

Tema 1: Fundamentos en la construcción del software

Docente : Héctor Xavier Limón Riaño
email: xavier120@hotmail.com

Introducción

¿Qué es la construcción?

Introducción

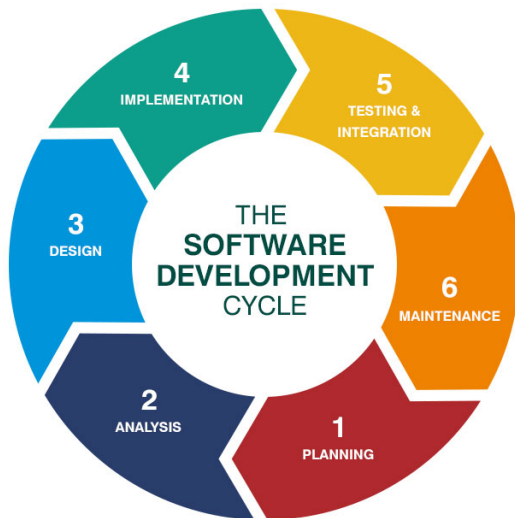
La construcción implica aspectos de:

- ▶ Planificación
- ▶ Diseño
- ▶ Control
- ▶ Creación

¿Qué es la construcción de software?

- ▶ El desarrollo de software es una tarea compleja que requiere un proceso
- ▶ En los últimos 35 años investigadores han identificado muchas actividades referentes al desarrollo de software:
 - ▶ Definición del problema
 - ▶ Desarrollo de requerimientos
 - ▶ Planificación de construcción
 - ▶ Arquitectura de software o diseño de alto nivel
 - ▶ Diseño detallado
 - ▶ Codificación y depuración
 - ▶ Prueba de unidad, integración y sistema
 - ▶ Integración
 - ▶ Mantenimiento correctivo

¿Qué es la construcción de software?



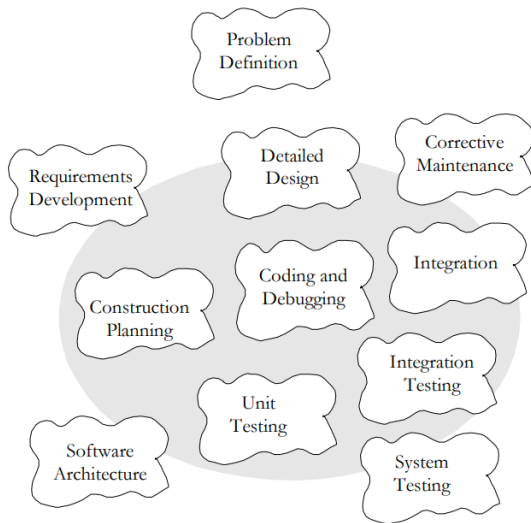
¿Qué es la construcción de software?

- ▶ La construcción de software es una disciplina de ingeniería de software
- ▶ Es la creación detallada de software significativo y funcional a través de la combinación de codificación, verificación, pruebas de unidad, pruebas de integración y depuración
- ▶ Está ligada a todas las demás disciplinas de ingeniería de software, más fuertemente a diseño y prueba de software

¿Qué es la construcción de software?

- ▶ Principalmente se enfoca a aquellas tareas que tienen que ver con la creación:
 - ▶ Codificación
 - ▶ Depuración
 - ▶ Planificación de la construcción
 - ▶ Pruebas de unidad
- ▶ Aunque también impacta en las demás tareas

¿Qué es la construcción de software?



¿Qué es la construcción de software?

- ▶ A la construcción se le suele llamar "Programar" o "Implementar"
- ▶ Algunas tareas específicas de construcción incluyen:
 - ▶ Determinar cómo probar el código
 - ▶ Diseñar e implementar clases y rutinas
 - ▶ Crear y nombrar variables y constantes
 - ▶ Seleccionar estructuras de control y organizar bloques de sentencias
 - ▶ Hacer pruebas de unidad e integración y depurar el código
 - ▶ Pulir código dándole formato y agregando comentarios
 - ▶ Integrar componentes de código que fueron creado de forma separada
 - ▶ Optimizar el código para hacerlo más compacto y eficiente

¿Porqué es la construcción de software importante

- ▶ La construcción es un gran parte del desarrollo de software. De 30% a 80% del tiempo de desarrollo total
- ▶ La construcción es la actividad central del desarrollo de software
- ▶ Con un enfoque de construcción, la productividad del programador individual puede mejorar mucho
- ▶ El código fuente es a menudo la única descripción totalmente acertada del software. Es imperativo que el código sea de la mayor calidad posible
- ▶ La construcción es la única actividad que se garantiza realizar

Minimizar complejidad

- ▶ La necesidad de reducir complejidad se deriva del hecho de que la mayoría de personas cuentan con una habilidad limitada para mantener en su memoria de trabajo estructuras e información compleja
- ▶ La reducción de complejidad se consigue enfatizando la creación de código que es simple y legible más que ingenioso

Minimizar complejidad

- ▶ La disminución de complejidad también se consigue utilizando estándares y técnicas de codificación
- ▶ También es soportada a través de técnicas enfocadas a la calidad de construcción

Anticipar cambios

- ▶ Anticipar cambios le ayuda a los ingenieros de software construir software extensible
- ▶ Esto significa mejorar el producto de software sin alterar la estructura central
- ▶ Investigaciones de 25 años muestran que el costo de retrabajar software puede ser de 10 a 100 veces más costoso que obtener los requerimientos de forma adecuada la primera vez

Anticipar cambios

- ▶ Dado que el 25% de los requerimientos cambian durante el desarrollo en promedio, la necesidad de reducir los costos de retrabajar muestran la necesidad de anticiparse a los cambios
- ▶ Esta anticipación se traduce a cuidar que el software pueda extenderse y modificarse con facilidad y debe ser un requerimiento de diseño

Construir para la verificación

- ▶ Es refiere a construir software de tal forma que los fallos puedan ser desprendidos por los ingenieros de software que lo escribieron, durante pruebas individuales o actividades operacionales
- ▶ Técnicas específicas que soportan la construcción de software para verificación incluyen aunque no están limitadas:
 - ▶ Seguir estándares de codificación para dar soporte a revisiones de código
 - ▶ Pruebas de unidad
 - ▶ Organizar código para dar soporte a pruebas automatizadas
 - ▶ Restringir el uso de estructuras complejas o difíciles de entender en el código

Reuso

- ▶ El reuso sistemático puede permitir un incremento significativo de la productividad, calidad, y reducción de costos
- ▶ El reuso tiene dos facetas cercanamente ligadas:
 - ▶ Construcción para el reuso: crear elementos de software reusables
 - ▶ Construcción con reuso: reusar elementos de software en la construcción de una solución nueva

Estándares en la construcción

- ▶ Los estándares pueden ser externos (creados por organizaciones internacionales) o internos (creados a nivel corporación)
- ▶ Los estándares afectan directamente los siguientes aspectos de construcción:
 - ▶ Métodos de comunicación: tales como estándares para formatos de documentos y contenido
 - ▶ Lenguajes de programación
 - ▶ Estándares de codificación
 - ▶ Plataformas
 - ▶ Herramientas: tales como estándares para la notación de diagramas como UML