

MODELOS Y METODOLOGIAS DE DESARROLLO DE SOFTWARE

Un modelo es la representación simplificada de los procesos para el desarrollo de software, no se tiene claramente cuál es el mejor pero si se sabe que se pueden utilizar diferentes modelos para producir un mismo tipo de sistema de software.

Las características de un proceso de desarrollo de software son:

- Claridad: Entender con facilidad.
- Visibilidad: Cuando producen resultados claros y de fácil identificación.
- Facilidad de soporte: Herramientas disponibles para dar soporte a las actividades que se desarrollan.
- Fiabilidad: El proceso es fiable cuando se detectan posibles errores.
- Facilidad de mantenimiento: Capacidad para modificar o actualizar.
- Rapidez: Reducir los tiempos de procesos.

Modelo en cascada o convencional

Es el primer modelo de desarrollo de software. Usado en la industria por su facilidad de gestión y visibilidad.

El principal problema es por su poca flexibilidad al separar el proceso de desarrollo en etapas distintas. Otros problemas son la dificultad para realizar prototipos, reutilizar software y realizar pruebas sin implementar el sistema.

Modelo evolutivo

Se entrelazan las actividades de las especificaciones, desarrollo y validación. Se desarrolla rápidamente un sistema inicial de una especificación muy abstracta. El sistema se va refinando con la información que suministra los clientes y/o usuarios que satisfaga las necesidades previstas.



Universidad Nacional Abierta y a Distancia – UNAD -

Escuela: Ciencias Básicas Tecnología e Ingeniería

Curso: Diseños de Sitios Web

Vicerrectoría Académica y de Investigación - VIACI

Programa: Ingeniería de Sistemas

Código: 301122

Existen dos tipos de procesos de desarrollo evolutivo:

Exploratorio: La principal función es trabajar con el cliente para identificar y construir el sistema final a partir de una especificación que se le informa.

Prototipado desechable: El otro proceso es entender los requisitos del cliente.


Los problemas principales son: escasa visibilidad; cambios continuos en los sistemas desarrollados estructurados; la necesidad de disponer, un equipo de desarrollo calificado. Esto hace que la aplicación suela limitar los sistemas interactivos de tamaños pequeños o medianos. Dificulta las tareas de mantenimiento que suele aplicar sistemas con una vida corta y partes de grandes sistemas, especialmente a sistemas de inteligencia artificial y a interfaces del usuario.

Modelo transformacional

Requiere de una especificación formal del sistema en transformar, mediante métodos matemáticos. Si las transformaciones son correctas se aplican y es posible asegurar que el sistema construido satisface las especificaciones.

Una de sus ventajas es la posibilidad de realizarle mantenimiento a nivel específico. Por lo cual se debe disponer de una especificación correcta y diseñadores altamente calificados. Este modelo se basa en la reutilización haciendo que algunos de los componentes del sistema final ya existen. El proceso de desarrollo se centra en integrar las partes ya existentes más que en construir todo el sistema desde el principio.

Las ventajas de este método es que desde el punto de vista económico se puede producir este modelo y empieza ser estudiado con profundidad. Casi no se tiene experiencia con este modelo.





Universidad Nacional Abierta y a Distancia – UNAD -

Escuela: Ciencias Básicas Tecnología e Ingeniería

Curso: Diseños de Sitios Web

Vicerrectoría Académica y de Investigación - VIACI


Programa: Ingeniería de Sistemas

Código: 301122

Modelo en espiral

Desde el año de 1988 Boehm, tuvo como objetivo el de reunir ventajas de los modelos de procesos de software en cascada y prototipado. Incluye en este el análisis de riesgo como parte importante.

El modelo tiene la forma de una espiral en la que cada vuelta representa las fases de la estructura del proceso de software y está organizado por:

1. Definición de objetivos, alternativas y restricciones de cada fase del proyecto.
 2. Evaluación de alternativas y análisis de riesgos.
 3. Desarrollo y validación. Se elige el modelo de proceso de desarrollo que se considere más adecuado.
 4. Planificación de las siguientes fases del proyecto.
- 

METODOLOGÍAS

La metodología para el desarrollo de software es un modo sistemático de realizar, gestionar y administrar un proyecto para llevarlo a cabo con altas posibilidades de éxito. Esta sistematización nos indica cómo dividiremos un gran proyecto en módulos más pequeños llamados etapas, y las acciones que corresponden en cada una de ellas, nos ayuda a definir entradas y salidas para cada una de las etapas y, sobre todo, normaliza el modo en que administraremos el proyecto.

En conclusión una metodología para el desarrollo de software son los procesos a seguir sistemáticamente para idear, implementar y mantener un producto software desde que surge la necesidad del producto hasta que cumplimos el objetivo por el cual fue creado.

Desde un punto de vista general puede considerarse que el ciclo de vida de un software tiene tres etapas claramente diferenciadas, las cuales se detallan a continuación:

- **Planificación:** idearemos un planeamiento detallado que guíe la gestión del proyecto, temporal y económicamente.
- **Implementación:** acordaremos el conjunto de actividades que componen la realización del producto.
- **Puesta en producción:** nuestro proyecto entra en la etapa de definición, allí donde se lo presentamos al cliente o usuario final, sabiendo que funciona correctamente y responde a los requerimientos solicitados en su momento.

Esta etapa es muy importante no sólo por representar la aceptación o no del proyecto por parte del cliente o usuario final sino por las múltiples dificultades que suele presentar en la práctica, alargándose excesivamente y provocando costos no previstos.



Universidad Nacional Abierta y a Distancia – UNAD -

Escuela: Ciencias Básicas Tecnología e Ingeniería

Curso: Diseños de Sitios Web

Vicerrectoría Académica y de Investigación - VIACI

Programa: Ingeniería de Sistemas

Código: 301122

CLASIFICACIÓN DE LAS METODOLOGÍAS

Existen dos metodologías que tienen analogía en la práctica con los paradigmas de programación. Metodología estructurada y metodología orientada a objetos.

- **Metodología estructurada:** la orientación de esta metodología se dirige hacia los procesos que intervienen en el sistema a desarrollar, es decir, cada función a realizar por el sistema se descompone en pequeños módulos individuales. Es más fácil resolver problemas pequeños, y luego unir cada una de las soluciones, que abordar un problema grande.

- **Metodología orientada a objetos:** a diferencia de la metodología mencionada anteriormente, ésta no comprende los procesos como funciones sino que arma módulos basados en componentes, es decir, cada componente es independiente del otro. Esto nos permite que el código sea reutilizable. Es más fácil de mantener porque los cambios están localizados en cada uno de estos componentes.



BIBLIOGRAFÍA

Israel Alcázar, M. A. (03 de Junio de 2014). *Introducción a Git y Github*. Obtenido de Desarrolloweb.com:
<http://www.desarrolloweb.com/articulos/introduccion-git-github.html>

UNAD. (Agosto de 2016). *Curso de Diseño de Sitios WEB*. Obtenido de <http://campus0b.unad.edu.co/campus0/>

