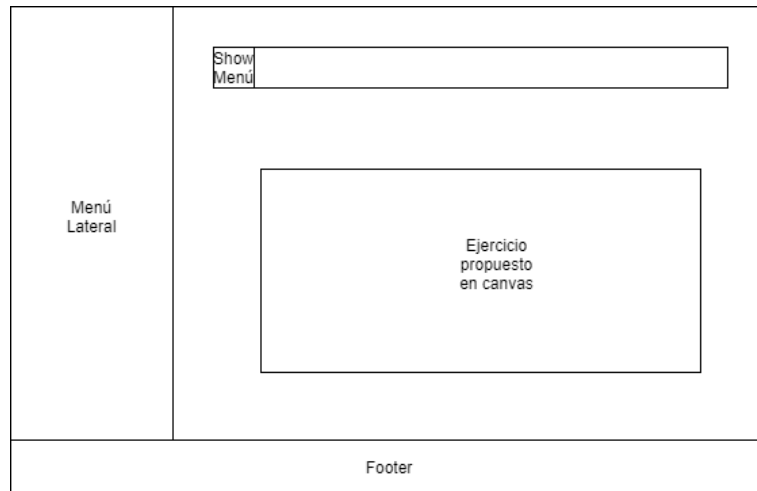
**Figura 11***Diseño de la interfaz abstracta 2*

*Nota:* El gráfico muestra una interfaz abstracta del proyecto.

**Figura 12***Diseño de la interfaz abstracta 3*

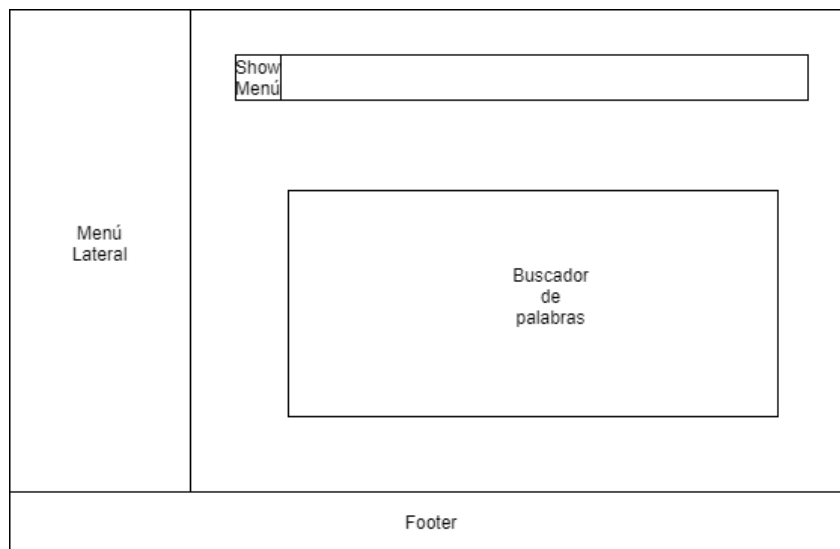
*Nota:* El gráfico muestra una interfaz abstracta del proyecto.

**Figura 13**  
*Diseño de la interfaz abstracta 4*



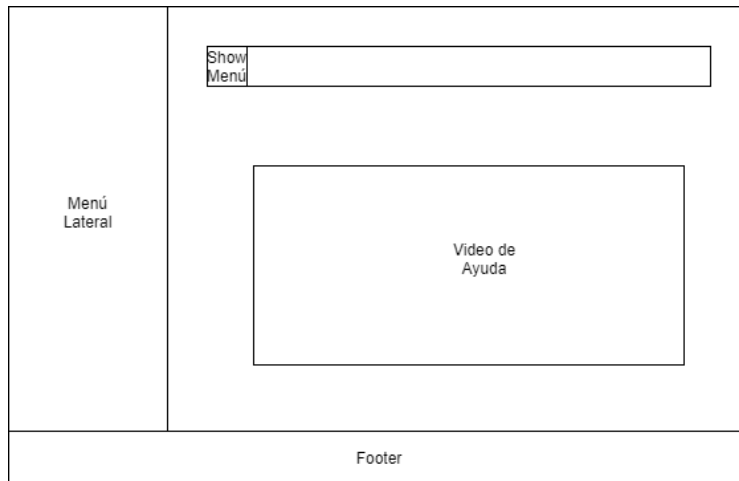
*Nota:* El gráfico muestra una interfaz abstracta del proyecto.

**Figura 14**  
*Diseño de la interfaz abstracta 5*



*Nota:* El gráfico muestra una interfaz abstracta del proyecto.

**Figura 15**  
*Diseño de la interfaz abstracta 6*



*Nota:* El gráfico muestra una interfaz abstracta del proyecto.

Análisis y diseño de los objetos de aprendizaje aplicando la metodología CROA

Análisis

Para este análisis se ha tomado como guía las preguntas más importantes de la metodología CROA que se irán respondiendo en el proceso.

¿Qué es lo que los destinatarios necesitan aprender?

El principal tema que deben aprender los estudiantes, es el desarrollo del pensamiento computacional mediante la programación de videojuegos.

¿Por qué se cree que es necesario utilizar un OA y no otro tipo de material educativo?

La principal necesidad es orientar el aprendizaje de los estudiantes mediante actividades y autoevaluaciones, no solo por información, además de que el OA será fácil de acceder, compartir y guardar en un repositorio para su reutilización.



¿A qué nivel educativo se orienta?

Los OA deben estar orientados a jóvenes aprendientes de 11 a 12 años de edad con un nivel de educación básica.

¿Cuál es el tema que el OA abordará?

Los temas que los OA abordan son el desarrollo de videojuegos en 3 diferentes plataformas empezando con Scratch, seguido por RPG Maker y terminando en AppInventor, todos estos temas se abordarán mediante paquetes SCORM creados por ExELearning.

¿Qué conocimientos previos debería tener el alumno para utilizar al OA?

Los conocimientos específicos previos que deben tener los jóvenes aprendientes se limitan solo al uso básico de un equipo computacional, y el uso de páginas web.

¿Con qué otros conocimientos se relacionan el OA que pueden ser adquiridos en forma posterior a trabajar con lo que se está diseñando?

Los posibles conocimientos que se pueden adquirir luego de trabajar con los OA son la programación en herramientas como Scratch, la creación de videojuegos en RPG Maker y la creación de videojuegos en aplicaciones móviles con AppInventor.

Diseño

En esta etapa consideramos tres sub etapas importantes que nos ayudaran con el diseño de los OA.

## Diseño instruccional del OA

¿Qué objetivo de aprendizaje específico se propone para el OA?

El objetivo principal es aprender a desarrollar videojuegos y el pensamiento computacional mediante tres herramientas diferentes.

¿Qué información proveerá el OA acerca del tema a tratar, como se organizará y presentará esa información?

El OA presentará información acerca de las 3 herramientas, Scratch, RPG Maker y AppInventor y su respectiva funcionalidad orientada a videojuegos, la cual se presenta mediante videos tutoriales.

¿Qué actividades se le propondrán al alumno?

Como actividades para el alumno se plantearon sopas de letras, crucigramas y el juego del ahorcado con temas relacionados a la lección vista previamente y sus conceptos.

¿En qué consistirá la autoevaluación?

Las autoevaluaciones consistirán en listas desordenadas sobre instrucciones de Scratch y preguntas de verdadero y falso para RPG Maker y AppInventor.

¿De qué manera se le dará feedback al estudiante sobre su desempeño en esta?

Al momento de resolver las listas desordenadas se muestran las respuestas correctas y en el caso de las preguntas, cada pregunta tiene su explicación de por qué es verdadera o falsa.

Estructura del OA

Para la estructura de los OA se ha tomado la estructura lineal ya que es necesario ir en un orden determinado para poder avanzar a las demás lecciones o temas y aplicar lo aprendido previamente, como se puede observar en la Figura 16.

**Figura 16**  
*Estructura Lineal*



Nota: El gráfico representa la estructura que llevan los objetos de aprendizaje.

Tomado de Metodología CROA, Anexo 2(p.1), por Sanz, C., Moralejo, L., Barranquero, F.,2014.

Diseño multimedia basado en escenarios

Para el curso de pensamiento computacional se ha propuesto un diseño basado en un escenario de contenido audiovisual ya que consta con la presentación de una introducción, el objetivo de la lección y un video tutorial, todo ensamblado mediante paquetes SCORM de la herramienta ExELearning.

Para las actividades se ha propuesto utilizar Ardora como herramienta principal para generar juegos de palabras como sopas de letras, crucigramas y el ahorcado los cuales tienen como tema la lección estudiada previamente, para las autoevaluaciones se ha propuesto utilizar ExELearning para generar paquetes SCORM que contengan las listas desordenadas y los cuestionarios de verdadero y falso.

## Capítulo IV Desarrollo e implementación

En este capítulo se va a mencionar el desarrollo e implementación de la solución tecnológica y el curso de pensamiento computacional.

### Desarrollo de la aplicación multimedia aplicando la metodología OOHDM

Para esta etapa se realizará la implementación del diseño del micrositio.

### Implementación del menú lateral

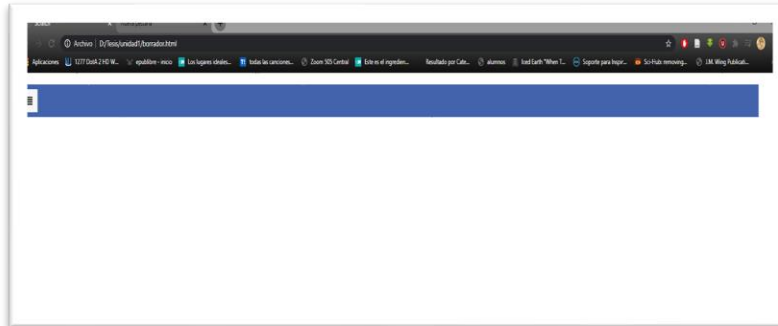
Para la implementación del diseño del menú lateral se ha utilizado Bootstrap como framework para adaptarlo a las necesidades del micrositio, como se puede apreciar en las Figuras 17 y 18, además se ha creado un botón que permite mostrar y ocultar el menú ubicado en la cabecera del micrositio, los colores y estilos se han aplicado mediante un archivo Css global, de igual manera para los demás micrositios se implementa el mismo estilo por lo que se pueden apreciar en la sección de Anexos.

**Figura 17**  
*Menú Lateral*



*Nota:* El gráfico muestra la implementación del menú lateral del micrositio

**Figura 18**  
*Menú Lateral Oculto*

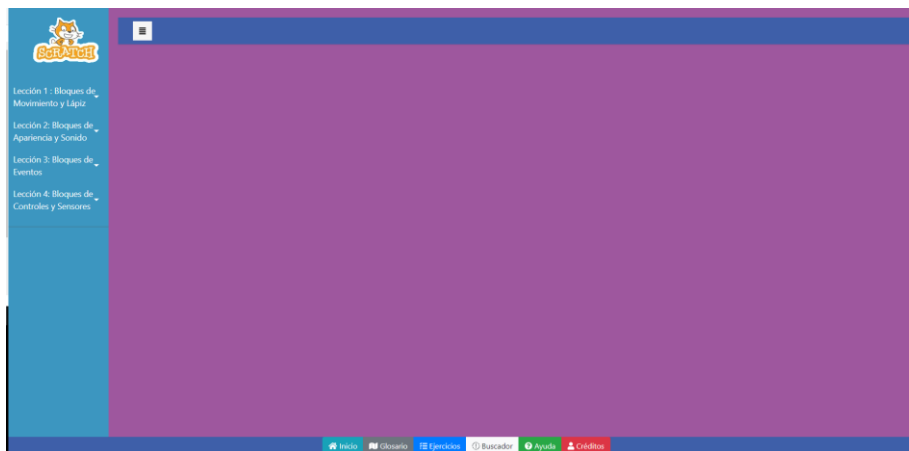


*Nota:* El gráfico muestra la implementación del menú lateral oculto del micrositio

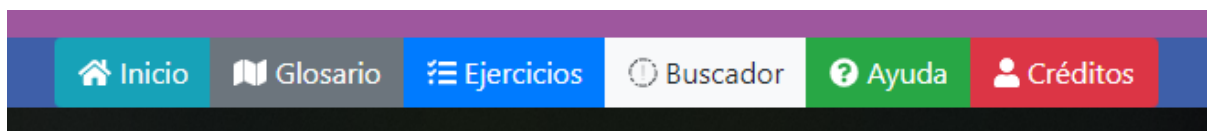
### Implementación de footer y links

Para el desarrollo de la sección del footer que contiene enlaces a páginas específicas dentro del micrositio se utiliza Bootstrap para personalizar los colores y tamaños de los botones además que se agregó diferentes íconos en los mismos para distinguir su apariencia y funcionalidad, como se puede observar en las Figuras 19 y 20.

**Figura 19**  
*Pantalla del footer*



*Nota:* El gráfico muestra la implementación del footer del micrositio

**Figura 20***Botones del footer*

*Nota:* El gráfico muestra la implementación del footer del micrositio

**Implementación del frame central y la función buscar**

Para la implementación del frame central se estableció una sección que se pueda editar y manipular de acuerdo al contenido de cada página, tal como se puede observar en las Figuras 21, 22, 23, 24, 25, 26 y 27, de la siguiente manera:

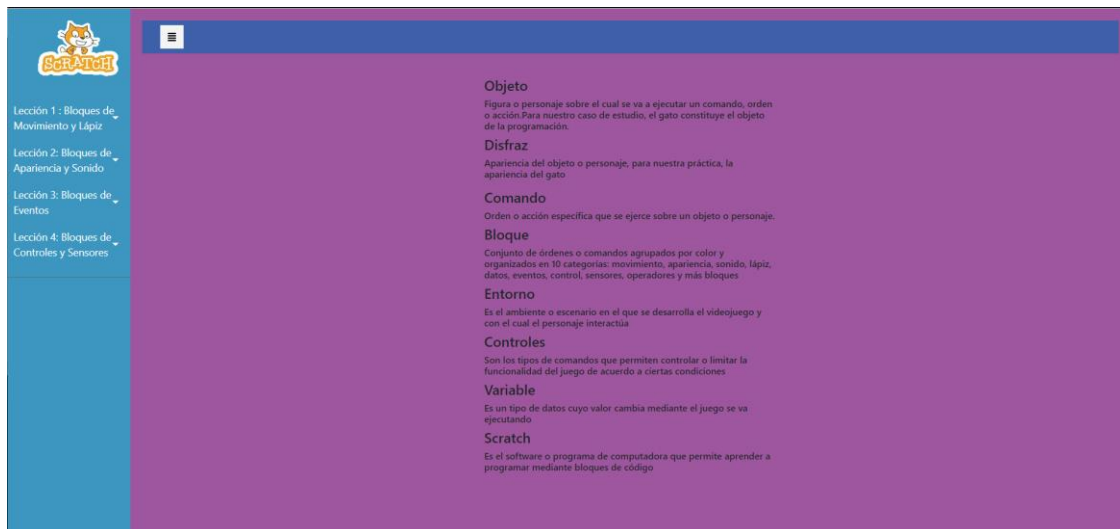
- Para las lecciones, en el frame se despliega el contenido generado por la herramienta de autor, ya sea en ExELearning o Ardora.
- Para la sección de glosario en el frame se despliega el contenido en forma de texto.
- Para la sección de ejercicios propuestos en el frame se despliega una imagen creada en canvas.
- Para la sección de ayuda en el frame se despliega un video mediante código embebido desde YouTube.
- Para la sección de créditos el frame desplegará contenido en formato de texto.

**Figura 21**  
Pantalla de inicio



*Nota:* El gráfico muestra la implementación de la pantalla principal del micrositio

**Figura 22**  
Pantalla de glosario



*Nota:* El gráfico muestra la implementación del glosario de términos del micrositio

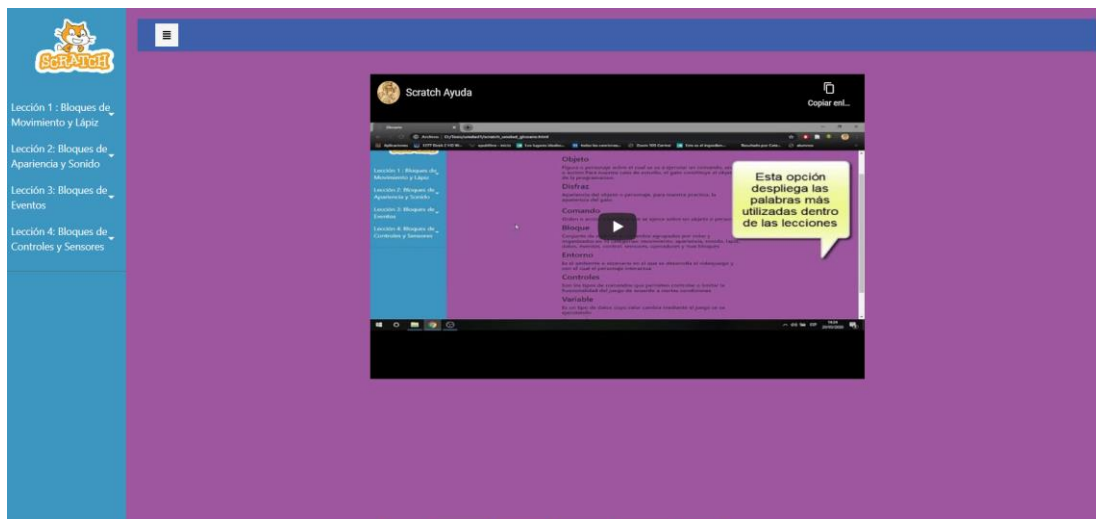


**Figura 23**  
Pantalla de ejercicios propuestos



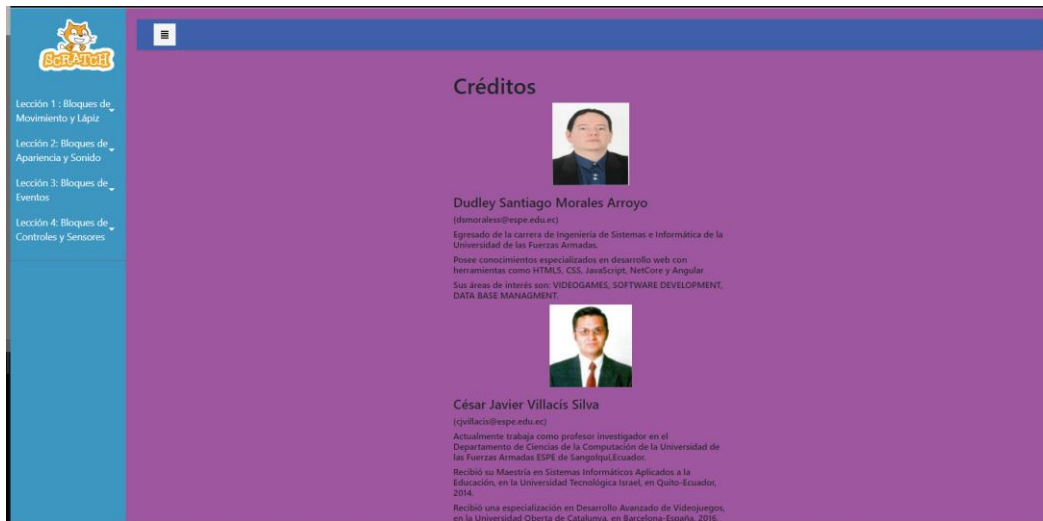
*Nota:* El gráfico muestra la implementación de los ejercicios propuestos del micrositio

**Figura 24**  
Pantalla de Ayuda



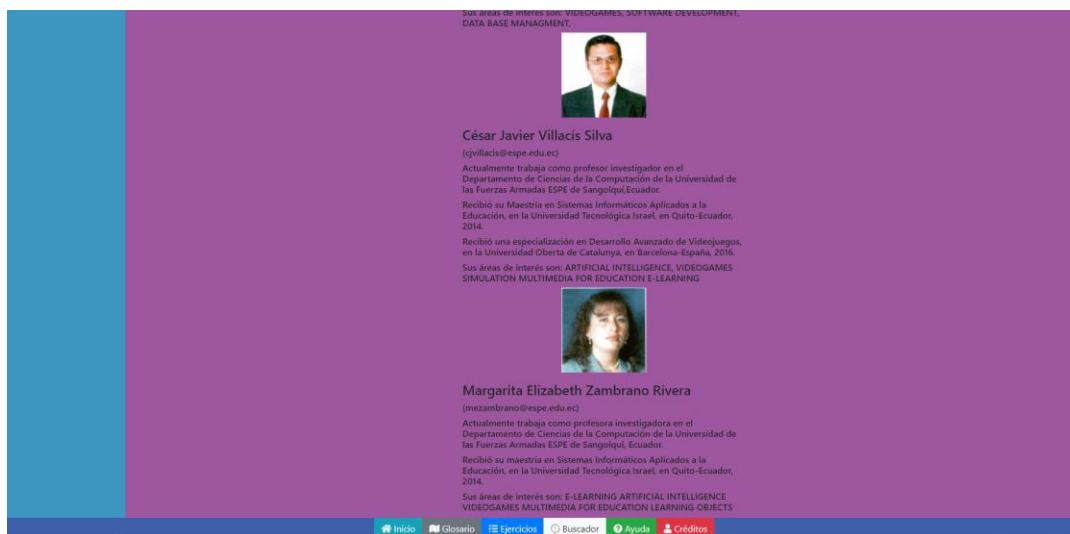
*Nota:* El gráfico muestra la implementación de la página de ayuda del micrositio

**Figura 25**  
Pantalla de créditos



*Nota:* El grafico muestra la implementación de la página de créditos del micrositio

**Figura 26**  
Pantalla de créditos



*Nota:* El grafico muestra la implementación de la página de créditos del micrositio

Para la implementación del buscador, se desarrollaron funciones en JavaScript para la búsqueda de palabras del glosario o elementos referentes a las lecciones, en la misma página de búsqueda se muestran todos los resultados obtenidos, como se puede observar en la Figura 27, los resultados están programados para enlazarse a páginas específicas mediante una pequeña base de datos hecha en JavaScript.

**Figura 27**  
*Pantalla del Buscador*



*Nota:* El grafico muestra la implementación del buscador de términos del micrositio

## **Desarrollo de los objetos de aprendizaje aplicando la metodología CROA**

En esta etapa se aborda la selección de recursos que se incorporan en los OA, además de utilizar las plantillas de las herramientas de autor para conformar el OA íntegramente.

## Selección de recursos

Como recursos principales para los OA se ha establecido utilizar videos tutoriales sobre los temas a tratarse, los cuales estarán almacenados en YouTube en el canal del siguiente enlace

[https://www.youtube.com/playlist?list=PLhOm7mpc\\_MFRGm9tGIEeAJ-](https://www.youtube.com/playlist?list=PLhOm7mpc_MFRGm9tGIEeAJ-)

[8aGU7h2h19](#), lo cual garantiza su disponibilidad en todo momento, además serán embebidos al micrositio.

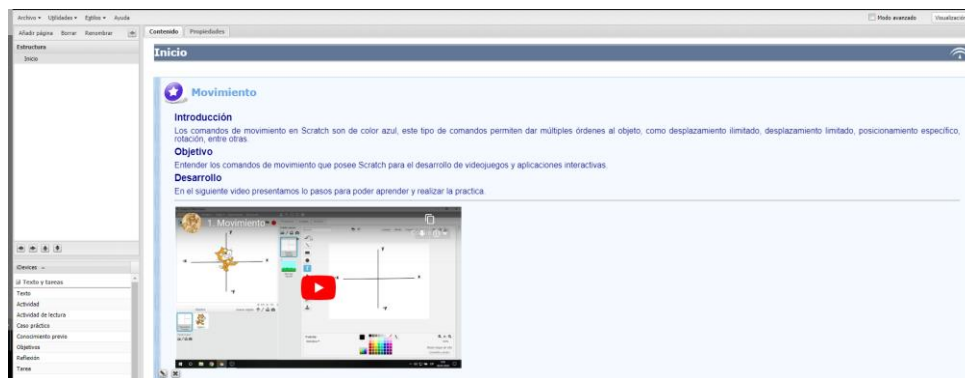
## Desarrollo de contenido basado en plantillas

Como principal herramienta para el desarrollo se utiliza ExE Learning, el cual se estructura de la siguiente manera, una breve introducción, el objetivo de la lección y el video tutorial.

Por otra parte, las actividades de aprendizaje están realizadas en Ardora y las autoevaluaciones en ExE Learning con una diferente estructura para ser embebidas en páginas diferentes en el micrositio, como se puede observar en las figuras 28 y 29.

### Figura 28

*Objetos de aprendizaje en ExE Learning*



*Nota:* El gráfico muestra la creación de las lecciones en ExE Learning

**Figura 29**  
*Plantilla de autoevaluación*

**Autoevaluación**

Ordene los elementos de tal manera que al moverse la ruta del personaje sea un rectángulo

- Mover 50 pasos
- Al presionar bandera
- Mover 100 pasos
- Girar a la izquierda 90 grados
- Mover 100 pasos
- Girar a la izquierda 90 grados
- Mover 50 pasos
- Girar a la izquierda 90 grados
- Fin

Comprobar Respuesta

*Nota:* El gráfico muestra la implementación de las autoevaluaciones en ExELearning

Para las actividades de aprendizaje se han planteado previamente tres diferentes tipos los cuales son, sopa de letras, crucigramas y el juego del ahorcado, cada una de estas actividades ayudan a reforzar el tema aprendido mediante el recordatorio de conceptos y funcionalidades vistos previamente.

Para el caso de la sopa de letra, se tiene palabras relacionadas con los conceptos de las lecciones, las cuales deben ser encontradas en un tiempo de 120 segundos, como se puede observar en la Figura 30.

**Figura 30**  
*Sopa de letras en Ardora*

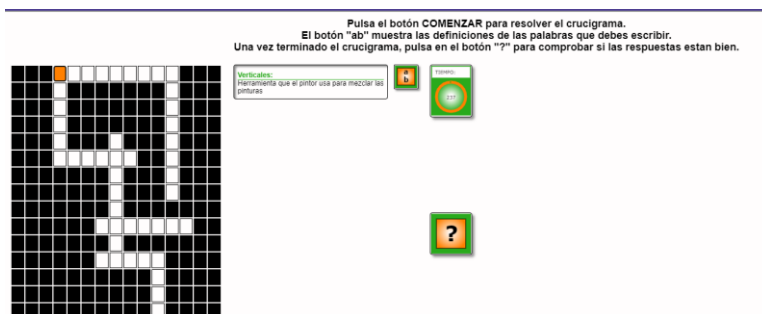
Pulsa en el botón **COMENZAR** para buscar en la sopa de letras las palabras de la lista, las cuales están relacionadas con los comandos de Movimiento y Lápiz en Scratch. Tienes 2 minutos para encontrarlas todas

S	C	Ú	Q	U	H	U	H	B	C	G	E	Y	Definición		
H	O	Ñ	F	L	U	A	R	B	G	Ñ	H	A	B	<input type="checkbox"/> MOVER	
N	L	H	C	I	D	E	R	E	C	H	A	S	U	W	<input type="checkbox"/> GIRAR
I	O	K	Ñ	S	E	L	L	A	R	B	D	M	A	A	<input type="checkbox"/> IZQUIERDA
K	R	O	A	U	N	N	I	M	B	I	H	N	I	I	<input type="checkbox"/> DERECHA
Á	Ó	K	Ñ	H	Á	N	G	Ú	D	A	I	D	E	U	<input type="checkbox"/> DESLIZAR
Ú	I	N	T	E	N	S	I	D	A	D	E	A	Q	J	<input type="checkbox"/> BORRAR
Ñ	G	Z	Ñ	P	Z	S	R	E	I	S	L	A	V	M	<input type="checkbox"/> SELLAR
B	L	Ú	I	V	Z	F	A	Ú	S	Ú	P	T	F	O	<input type="checkbox"/> COLOR
B	O	R	R	A	R	Y	R	O	C	S	F	I	O	M	<input type="checkbox"/> INTENSIDAD
I	X	I	P	H	T	M	D	E	S	L	I	Z	A	R	<input type="checkbox"/> TAMAÑO
Z	V	X	X	R	G	T	L	H	A	H	U	K	Q	R	
Ú	H	I	T	B	U	I	Z	Q	U	I	E	R	D	A	
Á	U	A	S	T	G	T	A	M	A	Ñ	O	G	T	B	
R	C	M	O	V	E	R	G	G	X	Q	E	H	J	U	

*Nota:* El gráfico muestra la implementación de la sopa de letras en Ardora para el micrositio

Para el caso del crucigrama, se tienen 9 palabras con sus respectivas definiciones y sinónimos sobre términos y elementos de las lecciones, este crucigrama debe ser resuelto en 240 segundos, como se puede observar en la Figura 31.

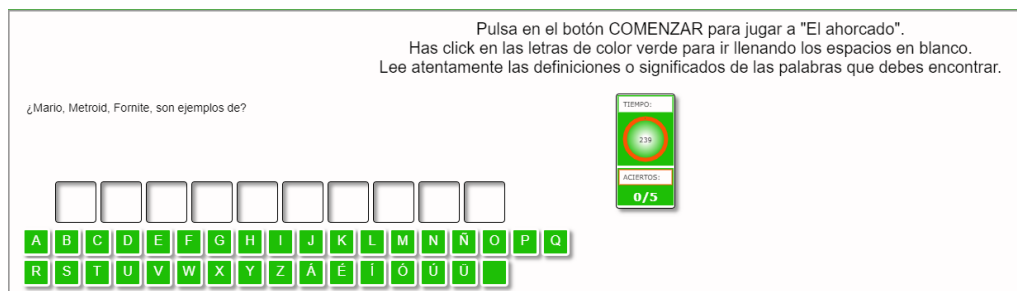
**Figura 31**  
*Crucigrama en Ardora*



*Nota:* El grafico muestra la implementación del crucigrama en Ardora para el micrositio

Finalmente, para el juego del ahorcado se tiene 5 palabras con sus respectivos sinónimos y significados, los cuales están relacionados a los elementos y términos de los juegos en las lecciones vistas previamente, su duración es de 240 segundos, como se puede observar en la Figura 32.

**Figura 32**  
*Juego del ahorcado en Ardora*



*Nota:* El grafico muestra la implementación del juego del ahorcado en Ardora para el micrositio

## Ingreso de metadatos del OA

Para el ingreso de metadatos en los OA se utiliza ExELearning el cual permite ingresarlos basándose en el estándar LOM, de acuerdo a la metodología CROA no todos los campos son necesarios, algunos se pueden omitir, quedando la información de metadatos para los OA de la siguiente manera como se puede observar en la Figura 33.

**Figura 33**  
*Metadatos en estándar LOM*

The screenshot displays the 'Propiedades' (Properties) window in ExELearning, specifically the 'Metadatos' (Metadata) tab. The form is organized into several sections:

- General (Obligatorio):**
  - Identificador (Obligatorio):** Código: 'Curso Scratch', Entrada: '4470759-3445-403a-a764-60388113623'.
  - Título (Obligatorio):** 'Curso de movimiento en Scratch'.
  - Idioma (Obligatorio):** 'Español'.
  - Descripción (Obligatorio):** 'Uso de los comandos Movimiento en Scratch, orientado a niños y jóvenes aprendices'.
- Palabras clave (Opcionales):** 'Scratch', 'Movimiento'.
- Colectores (Opcionales):** (Empty field).
- Estructura (Opcionales):** 'lineal'.
- Nivel de apropiación (Obligatorio):** 'objetos de aprendizaje'.

*Nota:* El gráfico muestra el ingreso de metadatos en los objetos de aprendizaje en ExELearning

## Evaluación

Para la evaluación del curso de pensamiento computacional se utilizó la matriz para la evaluación de objetos de aprendizaje basada en CODA, debido al cierre de los centros educativos por la pandemia global, se ha tomado como muestra jóvenes bachilleres recién ingresados a la carrera de Ingeniería de Software en la Universidad de las Fuerzas Armadas para realizar dicha evaluación, como se puede observar en la tabla 5, la matriz de evaluación consta de 30 preguntas con un valor de 1 a 5 puntos para cada una.

**Tabla 5**  
**Matriz de Evaluación CODA**

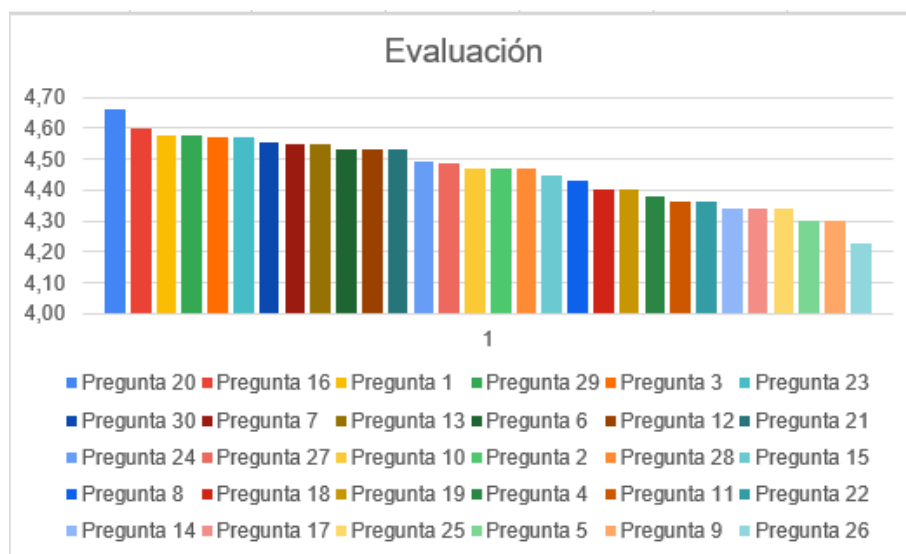
	Pregunta	Escala de Valoración				
	<b>Objetivos y Coherencia Didáctica</b>					
1	Existe coherencia entre los objetivos y contenidos del Objeto de Aprendizaje.	1	2	3	4	5
	<b>Calidad de Contenidos</b>					
2	La presentación del contenido es clara.					
3	Las actividades presentan instrucciones claras de cómo realizarlas.	1	2	3	4	5
4	El número y distribución de los conceptos dentro de los Objetos de Aprendizaje es equilibrado.	1	2	3	4	5
5	Se destacan las ideas clave de tal forma que se perciben las ideas fundamentales.	1	2	3	4	5
6	El contenido está de acuerdo a su perfil de conocimiento.	1	2	3	4	5
7	El contenido es coherente con los objetivos planteados.	1	2	3	4	5
8	La información es veraz, exacta y se presenta con un nivel de detalle suficiente.	1	2	3	4	5
9	El contenido está actualizado.	1	2	3	4	5
10	El contenido respeta los derechos de propiedad intelectual cuando utiliza otras fuentes.	1	2	3	4	5
	<b>Capacidad de generar reflexión, crítica e innovación</b>					
11	El Objeto de Aprendizaje estimula la reflexión sobre las ideas presentadas.	1	2	3	4	5
12	El Objeto de Aprendizaje promueve/facilita el descubrimiento, generación y adquisición de ideas de aprendizaje de forma autónoma.	1	2	3	4	5
13	El Objeto de Aprendizaje fomenta la capacidad de relacionar conceptos ya aprendidos con los nuevos conceptos.	1	2	3	4	5
	<b>Interactividad y Adaptabilidad del Objeto de Aprendizaje</b>					
14	El contenido que se presenta está relacionado con preguntas, respuestas o acciones realizadas anteriormente.	1	2	3	4	5
15	En su rol de participante siente que realmente controla y maneja su aprendizaje.	1	2	3	4	5
16	La presentación del contenido es automática, programada, o manual.	1	2	3	4	5



17	El Objeto de Aprendizaje propone diferentes contenidos/actividades que generen la competencia del participante.	1	2	3	4	5
18	El Objeto de Aprendizaje puede ser utilizado por el docente como herramienta de apoyo en su método de enseñanza/aprendizaje.	1	2	3	4	5
<b>Motivación</b>						
19	El Objeto de Aprendizaje hace referencia directa a su utilidad en el mundo real.	1	2	3	4	5
20	El Objeto de Aprendizaje le permite a usted como participante percibir que lo que está aprendiendo es relevante/significativo.	1	2	3	4	5
<b>Formato y Diseño</b>						
21	El Objeto de Aprendizaje presenta de forma innovadora o atractiva los contenidos o los procedimientos didácticos.	1	2	3	4	5
22	El diseño del Objeto de Aprendizaje es claro, conciso y bien organizado.	1	2	3	4	5
23	El diseño de los contenidos favorece la comprensión y asimilación del conocimiento.	1	2	3	4	5
24	Se utilizan texto imágenes, audio para aprovechar las diferentes formas de aprendizaje.	1	2	3	4	5
25	El Objeto de Aprendizaje es adecuado para el estudio y reflexión sin distraer el aprendizaje.	1	2	3	4	5
26	Los textos imágenes y audio son de buena calidad.	1	2	3	4	5
<b>Usabilidad</b>						
27	Es fácil navegar en el contenido digital del Objeto de Aprendizaje.	1	2	3	4	5
28	El Objeto de Aprendizaje, presenta una interfaz intuitiva e informa implícitamente a usted como participante, cómo interactuar sus instrucciones de uso, son claras.	1	2	3	4	5
29	Todos los enlaces funcionan correctamente, no hay enlaces rotos o que conduzcan a un contenido erróneo.	1	2	3	4	5
30	La accesibilidad de los contenidos de la web se muestra con facilidad.	1	2	3	4	5

*Nota:* Esta tabla muestra las preguntas y el puntaje para la evaluación de objetos de aprendizaje

**Figura 34**  
Resultados de la evaluación



Nota: La figura presenta los resultados de la evaluación a los objetos de aprendizaje

En la figura 34 se puede observar los resultados de la evaluación, con una calificación entre 1 y 5 para las 30 preguntas, de las cuales podemos notar que las preguntas con mejor puntuación son la 20,16 y 1, por lo que se puede entender que, los objetos de aprendizaje permiten percibir al estudiante que lo que está aprendiendo es relevante, la presentación del contenido educativo es automático y finalmente existe una gran coherencia entre los objetivos y contenidos de los objetos de aprendizaje.

Por otra parte, las preguntas que presentan la menor puntuación son la 5,9 y 26, por lo que se puede entender que las ideas claves deben destacar de una mejor manera y los elementos multimedia pueden ser de una mejor calidad.

## **Capítulo V**

### **Conclusiones y recomendaciones**

#### **Conclusiones**

- Para la creación de los objetos de aprendizaje, la apariencia, es uno de los parámetros más importantes que se deben tomar en cuenta, de acuerdo a la matriz de evaluación, la apariencia de los objetos de aprendizaje favorece la comprensión y asimilación del conocimiento.
- Al desarrollar el marco teórico sobre el pensamiento computacional y la educación STEAM, se puede apreciar que la educación a nivel mundial está cada vez más predispuesta a probar y utilizar nuevas metodologías, esto debido a los diferentes proyectos que se han promovido en países como Estados Unidos y España.
- El uso de los videos tutoriales como herramientas dentro del proceso de enseñanza y aprendizaje cumplen un rol importante, ya que esto le permite al estudiante llevar el curso a su ritmo y así comprender de mejor manera el contenido educativo.
- La metodología CROA posee el proceso más completo y adecuado para diseñar y crear OA, debido a la efectividad de sus subprocesos, ya que como la matriz de evaluación CODA lo muestra, los OA del curso de pensamiento computacional tienen en promedio puntajes mayores a 4 dentro del intervalo de 1 a 5 puntos.
- El método OOHDM posee el proceso más adecuado para diseñar y crear la estructura de los microsítios, de tal manera que los paquetes SCORM creados por las herramientas de autor se despliegan de manera óptima y permiten al estudiante navegar a gusto entre los diferentes contenidos educativos o páginas con otras funcionalidades.

- Los test para la autoevaluación están diseñados para que el conocimiento adquirido por los alumnos pueda ser confirmado mediante múltiples intentos para resolver los mismos, de tal manera que puede recibir una retroalimentación constante de lo aprendido.

### **Recomendaciones**

- Para la evaluación de objetos de aprendizaje se recomienda utilizar la matriz de evaluación CODA, ya que sus parámetros abarcan muchas características y aspectos de los objetos de aprendizaje.
- A pesar que los micrositos son intuitivos y amigables, pueden presentar ciertas dificultades para los futuros usuarios que no estén acostumbrados al uso de las tecnologías, por lo que se recomienda familiarizarse con los elementos y funcionalidades de los micrositos.
- Se recomienda que los futuros alumnos del curso de pensamiento computacional tengan sus propias herramientas informáticas ya que en el caso de Scratch y RPG Maker se pueden trabajar de manera local, mientras que AppInventor se puede utilizar en la nube, siendo solo necesario para ejecutar su código un dispositivo móvil con Android o el simulador instalado en un pc.

### **Trabajos Futuros**

Implementar y mejorar los elementos evaluados en la matriz de CODA que presenten el menor puntaje de la evaluación.

## Referencias bibliográficas

- Alfaro, M. (5 de Diciembre de 2019). *Mujer Peruana*. Obtenido de <https://www.mujerperuana.com/educacion/educacion-tradicional-versus-educacion-steam/>
- Alsina, Á. (2020). Conexiones matemáticas a través de actividades STEAM en Educación Infantil. *Revista Iberoamericana de educación matemática*, 168-190.
- AspGems. (05 de Abril de 2019). *ASPGems*. Obtenido de <https://aspgems.com/metodologia-de-desarrollo-de-software-iii-modelo-en-espiral/>
- Bruer, J. (2010). PREPARING THE NEXT GENERATION OF STEM INNOVATORS:Identifying and Developing our Nation's Human Capital. *National Science Board*, 62. Obtenido de <https://www.nsf.gov/nsb/publications/2010/nsb1033.pdf>
- Bushley, M., Gettings, M., Hancock, B., Fiege, W., & White, J. (27 de Agosto de 2013). *STEAM White Paper:Planning for the Plan*. Obtenido de <http://www.edleader21.org/info/CCPS-STEAM-whitepaper.pdf>
- Cilleruelo, L., & Zubiaga, A. (2014). Una aproximacion a la Educación STEAM. Practicas educativas en la encrucijada arte, ciencia y tecnologia. *Jornadas de Psicodidáctica*, 18.
- Code.org. (Enero de 2013). *Code*. Obtenido de <https://code.org/educate/spritelab>
- Connor, A., Karmokar, S., Whittington, C., & Walker, C. (2014). Full STEAM ahead a manifesto for integrating arts pedagogics into STEM education. *IEEE International Conference on Teaching*.
- Domínguez, E., Armas, I., & Fernández, A. (2012). Herramienta de Evaluación de la Calidad de Objetos de Aprendizaje. 16.
- Drake, J. (2008). Proceso de desarrollo de aplicaciones de software. 37-38.
- Educación, M. d. (15 de Junio de 2008). *Ministerio de Educación*. Obtenido de <https://educacion.gob.ec/>
- Fischer, G. (2001). The Software Technology of the 21st Century:From Software Reuse to Collaborative Software Design. 8.
- Greenberg, D. (2008). *Turning Learning Right Side Up: Putting education back on track*. New Jersey: Pearson Education.
- Guenaga, M. (15 de Diciembre de 2015). *DeustoTEch*. Obtenido de <https://pt.slideshare.net/innobasque/deustotech-trastea-club>

- Huertas, I., Moreno, J., Flor, P., & Moriana, M. (31 de Mayo de 2019). *Programamos*. Obtenido de <https://programamos.es/>
- Innobasque Agencia Vasca de la Innovación*. (8 de Abril de 2016). Obtenido de [https://www.innobasque.eus/microsite/escuela\\_innovacion/article/de-que-hablamos-cuando-hablamos-de-steam/](https://www.innobasque.eus/microsite/escuela_innovacion/article/de-que-hablamos-cuando-hablamos-de-steam/)
- Lapuenta, M. (8 de Diciembre de 2013). *Modelo OOHDM*. Obtenido de <http://www.hipertexto.info/documentos/oohdm.htm>
- Maldonado, J., Carvallo, J., & Siguencia, J. (2017). Metodologías y Propuestas Metodológicas Para el Diseño de Objetos de Aprendizaje: Un Estado del Arte en Iberoamérica. 45.
- Mitra, S. (2007). *LIFT*. Obtenido de [https://www.ted.com/talks/sugata\\_mitra\\_shows\\_how\\_kids\\_teach\\_themselves/transcript?source=email&language=es#t-676717](https://www.ted.com/talks/sugata_mitra_shows_how_kids_teach_themselves/transcript?source=email&language=es#t-676717)
- Parra, A. M. (21 de Septiembre de 2019). *Rockcontent*. Obtenido de <https://rockcontent.com/es/blog/pensamiento-computacional/>
- Patrovi, H., & Patrovi, A. (3 de Enero de 2013). *Code.org*. Obtenido de <https://code.org/>
- Perales, J., & Aguilera, D. (2019). Educación STEAM: algo más que unas siglas. *Ciencia Abierta*, 1.
- Sánchez, A., & Monteagudo, P. (4 de Octubre de 2017). *El video como medio de enseñanza*. Obtenido de [http://bvs.sld.cu/revistas/ems/vol21\\_2\\_07/ems06207.htm](http://bvs.sld.cu/revistas/ems/vol21_2_07/ems06207.htm)
- Sanchez, E. (2019). La educación STEAM y la cultura maker. *Innovación Educativa*, 7.
- Sanders, M. (2012). Integrative Stem Education as “Best Practice”. *Virginia Tech*, 15.
- Sanz, C., Barranquero, F., & Moralejo, L. (2016). Metodología CROA para la creación de Objetos de Aprendizaje. *Instituto de investigación en informática*, 7.
- Schools, C. C. (2013). *STEAM*. Obtenido de <https://www.collierschools.com/Page/13729>
- Secretaria de educación superior, c. t. (30 de Octubre de 2018). *STEM Ecuador incentiva el estudio de las ciencias en la niñez*. Obtenido de <https://www.educacionsuperior.gob.ec/stem-ecuador-incentiva-el-estudio-de-las-ciencias-en-la-ninez/>
- Silva, D. (2002). Construyendo aplicaciones web con una metodología de diseño orientada a objetos. 21.

- std, I. (2008). Especificacion de Requisitos segun el estandar de IEEE 830. *IEEE*, 9.
- Vergara, E. (13 de Febrero de 2013). *El Diario de Atacama*. Obtenido de <http://www.diarioatacama.cl/>
- Wing, J. (2006). Computational Thinking. *Communications of the acm*, 33.
- Wing, J. (2014). Computational Thinking Benefits Society. *Social Issues in Computing*, 5.
- Wing, J. (27 de Enero de 2018). *Social Issues in Computing*. Obtenido de <http://socialissues.cs.toronto.edu/2014/01/computational-thinking/>
- Yakman, G. (2012). Recognizing the A in STEM education. *Middle Ground*, 2.
- Yakman, G. (11 de Julio de 2017). *STEAM Education Professional Development Practicum & Research*. Obtenido de [https://d1wqtxts1xzle7.cloudfront.net/53836271/STEAM\\_Education\\_Professional\\_Development\\_Practicum\\_\\_\\_Research.pdf?1499856381=&response-content-disposition=inline%3B+filename%3DSTEAM\\_Education\\_Professional\\_Development.pdf&Expires=1599416439&Signature=VtAyLL](https://d1wqtxts1xzle7.cloudfront.net/53836271/STEAM_Education_Professional_Development_Practicum___Research.pdf?1499856381=&response-content-disposition=inline%3B+filename%3DSTEAM_Education_Professional_Development.pdf&Expires=1599416439&Signature=VtAyLL)
- Yee-King, M., Grierson, M., & d'Inverno, M. (2017). STEAM WORK: Student coders experiment more and experimenters gain higher grades. *IEEE Global Engineering Education Conference (EDUCON)*.
- Zapata, M. (2015). Pensamiento computacional: Una nueva alfabetización digital. *RED*, 47.
- Mujer Peruana. (2019). Educación Tradicional versus Educación STEAM, 5 diciembre 2019 [Tabla. Recuperado de : Mujer Peruana

## Anexos