[Cuadro comparativo](http://iswescom.blogspot.com/2013/02/cuadro-comparativo.html)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Modelo en cascada | Modelo en espiral | Modelo incremental | Modelo DRA (Desarrollo Rápido de Aplicaciones) | XP (Xtreme Programming) |
| ¿Qué es? | Es el enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la inmediatamente anterior. | Consiste en una serie de ciclos que se repiten en forma de espiral, comenzando desde el centro. El Espiral puede verse como un modelo evolutivo que conjuga la naturaleza iterativa del modelo MCP. con los aspectos controlados y sistemáticos del modelo cascada . | El incremental es un modelo de tipo evolutivo que está basado en varios ciclos Cascada realimentados aplicados repetidamente, con una filosofía iterativa | Es un modelo de proceso de desarrollo de software lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto. | Es una metodología ligera de desarrollo de software que se basa en la simplicidad, la comunicación, y realimentacion del código desarrollado. fue desarrollado por kent beck. |
| Fases del modelo | 1. Análisis de requerimientos: Contiene la especificación completa de lo que debe hacer el sistema sin entrar en detalles internos. 2. Diseño del Sistema Contiene la descripción de la estructura relacional global del sistema y la especificación de lo que debe hacer cada una de sus partes, así como la manera en que se combinan unas con otras. 3. Diseño del Programa Es la fase en donde se realizan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario. 4. Codificación 5. Pruebas 6. Implantación | 1. establecer la comunicación entre el cliente y el desarrollador. 2. definición de los recursos, tiempo y otra información relacionada con el proyecto. 3. evaluar los riesgos técnicos y de gestión del proyecto. 4. construir una o más representaciones de la aplicación software. 5. construir la aplicación, instalarla, probarla y proporcionar soporte al usuario o cliente 6. obtener la reacción del cliente, según la evaluación de lo creado e instalado en los ciclos anteriores. | Dentro de modelo incremental podemos encontrar el modelo DRA. | 1.Modelado de gestión: flujo de información entre las funciones de gestión responde las siguientes preguntas: ¿que información conduce al proceso de gestión?, ¿A dónde va la información?, ¿Quién la procesa?   2.-modelado de datos:flujo de información definido como parte de la fase del modelado de gestión se refina como un conjunto de objetos y datos necesarios para apoyar la empresa.  3.modelado de procesos: los objetos de datos definidos en la fase de modelado quedan transformados para lograr el fin deseado. | 1. planificación del proyecto.definir las historias de usuario con el cliente, las historias de usuario tienen la misma finalidad que los casos de uso, pero con algunas diferencias, constan de 3 o 4 lineas escritas por el cliente en un lenguaje no técnico, sin profundizar mucho en los detalles.  2. diseño.   3. codificación. Debe hacerse atendiendo estándares de codificación ya creados, para facilitar su comprensión y escalabilidad.   4 pruebas. Uso de test para comprobar el funcionamiento de los códigos que se van implementando. |
| Ventajas | Se tiene todo bien organizado y no se mezclan las fases. Es perfecto para proyectos que son rígidos, y además donde se especifiquen muy bien los requerimientos y se conozca muy bien la herramienta a utilizar | Reduce riesgos del proyecto Incorpora objetivos de calidad. Integra el desarrollo con el mantenimiento, etc. Además es posible tener en cuenta mejoras y nuevos requerimientos sin romper con la metodología, ya que este ciclo de vida no es rígido ni estático. | Se reduce el tiempo de desarrollo inicial, ya que se implementa la funcionalidad parcial. proporciona todas las ventajas del modelo en cascada realimentado, reduciendo sus desventajas sólo al ámbito de cada incremento. más rápido en comparación del modelo de cascada. Resulta más sencillo acomodar cambios al acotar el tamaño de los incrementos. | Permiten que los ingenieros de sw desarrollen versiones cada vez más completas del sw. Producen una versión completa en forma incremental con cada iteración | Programación organizada menor taza de errores satisfacción del programador |
| Desventajas | Un proyecto rara vez sigue una secuencia lineal, esto crea una mala implementación del modelo, lo cual hace que lo lleve al fracaso. El proceso de creación del software tarda mucho tiempo ya que debe pasar por el proceso de prueba y hasta que el software no esté completo no se opera. | Genera mucho tiempo en el desarrollo del sistema Modelo costoso Requiere experiencia en la identificación de riesgos. | El modelo Incremental no es recomendable para casos de sistemas de tiempo real, de alto nivel de seguridad, de procesamiento distribuido, y/o de alto índice de riesgos. Requiere de mucha planeación, tanto administrativa como técnica. Requiere de metas claras para conocer el estado del proyecto. | Para proyectos grandes, necesita suficientes recursos humanos para crear el número correcto de equipos DRA Si los desarrolladores y clientes no se comprometen con las actividades rápidas necesarias para completar un sistema en un marco de tiempo muy breve, los proyectos fallarán. Si un sistema no se puede modular en forma apropiada, la construcción de los componentes necesarios será problemática Inapropiado cuando los riesgos técnicos son altos… cuando se aplican muchas nuevas tecnologías | Es recomendable emplearlo solo en proyectos a corto plazo. Altas comisiones en caso de fallar. |
| Usos | El modelo en cascada se despeña bien en proyectos con requisitos claros o cuando se trabaja con herramientas técnicas y es des aconsejable cuando se necesita un rápido desarrollo. | El modelo en espiral es beneficioso en proyectos que necesitan reducción de riesgos. | El modelo incremental es útil sobre todo cuando el personal necesario para una implementación completa no esta disponible. | El modelo DRA es utilizado para ciclos de vida del software cortos. | Es utilizado para la creación y desarrollo practico de software, es utilizado mucho últimamente ya que es una metodología ágil para el desarrollo. |

Publicado 21st February 2013 por [Sanctus](http://www.blogger.com/profile/01026434786076508546)

FUENTE: <http://iswescom.blogspot.mx/2013/02/cuadro-comparativo.html#!/2013/02/cuadro-comparativo.html>

Principio del formulario



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **MODELO EN CASADA** | **MODELO ESPIRAL** | **MODELO INCREMENTAL** | **MODELO DRA**  **(desarrollo rápido de aplicaciones)** | **metodologia XP (extreming programming)** |
| QUE ES | es el enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del software, de tal forma que el inicio de cada etapa debe esperar a la finalización de la inmediatamente anterior. | Consiste en una serie de ciclos que se repiten en forma de espiral, comenzando desde el centro. El Espiral puede verse como un modelo evolutivo que conjuga la naturaleza iterativa del modelo MCP. con los aspectos controlados y sistemáticos del modelo cascada . | El incremental es un modelo de tipo evolutivo que está basado en varios ciclos Cascada realimentados aplicados repetidamente, con una filosofía iterativa | Es un modelo de proceso de desarrollo de software lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto. | es una metodología ligera de desarrollo de software que se basa en la simplicidad, la comunicación, y realimentacion del código desarrollado. fue desarrollado por kent beck. |
| FASES DEL MODELO | **1. Análisis de requerimientos:**  Contiene la especificación completa de lo que debe hacer el sistema sin entrar en detalles internos.  **2. Diseño del Sistema**  Contiene la descripción de la estructura relacional global del sistema y la especificación de lo que debe hacer cada una de sus partes, así como la manera en que se combinan unas con otras.  **3. Diseño del Programa**  Es la fase en donde se realizan los algoritmos necesarios para el cumplimiento de los requerimientos del usuario.  **4. Codificación**  **5. Pruebas**  **6. Implantación** | 1. establecer la comunicación entre el cliente y el desarrollador.  2. definición de los recursos, tiempo y otra información relacionada con el proyecto.  3. evaluar los riesgos técnicos y de gestión del proyecto.  4. construir una o más *representaciones* de la aplicación software.  5. construir la aplicación, instalarla, probarla y proporcionar soporte al usuario o cliente  6. obtener la reacción del cliente, según la evaluación de lo creado e instalado en los ciclos anteriores. | Dentro de modelo incremental podemos encotrar el modelo **DRA.** | **1.Modelado de gestión:** flujo de información entre las funciones de gestión responde las siguientes preguntas: ¿que información conduce al proceso de gestión?, ¿A dónde va la información?, ¿Quién la procesa?  **2.modelado de datos:**flujo de información definido como parte de la fase del modelado de gestión se refina como un conjunto de objetos y datos necesarios para apoyar la empresa. 3.**modelado de procesos:** los objetos de datos definidos en la fase de modelado quedan transformados para lograr el fin deseado. | **1. planificacion del proyecto.**definir las historias de usuario con el cliente, las historias de usuario tienen la misma finalidad que los casos de uso, pero con algunas diferencias, constan de 3 o 4 lineas escritas por el cliente en un lenguaje no tecnico, sin profundizar mucho en los detalles **2. diseño. 3. codificacion.**Debe hacerse atendiendo estándares de codificación ya creados, para facilitar su comprensión y escalabilidad. **4 pruebas**.Uso de test para comprobar el funcionamiento de los códigos que sevan implementando. |
| VENTAJAS | Se tiene todo bien organizado y no se mezclan las fases.  Es perfecto para proyectos que son rígidos, y además donde se especifiquen muy bien los requerimientos y se conozca muy bien la herramienta a utilizar | Reduce riesgos del proyecto  Incorpora objetivos de calidad.  Integra el desarrollo con el mantenimiento, etc.  Además es posible tener en cuenta mejoras y nuevos requerimientos sin romper con la metodología, ya que este ciclo de vida no es rígido ni estático. | se reduce el tiempo de desarrollo inicial, ya que se implementa la funcionalidad parcial.  proporciona todas las ventajas del modelo en cascada realimentado, reduciendo sus desventajas sólo al ámbito de cada incremento.  más rápido en comparación del modelo de cascada.  Resulta más sencilo acomodar cambios al acotar el tamaño de los incrementos. | permiten que los ingenieros de sw desarrollen versiones cada vez más completas del sw.  Producen una versión completa en forma incremental con cada iteración | - programacion organizada  - menor taza de errores - - satisfaccion del programador |
| DESVENTAJAS | un proyecto rara vez sigue una secuencia lineal, esto crea una mala implementación del modelo, lo cual hace que lo lleve al fracaso.  El proceso de creación del software tarda mucho tiempo ya que debe pasar por el proceso de prueba y hasta que el software no esté completo no se opera. | Genera mucho tiempo en el desarrollo del sistema  Modelo costoso  Requiere experiencia en la identificación de riesgos. | El modelo Incremental no es recomendable para casos de sistemas de tiempo real, de alto nivel de seguridad, de procesamiento distribuido, y/o de alto índice de riesgos.  Requere de mucha planeacion, tanto administrativa como técnica.  Requiere de metas claras para conocer el estado del proyecto. | Para proyectos grandes, necesita suficientes recursos humanos para crear el número correcto de equipos DRA Si los desarrolladores y clientes no se comprometen con las actividades rápidas necesarias para completar un sistema en un marco de tiempo muy breve, los proyectos fallarán. Si un sistema no se puede modular en forma apropiada, la construcción de los componentes necesarios será problemática inapropiado cuando los riesgos técnicos son altos… cuando se aplican muchas nuevas tecnologías | -es recomendable emplearlo solo en proyectos a corto plazo.  -altas comisiones en caso de fallar. |
| USOS | el modelo en cascada se despeña bien en proyectos con requisitos claros o cuando se trabaja con herramientas técnicas y es des aconsejable cuando se necesita un rápido desarrollo. | el modelo en espiral es beneficioso en proyectos que necesitan reduccion de riesgos . | el modelo incremental es útil sobre todo cuando el personal necesario para una implementacion completa no esta disponible. | el modelo DRA es utilizado para ciclos de vida del software cortos. | es utilizado parala creacion y desarrollopractico de software, es utilizado mucho ultimamente ya que es una metodologia agilparael desarrollo. |

**CONCLUSIONES:**

* La Ingeniería de software establece y utiliza una serie de modelos que muestran las distintas etapas y estados por lo que pasa un producto de software, desde su concepción inicial, pasando por su desarrollo, puesta en marcha y posterior mantenimiento, hasta la retirada del producto.
* **‍con el modelo en espiral es posible tener en cuenta mejoras y nuevos requerimientos sin romper con la metodología, ya que este ciclo de vida no es rígido ni estático por eso lo contemplamos como el mas apropiado para desarrollar nuestro proyecto WINTEXTIL.**

* **‍la ingenieria de software es una diciplina que nos afrece tecnicas y metodos para desrrollar software de calidad en el cual nos da uno modelos o paragigmas que nos ayudad a la realizacion de software ademas nos da unos procesos y metodoligias con el fin de mejorar la calidad de el mismo.**

FUENTE: <http://integrantes1.wikispaces.com/CUADRO+COMPARATIVO>

## Unidad 1: La Ingeniería de Software

## Modelos de Programación Exploratoria

## [<<Indice](http://ceidis.ula.ve/cursos/pgcomp/isoo/modelosind.html)

## *Consisten en desarrollar una versión preliminar del sistema que cumpla con lo siguiente:*

 es puesta en operación tan rápido como sea posible;

 evoluciona a través de su uso hasta alcanzar un nivel de satisfacción adecuado.

 sólo es posible definir los requerimientos del sistema en la medida que éste se utiliza.

###### Parten de una especificación muy somera y superficial de los requerimientos del sistema:

 la especificación no puede ser empleada para la verificación debido a su imprecisión.

 la validación es subjetiva y consiste en demostrar la adecuación del sistema al usuario.

#### El modelo de desarrollo en fases o programación exploratoria [Pfleeger, 1998]:

