



Ingeniería de Software I

Conceptos de Ingeniería de Software

2

Software - Naturaleza



- Potencial de computo
- Transformador de información



- Base para control de cómputos
- Sistemas operativos
- Comunicación
- Creación y control de otros prog

3

Software

¿Qué es?

Instrucciones (programas de cómputo), procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación (IEEE)

IEEE: Instituto de Ingeniería Eléctrica y Electrónica

Características del Software

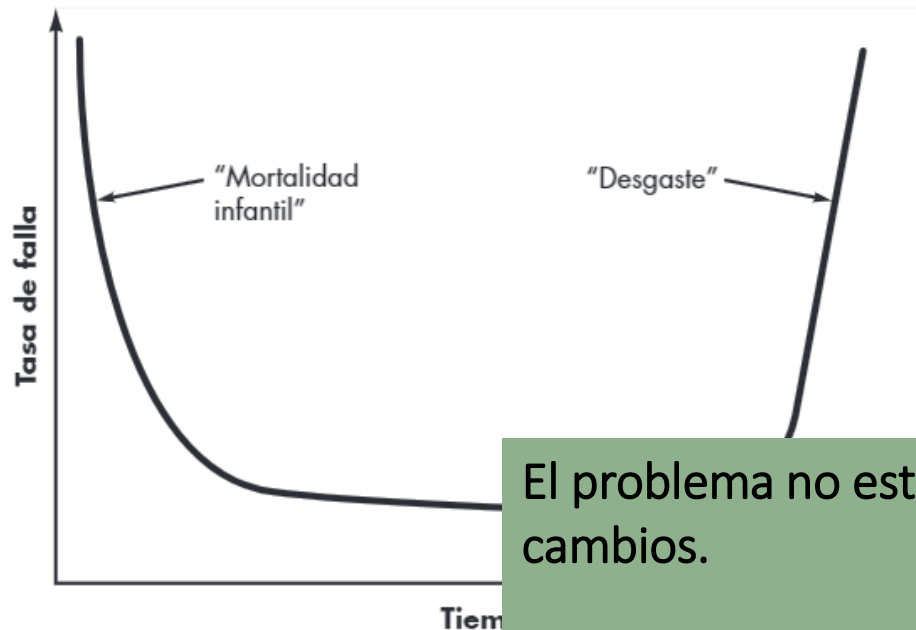


- ☐ elemento lógico.
- ☐ se desarrolla, no se fabrica
- ☐ no se desgasta

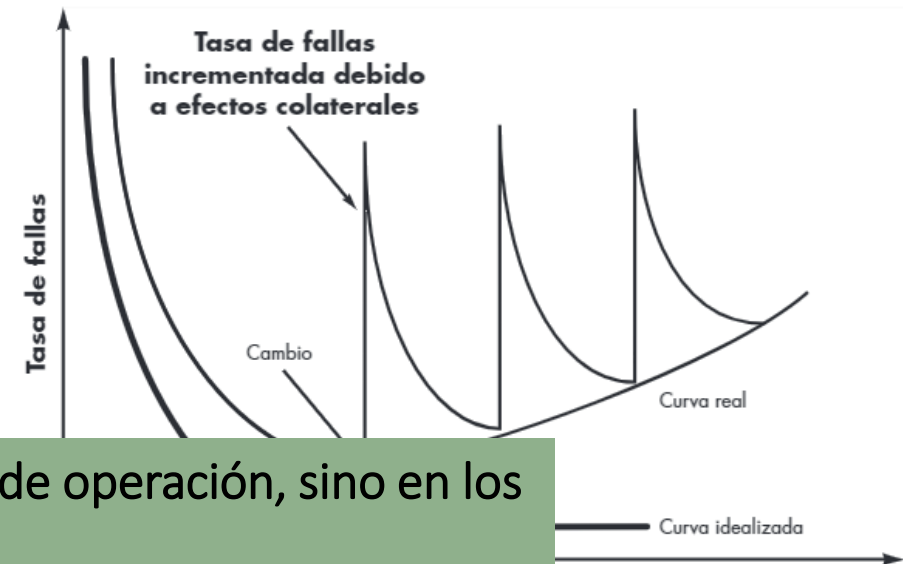
5

Características del Software

No sigue una curva clásica de envejecimiento.



Envejecimiento del hardware



Envejecimiento del software

El problema no está en el tiempo de operación, sino en los cambios.

6

Tipos de producto de software



Cada vez más sistemas son construidos por un producto genérico como base que luego es adaptado a los requerimientos del cliente.

ucidos por organizac
se venden en un m

idos por un

Clasificación del Software



8

Software

Nuevos retos




Computación Ubicua

World Wide Web

9

Software- Retos



Para hacer frente a estos retos

Ingeniería de Software

Enfrentar el futuro

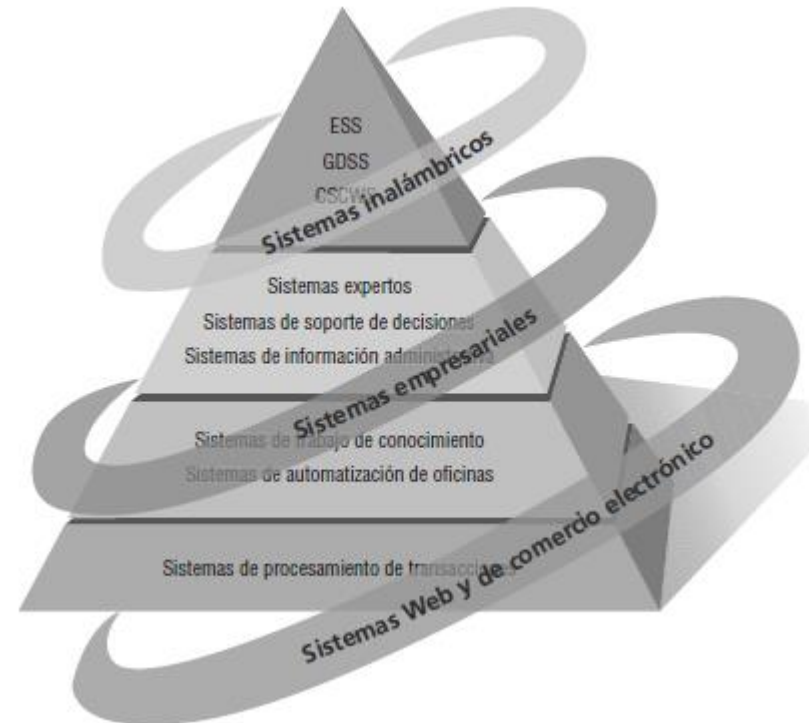
10

Software- Retos

- » A medida que los usuarios adoptan nuevas tecnologías, parte del trabajo consiste en integrar los sistemas tradicionales con los nuevos para asegurar un contexto útil.
- » Debemos estar conscientes de que al integrar tecnologías se ven afectados todos los tipos de usuarios y sistemas.

¿Qué hacer?

Ingeniería de Software



11

¿Qué es la Ingeniería de software?

» *Disciplina de la ingeniería que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema incluyendo la evolución de éste, luego que se comienza a ejecutar.*

Hace que las cosas funcionen.
Se aplican teorías, métodos y herramientas.

No sólo comprende los procesos técnicos del desarrollo de software, sino también se realizan actividades como la gestión de proyectos y el desarrollo de herramientas, métodos y teorías de apoyo a la producción de software

12

¿Qué es la Ingeniería de software?

» La IEEE define a la Ingeniería de Software como:

- 1. El uso de métodos sistemáticos, disciplinados y cuantificables para el desarrollo, operación y mantenimiento de software*
- 2. El estudio de técnicas relacionadas con 1*

13

¿Qué es la Ingeniería de software?

» Usa métodos sistemáticos cuantificables

» Dentro de tiempos y costos estimados”

» Para el “Desarrollo, operación y mantenimiento”

La cuantificación rigurosa de recursos, procesos y productos es una precondition para optimizar productividad y calidad. La “metrificación” y el control estadístico de procesos son claves en Ingeniería de Software

» Un Ingeniero de Software debe cumplir contratos en tiempo y costos como es normal en obras de Ingeniería. Ello presupone la capacidad de medir, estimar, planificar y administrar proyectos.

» La Ingeniería de Software se ocupa de todo el ciclo de vida de un producto, desde su etapa inicial de planificación y análisis de requerimientos hasta la estrategia para determinar cuándo y cómo debe ser retirado de servicio.

Fuente:



Facultad de
INFORMÁTICA
UNIVERSIDAD NACIONAL DE LA PLATA

Participantes en el Desarrollo del Software



15

Un poco de historia sobre la IS

»La Organización del Tratado del Atlántico Norte (OTAN) organizó un par de conferencias que tuvieron carácter fundacional para la *Ingeniería de Software* (Garmish 1968 y Roma 1969).

Propósito :

- ☐identificar la raíz de los problemas de la industria del software
- ☐Sentar las bases de procesos sistemáticos, repetibles y confiables.

Se comenzó a utilizar la expresión **“Ingeniería de Software”**.

16

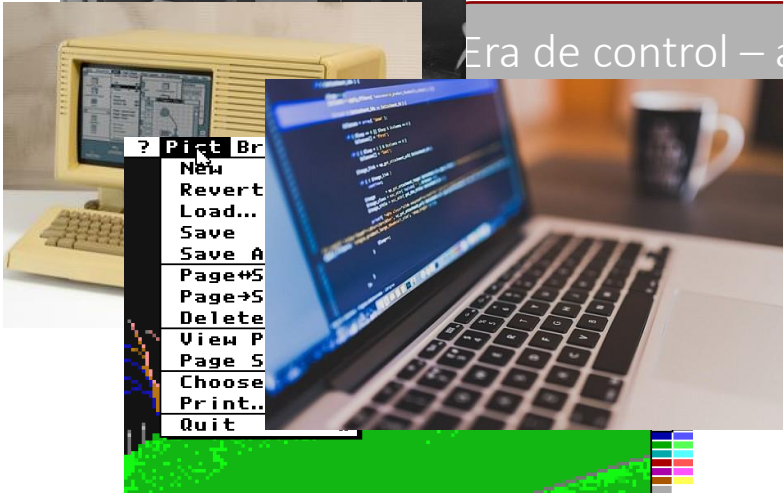
Un poco de historia sobre la IS



Era funcional – años 50 a 60
Se establecieron las necesidades

Era de calidad – años 90 a la actualidad

Se intensifica la necesidad de que el producto tenga atributos que satisfagan las necesidades explícitas e implícitas del usuario: mantenibilidad, confiabilidad, eficiencia, usabilidad



Era de control – años 70

Se establece la necesidad de desarrollar software en un entorno controlado y controlado. Se introduce el ciclo de vida en fases

Era de costos – años 80

La importancia de la productividad en el desarrollo de software se incrementa sustancialmente. Se ponen en práctica varios modelos de costos.

17



Facultad de
FORMÁTICA
UNIVERSIDAD NACIONAL DE LA PLATA

¿Qué conocimientos debe tener un IS?

- » El Ingeniero debe conocer las tecnologías y productos:
sistemas operativos, lenguajes, bases de datos, sistemas generadores de interfaces, bibliotecas de código.
- » El Ingeniero debe conocer técnicas de administración de proyectos:
planificación, análisis de riesgos, control de calidad, seguimiento de proyectos, control de subcontratistas, etc.

18

En los últimos años se observa una especialización de los ingenieros de software por dominio de aplicación o por actividad

Responsabilidad profesional y ética

» La Ingeniería de Software se desarrolla en un marco económico, social y legal.

Los IS deben aceptar responsabilidades más amplias que las responsabilidades técnicas

» No debe utilizar su capacidad y habilidades de forma deshonesto, o de forma que deshonre la profesión.

19

Responsabilidad profesional y ética

»Confidencialidad

Respetar la confidencialidad de sus empleados y clientes

»Competencia

No falsificar el nivel de competencia y aceptar responsabilidades fuera de su capacidad

»Derechos de la propiedad intelectual

Conocer la leyes vigentes sobre las patentes y copyright

»Uso inapropiado de las computadoras

No debe utilizar sus habilidades técnicas para utilizar de forma inapropiada otras computadoras

»Existen diferentes organizaciones como ACM o IEEE que sugieren diferentes códigos de ética a respetar

20

Técnicas de comunicación

21

Introducción



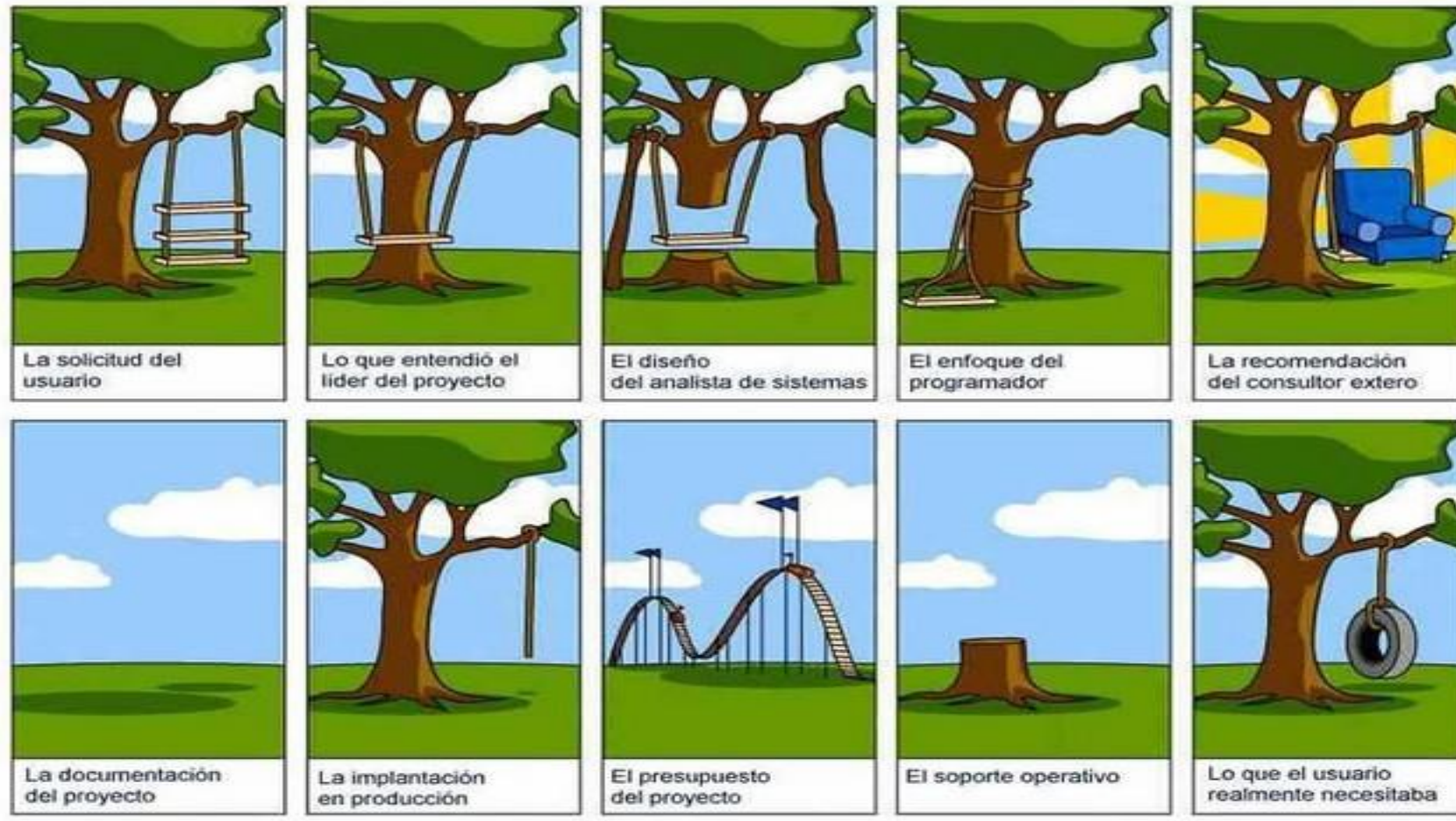
»Al iniciar un proyecto...
¿Cuál es la primera actividad?

Saber lo que el usuario quiere,
cómo lo quiere, cuándo y porqué.

22

Tenemos que Comunicarnos

El problema de la comunicación



23

¿Qué vemos?



24

La comunicación

- » La comunicación es la base para la obtención de las necesidades del cliente.
- » Es la principal fuente de error
- » Al hablar de necesidades, en términos más técnicos, estamos hablando de ***requerimientos.***

25

Requerimientos

» Un **Requerimiento** (o requisito) es una característica del sistema o una descripción de algo que el sistema es capaz de hacer con el objeto de satisfacer el propósito del sistema.

26

» Definición IEEE-Std-610

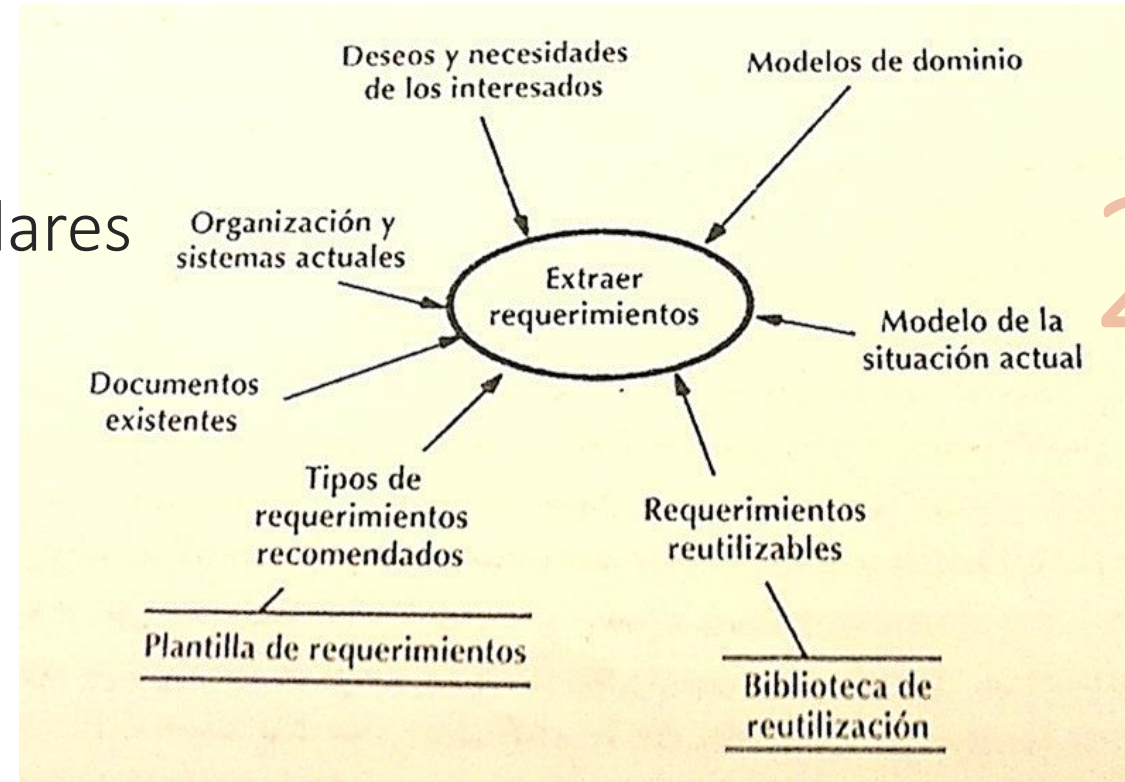
- ☐ *Condición o capacidad que necesita el usuario para resolver un problema o alcanzar un objetivo.*
- ☐ *Condición o capacidad que debe satisfacer o poseer un sistema o una componente de un sistema para satisfacer un contrato, un estándar, una especificación u otro documento formalmente impuesto.*
- ☐ *Representación documentada de una condición o capacidad como en 1 o 2.*

Fuentes de Requerimientos

Documentación

Stakeholders

Especificaciones de sistemas similares



27

Stakeholder

»El término **stakeholder** se utiliza para referirse a cualquier persona o grupo que se verá afectado por el sistema, directa o indirectamente.

»Entre los stakeholders se encuentran:

Usuarios finales

Ingenieros

Gerentes

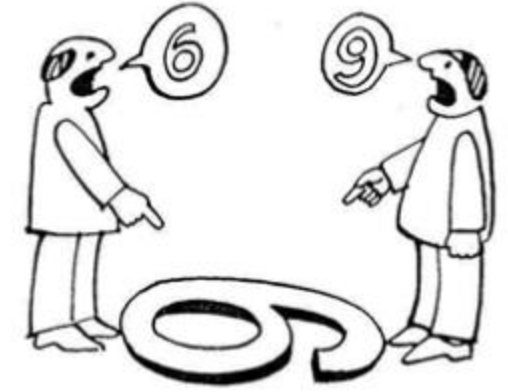
Expertos del dominio

→ diferentes visiones..



28

Puntos de Vista



» Existen tres tipos genéricos de puntos de vista:

Punto de vista de los **interactuadores**: representan a las personas u otros sistemas que interactúan directamente con el sistema. Pueden influir en los requerimientos del sistema de algún modo.

Punto de vista **indirecto**: representan a los stakeholders que no utilizan el sistema ellos mismos pero que influyen en los requerimientos de algún modo.

Punto de vista del **dominio**: representan las características y restricciones del dominio que influyen en los requerimientos del sistema.

29

Puntos de vista

» Su identificación puede ser difícil.

» Los más específicos son:

Los proveedores de servicios al sistema, los receptores de servicios del sistema.

Los sistemas que deben interactuar.

Las regulaciones y estándares a aplicar.

Las fuentes de requerimientos.

Los puntos de vista de las personas que lo van a desarrollar, administrar y mantener.

Puntos de vista del marketing y otros que generan requerimientos sobre las características del sistema.



30