

ESTRUCTURA DE MATERIALES DIDÁCTICOS PARA LA EDUCACIÓN VIRTUAL QUE FACILITEN LA COMPRENSIÓN DE CONTENIDOS TEMÁTICOS

R. Mora Reyes¹
N. Rodríguez Ventura²
L. M. García Martínez³

RESUMEN

Quintero (2015), dice: “Finalmente, es importante resaltar que con la Educación a Distancia, el Tecnológico Nacional de México, contribuye a la reducción de la brecha digital, a fortalecer la inclusión, la equidad, la diversidad; y a llevar a la educación superior tecnológica de México a ámbitos internacionales, desde un escenario nacional a uno mundial.” El alumno que inicia sus estudios en modalidad virtual, se enfrenta a retos, como: el uso de nuevas tecnologías, ser autodidacta, organizar su tiempo (entre trabajo, familia y estudio); por lo que, en muchas ocasiones, se les dificulta este paso y terminan por desertar de la carrera. Los profesionistas encargados de desarrollar los cursos virtuales deben de adoptar procesos estandarizados para el desarrollo de los objetos de aprendizaje para asegurar su efectividad, mediante trabajo en conjunto con especialistas de diferentes áreas. El Instituto Tecnológico Superior de Teziutlán consiente de esta situación, se ha preocupado por adoptar modelos de desarrollo que permitan elevar la calidad del proceso de enseñanza aprendizaje. Por lo que se presenta la experiencia obtenida al utilizar el modelo de desarrollo instruccional en la realización de los recursos virtuales de la carrera de Ingeniería en Sistemas Computacionales modalidad Virtual, que permitió disminuir el índice de reprobación en materias de razonamiento lógico, como Fundamentos de Programación, y por consiguiente bajar el índice de deserción.

ANTECEDENTES

El entorno global demanda formar profesionistas para desempeñarse por competencias de manera dinámica en diferentes contextos; los paradigmas deben permitir flexibilidad en el proceso de enseñanza-aprendizaje para estudiantes que no pueden asistir al aula. El Mtro. Manuel Quintero (2015), en su presentación del Modelo de educación a distancia del Tecnológico Nacional de México (TecNM), menciona que: “Ante los nuevos escenarios de la sociedad del conocimiento que traen consigo cambios radicales en todas las estructuras e instituciones, en el TecNM se tiene claro que es necesario ofrecer a la sociedad, alternativas de educación superior que permitan combinar la formación profesional con otras actividades, para atender a las generaciones de nativos digitales y a quienes por diferentes razones no pueden acudir a la educación escolarizada”.

El Instituto Tecnológico Superior de Teziutlán (ITST) preocupado por este reto, apertura en el 2014 la carrera de Ingeniería en Sistemas Computacionales (ISC) en modalidad Virtual; para lo cual cada docente asignado a las materias a impartir, desarrolla sus propios Objetos de Aprendizaje (OA), mediante diversas herramientas de elaboración de recursos virtuales.

Los objetos de aprendizaje se definen como cualquier entidad, digital o no digital, que puede ser utilizada, reutilizada o referenciada durante el aprendizaje apoyado en la tecnología... Como ejemplos de Objetos de Aprendizaje se incluyen los contenidos multimedia, el contenido instruccional, los objetivos de aprendizaje, el

¹ Profesor Asociado de tiempo completo B. Instituto Tecnológico Superior de Teziutlán. raulmorareyes@yahoo.com.mx

² Profesora Asociada de tiempo completo B. Instituto Tecnológico Superior de Teziutlán. naty075@yahoo.com.mx

³ Profesor Asociado tiempo completo A. Instituto Tecnológico Superior de Teziutlán. luismanuel.garcia@live.com.mx

software instruccional y las herramientas de software, así como a las personas, organizaciones o eventos referenciados durante el aprendizaje apoyado por la tecnología (IEEE, 2009).

Una vez estructurados los cursos y montados en la plataforma institucional (Moodle 2.7), se da seguimiento durante el primer semestre de la modalidad virtual (Agosto-Diciembre 2014). Para evaluar la eficacia del curso se realizaron encuestas a los estudiantes, y principalmente exhortaron que les fue difícil comprender temas de materias prácticas; debido a que las lecturas y ejercicios solicitados no les permitieron comprender los contenidos y posteriormente llevarlos a su aplicación. Lo anterior crea frustración y posteriormente abandono de los estudios, elevando índices de deserción y reprobación en modalidad virtual. Surge la necesidad de crear materiales de fácil comprensión, que no requiera de mucho tiempo para su entendimiento y aplicación. Sin ser los mismos utilizados en modalidad presencial para despertar interés y por consiguiente la obtención de aprendizaje significativo. Por lo que surge la pregunta: ¿La realización de objetos de aprendizaje mediante una metodología formal permitirá disminuir los índices de reprobación y deserción en la modalidad virtual de la carrera de Ingeniería en Sistemas Computacionales?

Se busca como objetivo, por lo tanto, desarrollar la estructura de materiales didácticos para la educación virtual con la finalidad de facilitar la comprensión de los contenidos temáticos. Considerando aspectos como: naturaleza del estudiante, perfil, rango de edades, hábitos de estudio, el estado actual y necesidades (Gros, 2012). Se consideró el uso de un modelo de diseño de objetos de aprendizaje, trabajando con la materia de Fundamentos de Programación donde se manejan conceptos abstractos y que deben ser aplicados en la resolución de problemas factibles a ser computarizados.

METODOLOGÍA

Para el diseño de los objetos de aprendizaje se utilizó el modelo de diseño ADDIE (Análisis, Diseño, Desarrollo, Implementación y Evaluación); debido a que el modelo tiene los pasos esenciales para cualquier diseño instruccional. En la Figura 1 se muestra el diagrama del modelo ADDIE.

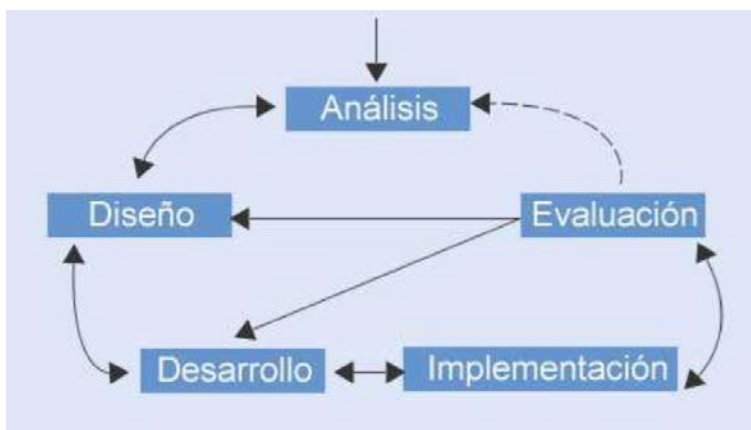


Figura 1. Modelo de diseño instruccional ADDIE (extraído de Sangrà 2005)

Se tomaron como documentos de inicio, en la fase de análisis, los planes de estudio; y mediante la conformación de cuerpos colegiados por áreas del conocimiento se consensó sobre la creación de la guía didáctica y el guion técnico, de las materias: Fundamentos de Programación, del área de conocimiento: Programación Básica; teniendo en mente la creación de recursos educativos para ambientes virtuales de aprendizaje que sean dinámicos, flexibles, atractivos e impactantes en todos los aspectos.

Fase de análisis

En esta fase se formula la estrategia de trabajo, efectuando los siguientes pasos:

- 1) Determinación de las áreas de conocimiento para lo cual se analizó la retícula de la carrera de ISC; carrera que se oferta en modalidad virtual en el ITST. Se decide trabajar con el área de Programación Básica, la cual incluye las materias de: Fundamentos de Programación, Programación Orientada a Objetos y Estructura de Datos; siendo estas las que presentan mayor dificultad, a los alumnos, en la comprensión de conceptos y aplicación de los mismo.
- 2) Se conforma el cuerpo colegiado para el área de conocimiento elegida, por tres docentes del área de programación y una pedagoga bajo la coordinación de la jefa de División de la carrera de Ingeniería en Sistemas Computacionales.
- 3) Se determinan las herramientas de infraestructura e infoestructura con la que se cuenta en el instituto, las cuales consisten en: equipo de cómputo personal, equipo de grabación de audio (micrófono y audífonos), cámara fotográfica y videocámara; conexión a internet, manuales de práctica y antologías de la materia.

Se identifican las competencias genéricas y específicas que deben aportar cada una de las materias que conforman el área de conocimiento Programación Básica; considerando las características del estudiante a atender, modalidad virtual. Tomando en cuenta los tipos de contenidos: conceptuales, procedimentales y actitudinales; para determinar las estrategias de aprendizaje y evaluación. El resultado de dicho análisis, con respecto a la materia de Fundamentos de Programación, se muestra a continuación en la Tabla 1.

Tabla 1. Análisis de la materia Fundamentos de Programación

Nombre	Fundamentos de programación
Aportación al perfil de egreso	Desarrollar el pensamiento lógico. Manejar el proceso de creación de un programa. Desarrollar algoritmos para resolver problemas.
Intención didáctica	Conceptos esenciales del diseño algorítmico: <ul style="list-style-type: none"> • Conceptos básicos de programación • Resolución de problemas utilizando el razonamiento lógico • Aplicar herramientas para diseñar e implementar soluciones en un lenguaje de programación • Manejo de estructuras de control y bloques de código reutilizables

	<ul style="list-style-type: none"> • Implementar arreglos.
Competencias generales y previas	<ul style="list-style-type: none"> • Habilidad para el uso de Internet y su principales herramientas • Manejo de software de ofimática • Razonamiento lógico básico
Diagnóstico de habilidades	Las habilidades con las que debe contar el alumno que cursa la materia son: autoaprendizaje, comunicación, uso de herramientas tecnológicas generales y especializadas.

Antes de iniciar la etapa de diseño se decide trabajar con la materia de Fundamentos de Programación, para poner a prueba los objetos de aprendizaje y recibir retroalimentación por parte de los alumnos; lo que permitirá mejorarlos y continuar con el resto de las materias.

Fase de diseño

El diseño de los objetos de aprendizaje está íntimamente ligado al enfoque didáctico que se desea lograr, es por eso que se trabaja en conjunto, el docente especialista en la materia Fundamentos de Programación y el diseñador instruccional; determinando las estrategias que faciliten el aprendizaje y permitan conocer el nivel de desempeño del estudiante. Las actividades se diseñan de acuerdo a la finalidad de alcanzar las competencias planeadas. En esta fase se investigan las diferentes herramientas de diseño, tanto de actividades multimedia que permitan acercar el conocimiento al alumno, como aquellas que refuercen los conceptos adquiridos mediante la resolución de ejercicios. Para cada actividad de enseñanza y de aprendizaje se describen los elementos, como se muestra en la Tabla 2 a continuación.

Tabla 2. Elemento que describen una actividad

Tema	4. Control de flujo
Subtema	4.1. Estructuras secuenciales
Competencia específica del tema	Construye programas utilizando estructuras secuenciales.
Competencias a desarrollar	<ul style="list-style-type: none"> • Capacidad de pensamiento lógico, algorítmico y analítico. • Resolución de problemas. • Capacidad de aplicar los conocimientos en la práctica.
Contenido del tema	<ul style="list-style-type: none"> • Operadores aritméticos (+, -, *, /) • Estructuras de asignación (=) • Paréntesis • Expresiones • Reglas de precedencia de operaciones
Actividades de enseñanza	Video-presentación con los conceptos del tema Antología con audio integrado y actividades de reforzamiento
Actividades de aprendizaje	Actividad de EducaPlay sobre la evaluación de expresiones Actividad de EducaPlay sobre la precedencias de operaciones

	Realización de un programa que utiliza estructuras secuenciales para la solución de un problema simple.
Programación de actividades	Revisión de video-presentación (1 hora) Actividades de EducaPlay (1 hora) Análisis y diseño del programa (1 horas) Codificación de la solución (2 horas)

También se diseñaron: etiquetas del curso (presentación del curso y por unidad), las secciones que conforman la unidad (Aprende, Aplica, Comunica), presentación del curso (incluye competencias, unidades temáticas, presentación del docente), y los documentos que incluyen: las instrucciones, rúbricas y ejemplos de las actividades.

Fase de desarrollo


Para llevar a cabo los diseños se utilizó dos equipos Mac Book Pro y los programas: Keynote, iMovie, GargeBand, EducaPlay, Page, Jing, entre otros. Se uniformó el diseño de etiquetas, documentos, presentaciones y actividades para no crear confusión en el alumno entre las diferentes unidades temáticas; permitiéndole así ubicar rápidamente las diferentes secciones tanto en el curso, como en los documentos que se utilizaron como guía para la realización de las actividades. En la Figura 2 se muestra la estructura de una actividad dentro de la sección Aprende y Aplica, la cual consiste en: etiqueta, instrucciones, presentación y liga donde el alumno sube la actividad.



Figura 2. Estructura de las actividades

En la Figura 3 se muestra un ejemplo de la estructura del documento de instrucciones el cual incluye: etiqueta de la unidad, tabla con número de actividad, intención didáctica, porcentaje de calificación, tiempo estimado, instrucciones, ejemplo y rúbrica de evaluación (la cual coincide con la utilizada para evaluarla en plataforma).

I.T.S.T. Fundamentos de Programación



Unidad IV
Control de flujo.

Competencia específica:
Construye programas utilizando estructuras condicionales y repetitivas para aumentar su funcionalidad.


M.S.C. Raul Mora Reyes

Actividad 1	Utiliza estructuras selectivas y funciones para resolver problemas.	
Porcentaje de Unidad 50%	Tiempo estimado 2 horas 30 minutos	

Instrucciones:

- Efectúa la lectura: **"Antología: Unidad 4"**, realiza los ejercicios que acompañan a algunos de los puntos, lo que te permitirá comprobar la comprensión de los mismos. Estos ejercicios debes realizarlos para que se registre su cumplimiento, **no** importa la calificación que obtengas o en que intento logres realizarla (cuentas con 10 intentos). El objetivo es que practiques lo aprendido.
- Observa la presentación **"Estructuras de Selección"**.
- Aplica al ejercicio, que encuentras después de las instrucciones, la metodología de las 6De y documenta cada una de ellas como se muestra en la presentación.
- Convierte el archivo a formato PDF y súbelo en la liga **"Ejercicio: Hospital"**.

Ejercicio:



En un hospital la cuenta de un paciente se calcula de la siguiente manera:

- Por cada día de estancia se le cobra un costo diario. El costo diario depende de la habitación: \$1200 privada (P) y \$500 sala general (G).
- Por cada estudio se le cobra \$450 si fueron menos de 5, en otro caso \$300.

M.S.C. Raul Mora Reyes 1


I.T.S.T. Fundamentos de Programación

Concepto	Sobresaliente	Notable	Aprobado
Fecha en que realiza la actividad	Realiza la actividad do la quinta a la séptima semana del curso. 20 puntos	Realiza la actividad en la octava semana del curso. 10 puntos	Realiza la actividad en la novena semana del curso. 5 puntos
Definición de la solución	Establece correcta y claramente los datos de entrada, salida y procesos. 20 puntos	Establece correcta y claramente los datos de entrada y salida, pero no los procesos. 10 puntos	No establece correcta y claramente los datos de entrada, salida y procesos. 5 puntos
Diseño de la lógica Diagrama UML	Utiliza correctamente el diagrama de clase UML e incluye adecuadamente los campos y métodos . 10 puntos	No utiliza correctamente el diagrama de clase UML e incluye adecuadamente los campos y métodos . 5 puntos	No utiliza correctamente el diagrama de clase UML y no incluye adecuadamente los campos y métodos . 0 puntos
Diseño de la lógica Diagramas de Flujo	Realiza correctamente todos los diagramas de flujo de los métodos que utilizan instrucciones selectivas . 20 puntos	Realiza correctamente el 60% de los diagramas de flujo de los métodos que utilizan instrucciones selectivas . 10 puntos	Realiza correctamente el 30% de los diagramas de flujo de los métodos que utilizan instrucciones selectivas . 5 puntos
Desarrollo de la codificación	El código de la clase propuesta y la principal están libres de errores . 20 puntos	El código de la clase propuesta está libre de errores , pero no la principal. 10 puntos	El código de la clase propuesta y la principal presentan errores . 5 puntos
Depuración y pruebas	Se incluyen todas las pantallas de ejecución. 10 puntos	Se incluyen algunas pantallas de ejecución. 5 puntos	Se omiten pantallas de ejecución. 0 puntos

M.S.C. Raul Mora Reyes 3

Figura 3. Ejemplo del documento de instrucciones de las actividades

En la Figura 4 se muestran ejemplos de los recursos digitales que se realizaron; el de la izquierda muestra la antología donde se puede navegar a través de los temas mediante el árbol de contenidos, cada tema cuenta con texto y audio del mismo. A la derecha una actividad EducaPlay que puede estar incrustada en la antología o como actividades independientes que refuerzan lo aprendido.



La Juventud de Hoy, Tecnología del Mañana.

Unidad 1

Presentación

Subtemas:

- 1.1. Clasificación del software
- 1.2. Algoritmo
- 1.3. Lenguaje de programación
- 1.4. Programa
- 1.5. Programación
- 1.6. Paradigmas de programación
- 1.7. Editores de texto
- 1.8. Compiladores e intérpretes
- 1.9. Ejecutables
- 1.10. Consola de línea de comandos

M.S.C. Raul Mora Reyes

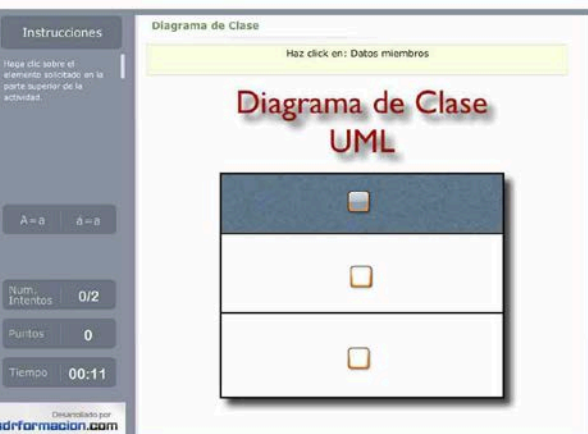
Instrucciones

Haga clic sobre el elemento solitado en la parte superior de la actividad.

Diagrama de Clase

Haz click en: Datos miembros

Diagrama de Clase UML



Num. Intentos: 0/2

Puntos: 0

Tiempo: 00:11

Desarrollado por adrfarmacion.com

Figura 4. Ejemplos de recursos digitales

Fase de implementación

Una vez configurado el curso en la plataforma institucional a distancia del instituto (Moodle 2.7), se procede a subir cada uno de los elementos diseñados en la etapa anterior: etiquetas, objetos de aprendizaje, documentos, presentaciones y actividades; se crean las categorías e ítems, se establecen las escalas de calificación, al igual que las rúbricas correspondientes para cada actividad. Se imparte el curso a los alumnos del primer semestre de la segunda generación, en el periodo Agosto-Diciembre 2015.

Fase de evaluación

Durante el semestre se les pidió a los alumnos que evaluaran los recursos didácticos de cada unidad, mediante pequeños cuestionarios y entrevistas en línea. Esto permitió conocer la efectividad de los OA. También se utilizaron en el curso de repetición, con los alumnos de la primera generación, los cuales pudieron dar una comparación entre los materiales nuevos y los anteriores.

DISCUSIÓN DE RESULTADOS

De acuerdo a las encuestas y entrevistas en línea realizadas a los alumnos que cursaron el primer semestre de la segunda generación (Agosto-Diciembre 2015) y los alumnos de la primera generación que repitieron el curso de Fundamentos de Programación, se pudo detectar que los recursos virtuales utilizados les permitieron comprender fácilmente los contenidos temáticos, de acuerdo a sus observaciones y a los resultados obtenidos en su ejecución; ya que les permitió realizar las actividades en los tiempos programados obteniendo puntajes altos.

Con respecto al índice de reprobación de la materia de Fundamentos de Programación, comparando la primera y segunda generación se disminuyó en un 62%; en la primera generación el índice de reprobación fue del 58% y en la segunda generación solo del 36%.

En lo que se refiere a deserción de la carrera de ISC, el índice disminuyó en un 43%; en la primera generación se tuvo un 47% y en la segunda generación solo el 27%.

CONCLUSIONES Y/O RECOMENDACIONES

La utilización de una metodología formal para el desarrollo de Objetos de Aprendizaje en nuestro caso el modelo de diseño ADDIE (Análisis, Diseño, Desarrollo, Implementación y Evaluación), permite entre otras cosas:

- Tener un mayor control y organización de los recursos virtuales disponibles para los cursos; ya que por cada recurso se debe especificar su descripción por escrito. Lo que permite tener actualizado el repositorio de elementos virtuales.
- Unificar los formatos de todos los elementos, desde etiquetas y presentaciones hasta instrucciones y rúbricas de las actividades, utilizados en los cursos; lo que beneficia al alumno al disminuir la curva de aprendizaje en el manejo de la plataforma y los OA.
- Participación de docentes especializados en diferentes áreas: pedagógica, diseño, uso de Tecnología de la Información y la Comunicación y, aquellos especialistas en el área del conocimiento; mediante el trabajo en equipo se logra, en consenso, la elaboración de materiales pensados de acuerdo a su objetivo, tomando en cuenta los

tipos de contenidos: conceptuales, procedimentales y actitudinales; y las estrategias de enseñanza, aprendizaje y evaluación.

Con la experiencia obtenida y la retroalimentación dada por los alumnos surgen nuevos retos por alcanzar y que se recomienda trabajar ellos al desarrollar los nuevos recursos virtuales:

- Capacitar a todo el personal involucrado en la modalidad virtual, sobre el uso del modelo de desarrollo, las herramientas de diseño y las estructuras de curso y documentos.
- Mejorar los recursos virtuales realizados al momento de acuerdo a las sugerencias que los alumnos hicieron.
- Crear redes de colaboración entre diferentes institutos que ofrezcan la modalidad virtual, para enriquecer mediante la experiencia, el proceso de desarrollo y los recursos virtuales.

BIBLIOGRAFÍA

- Gros, B. (Ed). (2011). *Evolución y retos de la educación virtual. Construyendo el e-learning del siglo XXI*. Barcelona. Editorial UOC.
- IEEE. (2009). *Estándar para Metadatos de Objetos Educativos*. Obtenida el 10 de junio del 2016, de: <https://standards.ieee.org/findstds/standard/1484.12.1-2002.html>
- Quintero, M. (2015). *Carta de presentación del Modelo de educación a distancia del Tecnológico Nacional de México*. México. TecNM.
- Sangrà, A. (2005). *Los materiales de aprendizaje en contextos educativos virtuales. Pautas para el diseño tecnopedagógico*. Barcelona: Editorial UOC.
- Tecnológico Nacional de México. (2015). *Modelo de educación a distancia del Tecnológico Nacional de México*. México. TecNM.