

Los proyectos y las metodologías / Modelos de procesos / Ciclos de vida de desarrollo de software

Yohelis González , Jean MC Ilenan, Alexis Abrego 

Docente: Luis Bernuil

Sede Veneto, Facultad de Tecnología, Construcción y Medio Ambiente, Licenciatura en Sistema y Programación, Asignatura: Ingeniería de Software

Yohelisgonzalez90@gmail.com, jeanbennett23@gmail.com, abregoalexis55@gmail.com

Resumen

El objetivo de este artículo investigativo fue relacionar el tema de Los proyectos y las metodologías / Modelos de Procesos / Ciclos de Vida de Desarrollo y cuál es su importancia en la ingeniería del software esto implica dar conocer la serie de etapas que debe pasar un proyecto de software el cual busca lograr eficacia y calidad para el usuario final también tomando en cuenta que existen diferentes modelos para implementación de un proyecto de software esto da como ventaja cual sería la mejor opción para utilizar dependiendo el proyecto que se quiera desarrollar.

Palabras clave: desarrollo, metodología, implementar, modelo de proceso, fases

Abstract

The objective of this investigative article was to relate the topic of Projects and methodologies / Process Models / Development Life Cycles and what is their importance in software engineering. This implies making known the series of stages that a project must go through. which seeks to achieve efficiency and quality for the end user, also taking into account that there are different models for implementing a software project, this gives the advantage of which would be the best option to use depending on the project to be developed.

Keywords: development, methodology, implement, process model, phases.

1. Introducción

El Desarrollo de software como dice la palabra es un proceso también llamado ciclo de vida de ahí parte la manera que se ejecutara el proyecto o el sistema, como alcance el tema busca discernir el proceso que conllevara el proyecto así asegura que el mismo cumplas con los requisitos solicitados por el usuario con esto tomar cuenta que fases conforman este proceso, conocer qué modelo favorece la implementación del sistema esto quiere decir cuál es más adecuado para su ejecución.

1.1. Naturaleza y alcance

Este trabajo se desarrolló de acuerdo a las líneas de investigación de la UMECIT, en este caso: Cibersociedad y Desarrollo Digital.

1.2. Objetivos

Objetivo General

Dar a conocer la relación que tiene los proyectos y las metodologías / Modelos de Procesos / Ciclos de Vida de Desarrollo de Software y su importancia en la ingeniería de Software.

Objetivos Específicos:

- Presentar como el concepto desarrollo de software busca un proceso formal, con resultados predecibles, que permitan obtener un producto final de alta calidad y satisfaga las necesidades y expectativas del cliente.
- Dar una visión general enmarcando detalles de valor para la comprensión del lector.

1.3. Justificación

El propósito del trabajo es proporcionar al lector la relación que guarda cada concepto, y la importancia de cómo se ejecuta cada fase el cual busca lograr eficacia y calidad para el usuario final también tomando en cuenta que existen diferentes modelos para implementación de un proyecto de software esto da como ventaja cual sería la mejor opción para utilizar dependiendo el proyecto que se quiera desarrollar.

El beneficiario de este trabajo en primera instancia sería los usuarios interesados en tomar una mejor decisión y los estudiantes que deseen tomar este estudio como un antecedente para proyectos más complejos.

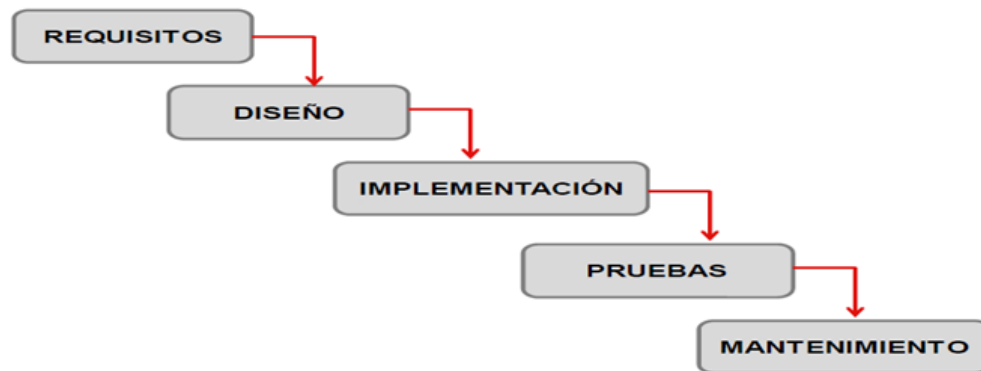
1.4. El Estado del Arte

La construcción de un software se efectúa a través de una serie de fases o etapas la cual lleva como nombre ciclos de vida cabe resaltar que los SLCD deben incluir el proceso de desarrollo y el mantenimiento del mismo durante su producción.

Las fases del desarrollo del software son las siguientes:

- Análisis
- Diseño
- desarrollo
- implementación

- Mantenimiento



(Gomez.S.) dice que cada fase mencionada cumple con un rol en específico, que nos generan resultados del sistema que se está implementando; el proceso del software lleva a cabo las fases antes mencionadas que a continuación daremos su descripción de lo que ejecuta cada una de esas etapas:

1. **Análisis:** como dice la palabra nos detallada la característica del proyecto a implementar cuales son esos requisitos o funcione que el usuario final quiere que genere de acuerdo a lo solicitado por el mismo el equipo procede a la siguiente.
2. **Diseño:** detalla los requerimientos y funciones del sistema a implementar esta fase toma en cuenta los elementos de hardware y software para construcción.
3. **Desarrollo:** esta fase empleas a especialistas para la realización del sistema los cual incluye la realización de diagramas de flujos y la cual da inicio a la producción del proyecto.
4. **Implementación:** se procede a la instalación de los componentes de hardware y software ya seleccionados.
5. **Mantenimiento:** en esta etapa necesario detectar y corregir las fallas que no fueron detectadas en el desarrollo del proyecto o la introducción de mejoras del mismos con estos evitar atrasos u otras deficiencias.

Para continuar resaltamos que los ciclos de vida garantizan la eficacia y calidad del software en proceso para el usuario que lo solicita.

Las metodologías para el desarrollo del software

Nos permiten la organización y la gestión de un proyecto con la finalidad de brinda un producto de calidad y la satisfacción del cliente.

En el presente se menciona dos tipos de metodologías las cuales son: las ágiles y las tradicionales según (Darlyn) Metodología ágil: nos permiten una mejor organización para el desarrollo de un producto eficiente y de calidad y la metodología tradicionales tiene un enfoque más acertado ya que son las más utilizadas.

Recordemos que a base de los ciclos de vida se crearon las metodologías. Brevemente se detallan algunos de los ciclos de desarrollo de software de las más utilizados:

Ciclo de desarrollo:

- **Modelo cascada** esta metodología consiste en la ejecución de etapas unas tras otras de arriba hacia abajo esto quiere antes de pasar a la siguiente fase esta requiere que la primera fase haya concluido para proseguir con la siguiente etapa

Ventajas

- Nos permite la supervisión del avance del proyecto
- El costo se puede atribuir al inicio del proyecto

Desventajas

- Presenta dificultad en caso que se dé un cambio en el desarrollo del mismo
- Excluye al usuario final.



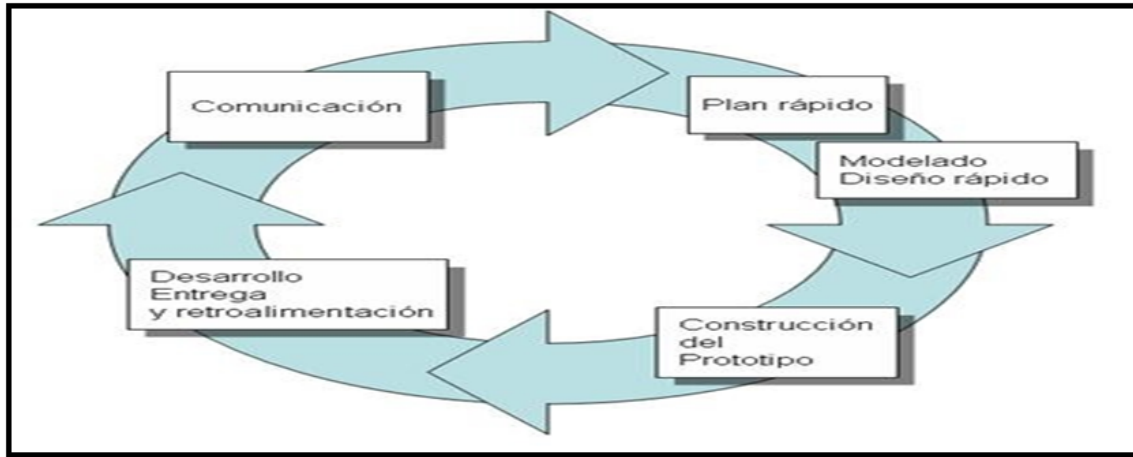
- **Modelo evolutivo** el progreso de este modelo está en una mejora continua desde el inicio de su implementación acorde a las exigencias del usuario final

Ventajas

- Cuenta con la colaboración del cliente a diferencia del modelo anterior
- A comparación de la modelo cascada este se adapta a los requerimientos del usuario final.

Desventajas

- Frecuentemente el sistema sufre fallas por la integración de cambios lo que dificulta el desarrollo del producto.
- Los requerimientos no son fijos desde el inicio del proyecto.



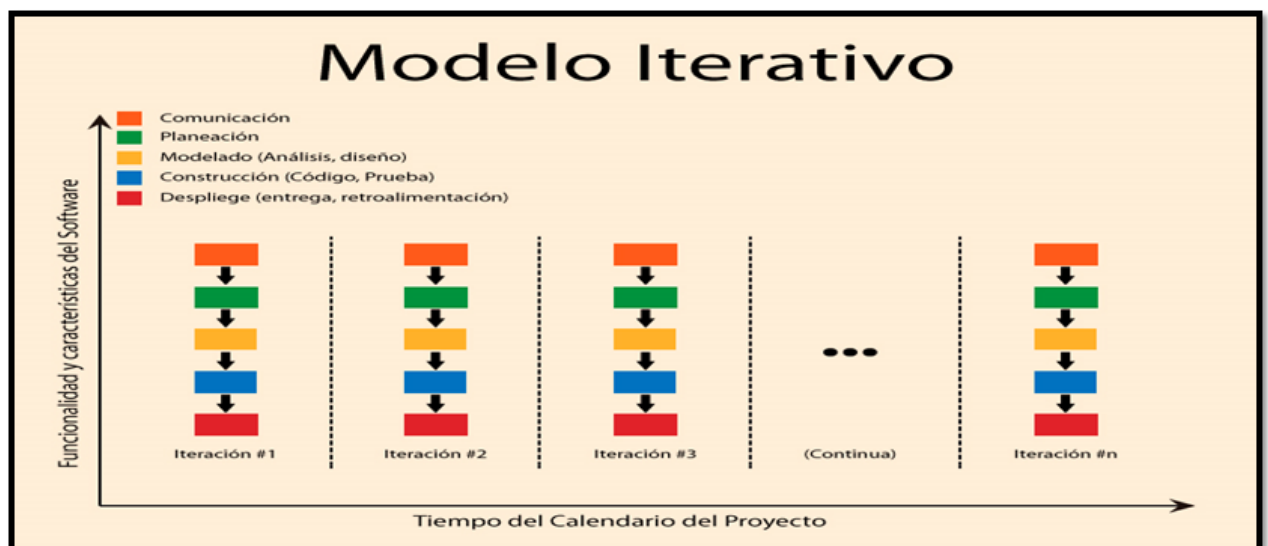
- **Modelo de desarrollo iterativo** este modelo se desarrolló en base a la deficiencia de la modelo cascada con el fin de dar una reducción de las irregularidades del producto de esta manera entregar un producto de calidad.

Ventajas

- Favorece las mejoras del sistema sin que este esté en proceso de culminación
- Los riesgos se reducen a través de la iteración

Desventajas

- No es el más apropiado para un proyecto más corto
- Limita el tema de documentación.



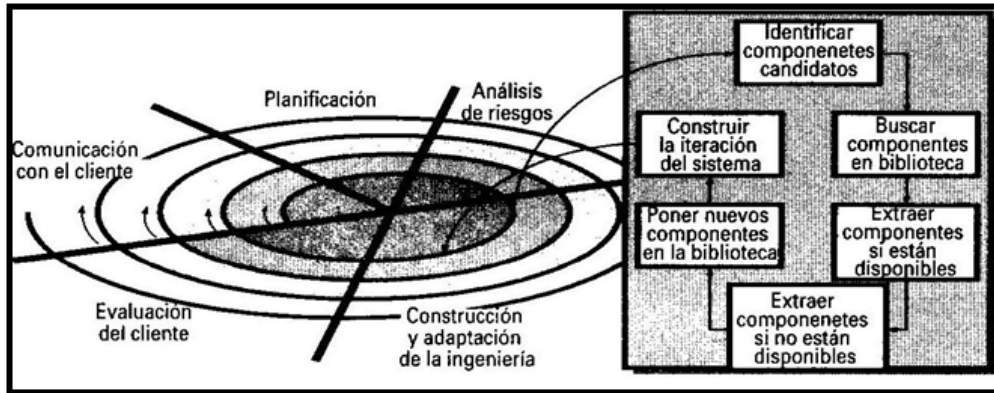
- **Modelo basado en componentes** se basa en la reutilización del software la cual aporta efectividad en la implementación de un sistema

Ventajas

- agiliza las pruebas a ejecutar
- aporta una mayor calidad al producto

Desventajas

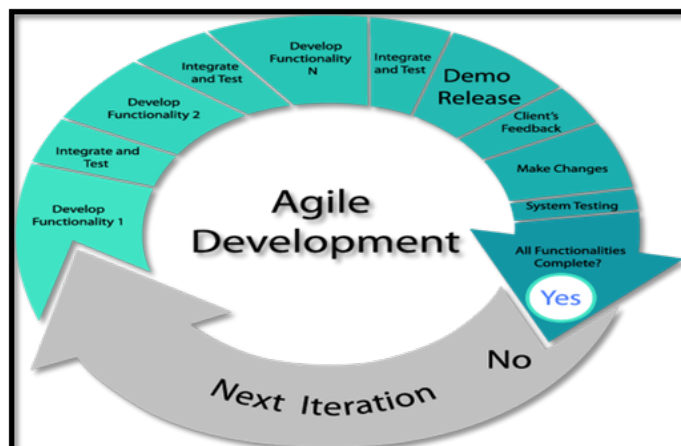
- solicita experiencia en la gestión de riesgo



- **Modelo de desarrollo ágil** nos ayuda en la generación de proceso más eficaces en la solución de problemas del sistema.

Mencionamos los 12 principios del manifiesto ágil:

1. Satisfacer al cliente
2. Cambios, incluso en las etapas tardías
3. Software funcionando
4. Los responsables de negocio y desarrollo trabajan en conjunto
5. Motivación
6. Comunicación cara a cara
7. Software como medida
8. Desarrollo sostenible
9. Excelencia técnica
10. Simplicidad
11. Equipos auto-organizados
12. Intervalos regulares



Ventajas

- se alcanzan los resultados deseados más rápido.
- Se trabaja en conjunto

Desventajas

- Limitación en la documentación del proyecto

Importancia

Actualmente existen diferentes ciclos de desarrollo de software, la normativa ISO/IEC/IEEE 12207:2017 establece: *“Un marco común para los procesos del ciclo de vida de los programas informáticos, con una terminología bien definida, a la que pueda remitirse la industria del software. Contiene procesos, actividades y tareas aplicables durante la adquisición, el suministro, el desarrollo, el funcionamiento, el mantenimiento o la eliminación de sistemas, productos y servicios informáticos. Estos procesos del ciclo de vida se llevan a cabo mediante la participación de los interesados, con el objetivo final de lograr la satisfacción del cliente”*.

Los ciclos de vida nos permiten la captación de fallos, dar garantía en la calidad del software. Según (Pressman) [1] y [2] la calidad del software es *“la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente”*.

Generalmente se reutiliza parte de un proyecto ya realizados esto nos brinda ventaja para proyecto futuros por ultimo podemos dar mención a renombrados autores:

- Ingeniería de software es el estudio de los principios y metodologías para el desarrollo y mantenimiento de sistemas software (Zelkovitz, 1978)
- Ingeniería de software trata del establecimiento de los principios y métodos de la ingeniería a fin de obtener software de modo rentable, que sea fiable y trabaje en máquinas reales (Bauer, 1972)
- Es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software; es decir, la aplicación de la ingeniería al software (IEEE, 1993).

Gracias a investigaciones realizadas por estas personas y otras con la realización de esta investigación de esta manera resaltado los conceptos dados por cada uno de ellos.

1.5. Antecedentes

La tecnología a pasar de los años ha ido evolucionando para hacer de nuestro diario vivir algo más simple y brindando un apoyo adicional en los años 40 se da inicio a los primeros ordenadores y así sucesivamente los programas que estas requieren para su funcionamiento.

Con anterioridad se definieron varios modelos de proceso de desarrollo de software, sus características, ventajas y desventajas de cada uno de ellos dando a conocer que estos modelos siguen una serie de etapas que reúnen lo esencial para asegurar el éxito en la implementación del software, cabe resaltar un modelo de proceso de desarrollo cada uno de ellos va adecuado al tipo de proyecto que se vaya a ejecutar.

2. Materiales y Métodos

2.1. Procedimiento

La función básica es la de informar y aportar una descripción de lo planificado, ya que a través de una intensiva lectura e investigación avanzamos en la recopilación de los datos que tomamos como fuente de información para que exista una mejor comprensión del tema.

2.2. Otros aspectos metodológicos

Responsable de su ejecución tomamos en cuenta, los actores del sistema, la mensajería intercambia entre los objetos que sean utilizados para escribir, escenarios y su notación es sencilla de entender, por lo cual no hay mucho que explicar.

Los modelos de comportamiento sirven también como aspectos metodológicos para explicar los modelos de procesos y los ciclos de vida de desarrollo del software. Tomando en cuenta la ejecución los cambios de los objetos que son parte de cada uno de los eventos que disparan generan dichas transiciones.

2.3. Aspectos éticos

Los progresos científicos y tecnológicos que día a día aportan el conocimiento de modernos procedimientos y avances en la medicina, por una parte, conllevan un sinnúmero de beneficios para las personas sanas o enfermas; pero, por otra, surgen de las investigaciones que se realizan en seres humanos diversos dilemas éticos, como producto, en determinadas ocasiones, del no cumplimiento de normas, códigos o reglamentación en la investigación experimental, si bien es cierto la experimentación en seres humanos tiene como propósito contribuir al mejoramiento de procedimientos diagnósticos y profilácticos y además la comprensión de la etiología y patogenia de una enfermedad, no es menos cierto que estos protocolos deben pasar por un cauteloso filtro

que proteja los derechos de los probando y, en especial, de aquellas poblaciones vulnerables donde pudiera ser violentada la dignidad de la persona humana; las normas éticas han expresado que los experimentos deben hacerse primariamente en animales y, según los efectos por analogía, se podrían emplear en seres humanos. (Paul Ehrlich) plantea la tesis que no es suficiente el principio de analogía para conocer cuál es el comportamiento de los fármacos en los seres humanos, sino que habría que ensayar en seres humanos para conocer las propiedades y el comportamiento farmacológico de los productos terapéuticos. Bradford Hill definió el método del ensayo clínico como *“un experimento cuidadoso y éticamente diseñado, con el fin de poder contestar a preguntas concretas formuladas previamente”*.

3. Resultados

Basándonos en información aportada por diferentes autores que se dedicaron y se dedican a ofrecer aportes en este tema tenemos como resultado que los proyectos requieren su manera para la implementación adecuada de un software todos los puntos resaltados alrededor de la investigación se relacionan con el fin de ofrecer alternativas de productos de calidad adoptando el modelo adecuado para su implementación gracias a esto se ha desarrollado diferentes tipos de software con la utilización de estas metodologías que se crearon a partir de brinda una solución más rápida y eficaz.

El análisis de los modelos para el proceso de desarrollo del software dio contribución que los modelos son actividades que están relacionadas con la especificación del software como el diseño, investigación y el desarrollo para la elaboración de pruebas que evidencien la calidad del software y la implementación del producto que este caso es un software y la cual persona dedican a la ingeniera de software cumplen su rol dentro d estos procesos.

4. Conclusiones

El desarrollo de software se ha convertido en una faceta importantes para la sociedad en donde la inteligencia artificial y la tecnología constituyen una parte fundamental de nuestras vidas, desarrollar software seguro de alta calidad será cada vez más importante.

Los modelos de desarrollo de software más empleados concluimos que principalmente se debe evaluar la magnitud del proyecto así elegir el modelo que mejor se adecue al proyecto agregando detalles que debemos tomar en cuenta como el tiempo pactado para la entrega del sistema culminado al usuario final, los requerimientos propuestos por el usuario entre otras características que se deben analizar para un producto de calidad.

Por otro lado, se destaca que la ingeniería de software forma parte en cada uno de los ciclos de vida ya que esta va orientada a resultados eficientes y de calidad.

Agradecimiento

Nuestro agradecimiento va dirigido a la universidad UMECIT nuestra casa de estudio por brindamos conocimiento a través del docente Luis Bernuil responsables de inducirnos a este artículo de investigación incitarnos al área de investigación para promover nuestro conocimiento, análisis que se van formado a través de estas actividades de investigación.

Referencias Bibliográficas

- Sáez, P., Rodríguez, V., Villanueva, J. & Cueto, M. (2014). Selección de Modelos y Metodologías Ágiles en Proyectos de Software. Paper presentado en el 18th International Congress on Project Management and Engineering. Alcañiz, España. Recuperado de: http://www.aepro.com/files/congresos/2014alcaniz/CIDIP2014_1862_1873.4302.pdf
- Amador Duran Toro, Beatriz Bernárdez Jiménez, “Metodología para el análisis de requisitos de sistemas de software versión 2.2”, Universidad de Sevilla, departamento de Lenguajes y Sistemas informáticos, escuela Técnica superior de Ingeniería Informática, diciembre de 2001.
- Amaya, Y. (2013). Metodologías ágiles en el Estado actual. Revista Tecnología. Vol. 12, No. 2, 111-124.
- Cadavid, A., Fernández, D. & Morales, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. Revista Prospectiva. Vol. 11, No. 2, 30-39.
- Cadavid, A., Fernández, D. & Morales, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. Revista Prospectiva. Vol. 11, No. 2, 30-39.
- Pressman, R. S. “Ingeniería del Software: Un Enfoque Práctico”. 6ª Edición. McGraw-Hill. 2006.
- Manuales UNBD. Ciclo de vida de software. Recuperado de <http://dis.unal.edu.co/grupos/unbd/manuales/ciclo/cap6.htm>
- Rubby Casallas, Andrés Yie. Ciclos de vida y metodología. Recuperado de
- <http://sistemas.uniandes.edu.co/~isis2603/dokuwiki/lib/exe/fetch.php?media=principal:isis2603-modelosciclosdevida.pdf>
- Pressman, 2005. Ingeniería del Software: Un Enfoque Práctico. 6ª Edición. McGraw-Hill, 2005.
- Luis E. Mendoza/Maria A. Pérez/Anna C. Griman, Prototipo de Modelo sistémico de calidad (MOSCA) del software Computación y Sistemas, vol. 8, Num. 3, pp.196-217. Año 2005. Instituto Politécnico Nacional. http://redalyc.uaemex.mx/redalyc/pdf/615/6158_0304.pdf.