

Unidad I

Introducción a la ingeniería del software y sistemas de información

Las economías de todos los países son cada vez más y más dependientes del Software Importancia del Software 10

Cada vez más y más sistemas están controlados por software El gasto en desarrollo de software está aumentando su porcentaje en el PIB de todos los países

1.1. Conceptos de Ingeniería del Software: mitos, paradigma, ingeniería de software, calidad, proceso, método, herramienta, espectro de gestión.

MITO: "Mito del cliente":

- Mito.- "Una declaración superficial de los objetivos es suficiente para empezar a escribir los programas".

- Realidad.- La mala definición inicial es la principal causa de baja calidad. Se requiere un conocimiento formal y detallado de los hechos y procesos y amplia comunicación con el cliente.

Mitos de los Desarrolladores.

- Mito.- "Lo único que se entrega al terminar el proyecto es el programa funcionando".
- Realidad.- El software funcionando es solo una parte de una CONFIGURACION DE SOFTWARE.

La documentación es la base de un buen desarrollo y guías para las tareas de mantenimiento.

Paradigma: La ingeniería de software surge de la ingeniería de sistemas y de hardware.

Abarca un conjunto de tres elementos que facilitan el control sobre el proceso de desarrollo de software y suministran las bases para construir software de calidad de una forma productiva:

- Métodos
- Herramientas
- Procedimientos

Métodos que indican cómo construir el software técnicamente e incluyen un amplio espectro de métodos para la planificación, la estimación, el análisis, el diseño, codificación, prueba y mantenimiento.

Herramientas automáticas y semiautomáticas que apoyan a la aplicación de los métodos.

Cuando se integran las herramientas de forma que la información creada por una herramienta puede ser usada por otra, se establece un sistema para el soporte del desarrollo de software, llamado Ingeniería de Software Asistida por Computadora (CASE).

Procedimientos que definen la secuencia en la que se aplican los métodos, las entregas, los controles de calidad y guías para evaluación del progreso.



Ingeniería del software: Disciplina que establece el uso de principios de ingeniería robustos, orientados a obtener software económico, que sea confiable y funcione de manera eficiente.

Calidad: Algunas características de calidad fundamentales en todo producto de programación son : utilidad, claridad, confiabilidad, eficiencia y economía.

•Proceso:Conjunto de actividades que conducen a la creación de un producto de software.

Depende de personas que toman decisiones y juicios.

•No existe proceso ideal

- Para los sistemas criticos se requiere un proceso de desarrollo muy estructurado
- Para los sistemas de negocio con requerimientos rapidamente cambiantes,un proceso flexible y agil probablemente sea mas efectivo.

Metodo: Estructurado para el desarrollo de software, facilita la produccion de software de alta calidad de una forma costeable.No existe un metodo ideal.Metodos se basan en la idea de modelos graficos.Dependen de personas que toman decisiones y juicios.

Herramienta: En los cursos de ingeniería de software se utilizan varias herramientas de desarrollo y gestión para mejorar la producción de software. Estas cubren distintas actividades del ciclo de desarrollo: requerimientos, diseño, construcción, pruebas, SQA, SCM.

Cuando se integran las herramientas de forma que la información creada por una herramienta puede ser usada por otra, se establece un sistema para el soporte del desarrollo de software, llamado Ingeniería de Software Asistida por Computadora (CASE).



El espectro de la gestion: La gestión eficaz de un proyecto de software se centra en las cuatro P's: personal, producto, proceso y proyecto. El orden no es arbitrario.

1.2. La importancia de la ingeniería del software.

Producir software costeable es esencial para el funcionamiento de la economía nacional e internacional.

Este es abstracto e intangible. No esta restringido por materiales, o gobernado por leyes físicas o por procesos de manufactura. Esto simplifica la ingeniería de

software ya que no existen limitaciones físicas del potencial del software.

Sin embargo, esta falta de restricciones naturales significa que el software puede llegar a ser extremadamente complejo.

Hemos desarrollado métodos efectivos de especificación, diseño e implementación del software. Las nuevas notaciones y herramientas reducen el esfuerzo requerido para producir sistemas grandes y complejos.

Los ingenieros de software pueden estar orgullosos de sus logros. Sin software complejo no habríamos explorado el espacio, no tendríamos Internet y Telecomunicaciones modernas, y todas las formas de viajar serían más peligrosas y caras. Dicha ingeniería ha hecho enormes contribuciones en su corto periodo de vida.

1.3. Historia de la Ingeniería del Software.

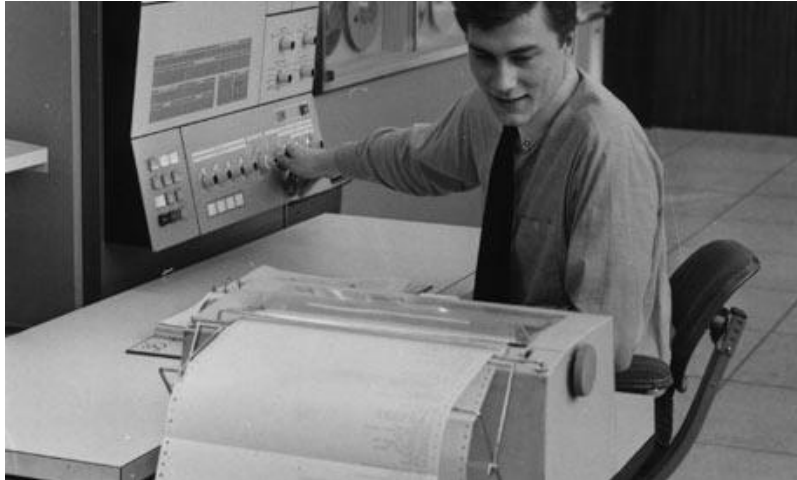
Desde sus inicios en la década de 1940, escribir software ha evolucionado hasta convertirse en una profesión que se ocupa de cómo crear software y maximizar su calidad. Surgimiento como una profesión: A principios de los 1980, la ingeniería del software ya habían surgido como una genuina profesión, para estar al lado de las ciencias de la computación y la ingeniería tradicional.

El papel de la mujer: en las década de los años 1940, 1950 y 1960, a menudo los hombres llenaron los roles más prestigiosos y mejor pagados en la ingeniería de hardware, pero a menudo delegaron la escritura de software a las mujeres. Grace Murray Hopper, Jamie Fenton y muchas otras mujeres anónimas llenaban

muchos trabajos de programación durante las primeras décadas de la ingeniería de software.

Costo de hardware: el costo relativo del software versus el hardware ha cambiado sustancialmente en los últimos 50 años. Cuando los mainframes eran costosos y requerían una gran cantidad de personal de soporte, las pocas organizaciones que los compraban también tuvieron los recursos para financiar proyectos de ingeniería de software a la medida, grandes y costosos.

El mercado más grande puede soportar grandes proyectos para crear software comercialmente, como los hechos por empresas como Microsoft. Las máquinas baratas permiten a cada programador tener un terminal capaz de una compilación bastante rápida.



1.4. **Los sistemas de información: concepto, características, estructuras, procesos, clasificación, ERP´s, CRM, SCM.**

Concepto: Un sistema de información es un conjunto de recursos humanos, materiales, financieros, tecnológicos, normativos y metodológicos, organizado para brindar, a quienes operan y a quienes adoptan decisiones en una organización, la información que requieren para desarrollar sus respectivas funciones.

Un sistema de información no requiere necesariamente el uso de la tecnología de computación. Ha habido sistemas de información antes de que se crearan las computadoras.

Características: Variedad en la presentación, disponibilidad de información, información selectiva, tiempo de respuesta, generalidad, seguridad, exactitud, flexibilidad, Amigabilidad.

Estructuras: Usando esta orientación de producto, un sistema de información se puede representar como una estructura jerárquica de cuatro niveles:

- NIVEL 1 – representación general del sistema (el producto).
- NIVEL 2 – representa los subsistemas contenidos dentro del sistema. Cada subsistema es un proceso del negocio para recoger, almacenar y recuperar datos dentro de un período de tiempo específico (por demanda).
- NIVEL 3 – representa los procedimientos necesarios para implementar cada subsistema
- NIVEL 4 – representa los pasos necesarios para poner cada procedimiento en ejecución.

Procesos: Conjunto estructurado de actividades requeridas para desarrollar un sistema de software.

Especificación- que debe hacer el software y cuales son sus especificaciones de desarrollo.

Desarrollo – producción del sistema de software.

Validación – verificar que el software hace lo que el cliente pide.

Evolución – cambiar/adaptar el software a las demandas.

Confiable: Los errores del proceso son descubiertos antes de que se conviertan en errores del producto

Robusto: Puede continuar el proceso a pesar de problemas inesperados

Mantenible: Puede el proceso evolucionar para cumplir con los objetivos organizacionales

Rapidez: Que tan rápido puede producirse el sistema ?.

Entendible: Se encuentra el proceso bien definido y es entendible ?.

Visible: El proceso es visible al exterior ?.

Soportable: Puede el proceso ser soportado por herramientas CASE ?.

Aceptable: El proceso es aceptado por aquellos involucrados en el ?.

Clasificación: Esta función consiste en identificar los datos, agruparlos en conjuntos homogéneos, y ordenarlos teniendo en cuenta la manera en que será necesario recuperarlos.

El almacenamiento de datos en archivos computadorizados dispone de técnicas que han permitido alcanzar un elevado nivel de refinamiento en este sentido.



ERP: corresponden a Enterprise Resource Planning (Planificación de Recursos Empresariales) constan de una serie de módulos que se pueden adquirir o no, en función de las necesidades exactas que tengamos en nuestra organización.

Suele ser habitual que estos módulos estén personalizados por sector productivo.

Excepto en el caso en que optemos por un desarrollo a medida interno del ERP (que no suele ser muy frecuente), el ERP será un producto estándar cuya adaptación a cada organización se realizará en un proceso conocido como parametrización.

ERP

PLANIFICACION DE RECURSOS EMPRESARIALES

CRM: Las siglas CRM, se trata de la gestión de las relaciones con los clientes, Customer Relationship Management. Todo proceso de una empresa debería estar enfocado a la venta de un determinado producto.

De esta forma, el concepto CRM define una estrategia de organización enfocada a la satisfacción de las necesidades del cliente.



SCM: Software Configuration Management (SCM) ó Gestión de configuración de software es una especialización de la Gestión de configuración a todas las actividades en el sector del desarrollo de software.

Conjunto de actividades tendientes a :

- Identificar un cambio.
- Controlar un cambio.
- Asegurar la correcta implementación.
- Comunicar los cambios realizados.

¿Por que es necesario usar SCM? :

- Cambios en el negocio dictan cambios en requisitos.
- Nuevas necesidades de información para funcionalidades existentes.

- Reducción o ampliación del negocio provoca cambios en prioridades o integrantes del proyecto.

- Cambios presupuestarios imponen redefinir en producto/proyecto.

Se puede clasificar en:

- Código fuente
- Datos
- Documentación

Debe proporcionar información sobre:

- Tipo de elemento
- Proyecto al que pertenece
- Versión
- Código fuente
- Datos
- Documentación

Líneas de base:

- Punto de inicio para la evolución controlada del software.
- Solo se incorporan productos del trabajo luego de la revisión y aprobación de los mismos.
- Puede cambiarse solamente a través de procedimientos formales.
- Generalmente constituye un milestone.

La Planificación de SCM Define:

- Marco de desarrollo del proceso.
- Alcance.
- Actores.
- Responsabilidades.
- Estándares.
- Herramientas.

Proceso SCM, define tareas con objetivos concretos:

- Identificar elementos de la configuración.
- Controlar los cambios.
- Controlar las versiones.