

Artículo de Investigación

Análisis comparativo entre modelos metodológicos para el desarrollo de aplicaciones web

Antonio David Ruiz Díaz Medina^{1*} y Nazario Luiz Ayala Frasnelli²

^{1,2} Universidad Nacional de Canindeyú – Facultad de Ciencias y Tecnología, Paraguay.

*Autor correspondiente: Antonio David Ruiz Díaz Medina; adavidruizdiaz@gmail.py

Recibido: 21/08/2020 **Aceptado:** 09/08/2021

Resumen

El buen inicio de un proyecto de software depende en gran medida de la elección adecuada de un modelo metodológico que se adecúe a las características y circunstancias del proyecto. En este sentido el presente trabajo se centra en el estudio de metodologías de desarrollo de aplicaciones web con el objetivo de analizar las más utilizadas, clasificarlas, compararlas y sintetizar los resultados para obtener tendencias que ayuden a facilitar su elección. Se trata de una investigación cualitativa basada en una revisión sistemática de la literatura con una serie de procesos bien definidos que llevaron a concluir que las metodologías de desarrollo de aplicaciones web tienden a aplicar procesos tanto lineales como iterativos y que se concentran en las primeras tres etapas del ciclo de vida del desarrollo de software. Además de observar la necesidad de establecer mecanismos de verificación de la efectividad de su empleo.

Palabras clave: ingeniería de software, ingeniería web, metodologías de desarrollo de software, aplicaciones web.

Abstract

The successful start of a software project depends to a great extent on the appropriate choice of a methodological model that suits the characteristics and circumstances of the project. In this sense, the present work focuses on the study of web application development methodologies with the objective of analyzing the most used ones in order to classify them, to compare and to synthesize the results and thus obtain trends that help facilitate their choice. This is a qualitative research based on a systematic review of the literature with a series of well-defined processes that led to conclude that web application development methodologies tend to apply both linear and iterative processes that focus on the first three stages of the

software development life cycle. In addition, it was also observed the necessity of establishing verification mechanisms on the effectiveness of their employment.

Keywords: software engineering, web engineering, software development methodologies, web applications.

1. Introducción

Se puede afirmar que el éxito de los proyectos de software depende en gran medida de que tengan un buen inicio. Uno de los factores clave para un buen inicio es que la elección del modelo de desarrollo se adecúe a las características y circunstancias del proyecto.(1, p. 1)

Este trabajo de investigación se centra en el estudio de los modelos metodológicos para el desarrollo o construcción de aplicaciones web. Para ello se sigue una serie de procesos que ayudan al análisis cualitativo, de las metodologías de desarrollo web, con la finalidad de obtener conclusiones que permitan la comprensión de estos modelos.

Hay una amplia variedad de metodologías para la producción de software de escritorio como para aplicaciones basadas en diferentes plataformas. Cada una con sus propios procesos, artefactos y paradigmas. Siendo algunos más genéricos y otros más específicos. Esta amplia cantidad y variedad hace difícil la selección asertiva de una de ellas para un proyecto específico (2) (3).

La presente investigación busca aportar conocimientos que ayuden a los equipos de desarrollo a elegir metodologías de desarrollo web que se ajusten a sus contextos. El aporte esperado se da mediante comparaciones de las técnicas de modelado, el tipo de flujo de procesos utilizados, artefactos de ingeniería de software empleadas, y las fases del ciclo de vida del desarrollo de software que abarcan las metodologías estudiadas. También se incluye un cuadro comparativo de ventajas y desventajas y un análisis de la tendencia que se pudo observar mediante las comparaciones.

Las metodologías de desarrollo web engloban actividades agrupadas en fases que incluyen las especificaciones necesarias para obtención de un producto de software de calidad. Algunas de estas actividades son: el diseño de la vista, distribución y navegación entre las interfaces de usuarios, definición de la estructura arquitectónica de la aplicación, desarrollo del contenido y la funcionalidad (4) (5). Algunas de las metodologías más extendidas para el desarrollo de aplicaciones web son las siguientes.

OOHDM (Object-Oriented Hypermedia Design Model)

Este modelo fue formulado por Schwabe y Rossi (6), (7). Ellos proponen la construcción de aplicaciones web utilizando un proceso de cuatro fases. Estas fases están basadas en los modelos de proceso incremental y por prototipos, cuyos flujos son iterativos y evolutivos. Esta metodología combina conceptos de orientación a objetos, como las clases, objetos y mecanismos de abstracción, como la agregación y la generalización, con conceptos de hipertexto, como estructuras jerárquicas, nodos y enlaces (8) (9).

WAE (Web Application Extension)

Es una metodología para el proceso de desarrollo de aplicaciones web propuesta por Conallen (10), basada en tres metodologías; clásicas, orientadas a objetos e iterativas; la *Rational Unified Process* (11), el *Use Case Driven Object Modeling with UML* (12) y el *Unified Software Development Process* (13).

La administración del proyecto se realiza estableciendo una serie de planificaciones que incluyen típicamente la planificación general del proyecto, la planificación de la iteración, la administración de riesgo y el monitoreo del proyecto.

UWE (UML-based Web Engineering)

“Es un enfoque de ingeniería de software para el dominio web cuyo objetivo es cubrir todo el ciclo de vida del desarrollo de aplicaciones web” (14). Utiliza notaciones y diagramas UML¹ para el análisis y diseño de aplicaciones web siempre que sea posible.

La metodología divide la aplicación en varios modelos. Por ejemplo, los modelos de contenido que definen la estructura de los datos manejados por la aplicación y el modelo de navegación, que define la navegación del usuario dentro de la aplicación (12, pp. 78-85) (9).

SOHDM (Scenario-based Object-oriented Hypermedia Design Methodology)

Tiene varias características nuevas entre las metodologías para el desarrollo de software, como, estrategias para la identificación de requisitos, mejoras en la calidad de los modelos y la utilización de la orientación a objetos para la definición de vistas, lógica de navegación, etc. (9). Fue propuesta por Lee, H., Lee, C., y Yoo (16).

¹*Unified Modeling Language*, ayuda a especificar, visualizar y documentar modelos de sistemas de software, incluida su estructura y diseño basado en la orientación a objetos.

WSDM (Web SemanticsDesignMethod)

Es una metodología secuencial que permite, a los desarrolladores de sistemas web, obtener productos de calidad siguiendo una filosofía de diseño bien definida y apoyada adecuadamente. Este método fue propuesto por De Troyer, Casteleyn, y Plessers (17).

RNA (RelationshipNavigationalAnalysis)

La metodología RNA fue ideada por Bieber (18). En esta propuesta se analizan los objetos, componentes y operaciones que se interrelacionan para conocer y entender todos los elementos de la aplicación. Con esto se espera que el análisis de relaciones o análisis de hipertexto tenga un papel importante en el diseño de aplicaciones que tengan una alta interacción con el usuario.

RMDM (Relationship Management Data Model)

La metodología RMDM provee un lenguaje para describir las unidades de información y los mecanismos de navegación necesarios para una aplicación web. La metodología fue desarrollada por Isakowitz, Stohr, y Balasubramanian (19). Ofrece una secuencia de pasos y recomendaciones que son basadas en los modelos de procesos incrementales. Se enfatiza en la obtención de modelos representados gráficamente, así como en los procedimientos paso a paso para el diseño y el desarrollo de aplicaciones web (9).

2. Materiales y Métodos

El presente trabajo se trata de una revisión sistemática de la literatura que busca obtener el estado del arte de los modelos metodológicos utilizados para la construcción de aplicaciones web. El enfoque cualitativo de esta investigación ayuda al análisis flexible y subjetivo de las metodologías de desarrollo identificando tendencias en la utilización de estas y analizando los procesos y características de cada una. El estudio consistió en las siguientes fases.

2.1. Fase de búsqueda y selección de los modelos metodológicos

Fueron consultados bibliotecas digitales, portales y plataformas de consulta de revistas indexadas y libros electrónicos.

Los términos principalmente buscados guardan relación con “Procesos de desarrollo de software”, “Procesos de desarrollo de aplicaciones web”, “Metodologías para el desarrollo de software”, “Metodologías para el desarrollo de aplicaciones web”. Además los materiales consultados debían estar escritos en español, inglés o portugués, la fecha de publicación

debía estar entre los años 1980 a 2019 y los tipos de materiales fueron estudios primarios que guarden relación con el tema de investigación.

Siguiendo con los criterios de selección descritos anteriormente, se seleccionaron 60 documentos. Luego de la revisión de los documentos se logró identificar un total de quince metodologías de desarrollo de software, los cuales se redujeron a siete ya que las metodologías seleccionadas debían estar enfocadas a entornos de ejecución web, se descartaron aquellas con escasa documentación y también debía de tener soporte para la aplicación de estándares en el ciclo de vida del software a desarrollar.

2.2. Fase de extracción y síntesis de datos

Consistió en el estudio de las metodologías seleccionadas, la construcción de esquemas y resúmenes compuestos de sus descripciones, conceptos principales y de las fases de cada una de ellas.

2.3. Fase de clasificación de las metodologías

La clasificación de los modelos metodológicos fue realizada en base a la naturaleza y características de los modelos estudiados. Esto permitió una mejor comprensión de cada tipo de proceso, así como del enfoque sobre el cual fue concebido.

2.4. Fase de análisis comparativo y síntesis realista

Luego, el análisis comparativo permitió la comprensión de las conexiones que existen entre cada metodología o proceso metodológico. El análisis se basó en la comparación de los artefactos utilizados en cada metodología, estos artefactos se describen a continuación.

Tabla 1. Artefactos de los procesos de desarrollo estudiados.

Código	Artefacto	Descripción
A01	Requisitos	Son las condiciones necesarias que se debe satisfacer para alcanzar la calidad de un producto.
A02	Diagrama de arquitectura	Representación gráfica de la estructura general de la aplicación y las relaciones entre sus componentes.
A03	Diagramas de casos de usos	Representación gráfica de las acciones que un usuario puede realizar en la aplicación.
A04	Diagramas de clases	Representación de las relaciones que existen entre los objetos identificados en un sistema.

Código	Artefacto	Descripción
A05	Diagramas de secuencias	Ilustra la secuencia de acciones, eventos y funciones que serán ejecutadas en un sistema cuando el usuario realice una acción.
A06	Diagramas de colaboración	Ilustra la relación de colaboración entre los componentes de un sistema.
A07	Diagramas de estados	Representación de los estados que puede tomar el sistema según las acciones que realice el usuario.
A08	Diseño de navegación	Grafico que ilustra el esquema de navegación de la aplicación.
A09	Diseño de interfaz	Representación abstracta de la disposición de cada componente visual dentro de la aplicación.
A10	Modelos E-R	Diagrama que muestra la relación de las entidades y sus propiedades dentro en un sistema de persistencia de datos.
A11	Diagrama de componentes	Ilustra la relación que existe entre los componentes visuales de la aplicación.
A12	Diagramas de Flujo de datos	Representación gráfica del movimiento que realizan los datos dentro y fuera del sistema.
A13	Pruebas	Diseño de actividades que deben ser realizadas para demostrar el buen funcionamiento de los componentes lógicos del sistema.
A14	Artefactos propios de la metodología	Incluye notaciones enriquecidas o propias de la metodología que ayudan al análisis y diseño específicos para entornos web.

Por último se realizó un análisis subjetivo de los resultados obtenidos con el que se explica el estado del arte de los modelos metodológicos de desarrollo de aplicaciones web, sus diferencias y similitudes, así como su aplicabilidad en el contexto actual en el que se lleva a cabo esta investigación.

3. Resultados y discusión

Se logró identificar un total de quince metodologías de los cuales aplicando los criterios de selección establecidos en la fase de búsqueda y selección se obtuvieron las siguientes:

- OOHDM (*Object-Oriented Hypermedia Design Model*).
- WAE (*Web Application Extension*).
- UWE (*UML-based Web Engineering*).
- SOHDM (*Scenario-based Object-oriented Hypermedia Design Methodology*).
- WSDM (*Web Semantics Design Method*).
- RNA (*Relationship Navigational Analysis*).
- RMDM (*Relationship Management Data Model*).

3.1. Clasificación de los modelos metodológicos para el desarrollo de aplicaciones web

Los modelos metodológicos estudiados se clasifican según, la técnica de modelado utilizada (Tabla 2) y el flujo del proceso de desarrollo (Tabla 3).

Tabla 2. Clasificación de metodologías de desarrollo web según la técnica de modelado.

Metodología \ Paradigma	Orientado a objetos	Orientado a Relaciones	Orientado a escenarios
OOHDM	✓		
WAE	✓		
UWE	✓		
SOHDM	✓		✓
WSDM	✓	✓	
RNA		✓	
RMDM		✓	

La Tabla 2 muestra que las metodologías estudiadas se clasifican en tres tipos, orientadas a objetos, a relaciones y a escenarios. Además se puede observar que las metodologías SOHDM y WSDM metodologías se encuadran en dos clasificaciones diferentes y que la mayoría de las metodologías, aquí seleccionadas, son del tipo orientadas a objetos.

Tabla 3. Clasificación de metodologías de desarrollo web según el flujo de proceso.

Metodología \ Flujo	Lineal	Iterativo	Evolutivo	Paralelo
OOHDM		✓	✓	
WAE		✓		
UWE	✓	✓		
SOHDM		✓	✓	
WSDM	✓			
RNA	✓			
RMDM	✓	✓		

La clasificación según el flujo de proceso expone que las metodologías estudiadas en este trabajo, son mayormente de tipos lineales e iterativas. Se considera que para establecer un flujo de proceso paralelo, las fases iniciales del modelo metodológico empleado, deben haber sido realizadas y verificadas siguiendo un proceso lineal o iterativo previo.

3.2. Análisis comparativo

Los resultados del análisis comparativo se incluyen en la Tabla 4 Tabla 4. Artefactos de la ingeniería de software incluidos en la metodología., Tabla 5 y Tabla 6.

Tabla 4. Artefactos de la ingeniería de software incluidos en la metodología.

Metodología	Artefactos													
	A01	A02	A03	A04	A05	A06	A07	008	A09	A10	A11	A12	A13	A14
OOHDM				✓		✓		✓	✓					
WAE	✓		✓	✓	✓	✓	✓				✓		✓	
UWE	✓		✓	✓			✓	✓						✓
SOHDM	✓	✓		✓		✓		✓				✓		
WSDM	✓			✓				✓		✓	✓			
RNA		✓			✓			✓		✓		✓		
RMDM								✓		✓	✓		✓	✓

Uno de los artefactos más utilizados por las metodologías es el diseño de la navegación de la aplicación. Por otra parte la metodología que aplica mayor número de artefactos en sus procesos es la WAE y la que menos artefactos utiliza es la OOHDM, ambas orientadas a objetos. Se puede notar que las metodologías orientadas a relaciones también aplican artefactos orientados a objetos como una forma de ampliar los mecanismos de diseño y especificación funcional. Otro aspecto a tener en cuenta es la existencia de metodologías que implementa sus propios artefactos complementarios a los usuales.

Tabla 5. Cantidad de fase y enfoque en el ciclo de vida del desarrollo de software.

Metodología	Cant. Fases	Especif.	Análisis	Diseño	Implemt.	Pruebas	Mantenim.
OOHDM	4		✓	✓	✓		
WAE	7	✓	✓	✓	✓	✓	✓
UWE	6	✓	✓	✓			
SOHDM	6	✓	✓	✓	✓		
WSDM	6	✓	✓	✓	✓		
RNA	5		✓	✓	✓		
RMDM	7	✓	✓	✓	✓	✓	✓

La Tabla 5 ilustra la cantidad de fases de cada metodología y los ámbitos de enfoque en cuanto al ciclo de vida del desarrollo de software. Se puede observar que la metodología WAE abarca todos los procesos del ciclo de vida a diferencia de las demás que se enfocan más en el análisis, diseño e implementación.

La Tabla 6 muestra el resultado del análisis comparativo realizado que se origina de las características de cada una.

Tabla 6. Ventajas y desventajas de las metodologías para el desarrollo de aplicaciones web.

Metodol.	Ventajas	Desventajas
OOHDM	<ul style="list-style-type: none"> - Tiene un flujo iterativo e incremental lo que permite una alta calidad del software - Incluye UML estándar de análisis y diseño, el cual es ampliamente aceptado por la comunidad - Número considerable de fases adecuada para equipos de desarrollo pequeños. - Basada en varias metodologías clásicas. - Incluye notaciones UML 	<ul style="list-style-type: none"> - No contempla artefactos para la fase de especificación de requisitos - No incluye fase de prueba
WAE	<ul style="list-style-type: none"> - Procesos iterativos - Abarca todo el ciclo de vida del desarrollo de software. - Maneja gran cantidad de artefactos para el análisis y el diseño. - Incluye notaciones UML enriquecidas para la web. 	<ul style="list-style-type: none"> - Gran cantidad de fases - Necesita planificación previa. - Pensada para grandes equipos de desarrollo.
UWE	<ul style="list-style-type: none"> - Es flexible en cuanto al flujo de los procesos pudiendo adoptar flujos lineales como iterativos. - Maneja gran cantidad de artefactos para el análisis y el diseño. - Incluye artefactos propios - Incluye notaciones UML enriquecidas para la web. 	<ul style="list-style-type: none"> - Pensada para grandes equipos de desarrollo. - No abarca todo el ciclo de vida del desarrollo. - Necesita un estudio previo de los artefactos propios de la metodología.
SOHDM	<ul style="list-style-type: none"> - Tiene un flujo iterativo e incremental. - Maneja los artefactos de ingeniería de software esenciales. - Adecuada para equipos de desarrollo pequeños. 	<ul style="list-style-type: none"> - No abarca todo el ciclo de vida del desarrollo. - Necesita muchos diseños antes de la implementación.
WSDM	<ul style="list-style-type: none"> - Paradigma mixto, relacional y orientado a objetos. - Puede manejar múltiples notaciones y estándares para el análisis y diseño. 	<ul style="list-style-type: none"> - Flujo de proceso lineal - Pocos artefactos de ingeniería de software. - Gran cantidad de sub-fases.
RNA	<ul style="list-style-type: none"> - El enfoque fuerte en la fase de análisis permite la calidad de las siguientes fases. - Adecuada para equipos de desarrollo pequeños. 	<ul style="list-style-type: none"> - Flujo de proceso lineal - No abarca todo el ciclo de vida del desarrollo. - Pocos artefactos de ingeniería de software.

Metodol.	Ventajas	Desventajas
RMDM	<ul style="list-style-type: none"> - Incluye artefactos propios - Es flexible en cuanto al flujo de los procesos pudiendo adoptar flujos lineales como iterativos - Abarca todo el ciclo de vida del desarrollo de software 	<ul style="list-style-type: none"> - Pocos artefactos de ingeniería de software - Gran cantidad de fases - Pensada para grandes equipos de desarrollo

3.3. Síntesis y tendencias

Se puede notar que la tendencia evolutiva de estas metodologías apunta hacia los procesos de desarrollo más flexibles, como los procesos iterativos que permiten una amplia facilidad para adaptarse a los proyectos ágiles (20).

Las fases de las metodologías estudiadas tienen mayor enfoque en los tres primeros procesos del ciclo de vida del desarrollo de software, la especificación de requisitos, el análisis y el diseño. Se puede considerar estos primeros procesos como los más importantes para la ingeniería de software ya que ayudan al entendimiento del problema y el planteo de la solución a construir. Por otra parte existen metodologías que dan apoyo a las etapas más avanzadas del desarrollo, que tampoco se deben descuidar.

La aplicabilidad de una u otra metodología depende, tanto, de la experiencia del equipo, como de la disponibilidad de recursos y del enfoque de la solución planteada. Es por ello que no es posible determinar una metodología apropiada para todos los proyectos.

4. Conclusión

La clasificación de los modelos metodológicos estudiados, según características fácilmente observables permitió el entendimiento de la naturaleza de las metodologías.

El análisis comparativo posibilitó la diferenciación de cada una para lograr determinar sus características, ventajas y desventajas. En este punto se pueden destacar dos metodologías mejor valoradas en cuanto a sus características, la WAE y la SOHDM.

Además, con los resultados obtenidos de las comparaciones entre los diferentes aspectos de estas metodologías, el trabajo permite visualizar sus diferencias y similitudes para así poder facilitar la elección de una de ellas dependiendo del tipo de proyecto a emprender y la disponibilidad de recursos.

Lo anterior puede ayudar a los equipos de desarrollo a identificar y elegir rápidamente una de estas metodologías para aplicar a proyectos que se encuadren en una de ellas.

Gracias a la síntesis de lo aprendido es posible concluir que las metodologías de desarrollo de aplicaciones web tienden hacia los procesos iterativos y que concentran el esfuerzo

requerido para la construcción del software hacia las primeras etapas del ciclo de vida del desarrollo, principalmente porque son las etapas que definen la dirección del proyecto.

Con todo se puede decir que, no se puede señalar una única metodología aplicable a todos los proyectos, sino que se debe decidir, entre una u otra, teniendo en cuenta las características de la aplicación a desarrollar. Y que además se necesitan establecer de métricas para determinar su efectividad.

Conflicto de interés. Los autores declaran que no existe ningún conflicto de interés con respecto a la publicación de este artículo.

Referencias bibliográficas

1. CERVANTES OJEDA, J. y GÓMEZ FUENTES, M. del C. Taxonomía de los modelos y metodologías de desarrollo de software más utilizados. *Universidades*. 2012. No. 52, p. 37-47.
2. ZUMBA, J. y LEÓN, C. Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software. Evolution of the Methodologies and Models used in Software Development. *INNOVA Research Journal*. 2018. Vol. 3, no. 10, p. 20-33.
3. RIVAS, C.I., CORONA, V.P., GUTIÉRREZ, J.F. y HERNÁNDEZ, L. Metodologías actuales de desarrollo de software. *Tecnología e Innovación*. 2015. Vol. 2, no. 5, p. 980-986.
4. UIKEY, N. y SUMAN, U. A Lifecycle Model for Web-based Application Development: Incorporating Agile and Plan-driven Methodology. *International Journal of Computer Applications*. 2015. Vol. 117, no. 19, p. 28-36.
5. ISO, IEC y IEEE. *ISO/IEC/IEEE 24765:2017 International Standard - Systems and software engineering - Vocabulary*. 2a. Switzerland : ISO/IEC, IEEE, 2017.
6. SCHWABE, D. y ROSSI, G. Building hypermedia applications as navigational views of information models. En : *28th Annual Hawaii International Conference on System Sciences*. Wailea : IEEE Computer Society, 1995. p. 231-240.
7. SCHWABE, D. y ROSSI, G. The Object-Oriented Hypermedia Design Model (OOHDM). *CACM*. 1995. Vol. 38, no. 8, p. 45-46.
8. KURNIATI, R. y MAWARNI, S. The Design of Application for Accelerating Funding Proposal Selection Using the Object-Oriented Hypermedia Design Method. *Proceedings of the International Conference on Engineering, Technology and Social Science*. 2021. Vol. 529, p. 720-726.
9. MOLINA-RÍOS, J. y PEDREIRA-SOUTO, N. Comparison of development methodologies in web applications. *Information and Software Technology*. 2020.

Vol. 119.

10. CONALLEN, J. *Building Web applications with UML*. México : Addison-Wesley, 2000. ISBN 0201615770.
11. KRUCHTEN, P. *The Rational Unified Process: An Introduction*. 3a. México : Addison-Wesley, 2003.
12. ROSENBERG, D. y KENDALL, S. *Use Case Driven Object Modeling with UML*. México : Addison-Wesley, 2000.
13. JACOBSON, I., BOOCH, G. y RUMBAUGH, J. *Unified Software Development Process*. México : Addison-Wesley, 1999.
14. LUDWIG-MAXIMILIANS UNIVERSITÄT MÜNCHEN. UWE – UML-based Web Engineering. [en línea]. 2016. [Consulta: 11 octubre 2019]. Disponible en: <http://uwe.pst.ifi.lmu.de/>
15. BUSCH, M. *Evaluating & Engineering: an Approach for the Development of Secure Web Applications*. Munich, 2016.
16. LEE, H., LEE, C. y YOO, C. Scenario-based object-oriented methodology for developing hypermedia information systems. *Proceedings of the Hawaii International Conference on System Sciences*. 1998. Vol. 2, p. 47-56.
17. DE TROYER, O., CASTELEYN, S. y PLESSERS, P. WSDM: Web Semantics Design Method. *Web Engineering*. 2007. Vol. 12, p. 303-352.
18. BIEBER, M. 1: *Relationship-Navigation Analysis for developing World Wide Web*. New Jersey, 1997. 1.
19. ISAKOWITZ, T., STOHR, E.A. y BALASUBRAMANIAN, P. RMM: A Methodolog for Structured Hypermedia Design. *Communications of the ACM*. 1995. Vol. 38, no. 8, p. 34-44.
20. PROYECTOSAGILES.ORG. Desarrollo iterativo e incremental. [en línea]. [Consulta: 19 noviembre 2019]. Disponible en: <https://proyectosagiles.org/desarrollo-iterativo-incremental/>