



Revista Iberoamericana de Tecnología en Educación y
Educación en Tecnología

ISSN: 1851-0086

ISSN: 1850-9959

editor-teyet@lidi.info.unlp.edu.ar

Universidad Nacional de La Plata
Argentina

Bertossi, Valeria Iliana; Gutiérrez, María de los Milagros
Prácticas ágiles en el desarrollo de objetos de aprendizaje: estado del arte
Revista Iberoamericana de Tecnología en Educación y
Educación en Tecnología, núm. 31, Esp., 2022, pp. 121-132
Universidad Nacional de La Plata
Argentina

DOI: <https://doi.org/10.24215/18509959.31.e12>

- Número completo
- Más información del artículo
- Página de la revista en redalyc.org



Prácticas ágiles en el desarrollo de objetos de aprendizaje: estado del arte

Agile practices in LO development: state of the art

Valeria Iliana Bertossi¹, María de los Milagros Gutiérrez¹

¹ Universidad Tecnológica Nacional Facultad Regional Santa Fe – Centro de Investigación y Desarrollo en Ingeniería en Sistemas (CIDISI), Argentina

vbertossi@fsf.utn.edu.ar, mgutier@fsf.utn.edu.ar

Recibido: 21/04/2021 | Corregido: 02/08/2021 | Aceptado: 08/09/2021

Cita sugerida: V. I. Bertossi, M. de los M. Gutiérrez, “Prácticas ágiles en el desarrollo de objetos de aprendizaje: estado del arte,” *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, no. 31, pp. 121-132, 2022. doi: 10.24215/18509959.31.e12

Esta obra se distribuye bajo **Licencia Creative Commons CC-BY-NC 4.0**

Resumen

Las prácticas ágiles surgieron como alternativa a las metodologías convencionales de desarrollo de software para lograr resultados más rápidos sin disminuir su calidad, otorgándole centralidad al factor humano que participa en el proceso y a los incrementos del producto en iteraciones muy cortas como fundamento para la generación de valor. Los objetos de aprendizaje, en cuanto productos de software cuya concepción y construcción demanda esfuerzo intelectual y creativo, y cuyo destino final se enmarca en el ámbito del conocimiento, son pasibles de capitalizar las ventajas de la agilidad. En este trabajo se presenta una revisión sistemática de la bibliografía referida a metodologías de desarrollo de objetos de aprendizaje con el propósito de detectar si están concebidas bajo el modelo de gestión ágil.

Palabras clave: Metodología de desarrollo de objetos de aprendizaje; Objeto de aprendizaje; Agilidad; Prácticas ágiles; Metodologías tradicionales.

Abstract

Agile practices emerged as an alternative to conventional software development methodologies to achieve faster results without reducing their quality, giving centrality to the human factor that participates in the process and to product increases in very short iterations as a basis to generate value. Learning objects, as software products whose conception and construction demand intellectual and creative effort, and whose final purpose is framed in the field of knowledge, are capable of capitalizing on the advantages of agile approach. This paper presents a systematic review of the bibliography referring to learning objects development methodologies in order to detect whether they are conceived under the agile management model.

Keywords: Learning object development methodology; Learning object; Agile framework; Agile practices; Traditional methodologies.

1. Introducción

La agilidad remonta sus comienzos a los años 90, donde empieza a experimentarse una nueva manera de trabajar basada en la comunicación horizontal, con talleres de reflexión y desarrollo de software incremental; aunque el puntapié inicial ocurre en 2001 con la firma del Manifiesto por el desarrollo ágil de software [1], que proclama estos cuatro valores: (i) los individuos e interacciones por encima de los procesos y herramientas, (ii) software funcionando por encima de la documentación exhaustiva, (iii) la colaboración con el cliente por encima de la negociación contractual, (iv) la respuesta al cambio por encima del seguimiento de un plan.

Por aquellos años también comienza a delinearse un nuevo modelo de gestión, adoptado por la agilidad, que cambió el foco de atención de los procesos a las personas. Se basa en la premisa que el conocimiento está en las personas que hacen la tarea y la toma de decisiones no debe pasar por alto esa realidad. Este modelo, consolidado bajo el nombre Management 3.0 [2] [3] [4], asume que la productividad de una persona está condicionada no sólo por la actividad laboral sino también por su situación personal. Por ello, se aplican técnicas que permiten a los equipos hacer una pausa en el trabajo para reflexionar cómo se lo está realizando, cómo mejorar las relaciones interpersonales y también para detectar la motivación de las personas con el objetivo, en primer lugar, de ayudarlas a exteriorizar sus valores y hacer uso de sus habilidades emocionales para integrarse al equipo, para conocerse entre sí y comprender actitudes de los otros sin juzgar; y, en segundo lugar, de posibilitar la definición de planes de acción que contribuyan a cumplir esas motivaciones y solucionar conflictos. La gestión es una responsabilidad compartida por el equipo; lo importante son las personas y la estructura de la organización se crea a partir de ellas, dotándolas de recursos para comunicarse y colaborar. De este modo, se genera una cultura de trabajo sinérgico que conduce a la satisfacción de las personas (miembros del equipo, de la organización y los clientes), que propicia el compromiso, el surgimiento de ideas superadoras, la aceptación del cambio y los errores como oportunidad de mejora, ganando en agilidad y creación continua de valor.

Según [5] los tres indicios a tener en cuenta para decidirse por un modelo de gestión ágil son:

- 1) Inestabilidad de las necesidades. Cuanto más inestables son las necesidades/funcionalidades del producto de software, más ágil se debe ser. El lema es ¡bienvenido el cambio, el cambio no es un problema!
- 2) Cuán costoso resulta hacer un cambio en el producto. Si es altamente costoso modificarlo se debería optar por un modelo tradicional. En tanto que la agilidad se adecua a productos que son fáciles de modificar.
- 3) En qué medida es clave en la construcción del producto el componente humano, el trabajo en

grupo de carácter intelectual; es decir, hasta qué punto se requiere del conocimiento de todos y cada uno para producir un único producto. Cuando el producto se basa en la intelectualidad se necesita un modelo de trabajo centrado en las personas (las personas por encima del proceso).

Atendiendo a estos consejos, los objetos de aprendizaje (OA), en tanto productos de software, son factibles de desarrollarse aplicando agilidad. Aunque si bien las necesidades son relativamente estables, las prácticas ágiles bien podrían implementarse en virtud de que el producto es fácil y económico de modificar y, lo más determinante, es que el componente humano en su construcción y destino final adquiere una contundencia innegable, tanto más en el ámbito universitario en el que el intelecto es la herramienta de trabajo por excelencia: se requiere del conocimiento del experto en sistemas, del experto en la disciplina que se enseña, del experto en Pedagogía que asesora en lo referente a Teorías del Aprendizaje, del experto en Bibliotecología, de la creatividad del diseñador gráfico y, fundamentalmente, de la mirada del alumno, el usuario a quien está destinado el OA para ayudarlo a desarrollar las competencias esperadas y lograr los niveles de aprendizaje planteados en los objetivos.

En este artículo se presenta una revisión sistemática de las propuestas metodológicas para el desarrollo de OA diseñadas en los últimos veinte años por grupos de investigación, con el objetivo de detectar si en ellas se adoptan prácticas ágiles como marco de trabajo. Para ello, el resto de esta comunicación se organiza como sigue: en la sección 2 se explica la metodología empleada para la revisión de la literatura; en la sección 3 se expone el marco conceptual; en la sección 4 se detallan las principales características de las metodologías de desarrollo de OA desde la perspectiva en la que se abordó la búsqueda; finalmente, en la última sección se vierten las reflexiones sobre la exploración realizada.

2. Metodología

Este trabajo se llevó a cabo siguiendo la metodología propuesta en [5] para revisiones sistemáticas de literatura, la cual consta de cinco etapas: (i) Identificación del campo de estudio y del período a analizar; (ii) Selección de las fuentes de información; (iii) Realización de la búsqueda (qué, dónde y cómo); (iv) Gestión y depuración de los resultados de la búsqueda; (v) Análisis de los resultados.

El campo de estudio definido en la etapa (i) versa sobre metodologías de desarrollo de OA diseñadas por grupos de investigación de universidades de Iberoamérica; y el período de análisis establecido abarca los últimos 20 años. Las fuentes seleccionadas en la etapa (ii) involucran artículos de revistas, comunicaciones en congresos y tesis académicas, revistas relevantes en materia de tecnología educativa y de educación a distancia, actas de congresos, repositorios digitales abiertos de universidades, portales bibliográficos y bases de datos como ELSEVIER, Scielo, EBSCO, SCOPUS, ScienceDirect, Redalyc, Dialnet,

Latindex, DOAJ, REDIB y Crossref; se apeló también al buscador Google Académico y a las redes sociales académicas Mendeley y ResearchGate. En la etapa (iii) se realizaron búsquedas automáticas y manuales estableciendo los siguientes criterios de búsqueda: *Objeto de aprendizaje*, [sinónimo de OA] (sinónimo de OA puede ser: objeto virtual de aprendizaje, objeto digital educativo, recurso educativo digital, recurso educativo abierto, material educativo digital, material didáctico interactivo, etc.), *Metodología de desarrollo de* [sinónimo de OA], *Diseño de* [sinónimo de OA]. Durante la etapa (iv), mediante lectura rápida de título, resumen y palabras clave, el material recopilado se clasificó en tres categorías y se almacenó en sendas carpetas del sistema de archivos de la computadora: OA, No-OA y Dudosos. En la primera se agruparon las publicaciones referidas a materiales educativos que se encuadran dentro de la definición de OA adoptada en la subsección 3.2 de esta comunicación; mientras que en la segunda se colocaron los falsos positivos (divulgaciones sobre materiales educativos que no califican como OA según la definición mencionada); y en la tercera se ubicaron aquellos trabajos para los que con la lectura rápida resultó dudoso clasificarlos en OA o No-OA y requirieron de una revisión ulterior del documento completo. Posteriormente se procedió a depurar la carpeta OA mediante el uso de los términos de búsqueda *metodología*, *desarrollo* y *diseño* en el cuadro de búsqueda del explorador de archivos de Windows, con el propósito de obtener los documentos referidos exclusivamente a metodologías de desarrollo de OA y reubicarlos en la subcarpeta *Metodologías* creada para tal fin. La etapa (v) se centró en la lectura compresiva de los artículos alojados en dicha carpeta, prestando especial atención a las referencias bibliográficas como fuente eventual de trabajos pertenecientes al campo de estudio que no fueron detectados en la etapa (iii), llevando adelante un ciclo iterativo entre las etapas (iii), (iv) y (v).

3. Marco Conceptual

El marco teórico que sustenta la investigación bibliográfica presentada en este trabajo está conformado por las características distintivas de las prácticas ágiles y por la conceptualización de OA y sus propiedades.

3.1. Conceptos Clave de las Prácticas Ágiles en el Desarrollo de Software

La agilidad asume una nueva manera de trabajar, diferente a la de la visión clásica del desarrollo de software, entendiéndose por ésta el uso de procesos rigurosos orientados por la documentación exhaustiva y el cumplimiento meticuloso de todos los pasos prescriptos, que tienen sus raíces en los procesos industriales de elaboración de productos físicos [6]. La comunidad ágil prefiere la terminología *framework* ágil en lugar de *metodología* ágil para denotar que no se prescriben etapas ni artefactos, sino que se trata de una cultura de trabajo guiada por los valores del manifiesto ágil y que se caracteriza por:

- Llevar a cabo el desarrollo de productos de software en ciclos iterativos, denominados *sprints*, que duran unas pocas semanas. Esas iteraciones son incrementales, ya que a medida que transcurren se va limpiando y refinando el producto, no sólo porque se agregan funcionalidades sino porque, al obtener retroalimentación temprana y frecuente del cliente, se mejoran aquellas cuestiones que en iteraciones previas no cubrieron sus expectativas [6][7].
- Hacer foco en una muy buena gestión de los equipos, teniendo en cuenta que al software no lo fabrica una cadena de montaje, lo desarrollan personas a través de un proceso complejo que se identifica con el mundo del conocimiento, el intelecto y la creatividad [6][8].
- Gestionar los problemas de motivación de las personas porque los mismos impactan en la productividad [6][8].
- Trabajar en equipos que se distinguen por ser [6][9][10][11][12]:
 - pequeños: entre 5 ó 7 personas.
 - multifuncionales: no hay un único responsable del cual se depende, la responsabilidad es compartida por todos los miembros del equipo. Se procuran y valoran las habilidades en forma de “T”: la pata vertical de la “T” representa la especialidad (conocimiento en profundidad de un área determinada) y la pata horizontal, la amplitud (conocimiento en otras áreas). Una persona puede, por ejemplo, dedicarse a testear porque conoce mucho sobre la tarea, pero ello no le quita corresponsabilidad por el trabajo de los demás.
 - estables: todos los miembros trabajan juntos en un ritmo sostenible (no hay gente que entre y salga del equipo según el proyecto), se conocen mutuamente e intentan mejorar continuamente la forma de trabajar.
 - autoorganizados: deciden cómo realizar las tareas de la mejor manera para que en determinado período de tiempo se obtenga el máximo de trabajo completado.
- Procurar entornos óptimos para el desarrollo de software, sin ruidos ni interrupciones. Se estima que por cada interrupción una persona pierde 15 minutos en retomar su actividad con el nivel de concentración que tenía al momento de dejarla, ocasionando desperdicios que frenan la productividad [6].
- Usar un tablero de trabajo que visibiliza a todo el equipo de desarrollo el grado de avance del proyecto completo. Permite identificar claramente el trabajo por hacer, el que se está llevando a cabo en ese momento y el trabajo que ya está completado [6][7].
- Reemplazar la palabra requisito por necesidad,

debido a que la primera tiene la connotación de especificación acabada, documentada e inamovible; en tanto que la segunda sólo refleja una necesidad del usuario que debe invitar a sucesivas conversaciones con él para ir entendiendo qué es lo que realmente quiere. Las necesidades se capturan a través de la herramienta historias de usuario que responde a la estructura “como [rol] quiero [necesidad] para [beneficio o valor]” y debe ser escrita sólo en una tarjeta que se ubicará en el tablero de trabajo con la intención de no dejar su interpretación cerrada, sino que sirva como disparador para recordar lo que se habló con el cliente y, en caso de necesitarlo, volverlo a discutir [6][11][12][13][14].

- Trabajar en un flujo continuo de tareas. No hay una división en las clásicas etapas secuenciales del ciclo de vida en cascada, sino que en cada sprint las tareas (análisis, diseño concurrentemente, tanto como sea posible, para obtener un incremento en el que, a partir de una necesidad de negocio priorizada, se diseña l, desarrollo y pruebas) se llevan a cabo casi a solución, se la programa y pasa las pruebas, obteniendo así software potencialmente desplegable [6][7].
- Adoptar el concepto de working software como valor entregado al cliente, esto es, software funcionando que le permite obtener ventaja competitiva [6].
- Adoptar el concepto de velocidad como trabajo completado por unidad de tiempo, es decir, valor entregado al cliente por sprint [6][7][12].
- Aceptar los cambios y ser flexibles para mejorar a partir de ellos [6][7][15].
- Primar la comunicación cara a cara. No hay una exigencia de entregables que pasan de un integrante del equipo a otro, a menos que la singularidad de la tarea y el producto lo ameriten [6][13][14].
- Realizar reuniones periódicas para analizar cómo se está trabajando y encontrar caminos de mejora [6][8] [11][12].
- Tener presente que la buena calidad del software (también llamada deuda técnica o excelencia técnica) es un continuo a lo largo de todo el desarrollo del producto; no implica un hito único de evaluación de calidad cuando el mismo ya está terminado [6].
- Basar en la confianza la relación con el cliente, involucrándolo y contando con su colaboración en el desarrollo del producto de software [6][16].
- Priorizar las necesidades junto al cliente para optar por darles solución en los sucesivos sprints a las que más valor a su negocio aportan [6].

La tabla 1 resume un análisis comparativo de los aspectos más relevantes en los que se diferencian las metodologías tradicionales y la agilidad, elaborado por [17].

Tabla 1. Comparación entre agilidad y metodologías tradicionales. Fuente: [17].

Metodologías Ágiles	Metodologías Tradicionales
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el grupo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos

3.2. Conceptos Clave de los OA

En [18] se adopta la siguiente definición de OA: “Los objetos de aprendizaje son un tipo de material educativo, abierto y digital, compuesto por una estructura interna y otra externa. La primera está conformada por un objetivo de aprendizaje, un contenido alineado al objetivo, un conjunto de actividades para aprender el contenido y un instrumento de evaluación que mide el logro del objetivo planteado; la segunda, por un conjunto de metadatos que facilitan su almacenamiento, búsqueda y recuperación en repositorios de la Web, con el objetivo de reutilizarlos en cualquier plataforma de software y en una diversidad de situaciones pedagógicas.”

Esta conceptualización reúne la dimensión tecnológica y pedagógica de los OA. Objetivo, contenido, actividades y evaluación abonan la dimensión pedagógica; mientras que los metadatos, estándares para su anotación, estándares de empaquetamiento y repositorios refieren a la segunda.

Atendiendo a estas dos dimensiones, diversos autores han caracterizado a los OA a través de una serie de propiedades que, en su conjunto, los distinguen de cualquier otro recurso educativo:

- 1) Reutilizable: el OA puede emplearse dentro de distintos sistemas de gestión del aprendizaje o entornos virtuales, en una variedad de situaciones pedagógicas (otra institución, otra asignatura, otro grupo de alumnos con nivel de conocimientos previos diferente, etc.) y con diferentes propósitos (como disparador de un tema, para desarrollar un concepto, para afianzarlo o repasar, para entrenamiento, etc.) para los cuales estaba originalmente destinado cuando se lo creó [19] [20].
- 2) Interoperable: un sistema de empaquetamiento normalizado garantiza que el OA funcione en diferentes plataformas de software sin problemas de compatibilidad [21][22][23][24].
- 3) Accesible: gracias al uso de metadatos, es posible acceder a los OA desde una ubicación remota [25].
- 4) Durable: el OA no requerirá de rediseño ni recodificación aunque cambie la tecnología base [25], además debe contar con información vigente [19] o, al menos, recomendaciones sobre su actualización.
- 5) Autocontenido: también llamado “autónomo” o “independiente” [26][27], significa que el OA contiene

todo lo necesario para contribuir al logro del objetivo educativo para el cual fue pensado [20], posibilitando experiencias de aprendizaje íntegras.

6) **Adaptable:** también denominado “personalizable” por [19][27], alude a que el OA es fácilmente modificable para adaptarse a necesidades individuales singulares (estilo personal de aprendizaje) y situacionales (grupo de alumnos, programa formativo, institución) [19][25].

7) **Generativo:** que pueda ensamblarse con otros OA para construir colecciones de mayor tamaño como unidades, lecciones, módulos [23][28]. Esta propiedad es denominada “ensamblable” por [21].

8) **Localizable:** el OA debe tener un adecuado esquema descriptivo de metadatos [23][25] para que sea fácilmente encontrado utilizando términos de búsqueda simples y comprensibles. El autor [29] lo llama “descubrible”.

9) **Recuperable:** significa que el OA pueda obtenerse del repositorio que se desea y en el momento que se lo precisa [25], para lo cual es necesario el uso de metadatos. Esta propiedad es considerada por [27] como parte de la descripción de “accesibilidad”.

10) **Granularidad:** esta propiedad responde a preguntas tales como “¿qué tanto contenido?, ¿qué tan profundo?,

¿qué tan extenso [debe ser el OA]?” [20]. Define la “atomicidad” que debe tener un OA. [19].

11) **Asequible:** alude a que el OA permita mejorar la eficacia del aprendizaje a menor tiempo y costo [25].

12) **Escalable:** el OA debe poder extenderse a grandes audiencias sin un aumento proporcional en el costo [22].

13) **Evaluable:** susceptible de ser evaluado por docentes, alumnos, equipo de desarrollo e instituciones educativas, cada uno según rol y utilizando criterios predefinidos [25].

14) **Abierto:** debe contar con licencia de autor abierta, es decir, que no sea necesario tener que solicitar los derechos de autor para poder usar, reusar o adaptar el OA [30]. Este concepto facilita la adaptabilidad y generatividad.

15) **Digital:** propiedad que se desprende del marco de aprendizaje mediado por las tecnologías de la información y comunicaciones.

4. Metodologías de Desarrollo de OA Propuestas en la Literatura

En esta sección se describen, desde la perspectiva del proceso de desarrollo de software, las metodologías para la construcción de OA encontradas en la literatura, con el propósito de detectar si están concebidas bajo el paraguas de la agilidad.

Universidad Nacional del Litoral [31]: Identifica 7 fases para las cuales prescribe las actividades a desarrollar y los entregables: 1) Identificación de competencias, 2) Especificación, 3) Diseño de contenido, 4) Puesta visual, 5) Producción, 6) Evaluación y 7) Publicación. Desde la fase 2 a la 6 el ciclo de vida es iterativo e incremental. El equipo de desarrollo está integrado por: experto temático, diseñador didáctico instruccional, experto informático, diseñador gráfico, coordinador, administrador del repositorio, docente responsable.

OAULA (OA de la Universidad de Los Andes) [32]: Es una metodología de prototipado evolutivo en el que el proceso es controlado por una fase central de Validación que se lleva a cabo en paralelo a las siguientes fases: 1) Análisis, 2) Diseño, 3) Desarrollo, 4) Implementación, 5) Evaluación. El equipo de desarrollo está conformado por: docente experto en la disciplina, experto en diseño instruccional, experto en pedagogía, experto en psicología, diseñador gráfico, experto en herramientas tecnológicas, desarrollador Web, estudiantes.

Universidad Militar Nueva Granada [33]: Presenta un conjunto de pasos secuenciales: 1) Identificación del OA, 2) Identificación de la población objetivo, 3) Formulación del objetivo pedagógico, 4) Elaboración del contenido, 5) Definición de la estrategia pedagógica, 6) Formulación de actividades de aprendizaje, 7) Evaluación con estudiantes, 8) Desarrollo, 9) Revisiones y pruebas, 10) Implementación del OA en un curso. El docente y especialistas en tecnología y multimedia integran el equipo de desarrollo.

INTERA (INteligencia, Tecnologías Educativas y Recursos Accesibles) [34]: Está compuesta de 3 fases secuenciales: Inicial, Intermedia y de Transición. Se definen las siguientes etapas, que son iterativas y pueden pertenecer a más de una fase: 1) Contextualización, 2) Requisitos, 3) Arquitectura, 4) Desarrollo, 5) Ambiente, 6) Estándares, 7) Pruebas y calidad, 8) Liberación, 9) Evaluación, 10) Gestión de proyectos. Cada una de estas etapas tiene definidos los artefactos de entrada, las técnicas a emplear y los artefactos de salida. Los diferentes roles dentro del equipo de desarrollo son: analista, compilador de contenido, gerente de proyectos, solicitante, diseñador de interfaces, diseñador instruccional, equipo de desarrollo y equipo de prueba.

DICREVOA (Diseño, CReación y EValuación de OA) [35]: Está destinada a docentes que no disponen en sus instituciones de un equipo multidisciplinar que dé soporte al desarrollo de OA, por lo cual, es él quien realiza todo proceso y utiliza alguna herramienta de autor. Se compone de 5 fases: 1) Análisis, 2) Diseño, 3) Implementación, 4) Evaluación, 5) Publicación.

Universidad Nacional de Loja [36]: Está destinada a docentes para que utilicen una herramienta de autor. Se compone de 6 fases, cada una de las cuales con un conjunto de actividades: 1) Planificación, 2) Diseño, 3) Maquetación del OA (se puede retornar a la fase de diseño para hacer reajustes), 4) Empaquetamiento del OA, 5)

Integrar el OA al LMS/LCMS, 6) Evaluación (retroalimenta la fase de Diseño).

UBoa (Universidad de Bocayá – objeto de aprendizaje) [37]: Incorpora el modelo pedagógico de la Universidad de Bocayá y se compone de 5 fases cada una con sus respectivas actividades y especificación de resultados: 1) Conceptualización, 2) Diseño, 3) Producción, 4) Publicación, 5) Control de Calidad. La calidad está concebida como seguimiento y control de todas las fases. El equipo de desarrollo es interdisciplinar: docente, asesor pedagógico, diseñador instruccional, desarrolladores, evaluadores.

CROA (CReación de OA) [38]: Está pensada para que el docente cree los OA con alguna herramienta de autor, sin necesidad de un equipo de desarrollo; aunque ello no invalida que equipos multidisciplinares que producen materiales educativos hagan uso de esta metodología. Presenta 5 etapas, que si bien son secuenciales, prevé realimentación entre ellas: 1) Análisis, 2) Diseño: se lleva a cabo a través de plantillas y establece un punto de revisión de calidad de coherencia objetivo-contenido-actividades-evaluación, 3) Desarrollo, 4) Publicación (en un repositorio o LMS) y 5) Evaluación: luego de que el OA entra en producción se vuelve a valorar la coherencia objetivo-contenido-actividades-evaluación y se realiza una evaluación de la calidad en uso por parte de docentes y alumnos. La salida de cada etapa tiene previsto un entregable que constituye la documentación del OA.

MEDOA (MEtología para el Desarrollo de OA) [39]: Combina cascada y desarrollo en espiral. La metodología comienza con la fase 1) Planeación y a partir de allí se ejecutan en espiral las fases 2) Análisis, 3) Diseño, 4) Implementación y 5) Validación, donde el primer ciclo corresponde al desarrollo del contenido del OA, el segundo a las actividades y el tercero a la evaluación. Una vez que se completan los tres componentes del OA, se continúa en cascada con las fases 6) Implantación y 7) Mantenimiento. Con el objetivo de controlar el proceso todas las fases son documentadas en una Base de Datos integrada. Los usuarios de la metodología son estudiantes y profesores, con o sin experiencia en el empleo de aplicaciones computacionales.

ISDOA (Ingeniería de Software para Desarrollar OA) [40]: Usa el ciclo de vida en Cascada con Verificación y Validación (V&V), que se compone de 4 fases secuenciales y 2 fases que desarrollan paralelamente a aquéllas. Las fases secuenciales son: 1) Análisis e ingeniería de requisitos, 2) Diseño, 3) Desarrollo e implementación, 4) Evaluación de vida útil. Las fases paralelas son: 1) Plan de pruebas, 2) Evaluación de calidad. La metodología prescribe las actividades y entregables de cada fase. No describe el equipo de desarrollo pero se infiere que está formado por el docente especialista en la disciplina y desarrolladores de software.

MESOVA (MEtología de desarrollo de Software para Objetos Virtuales de Aprendizaje) [41]: El autor declara recoger algunos elementos del *framework* Programación Extrema (PE), Proceso Unificado Racional (RUP, por su

sigla del inglés Rational Unified Process) y modelo evolutivo en espiral. Pero de PE adopta sólo la declaración de un conjunto de principios de desarrollo aplicados a OA y del modelo en espiral toma sólo iteraciones en la fase de diseño, porque, en definitiva, está estructurada en 6 fases secuenciales, cada una con sus respectivas actividades y especificación de resultados: 1) Concepción del objeto, 2) Diseño y desarrollo modular evolutivo, 3) Integración y despliegue, 4) Pruebas de aprendizaje, 5) Consolidación. La calidad se gestiona estableciendo puntos de control al finalizar cada fase. Exige la documentación del proceso, la arquitectura del OA y el manual de usuario. El equipo de desarrollo está conformado por: líder de proyecto, docente, pedagogo, técnicos especializados en software y repositorios de OA.

MPOBA (Modelo de Proceso para el desarrollo de OA) [42]: Es una metodología de prototipado evolutivo, comparte el valor de la filosofía ágil de trabajar junto al cliente al emplear el enfoque del diseño centrado en el usuario, implicando su participación desde etapas tempranas del proceso de desarrollo. Pero los artefactos generados en cada iteración son sólo prototipos, no hay una entrega al usuario de producto funcionando. Una de las técnicas empleadas para la construcción de prototipos es la de Escenarios, que constituyen descripciones parciales del funcionamiento del sistema en un momento específico de la aplicación. Esta técnica se asemeja, sin serlo, a la de historias de usuarios. Recién luego de realizadas todas las iteraciones de prototipado, el OA se programa con alguna herramienta de autor y se entrega completo para ser evaluado. En caso de necesitar modificaciones o mejoras como resultado de dicha evaluación, el OA debe retomar nuevamente al proceso de desarrollo. El equipo de desarrollo es multidisciplinar, aunque no se especifican los roles.

Tecnopedagógica [43]: Integra el conocimiento de las áreas de Educación, Ingeniería de Software e Interacción Humano-Computador. Se desarrolla en 7 pasos: 1) Diseño Instruccional, 2) Modelado de las funcionalidades con UML, 3) Modelado de la interfaz, 4) Selección de las herramientas tecnológicas, 5) Codificación e implementación, 6) Estandarización del OA según LOM y SCORM, 7) Aplicación de un instrumento de calidad del OA. No se menciona la conformación del equipo de desarrollo, pero se infiere del trabajo leído la necesidad de los siguientes roles: diseñador instruccional, analista, diseñador de interfaz, experto en tecnologías, programador, evaluador.

MACOBA (Metodología de Aprendizaje Colaborativo fundamentada en patrones para la producción y uso de OA) [44]: Especifica un proceso por niveles que, en definitiva, son etapas secuenciales en las que se usan diferentes patrones: 1) Nivel requerimientos: los patrones didácticos (plantillas) guían al diseñador instruccional, 2) Nivel análisis: patrones de casos de uso y diagramas de secuencia en UML se usan para comunicarse con los diseñadores; 3) Nivel diseño y desarrollo: diseñador tecnológico usa patrones de software; 4) Nivel implementación, 5) Nivel evaluación. El equipo de

desarrollo está formado por el docente, el diseñador instruccional y el diseñador tecnológico.

Plan Ceibal – Uruguay [45]: Concibe el ciclo de vida en 3 etapas: 1) Diseño: se lleva a cabo el diseño instruccional, creación y elaboración del OA, 2) Almacenamiento: se realiza el empaquetamiento, catalogación y publicación en un repositorio, 3) Presentación/difusión: refiere al uso en distintos contextos (aula, cursos a distancia). La metodología está destinada a docentes.

DINTEV (Dirección de Nuevas Tecnologías y Educación Virtual de la Universidad del Valle) [46]: Se utiliza el modelo de ciclo de vida incremental IWeb, al que se integra un modelo pedagógico para el diseño de OA, y se divide en 5 fases: 1) Formulación y planificación, 2) Análisis, 3) Ingeniería, 4) Generación de páginas y pruebas, 5) Evaluación del cliente. El equipo de desarrollo es multidisciplinar: docente, asesor pedagógico, diseñador gráfico, ingeniero de sistemas y comunicador.

AODDEI (Análisis, Obtención, Diseño, Desarrollo, Evaluación, Implantación) [47]: Está pensada para guiar a los docentes en la parte pedagógica, para lo que provee una serie de plantillas. Las fases que involucra son: 1) Análisis y Obtención, 2) Diseño, 3) Desarrollo, 4) Evaluación, 5) Implantación. El equipo de desarrollo lo conforman: docente, técnico en diseño y evaluadores.

Univap Virtual [48]: La metodología tiene cinco fases, cada una con actividades prescriptas: 1) Análisis, 2) Planificación y desarrollo educativo, 3) Pre-producción, 4) Producción, 5) Integración. El equipo de desarrollo está integrado por: pedagogos, profesores con conocimientos en el área del contenido del OA, coordinadores del curso/proyecto, programador.

Universidad Autónoma Metropolitana Cuajimalpa [49]: La metodología está destinada a docentes no expertos en aspectos técnicos y describe el formato de la documentación y productos que se obtienen en cada una de las 5 etapas: 1) Análisis, 2) Diseño, 3) Desarrollo, 4) Implementación y 5) Evaluación.

Facultad de Ingeniería de la Universidad Nacional de la Patagonia San Juan Bosco [50]: Propone una metodología de 5 etapas secuenciales: 1) Crear los contenidos, 2) Generar los OA, 3) Almacenarlos en un repositorio, 4) Buscar en el repositorio los OA que se necesitan, 5) Montar el curso con esos OA más los que se necesiten crea. Las autoras no especifican el destinatario de la metodología, pero se infiere de la lectura del trabajo que es un docente con conocimientos en informática.

Universidad de Guadalajara [51]: Metodología para la producción de OA basada en la utilización patrones, destinada a docentes con conocimientos de informática. Los patrones se utilizan en la selección de las dimensiones de aprendizaje y actividades cognitivas. El proceso involucra 3 fases: 1) Selección del Patrón, 2) Selección y elaboración de contenidos, 3) Parametrización del patrón.

Universidad Politécnica de Valencia [52]: Está destinada al docente con conocimientos en informática. Se basa en

9 pasos: 1) Determinar el tipo de objetivo a alcanzar, 2) Seleccionar los contenidos, 3) Elegir el formato digital del OA, 4) Realizar la introducción, 5) Desarrollar el contenido del OA; 6) Cerrar el OA, 7) Elaborar la ficha de metadatos; 8) Evaluar el OA.

MIDOA (Modelo Instruccional para el Diseño de OA) [53]: La metodología se compone de 5 fases que se desarrollan en un proceso en espiral incremental con un mínimo de 3 ciclos: 1) Análisis, 2) Diseño; 3) Desarrollo; 4) Utilización, 5) Evaluación. En el primer ciclo se desarrollan los contenidos por primera vez, en el segundo ciclo se optimizan los contenidos para mejorar su calidad académica, pedagógica, ilustrativa, etc., y en el tercer ciclo se optimiza la interfaz para aumentar su atractivo.

Cada una de las fases tiene estipulado un conjunto de artefactos de entrada y de salida. El equipo de desarrollo está conformado por: experto en contenido, pedagogo, analista, diseñador instruccional, desarrollador, evaluador.

Universidad Austral de Chile [54]: La construcción del OA se realiza en 7 etapas: 1) Diseño del modelo del OA, 2) Construcción de la webgrafía del OA, 3) Formulario de metadatos, 4) Diseño material pedagógico, 5) Elaboración multimedial del OA, 6) Diseño de la encuesta de validación, 7) Evaluación del OA por parte de los estudiantes. La metodología está destinada a docentes y estudiantes.

LOCoME (Learning Object Construction Methodology) [55]: Los autores no especifican usuarios de la metodología, pero se infiere de la lectura de su trabajo que es un equipo multidisciplinar. Incorporan consideraciones pedagógicas y de diseño instruccional al RUP. Es una metodología iterativa compuesta de 4 fases para las cuales se prescribe el conjunto de criterios de evaluación y de artefactos de salida: 1) Análisis, 2) Diseño Conceptual, 3) Construcción, 4) Evaluación Pedagógica. Bajo la consideración de calidad sistémica, las iteraciones ocurren entre las fases y dentro de cada una.

APROA (APREndiendo con OA) [56]: Creada por un grupo de investigación de educación a distancia en Chile, hace uso de una plataforma que da soporte al docente en la construcción del OA. Se compone de un conjunto de pasos que se realizan por dos vías paralelas: 1.1) Definición del objetivo, 1.2) Llenado del formulario, 1.3) Elaboración de la aplicación, 1.4) Elaboración de la evaluación, 2.1) Elaboración del contenido (texto y multimedia), 2.2) Incorporación a la plantilla, 2.3) Adjuntado en la plataforma, 2.4) Generación del metadato, 2.5) Empaquetado del objeto. Los pasos 1.1, 2.1 y 2.2 se realizan fuera de la plataforma. El equipo de desarrollo lo conforma el docente (oficia el rol de desarrollador de contenido) y el desarrollador multimedia.

ISDMeLO (por su sigla del inglés Instructional System Development Methodology based on e-Learning Objects) [57]: Incluye 5 fases en las que se prescribe un conjunto de procedimientos junto a la documentación y artefactos de salida: 1) Análisis, 2) Diseño, 3) Desarrollo, 4) Implementación, 5) Evaluación. En función de la

evaluación el OA puede tener que ser actualizado. El equipo de desarrollo es multidisciplinar: diseñador instruccional, docente, desarrollador, evaluador.

Universidad Nacional de La Plata [58]: Especifica una matriz de actividades para los procesos de cada etapa: 1) Factibilidad, 2) Definición de requisitos del sistema, 3) Especificación de los requisitos del prototipo, 4) Diseño del prototipo, 5) Diseño detallado el prototipo, 6) Desarrollo del prototipo, 7) Implementación y prueba del prototipo, 8) Refinamiento iterativo de las especificaciones del prototipo, se puede volver a la etapa 2

o continuar si se logró el objetivo y alcance deseados, 9) Diseño del sistema final, 10) Implementación del sistema final, 11) Operación y mantenimiento, 12) Retiro (si corresponde). El equipo de desarrollo está integrado por: docente, pedagogo, analista, programador, coordinador, diseñador gráfico, especialista en sonido y video.

En la tabla 2 se extraen las prácticas ágiles aplicadas en cada metodología. El modelo de ciclo de vida en Cascada, en su forma pura o con leves modificaciones, es utilizado por 21 metodologías de las 28 recabadas.

Tabla 2. Prácticas ágiles aplicadas en las metodologías de desarrollo de OA recabadas en la bibliografía.

Metodología	País, año	Modelo de ciclo de vida	Usuario de la metodología	Prácticas ágiles
Univ. Nac. del Litoral [30]	Argentina, 2020	Iterativo incremental	Equipo multidisciplinar	Desarrollo iterativo incremental
OAULA [31]	Venezuela, 2019	Prototipo evolutivo	Equipo multidisciplinar	Gestión de la calidad durante todo el proceso de desarrollo
Univ. Militar Nueva Granada [32]	Colombia, 2016	Cascada	Equipo multidisciplinar	Ninguna
INTERA [33]	Brasil, 2016	Cascada con iteración de las etapas	Equipo multidisciplinar	Iteraciones
DICREVOA [34]	Ecuador, 2015	Cascada	Docente no informático	Ninguna
Univ. Nac. de Loja [35]	Ecuador, 2015	Cascada con retroalimentación entre algunas etapas	Docente con conocimientos en informática	Ninguna
UBoa [36]	Colombia, 2014	Cascada	Equipo multidisciplinar	Seguimiento continuo de la calidad en todo el proceso de desarrollo.
CROA [37]	Argentina, 2014	Cascada con retroalimentación entre las etapas	Docente o equipo multidisciplinar	Ninguna
MEDOA [38]	México, 2013	Cascada combinado con Espiral	Docentes y alumnos con o sin experiencia en informática	Iteraciones incrementales del conjunto de fases 2) Análisis, 3) Diseño, 4) Implementación y 5) Validación
ISDOA [39]	Colombia, 2012	Cascada con V&V	Equipo multidisciplinar	La calidad es transversal al proceso de desarrollo
MESOVA [40]	Colombia, 2011	Cascada, con Prototipo evolutivo sólo en la fase de diseño	Equipo multidisciplinar	Gestión continua de la calidad mediante puntos de control al final de cada fase. Desarrollo de prototipos de forma evolutiva (producto incremental)
MPOBA [41]	Argentina, 2011	Prototipo evolutivo	Equipo multidisciplinar	Involucramiento del usuario desde el inicio del proceso de desarrollo. Uso de prototipos para refinar necesidades
Tecnopedagógica [42]	Venezuela, 2011	Cascada	Equipo multidisciplinar	Ninguna
MACOBA [43]	México, 2009	Cascada	Equipo multidisciplinar	Ninguna
Plan Ceibal [44]	Uruguay, 2009	Cascada	Docente	Ninguna
DINTEV [45]	Colombia, 2009	Incremental IWeb	Equipo multidisciplinar	Desarrollo incremental
AODDEI [46]	México, 2008	Cascada, con retroalimentación entre la fase de Implantación y la de Diseño	Equipo multidisciplinar	Ninguna
Univap Virtual [47]	Brasil, 2008	Cascada	Equipo multidisciplinar	Ninguna
Univ. Autónoma Metrop. Cuajimalpa [48]	México, 2008	Cascada	Docente no informático	Ninguna
Univ. Nac. de la Patagonia San Juan Bosco [49]	Argentina, 2008	Cascada	Docente con conocimientos en informática	Ninguna
Univ. de Guadalajara [50]	México, 2007	Cascada	Docente con conocimientos en informática	Ninguna
Univ. Politécnica de Valencia [51]	España, 2007	Cascada	Docente con conocimientos en informática	Ninguna
MIDOA [52]	México, 2007	Espiral	Equipo multidisciplinar	Desarrollo incremental
Univ. Austral de Chile [53]	Chile, 2006	Cascada	Docentes y estudiantes	Ninguna
LOCoME [54]	Venezuela, 2006	RUP	Equipo multidisciplinar	Desarrollo iterativo. Concepción sistémica de la calidad
APROA [55]	Chile, 2005	Cascada	Docente y desarrollador multimedia	Ninguna
ISDMLO [56]	Brasil, 2004	Cascada	Equipo multidisciplinar	Ninguna
Univ. Nac. de La Plata [57]	Argentina, 2000	Prototipo evolutivo	Equipo multidisciplinar	Desarrollo incremental

Conclusiones

La mayoría de las metodologías analizadas aplican el ciclo de vida en cascada con algunas leves modificaciones en cuanto a nombre y número de etapas y eventual retroalimentación entre ellas. Como primer paso especifican todos los requisitos de la aplicación; en este sentido, cuentan con la ventaja que dichos requisitos son relativamente estables, no están sometidos al ritmo de cambio ni al nivel de incertidumbre que experimentan en las aplicaciones comerciales, por lo que se justifica no recurrir a la agilidad.

En general llevan control del proceso de desarrollo a través de la generación de documentación. La misma consiste en un conjunto entregables, usualmente en forma de plantillas, generados luego de finalizada cada fase/etapa para servir como insumos de entrada en la fase/etapa siguiente. No se prioriza la comunicación cara a cara.

Aquellas metodologías que son iterativas comparten con el modelo ágil la idea de iteración, aunque no con la misma concepción. No tienen definida su duración en el tiempo y, además, el valor entregado al usuario luego de cada iteración nunca es software funcionando, son sólo prototipos refinados o especificaciones de requisitos para tenerlos cerrados antes de empezar a programar. Por otra parte, la actividad de planificación se da al principio del proceso de desarrollo, no de cada iteración.

Son muy pocas las que conciben la calidad como un continuo transversal a todo el proceso de desarrollo. En general, consideran que la evaluación de calidad debe ser realizada por el alumno o docente que lo emplea en sus clases luego de que el OA está terminado y en producción.

Las metodologías que dicen aplicar agilidad no explicitan cómo lo hacen ni tampoco se evidencia este hecho en la descripción de las fases. Por lo que el uso de un *framework* ágil se reduce a una mera expresión de deseos.

En ninguna de las metodologías observadas se implementan reuniones de reflexión para mejora continua ni tampoco se contempla la noción de equipo estable,

multifuncional y autoorganizado. En algunas ni siquiera hay equipo, sólo es el docente que, con nociones de informática o cierto dominio en herramientas de autor, construye el OA. Pero aún en aquellas que sí identifican equipo, si bien hay especificación de roles (diseñador instruccional, programador, docente, etc.) que evidencian la multidisciplinariedad en el desarrollo de OA, es un aspecto a revisar si se entiende la multifuncionalidad como responsabilidad compartida, esto es, que si una persona falta en el equipo por alguna causa otro miembro puede suplantarla. Porque en desarrollo de OA el conocimiento en la disciplina o en pedagogía recae sobre los expertos en dichas áreas, su trabajo no puede ser realizado por otros; incluso el concepto de estabilidad se pone en duda por este motivo ya que para OA destinados a distintas asignaturas se necesitarán expertos diferentes, al menos uno por cada disciplina, lo que implica repensar las posibilidades de aplicar la agilidad, o bien, considerar su adaptación a esta situación particular.

Sin embargo, si se considera que un OA se compone de objetivos, contenido, actividades, evaluación y metadatos, un marco de trabajo ágil podría aprovechar dicha estructura para desarrollarla en iteraciones incrementales bajo los criterios de calidad tecnológica y pedagógica propuestos por [59]. Esto es, en los primeros *sprints* sería factible priorizar objetivos y contenido; en los siguientes, las actividades y así sucesivamente hasta llegar a la integración del OA en un sistema de gestión de aprendizaje, de modo que al finalizar cada iteración se obtenga un incremento de producto probado. Asimismo, el alto compromiso creativo e intelectual que involucran todas las tareas vinculadas a la concepción, desarrollo y uso de un OA es un indicador más por el que la opción por la agilidad podría resultar beneficiosa.

Agradecimientos

Las autoras agradecen a los siguientes proyectos de investigación de la UTN Santa Fe, Argentina: PID 7757 Aplicación de Técnicas de Inteligencia Artificial en plataformas de e-learning para dar soporte a estrategias Pedagógicas; PID 7688 Análisis de las Políticas de Accesibilidad en Carreras de Ingeniería de la Universidad Tecnológica Nacional Facultad Regional Santa Fe.

Referencias

[1] B. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. Martin, S. Mellor, K. Schwaber, J. Sutherland and D. Thomas, “Manifiesto por el desarrollo ágil de software”, 2001. [Online]. Disponible: <http://agilemanifesto.org/iso/es/manifesto.html> [Accedido Abril, 16, 2021].

[2] J. Appelo, *#Workout: games, tools & practices to engage people, improve work, and delight clients*. Happy Melly Express, 2014.

[3] J. Appelo, *How to Change the World: Change Management 3.0*. Jojo Ventures BV, 2012.

[4] J. Appelo, *Management 3.0: Leading Agile Developers, Developing Agile Leader*. Pearson Education, 2010.

[5] C. Medina-López, J. Marín-García and R. Alfalla-Luque, “Una propuesta metodológica para la realización de búsquedas sistemáticas de bibliografía,” *WPOM-Working Papers on Operations Management*, vol. 1, no. 2, pp. 13–30, 2010. doi: <https://doi.org/10.4995/wpom.v1i2.786>

[6] J. Garzías, Gestión de proyectos ágil y las experiencias de más de 12 años de proyectos ágiles, 233 grados de TI, 2011.

[7] H. Kniberg and M. Skarin, *Kanban y Scrum – obteniendo lo mejor de ambos*. InfoQ, 2010.

[8] L. Adkins, *Coaching Agile Teams: a Companion for ScrumMasters, Agile Coaches and Project Managers in Transition*. Addison-Wesley, 2010.

[9] A. Cockburn, *Crystal Clear: A Human-Powered Methodology for Small Teams: A Human-Powered Methodology for Small Teams*. Agile Software Development Series, 2004.

[10] M. Cohn, *Succeeding with Agile: Software Development Using Scrum*. Addison-Wesley, 2009.

[11] K. Schwaber y J. Sutherland. (2017). La guía de Scrum. [Online]. Disponible: <https://scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Spanish-SouthAmerican.pdf>

[12] J. Sutherland, *Scrum: The Art of Doing Twice the Work in Half the Time*. Crown Publishing Group, 2014.

[13] M. Cohn, *User Stories Applied: For Agile Software Development*. Addison-Wesley, 2004.

[14] J. Garzías, *La guía de historias de usuario*. 233 grados de TI, 2021.

[15] K. Beck and C. Andres, *Extreme Programming Explained: Embrace Change*, 2^{da} ed. Addison-Wesley, 2004.

[16] R. Pichler, *Agile product management with Scrum: creating products that customers love*. Pearson Education, 2010.

[17] J. Canós, P. Letelier and M. Penadés, “Metodologías ágiles en el desarrollo de software,” in *VIII Jornadas de Ingeniería del Software y Bases de Datos*, 2003.

[18] V. Bertossi and M. Gutiérrez, “Objetos de aprendizaje, Estado del arte,” in *IEEE Argencon*, Resistencia, 2020. [En proceso de publicación].

- [19] A. Chiappe Laverde, “Acerca de lo pedagógico en los objetos de aprendizaje. Reflexiones conceptuales hacia la construcción de su estructura teórica,” *Estudios Pedagógicos*, vol. 35, no. 1, pp. 261-272, 2009.
- [20] A. Pinilla Gómez, M. Gutiérrez and L. Ballejos, “AMELOIR: algoritmo para la extracción automática de metadatos a partir de objetos de aprendizaje en un repositorio institucional,” Tesis de Maestría, Universidad Tecnológica Nacional, Santa Fe, Argentina, 2017.
- [21] C. Barritt, D. Lewis and W. Wieseler, “Cisco Systems Reusable Information Object Strategy: Definition, Creation Overview, and Guidelines,” Cisco System, Inc., 1999.
- [22] H. W. Hodgins. (2000). Into the Future: A Vision Paper. [Online]. Disponible: <http://members.aect.org/publications/InstructionalUseofLearningObjects.pdf#page=28>
- [23] S. Martínez Naharro, P. Bonet, P. Cáceres, F. Fargueta and E. García, “Los objetos de aprendizaje como recurso de calidad para la docencia: criterios de validación de objetos en la Universidad Politécnica de Valencia,” in *IV Simposio Pluridisciplinar sobre Diseño y Evaluación de Contenidos Educativos Reutilizables*, 2007.
- [24] M. Zapata Ros, “Secuenciación de contenidos y objetos de aprendizaje,” *RED: Revista de Educación a Distancia*, no.13, 2005.
- [25] M. D. Merrill, “Components of instruction toward a theoretical tool for instructional design,” *Instructional Science*, vol. 29, no. 4, pp. 291–310, 2001. doi: <https://doi.org/10.1023/A:1011943808888>
- [26] J. Maldonado, C. Sanz and A. Fernández-Pampillón, “Desarrollo de un Marco de Análisis para la Selección de Metodologías de Diseño de Objetos de Aprendizaje (OA) basado en criterios de calidad para contextos educativos específicos,” Tesis de Maestría, Universidad Nacional de La Plata, La Plata, Buenos Aires, 2015.
- [27] R. Mason, M. Weller and C. Pegler, *Learning in the Connected Economy*. Londres: Open University, 2003.
- [28] M. Gértrudix, S. Álvarez, A. Galisteo, M. Gálvez and F. Gértrudix, “Actions in the design and development of digital educational objects: institutional programmes,” *RUSC Universities and Knowledge Society Journal*, vol. 4, no. 1, pp. 14–25, 2007. doi: <https://doi.org/10.7238/rusc.v4i1.296>
- [29] L. García Aretio, Objetos de aprendizaje. Características y repositorios, En Boletín Electrónico de noticias de Educación a Distancia, 2005. [Online]. Disponible: <http://espacio.uned.es/fez/eserv/bibliuned:327/editaabril2005.pdf>
- [30] N. Butcher, A. Kanwar and S. Uvalić-Trumbić, “A basic guide to open educational resources (OER),” in *Commonwealth of Learning – UNESCO – Section for Higher Education*, Vancouver, París, 2011.
- [31] L. Romero, “Objetos de Aprendizaje basados en Competencias: Una metodología para su desarrollo en carreras de Ingeniería,” in *IEEE Argencon*, Resistencia, 2020. [En proceso de publicación].
- [32] B. Sandia Saldivia J. Pérez Correa and O. Derwis Rivas, “Propuesta metodológica para la creación de Objetos de Aprendizaje,” *Revista Electrónica de Enseñanza de las Ciencias*, vol. 18, no. 3, pp. 521-542, 2019.
- [33] L. Morales, L. Gutiérrez and L. Ariza, “Guía para el diseño de objetos virtuales de aprendizaje (OVA). Aplicación al proceso enseñanza-aprendizaje del área bajo la curva de cálculo integral,” *Revista Científica General José María Córdova*, vol. 14, no. 18, pp. 127-147.
- [34] J. Braga, *Objetos de Aprendizaje Volumen 2: Metodología de Desarrollo*. Santo André: UFABC, 2016.
- [35] J. Maldonado, J. Bermeo and M. Mejía, “DICREVOA: A Proposal for the Design, Creation and Evaluation of Learning Objects,” in *XLI Latin American Computing Conference (CLEI)*, 2015. doi: <https://doi.org/10.1109/CLEI.2015.7360035>
- [36] P. Collaguazo, A. Padilla y L. Chamba-Eras. “Propuesta de un Modelo Genérico para el Diseño y Valoración de Objetos de Aprendizaje basado en Estándares E-Learning,” in *X Conferencia Latino-Americana de Objetos y Tecnologías de Aprendizaje (LACLO)*, 2015, pp. 227-236. <https://www.br-ie.org/pub/index.php/teste/article/view/5803/4093>
- [37] L. Bernal y J. Ballesteros, “UBoa, un Referente Metodológico para la Construcción de Objetos Virtuales de Aprendizaje,” *INGE CUC*, vol. 10, no. 2, pp. 67–75, 2014.
- [38] C. Sanz, F. Barranquero y L. Moralejo, *Metodología Croa*, 2014.
- [39] M. Alonso, I. Castillo, V. Martínez and Y. Muñoz, “MEDOA: Metodología para el Desarrollo de Objetos de Aprendizaje,” in *XII Conferencia Iberoamericana en Sistemas, Cibernética e Informática (CISCI)*, 2013. https://www.iiis.org/CDs2013/CD2013SCI/CISCI_2013/PapersPdf/XA247VX.pdf
- [40] E. Sema, C. Castro y R. Botero, “ISDOA: Ingeniería de Software para Desarrollar Objetos de Aprendizaje,” in *Euro American Conference on Telematics and Information Systems (EATIS)*, 2012.
- [41] E. Parra, “Propuesta de metodología de desarrollo de software para objetos virtuales de aprendizaje - MESOVA-,” *Revista Virtual Universidad Católica del Norte*, no. 34, pp. 113-137, 2011.

- [42] S. Massa, A. De Giusti y P. Pesado, “MPOBA: un Modelo de Proceso para el desarrollo de Objetos de Aprendizaje”, En XVII Congreso Argentino De Ciencias De La Computación (CACIC), 2011. http://sedici.unlp.edu.ar/bitstream/handle/10915/18712/Documento_completo.pdf?sequence=1
- [43] Y. Hernández y A. Silva, “Una experiencia tecnopedagógica en la construcción de objetos de aprendizaje web para la enseñanza de la matemática básica,” *Revista de Tecnología de Información y Comunicación en Educación*, vol. 5, no. 1, pp. 57-72, 2011.
- [44] M. Margain Fuentes, J. Muñoz Arteaga and F. Álvarez Rodríguez, “Metodología de Aprendizaje Colaborativo fundamentada en patrones para la producción y uso de Objetos de Aprendizaje,” *Investigación y Ciencia*, vol. 17, no. 44, pp. 22-28, 2009.
- [45] Plan Ceibal, *Manual para el diseño y desarrollo de objetos de aprendizaje*, 2009. [Online]. Disponible: <https://blogs.ceibal.edu.uy/formacion/wp-content/uploads/2015/06/356059053GUIAObjetosCeibal09-pdf>
- [46] M. Borrero, E. Cruz, S. Mayorga and K. Ramírez, “Una metodología para el diseño de Objetos de Aprendizaje. La experiencia de la Dirección de Nuevas Tecnologías y Educación Virtual, Dintev, de la Universidad del Valle,” in *Objetos de aprendizaje: prácticas y perspectivas educativas*, C. Valencia y A. Jiménez, Eds. Pontificia Universidad Javeriana, 2009, pp. 37- 59.
- [47] B. Osorio Urrutia, J. Muñoz Arteaga, F. Álvarez Rodríguez y C. Arévalo Mercado. (2008). “Metodología para elaborar Objetos de Aprendizaje e integrarlos a un Sistema de Gestión de Aprendizaje” [Online]. Disponible: https://campusmoodle.proed.unc.edu.ar/pluginfile.php/51346/mod_folder/content/0/Complementario/OADDIE.PDF?forcedownload=1
- [48] S. Fernandes Bicudo, I. da Silva, I. Ricardi León, T. Nogueira, M. de Paula y M. Prado, M., “Metodología para la construcción de objetos de aprendizaje para educación a distancia,” in *IX Encuentro Internacional Virtual Educa*, Zaragoza, 2008.
- [49] E. Peñaloza and P. Landa, “Objetos de aprendizaje: una propuesta de conceptualización, taxonomía y metodología,” *Revista Electrónica de Psicología Iztacala*, vol. 11, no. 3, pp. 19-49, 2008.
- [50] Z. Rosanigo, G. Bianchi and M. Saenz López, “Diseño de objetos de aprendizaje,” in *III Congreso de Tecnología en Educación y Educación en Tecnología*, 2008.
- [51] J. Delgado, R. Morales, S. González and M. Chan, “Desarrollo de objetos de aprendizaje basado en patrones,” in *Virtual Educa*, Brasil, 2007.
- [52] S. Martínez, P. Bonet, P. Cáceres, F Fargueta y E. García, “Los objetos de aprendizaje como recurso de calidad para la docencia: criterios de validación de objetos en la Universidad Politécnica de Valencia,” in *IV Simposio Pluridisciplinar sobre Diseño, Evaluación y Desarrollo de Contenidos Educativos Reutilizables (SPDECE)*, 2007. <http://ceur-ws.org/Vol-318/Naharro.pdf>
- [53] A. Barajas Saavedra, J. Muñoz Arteaga and F. Álvarez Rodríguez, “Modelo Instruccional para el Diseño de Objetos de Aprendizaje: Modelo MIDOA,” in *Virtual Educa*, Brasil, 2007. <http://hdl.handle.net/20.500.12579/1232>
- [54] S. Bucarey and L. Álvarez, “Metodología de Construcción de Objetos de aprendizaje para la Enseñanza de Anatomía Humana en Cursos Integrados,” *International Journal of Morphology*, vol. 24, no. 3, pp. 357-362, 2006.
- [55] M. Medina and M. López, “LOCoME: Metodología de Construcción de Objetos de Aprendizaje”, En III Simposio Pluridisciplinar sobre Diseño, Evaluación y Descripción de Contenidos Educativos Reutilizables (SPDECE), 2006.
- [56] APROA, *Manual De Buenas Prácticas Para El Desarrollo De Objetos De Aprendizaje*, 2005. [Online]. Disponible: http://formacionprofesional.homestead.com/objetos_de_aprendizaje.pdf
- [57] L. Blondet and R. Nascimento, “Learning Theory and Instruction Design Using Learning Objects,” *Journal of Educational Multimedia and Hypermedia*, vol. 13, no. 4, pp. 343-370.
- [58] Z. Cataldi, “Una metodología para el diseño y desarrollo de software educativo,” Tesis de Maestría, Universidad Nacional de la Plata, La Plata, Buenos Aires, 2000. [Online]. Disponible: <http://sedici.unlp.edu.ar/handle/10915/4055>
- [59] V. Bertossi, K. Martínez, M. Gutiérrez and L. Romero, “Análisis de calidad de objetos de aprendizaje en contextos universitarios,” *Latin-American Journal of Computing*, vol. 7, no. 1, pp. 100-113, 2020

Información de Contacto de los Autores:

Valeria Iliana Bertossi

Talcahuano 6751

Santa Fe

Argentina

vbortossi@frsf.utn.edu.ar

ORCID ID: <https://orcid.org/0000-0003-1904-3281>

María de los Milagros Gutiérrez

Defensa 7754

Santa Fe

Argentina

mgutierr@frsf.utn.edu.ar

ORCID ID: <https://orcid.org/0000-0002-3964-7931>

Valeria Iliana Bertossi

Ing. en Sist. de Información y becaria doctoral de la Universidad Tecnológica Nacional de Argentina. Es autora de diversos OA que se emplean en las cátedras donde se desempeña y de varios artículos publicados en revistas y actas de congresos.

María de los Milagros Gutiérrez

Dra. en Sist. de Información. Dirige proyectos de investigación y actualmente es directora del departamento de Ing. en Sist. de Información y profesora de grado y posgrado de la Universidad Tecnológica Nacional Regional Santa Fe de Argentina.