

Que es software?

Conjunto de programas, procedimientos, datos y documentacion que operan sobre un sistema de computo. (IEEE)

Tipos de software:

Genericos: Son organizaciones que se dedican a la produccion de software de proposito general digamos, no estan optimizados ni customizados para un cliente en particular

Personalizados: Software generado para una cliente/empresa por un contratista u organización que se decica al tema.

Atributos de un buen Soft:

(MA-CO-EFI-USA)

Evolucion del software:

1950-60: Hard de proposito general. Con continuos cambios, procesamiento batch.

Soft a medida, sin planificacion. "DOCUMENTACION INEXISTENTE".

60-70: Multiusuario, multiprogramacion. Lenguajes de programacion. Base de datos, tiempo real Aparece el "producto de software". "MANTENIMIENTO DE SOFT"

70-90: Microprocesadores, redes de datos, soft en todas las areas, baja considerable en el hardware. "PRODUCTIVIDAD".

90-2000: Tecnologias O.O, procesamiento paralelo, procesamiento ilimitado, capacidad de aprendizaje del usuario es un limite. "INGENIERIA".

2000-act: Aplicaciones web, sistemas expertos, Redes neuronales, Reconocimiento de patrones. Codigo abierto. "COMUNICACION Y DISTRIBUCION".

Que es Ing. De Soft.

Disciplina de la ingenieria que se encarga de **todos los aspectos de la produccion de soft.**

Disciplina de la ing.

-hace que las cosas funcionen

-se aplican te-me-he.

todos los aspectos de la produccion: no abarca solo los aspectos tecnico del desarrollo. tambien se realizan actividades, como GP, desarrollo de metodos y herramientas y teorias.

IEEE: aplicación de metodos "**sistematicos**" y "**cuantificables**" para el desarrollo, operación y mantenimiento. (DOM)

Richard Fairley: Disciplina tecnologica de administracion para la produccion y evolucion(**PE**) del software que incluye el desarrollo y las modificaciones en un tiempo y costo estimados.

Metodos sistematicos "Cuantificables"

La cuantificacion de recursos, productos y procesos, es muy importante para optimizar la productividad y calidad.

dentro de "tiempos y costos estimados"

Un ingeniero de soft debe cumplir con los tiempos de entrega y con los costos que se arreglaron, por eso debe contar con aptitudes como la medicion, estimacion, planificacion.

para el "desarrollo, operación y mantenimiento"

IG se ocupa de todo el ciclo de vida de un producto, desde etapas iniciales hasta de como debe ser retirado de servicio.

Como nacio la IGS.

La ingenieria de soft nacio ante las dificultades de producir software ante las capacidad y experiencias de los desarrolladores.

La aplicación de metodos "**sistematicos**" permite la produccion y evolucion(**PE**) por empresas distribuidas geograficamente y en el tiempo. La aplicación de estandares facilita la (**INREMA**) de

los productos.

En resumen: La IGS trata de dar **(PM)** para producir software confiable y eficiente al menor costo posible.

Para esto se desarrollan teorías, métodos y herramientas automáticas o semiautomáticas.

Evolución de la ingeniería de software

Era funcional 60: Se busca como explotar la tecnología para hacer frente a los problemas funcionales de las organizaciones.

Era de Control 70: Surge la necesidad de desarrollar soft de manera planificada y organizada. Aparece el modelo de ciclo de vida en fases

Era de Costos 80: La importancia de la productividad en el desarrollo aumenta sustancialmente. Aparecen los primeros modelos de costos.

Era de Calidad 90 a la actualidad: necesidad de que el producto desarrollado cumpla con las necesidades implícitas y explícitas del usuario: (MACOEFIUSA)

Que es un proceso de software:

Conjunto de actividades y resultados asociados que producen un producto de software.

Principales actividades:

- especificación del soft
- desarrollo del soft
- validación del soft
- evolución del soft.

Además de las actividades principales los procesos deben incluir:

Productos: son los resultados de llevar a cabo las actividades.

Roles: especifican el nivel de responsabilidad de todo aquel que participe en el proceso.

Pre condiciones y post condiciones: Son declaraciones válidas antes y después de llevar a cabo una actividad.

Características de cualquier proceso: (Al parecer no es lo mismo “proceso” que “proceso de software”).

- Establece todas las actividades.
- Utiliza recursos y está sujeto a restricciones, genera productos intermedios y finales.
- Puede estar compuesto por subprocesos.
- Cada actividad tiene entradas y salidas definidas.
- Las actividades se organizan en una secuencia.
- Hay principios que establecen las metas de cada actividad.
- Las restricciones pueden aplicarse a una actividad, recurso o producto.

Que es un modelo de proceso?

Es una representación simplificada de un proceso de soft, que presenta una visión de ese proceso.

Los modelos pueden incluir actividades que son parte de los productos y procesos y el papel de las personas que están involucradas.

La mayoría de los modelos de procesos se basan en uno de los siguientes modelos generales:

Modelo en cascada: Se define el sistema en un conjunto de fases.

Modelo iterativo: Se desarrolla rápidamente un sistema a partir de especificaciones abstractas. Luego se refina con las necesidades de los clientes.

IS basada en componentes: Se basa en que el sistema ya está hecho y en partes, y que solo falta

la unificación de esas partes.

Modelos de procesos:

- Modelo en cascada.
- Modelo en V.
- Modelo de prototipos.
- Desarrollo por fases.
 - Incremental.
 - Iterativo.
- Modelo en espiral.

Modelo en Cascada:

- Las etapas se representan cayendo en cascadas.
- Cada etapa se tiene que completar antes de que empiece la otra.
- Util para acordarse lo que hay q hacer.
- Util para explicárselo a los clientes.

Dificultades:

- No existen resultados hasta que no este todo terminado.
- Las fallas mas triviales se encuentran al inicio mientras que las mas importantes al final.
- Eliminar las fallas es muy difícil en las ultimas etapas.
- deriva de un mundo de hardware y presenta una vision de manufactura sobre el desarrollo de soft
- La necesidad de pruebas aumenta en las ultimas etapas.
- "congelar" es una etapa poco realista.
- Existen errores, cambio de parecer, cambios en el ambiente.

Modelo en V:

- Muestra como se relacionan las actividades de prueba con las de analisis y diseño.
- Sugiere que la prueba unitaria y de integracion tmb se aplique al diseño.
- La vinculacion entre los lados derecho e izquierdo hace que, si se encuentran errores durante la validacion y la verificacion el lado izq. de la V se puede volver a ejecutar.

Modelo de Prototipos:

- Un prototipo es un producto parcialmente desarrollado que permite a clientes y desarrolladores examinen algunos aspectos y decidan si este es correcto para seguir con producto.
- Es una alternativa para tratar la incertidumbre.

Tipos:

Evolutivos:

- El objetivo es obtener el sistema a entregar.
- Permite que todo el sistema o parte de el se desarrolle rapidamente para ayudar a comprender aspectos tanto al desarrollador, cliente y usuarios de manera de unificar la solucion

Descartables:

- No tiene funcionalidad
- Se utilizan herramientas de modelado.

Proyectos candidatos:

- Usuarios que no examinaran los modelos abstractos.
- Usuarios que no definieron sus requerimientos inicialente.
- Sistemas que prestan mas atencion a la E/S mas que a los detalles algoritmicos.
- Sistemas donde haya que explorar aspectos tecnicos.
- Usuarios con problemas con los modelos graficos para modelar los requerimientos.
- Si se enfatiza el aspecto de la interfaz humana.

Desarrollo por fases:

Incremental:

El sistema es particionado en subsistemas y en cada entrega q se hace se agrega un nuevo

subsistema.

Iterativo: Se entrega todo el sistema desde un inicio y luego se sacan versiones nuevas con mejoras y cambios.

Modelo en Espiral:

- Combina actividades de desarrollo con la gestion del riesgo.
- Trata de mejorar los ciclo de vida clasicos y de prototipos.
- Incorpora objetivos de calidad y gestion de riesgos.
- Elimina errores al comienzo.
- Permite iteraciones, vuelta atraz y finalizaciones rapidas.
- Cada ciclo comienza identificando.
 - Objetivos
 - Alternativas
 - Restricciones
- Un ciclo se completa con la revision del ciclo anterior y un plan para el siguiente.

Metodologias agiles:

Es un enfoque iterativo e incremental(evolutivo) para el desarrollo de software.

Iterativo: El tiempo se particiona para hacer un revision de las partes del sistema.

Incremental: El sistema se va desarrollando en tiempos distintos o en velocidades distintas luego de integra todo.

Objetivos:

- Hacer software de calidad con tiempo y costo apropiados.
- Afianzar los valores que permiten a los equipos desarrollar software rapidamente y adaptandose a los cambios que puedan surgir durante el proyecto.
- Es una alternativa a los procesos de desarrollo de software tradicionales caracterizados por ser rigidos y con documentacion en cada una de sus etapas.

[Fowler]

Una metodologia agil es aquella a la que se le presta atencion a aquellas actividades que producen resultados directos y que reducen la burocracia lo maximo posible, adaptandose a los cambios.

La comunicaci3n es la principal fuente de error en lo que respecta a las necesidades de los clientes.

Se producen por:

- 1) falta de procedimientos o guias formales
- 2) usuarios poco involucrados
- 3) interpretaciones malas de las necesidades
- 4) poca comunicacion

Al hablar de necesidades estamos refiriendonos a requerimientos/requisitos.

Requerimientos: condicion o requisito que tiene que saber llevar a cabo un sistema con el objetivo de satisfacer un proposito.

(IEEE)

- Condicion que necesita el usuario para resolver un problema.
- Condicion que necesita un sistema o un subconjunto del sistema para llevar a cabo la solucion de un problema.

Elicitacion de Requisitos/Requerimientos

La elicitation es el proceso por el cual se trata de “sacar” “sonsacar” todo aquel dato relevante que me sirba a la hora de emplear un modelo de solución para la resolución de un problema.

Objetivos:

- 1) conocer el dominio del problema para así poder comunicarse con usuarios y clientes y entender sus necesidades.
- 2) ver como se están manejando actualmente, con qué sistema ya sea manual o informatizado
- 3) entender tanto las necesidades implícitas y explícitas de clientes y usuarios y ver que esperan ellos del sistema.

La elicitation de requisitos es una actividad de carácter social más que tecnológico, por lo tanto los problemas que surgen tienen un carácter más psicológico que técnico.

A la hora de la elicitation pueden surgir una serie de problemas relacionados con la comunicación.

- 1) Dificultad para expresar claramente las necesidades
- 2) No ser conciente de las necesidades
- 3) No entender como la tecnología ayudaría en sus problemas
- 4) Miedo a parecer incompetente por ignorancia a la tecnología
- 5) No tomar decisiones por no poder prever las consecuencias.
- 6) vocabulario diferentes
- 7) visiones distintas del problema
- 8) medios de comunicación inadecuados (diagramas que no entienden ni clientes ni usuarios)
- 9) conflictos personales o políticos.

Impacto de los errores en la etapa de requerimientos

- El software resultante puede no cumplir las necesidades de los usuarios
- Las interpretaciones múltiples de requerimientos pueden causar problemas entre clientes y desarrolladores.
- Puede gastarse tiempo y dinero construyendo un sistema erróneo.

TECNICAS DE ELICITACION

- 1) Muestreo de la información, las formas y datos existentes
- 2) visitas al lugar
- 3) observación del ambiente de trabajo
- 4) cuestionarios
- 5) entrevistas
- 6) planeación conjunta de requerimientos JRP o JAD
- 7) brainstorming

Planeación conjunta de requerimientos (JRP o JAD)

- 1) Proceso mediante el cual se realizan reuniones altamente estructuradas con el objetivo de entender el problema en cuestión.
- 2) Necesitan un extenso entrenamiento
- 3) amplia participación de los usuarios
- 4) se trabaja sobre lo que se va generando
- 5) reduce el tiempo de exploración de requisitos.

Ventajas:

- usuarios y clientes involucrados en el proyecto
- reduce el tiempo de obtención de requerimientos
- desarrollo creativo.

Desventajas:

- difícil hacer coincidir los horarios de todas las personas involucradas.

-difícil llevar a cabo una reunión con tanta gente involucrada

Participantes del JRP:

- 1) **patrocinador:** es el que toma las decisiones finales, está a cargo de todos los departamentos involucrados.
- 2) **facilitador:** Posee amplios conocimientos de comunicación y negociación.
- 3) **usuarios y gerentes:** los usuarios transmiten los requerimientos a los gerentes y estos los aprueban.
- 4) **secretarios:** llevan nota de la sesión y publican los resultados con herramientas CASE.
- 5) **Equipo de TI:** Toman nota de los requerimientos.

Beneficios de JRP:

- 1) Todas las partes involucradas en el proyecto
- 2) Reducción de tiempo en lo que es la etapa de requerimientos
- 3) Si se incorporan prototipos, los mismos ya confirman el diseño del sistema.

Lluvia de ideas(Brainstorming)

- Técnica para generar tantas ideas como sean posibles en un cierto tiempo sin ningún análisis hasta que se hayan agotado las ideas-
- Promueve el desarrollo creativo.
- Se llevan a cabo reuniones con el personal en cuestión dirigidas por un director.

Los principios en los que se basa esta técnica son:

- Mientras más ideas se generen mejor va a ser la solución al problema.
- produce ideas por contagio
- es mejor elegir sobre un gran abanico de ideas.
- las mejores ideas por lo general aparecen tarde.
- es mejor el aporte de ideas en grupo que individual.
- clave para resolver la falta de consenso entre usuarios
- útil para entender el dominio del problema
- útil para aquellos participantes con problemas para transmitir sus ideas
- ayuda a entender tanto a usuarios como a analistas.

-útil combinarlo con la toma de decisiones.

Ingeniería de Requerimientos:

La ing de requerimientos es una disciplina para desarrollar una especificación completa y consisa (**COMCON**) y no ambigua, que va a servir como punto de consenso entre todas las partes involucradas y donde se describen las funcionalidades del sistema.

Proceso donde los requerimientos ya sean hablados o escritos se transforman en especificaciones precisas, no ambiguas consistentes y completas (**COMCONTES**).

Proceso mediante el cual se intercambian diferentes puntos de vista para modelar lo que el sistema va a realizar. Este proceso combina métodos, herramientas y actores cuyo resultado es un modelo del cual se genera un documento de requerimientos.

Es un enfoque para recolectar, organizar y documentar los requerimientos del sistema, también para mantener acuerdos sobre los cambios de requerimientos entre clientes y equipos de desarrollo.

Importancia:

- Permite gestionar las necesidades del proyecto de forma estructurada.
- Ayuda con la capacidad de predecir cronogramas de proyectos.
- Disminuye costos y retrasos.

- Mejora la calidad del soft.
- Mejora la comunicación entre equipos.
- Evita rechazos de usuarios finales.

La ingeniería de requerimientos incluye:

- Estudio de viabilidad ----> informe de viabilidad.
- Obtencion y analisis de requerimientos ---> modelos del sistema.
- Especificacion de requerimientos ----> requerimientos del usuario del sistema.
- Validacion de requerimientos ---->



documento de requerimientos.

Estudio de viabilidad: Por lo general se aplica en sistemas nuevos, consiste en a partir de una descripción resumida del sistema se elabora un informe breve donde se recomienda si realizar o no el proyecto.

Responde a ciertas preguntas:

El sistema contribuye con los objetivos de la empresa?

--Si no contribuye, se aborta el proyecto.

Se puede implementar con la tecnología actual?

Se puede implementar con las restricciones de tiempo y costo?

El sistema puede integrarse con otros sistemas que ya existen?

Una vez que se respondieron las preguntas anteriores se debe elevar un informe que detalla si se sigue con el proyecto o no.

Obtencion y analisis de requerimientos:

Propiedades de los requerimientos:

Necesario: que su omision trae problemas

Consiso: facil de entender.

Completo: que no necesita ampliarse.

Consistente: **que no es contradictorio con otro.**

No ambiguo: que tiene una sola implementacion.

Verificable: que pueden testearse a partir de un conjunto de pruebas.

Tipos de requerimientos:

Funcionales:

-Describe la interaccion entre el sistema y el ambiente. Como debe reaccionar ante determinados estímulos

-Detallan especificamente lo que debe hacer el sistema, incluso como NO debe comportarse.

-describen con detalle la funcionalidad del mismo.

-Son independientes de la implementacion de la solucion.

-Pueden expresarse de distintas maneras.

No funcionales: Representan las limitaciones o restricciones del sistema y limita nuestras soluciones a la hora de la resolución del problema.

Especificacion de requerimientos:

-Permitir a los desarrolladores explicarle a los clientes como han entendido el problema.

-explicarle a los diseñadores que funcionalidades y características tendrá el sistema.

-Permitir que los encargados de las pruebas puedan explicarle a los clientes que el sistema cumple con las necesidades que el pidió.

Documento de definicion de requerimientos:

- Listado de las funcionalidades que el cliente espera que haga el sistema.

Documento de especificacion de requerimientos:

- Listado pero en terminos mas tecnicos.

Documento de especificacion de requerimientos de software (SRS)

Objetivo:

- Brindar buenas practicas a la hora de realizar una buena especificacion de requerimientos de software.
- Se describen los contenidos y las cualidades de una buena especificacion de requerimientos de software.

Aspectos basicos de una especificacion de requerimientos:

-Funcionalidad:

- Que debe hacer el sistema.

-Interfases externas:

- Como interactua con el mundo exterior.

-Rendimiento:

- disponibilidad, velocidad, etc.

-Atributos:

- portabilidad, seguridad, etc.

-Restricciones del diseño:

- Lenguaje de programacion, limite de recursos.

Validacion de requerimientos:

- Es el proceso de certificar la correccion del modelo con las intenciones del usuario.
- Trata de mostrar que el sistema lleva a cabo los requerimientos definidos.
- Es una etapa importante porque los errores de los requerimientos nos pueden costar muy caro mas adelante.

(IEEE)

Validacion: Al final del desarrollo evaluar el software para ver si este cumple con los requerimientos.

Verificacion: Determinar si un producto de software de una fase cumple con los requerimientos de la fase anterior.

Sobre estas definiciones:

- La validacion se puede hacer solo con la participacion del usuario.
- validacion: hacer software correcto
- verificacion: hacer software correctamente.

TECNICAS DE ESPECIFICACION DE REQUERIMIENTOS

Estaticas:

- Se describe al sistema a travez de sus entidades, atributos y relaciones con otros. No describe como las relaciones cambian en el tiempo.
- Cando el tiempo no es un factor fundamental es una algo util y adecuado.

Ej: relaciones de recurrencia, expresiones regulares, etc.

Dinamicas:

- Se considera un sistema en funcion a los cambios que puedan ocurrir con el tiempo.
- Se considera que el sistema esta en un estado particular hasta que un estimulo lo obliga a

cambiar de estado.

Ej: redes de petri, diagramas de transición de estados, etc.

Técnicas de especificación de requerimientos Dinámicas:

DTE:

Petri:

- Utilizada para especificar sistemas de tiempo real, donde se deben representar aspectos de concurrencia.
- Los sistemas concurrentes están diseñados para permitir la ejecución de muchos procesos o tareas en varios procesadores o en uno solo intercalados.
- Los sistemas concurrentes deben estar sincronizados para poder permitir una comunicación entre ellas.
- Pueden ejecutarse varias tareas en paralelo pero son ejecutadas en un orden impredecible.
- No son secuenciales.

CU:

Beneficios:

- Captura requerimientos funcionales.
- Descompone el sistema en piezas más manejables.
- Medio de comunicación con los usuarios.
- Permite estimar el alcance del proyecto y el esfuerzo a realizar.
- Define una línea base para los planes de prueba.
- Define una línea base para la documentación del sistema.
- Provee de una herramienta para el seguimiento de requisitos.

DFD:

Gestión de los Requerimientos:

¿Por qué cambian los requerimientos?

- Porque se hicieron las preguntas inadecuadas a las personas inadecuadas.
- Porque clientes y usuarios son distintos.
- Porque cambió el problema.
- Porque usuarios cambiaron su forma de pensar.
- Porque el negocio tomó otro rumbo, pegó un giro.

Evolución:

-Requerimientos duraderos: Se mantienen a lo largo del tiempo, son relativamente estables.

-Requerimientos volátiles: cambian durante el desarrollo del sistema, o después de que se puso en operación.

Requerimientos Volátiles: (CAEMCONCOM)

- Requerimientos cambiantes: cambian porque se modifica el ambiente.
- Requerimientos emergentes: surgen por una ampliación (cliente empieza a entender un poco más del problema).
- Requerimientos consecuentes: Surgen como consecuencia de la puesta en marcha del sistema.
- Requerimientos de compatibilidad: cambian porque interactúa con otros sistemas que cambiaron.

Calidad:

- Calidad es una palabra usada con mucha frecuencia en estos tiempos.
- Al hablar de bienes y/o servicios de calidad la gente se refiere a bienes de lujo a precios muy

caros.

-Su significado sigue siendo ambiguo y muchas veces depende de cada uno su significado.

Se consulto a 4 filosofos expertos en este ambito:

Crosby: Conformidad de los requerimientos.

Deming: Calidad en terminos de la satisfacion del cliente.

Feingenbaum: Calidad como una relacion directa entre productos y servicios, y necesidades del cliente.

Ishikawa: Los requerimientos y necesidades de los clientes van cambiando lo que nos lleva a una definicion cambiante.

Conclusion:

-Se puede observar que todos coinciden en "satisfaccion de los requerimientos" en base a las "necesidades de los cleintes" y la relacion entre estos conceptos.

-Pero la evaluacion de los mismos continua dependiendo de la evaluacion de sus características particulares.

-Lo mas importarte es definir las características que queremos evaluar y su forma de evaluacion.

Definiciones de Calidad:

-Capacidad de un producto para servir satisfactoriamente las necesidades del usuario.

-Que cumple con los requisitos implicitos y explicitos del cliente.

-Ausencia de errores.

La calidad se divide en:

Calidad del producto.

Calidad del proceso.

Son dependientes.

Calidad del producto: Un producto es de buena calidad si satisfase las necesidades del usuario para los objetivos para lo que fue consebido.

Calidad del proceso:

-Un preceso mal consebido va a generar como resultado productos de mala calidad.

-Un proceso bien consebudo va a generar tantas veces como sea posible productos de buena calidad.

Modelos de calidad:

Modelos de calidad de producto:

La familia ISO 9000 es un conjunto de normas de "gestión de la calidad" aplicables a cualquier tipo de organización con el objetivo de obtener mejoras en ella y, eventualmente arribar a una certificación, punto importante a la hora de competir en los mercados globales.

Modelos de calidad de proceso:

CMM (Modelo de Capacidad y Madurez):

Es un modelo de evaluación de los procesos de una organización. Fue desarrollado inicialmente para los procesos relativos al desarrollo e implementación de software.

Este modelo establece un conjunto de prácticas o procesos clave agrupados en Áreas Clave de Proceso. Para cada área de proceso define un conjunto de buenas prácticas.

A su vez estas Áreas de Proceso se agrupan en cinco "niveles de madurez":

1 – Inicial:

- Desempeño basado en la competencia del personal*

- Frecuentemente la organización vive “apagando incendios”
- Aparecen héroes
- Dificultad para encarar mejoras a largo plazo
- La organización actúa esencialmente por reacción

2 – Repetible:

- La organización.
 - Estableció la gestión de los proyectos de software, y está documentado.
 - Usa políticas organizacionales.
 - Repite prácticas exitosas desarrolladas en proyectos previos.
 - Información (recursos, tiempo, esfuerzo) compartida por medios informales.

3 – Definido:

- El proceso para la gestión y las actividades de ingeniería está documentado e integrado en un proceso estándar para la organización.
- Todos los proyectos usan una versión documentada y aprobada del proceso estándar de la organización.
- La información del proceso se halla estandarizada y es compartida a través de la BDD de procesos de software.

4 – Gestionado:

- La organización aplica los principios de la gestión estadística de procesos para controlar el proceso del software
- La dirección tiene bases objetivas para tomar decisiones
- Puede predecir el desempeño en un entorno cuantificado realista
- Usa los datos como base para decisiones, objetivos y mejoras

5 – Optimizado:

- La organización
 - identifica y elimina causas de desempeño pobre
 - mejora continua del proceso en base a gestión del cambio del proceso y de la tecnología

CMMI :

Posee dos vistas que permiten un enfoque diferente según las necesidades de quien vaya a implementarlo.

- Escalonado
 - centra su foco en la madurez de la organización. Igual que CMM.
- Continuo
 - Enfoca las actividades de mejora y evaluación en la capacidad de los diferentes procesos. Presenta 6 (seis) niveles de capacidad. Los niveles de capacidad indican qué tan bien se desempeña la organización en un área de proceso individual.

Representación continua:

