

Tarea.

Investigación de metodologías

de Software.

Universidad Autónoma de la Ciudad de México, Campus Cuautepec

Leonardo Valdes Palafox

Grupo 1102

Matrícula: 18-011-0249

Introducción a la Ingeniería de Software

Profesor: Gerardo Hernández Hernández.

Fecha máxima de entrega: 10/Marzo/2020.

Introducción.

Televisa Presenta:

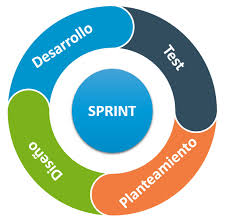
Dentro de este trabajo se busca el estudio y entendimiento de las metodologías para el desarrollo de software, así como la apreciación de sus elementos y la comprensión de sus manifiestos y procesos durante el desarrollo de un sistema de software.

Pasando por la investigación de procesos como:

* Extreme Programing.
* Agile Unified Process.
* Agile Modeling.

Sin olvidar el manifiesto del desarrollo de software y sus 12 elementos.

Metodologías Ágiles de Desarrollo de Software.

Históricamente, el desarrollo de software ha sido una tarea tediosa y compleja, donde muchas veces la rentabilidad de la misma estaba muy por debajo de su coste. Eran épocas de desarrollos interminables, de infinidad de líneas de código donde cualquier cambio en el proyecto inicial suponía un auténtico quebradero de cabeza para el desarrollador. La poca (o nula) tolerancia y/o previsión a cambios, los métodos arcaicos de programación, etc. no contribuían a generar un ecosistema adecuado para el correcto desarrollo de software.

Con el paso de los años, fueron surgiendo nuevos lenguajes de programación (en teoría más sencillos), entornos más amigables, frameworks que facilitaban muchas tareas, conceptos que las empresas dedicadas al desarrollo de software acogieron rápidamente para mejorar la calidad y los plazos de entrega de sus proyectos.

Repasando un poco la historia, en 2001, un grupo de expertos, a petición del ingeniero Kent Beck (creador de las metodologías XP y TDD), se reunieron para debatir la situación del desarrollo de software y acuñaron la expresión “metodologías ágiles”. La expresión hacía referencia a un marco de trabajo en el cual era posible aumentar la calidad del desarrollo, reducir los tiempos, eliminar la incertidumbre, aceptar la tolerancia a cambios… en definitiva, incrementar la satisfacción del cliente.

Los principios del manifiesto ágil de software.

1. Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.

2. Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.

3. Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.

4. Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.

5. Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.

6. El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.

7. El software funcionando es la medida principal de progreso.

8. Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.

9. La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.

10. La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.

11. Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.

12. A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.

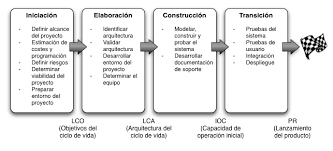
Programación extrema (Extreme programing).

Es una metodología de desarrollo de la ingeniería de software formulada por Kent Beck. Es el más destacado de los procesos ágiles de desarrollo de software. Al igual que éstos, la programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. Los defensores de la XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. Creen que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos.

Modelado ágil (Agile modeling).

Envuelve un enfoque para la toma de decisiones en los proyectos de software, que se refiere a métodos de ingeniería del software basados en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan con el tiempo según la necesidad del proyecto. Así el trabajo es realizado mediante la colaboración de equipos autoorganizados y multidisciplinarios, inmersos en un proceso compartido de toma de decisiones a corto plazo.

Proceso ágil unificado (Agile unified Process).

El Proceso Unificado Ágil de Scott Ambler es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP. El AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas (test driven development - TDD), Modelado Ágil, Gestión de Cambios Ágil, y Refactorización de Base de Datos para mejorar la productividad.

Conclusión.

Dirección: Enrique Segoviano.

El estudio de las metodologías para el desarrollo de software es extenso y digno de apreciación. Así como sus imponentes 12 elementos del manifiesto del desarrollo de un sistema de software.

Por lo cual, se concluye con este estudio de las metodologías antes mencionadas dentro del cuerpo del escrito.

Bibliografía.

<https://softwaretheory.wordpress.com/2014/11/17/20/>

<http://www.develapps.com/es/noticias/metodologias-agiles-en-el-desarrollo-de-software>

<https://es.wikipedia.org/wiki/Desarrollo_%C3%A1gil_de_software>