

# Testing y Calidad

Capítulo 2

Calidad



# Objetivos

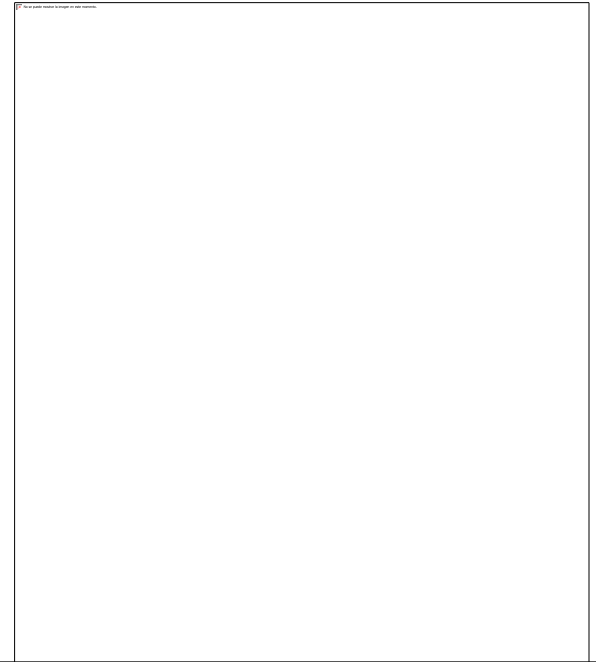
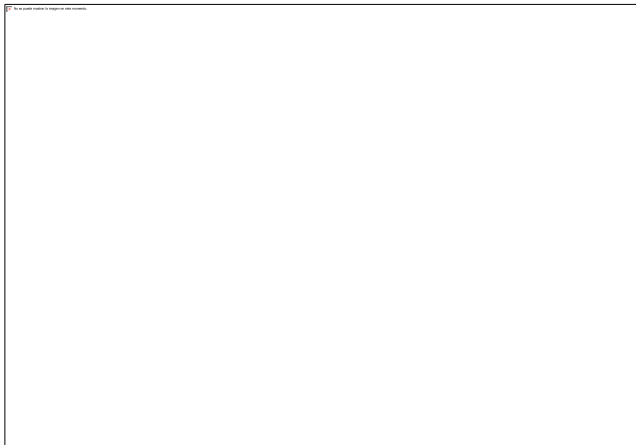
- Conocer, manejar y aplicar normas y estándares de aseguramiento de calidad de software.
- Diseñar, implementar y mejorar planes de aseguramiento de calidad de software.
- Evaluar procesos del aseguramiento de calidad de software.

# *Compraría un Producto si ...*

- *No tiene garantía con respecto al rendimiento, confiabilidad o calce para cualquier propósito específico*
- *No garantizamos que el producto o su documentación satisfaga sus requerimientos*
- *A pesar de haber pasado por un proceso de testing y documentación no aseguramos que esté libre de errores*
- *No nos hacemos responsables por daños (directos o indirectos), costos de recuperación de datos, pérdida de ganancias etc.*

# Parte intangible v/s la Física del SW

Si defectos físicos son descubiertos en la documentación o CD en que el SW es distribuido, nuestra compañía lo reemplazará sin cargo dentro de los 180 días de compra, siempre y cuando se presente prueba de esta.



***Los desarrolladores no declararán  
que su Software está libre de  
errores***

...FERENCIAS ESCENCIALES ENTRE SOFTWARE Y  
...QUIER OTRO PRODUCTO INDUSTRIAL.

# Principales Diferencias

- ✓ Complejidad del producto

Millones de posibilidades de operaciones de software

- ✓ Visibilidad del producto

Productos de software son invisibles

- ✓ Proceso de desarrollo y producción

# Principales Diferencias

## Proceso de Desarrollo y Producción de producto

### Otro Producto

- **Desarrollo de producto**
  - Detectar defectos en el prototipo
- **Planificación de producción del producto**
  - Herramientas y el proceso son preparadas y diseñadas.
- **Producción**
  - Procedimientos QA son aplicados para detectar fallas de productos.
  - Los defectos detectados se pueden corregir por cambiar diseño o materiales eliminandolos radicalmente.

### Software

- **Desarrollo del producto**
  - Detectar defectos inherentes
- **Planificación de producción del producto**
  - En algunos casos si.
- **Producción**
  - Se limita a copiar el producto e imprimir las copias de los manuales.

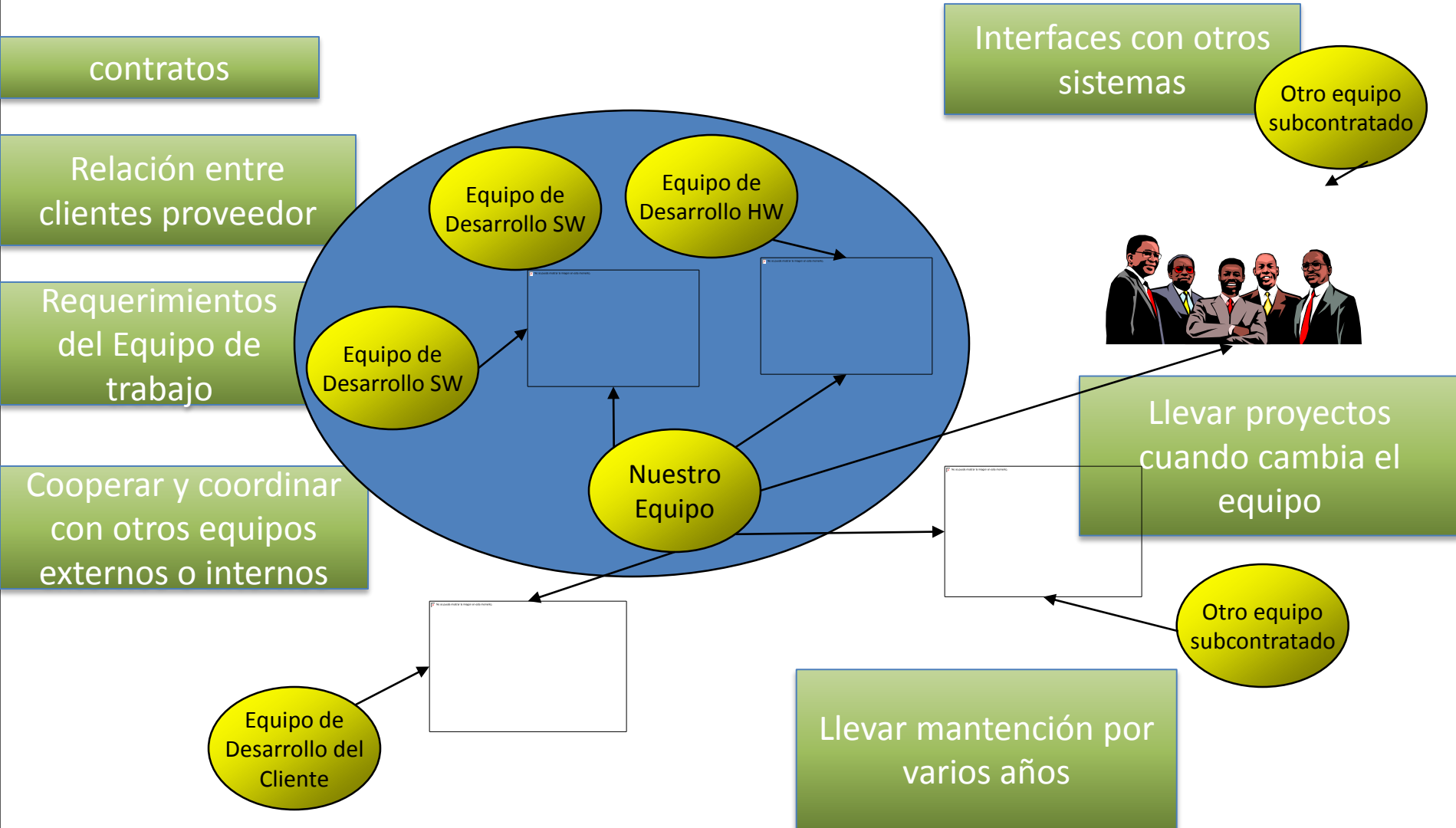
# Qué hace único al proceso de Desarrollo de Software?

- ✓ Alta complejidad – comparado a otros productos industriales
- ✓ Invisibilidad del producto
- ✓ Oportunidades de detectar defectos son limitadas a la fase de desarrollo del producto.





# Ambiente de desarrollo y mantención de Software



# Fallas, Defectos y Errores

- ✓ El origen de las fallas está en los **errores**
  - Error gramatical en una o más líneas de código
  - Error lógico en uno o mas requisitos de cliente
- ✓ No todo error llevará a un defecto
- ✓ Todo defecto lleva a una falla?

NO NECESARIAMENTE, SÓLO CUANDO ES ACTIVADO

# Ejemplo - Defecto

✓ Software desarrollado para apoyar las operaciones de una cadena de farmacia:

Defecto

- Evitar que ocurra una venta actual de productos a clientes cuya deuda excederá después de esta los 100.000 pesos
  - Lamentablemente el programador puso como límite 200.000 pesos
- Identificación de super clientes (al menos compren una vez al mes, sus compras los últimos M meses y es al menos N veces mayor que las del cliente promedio) – al pasar por la caja deben ser premiados por algún presente
  - Lamentablemente el analista solo especificó que el super cliente puede ser determinado sólo por la frecuencia de compras.

Error → Defecto

tos no serán fallas

# Ejemplo Meteoro-X

Los requerimientos para el firmware del equipo meteorológico Meteoro-X suponían que bloquearían la operación del equipo cuando su temperatura interna supere los 60 grados.

*Lamentablemente el programador lo puso a exactamente 60 grados, pero si el monitoreador recibía un 58 y luego un 61, no se suspendía la operación.*

# Errores vs Defectos vs Fallas

- ✓ Un *error de software* son secciones en el código que son parcial o totalmente incorrectos como resultado de un *error lógico* o gramatical hecho por un analista, programador u otro miembro del equipo
- ✓ Un *defecto de software* son *errores de software* que causan el funcionamiento incorrecto del software durante una aplicación específica.
- ✓ Un defecto de software se convierte en *falla* sólo cuando *es activado*



# Fallas, Defectos y Errores



# Clasificación de las Causas de los errores

- ✓ Un error es la causa de la **POBRE** calidad del software
- ✓ Un error de software puede ser un
  - Error de código
  - Error de procedimiento
  - Error de documentación
  - Error de datos

99,9% Causa humana  
El resto por ejemplo, a  
tráves de generadores  
de código ... etc

# Definición de Calidad Punto vista Cliente

## *Ejemplo*

Un editor de textos que sea incapaz de manejar archivos de más de 200 o 300 páginas, será un software de calidad para el que sólo escribe documentos de pocas páginas.

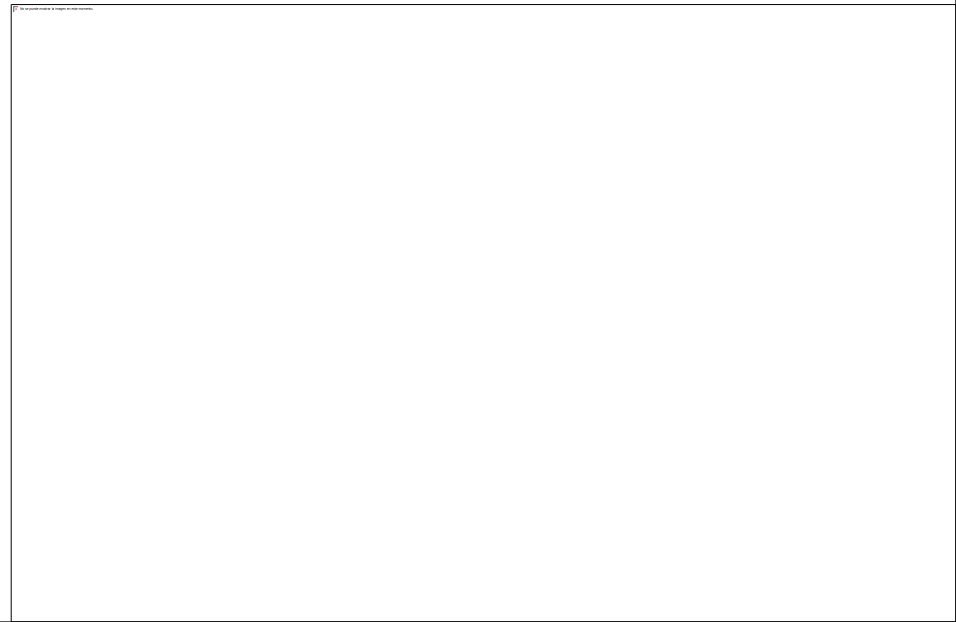
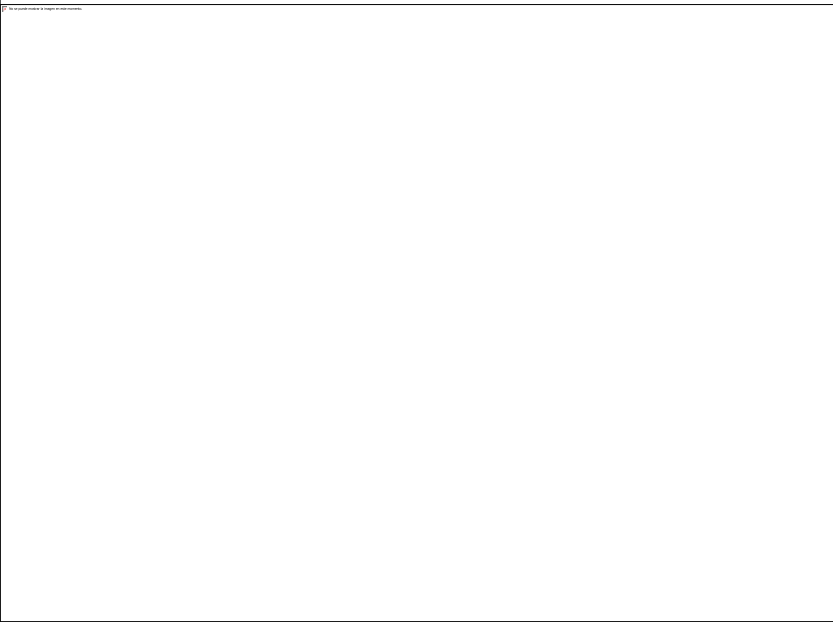
Sin embargo el mismo editor de textos será un software de mala calidad para el que escriba libros o documentos extensos.



# Diferencia entre Software de Calidad y Calidad del Software

**Software de Calidad** se entiende como la percepción que tiene un cliente respecto al software .

**Calidad de Software** es algo objetivo y cuantificado.



# Sistema de Gestión de la Calidad

## Sistema de Gestión de la Calidad

*Conjunto de elementos mutuamente relacionados que interactúan para establecer la política y los objetivos de calidad de una organización, y para dirigir y controlar dicha organización para lograr dichos objetivos.*

ISO 9000:2000

# Sistema de Gestión de la Calidad

Actividades coordinadas para dirigir y controlar una organización en lo relativo a la calidad.

Generalmente incluye el establecimiento de:

- Política de la calidad
- Objetivos de calidad
- Planificación de la calidad
- *Control de la calidad*
- Mejora de la calidad
- *Aseguramiento de la calidad*

ISO 9000:000

# Control de Calidad

## Juran:

El control de calidad es el proceso a través del cual se miden las **características** de la calidad en un momento dado, se comparan con **estándares**, y se toman **acciones** sobre las desviaciones.

## Shewhart:

Control significa **acción** para conseguir un determinado **propósito**.

Control de calidad son las acciones necesarias para mantener una determinada característica o características de la calidad dentro de unos límites preestablecidos.

# Aseguramiento de Calidad

*Parte de la gestión de la calidad orientada a proporcionar confianza en que se cumplirán los requisitos de la calidad.*

*ISO 9000:2000*

El aseguramiento de la calidad del software es el sistema de métodos y procedimientos empleados para asegurar que el producto software **satisface sus requisitos**. Este sistema implica la *planificación, medida y seguimiento* de las actividades de desarrollo llevadas a cabo.

# Control de Calidad versus Aseguramiento de calidad

## Definición Según ISO 9000

**Control de calidad:** Las técnicas y actividades que se utilizan para cumplir con los requisitos para la calidad de funcionamiento.

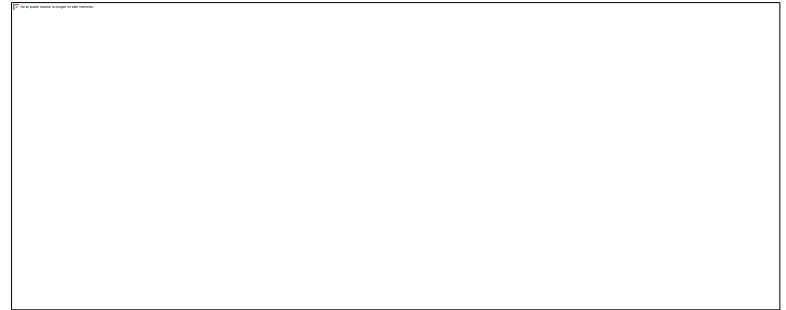
**Aseguramiento de la calidad:** Todas aquellas actividades planificadas y sistemáticas implementadas para proporcionar la confianza adecuada de que una entidad cumplirá con los requisitos de calidad.

### QC vs QA

Producto -> Proceso

Reactivo -> Proactivo

Encontrar Defectos -> Prevenir Defectos



# ***FACTORES DE CALIDAD DEL SOFTWARE***

# Medición de la Calidad del Software

La calidad al igual que la belleza, está en el ojo de quien lo observa, sin embargo desde el punto de vista de medición *“se debe tener una definición precisa en términos de atributos del software que sean de interés al usuario”*, éstos son **atributos externos** aunque, muchas propuestas miden y analizan **atributos internos** porque los consideran **predictores**, los atributos internos tienen dos ventajas:

- ✓ Están disponibles para medición más temprano
- ✓ Son más fáciles de medir

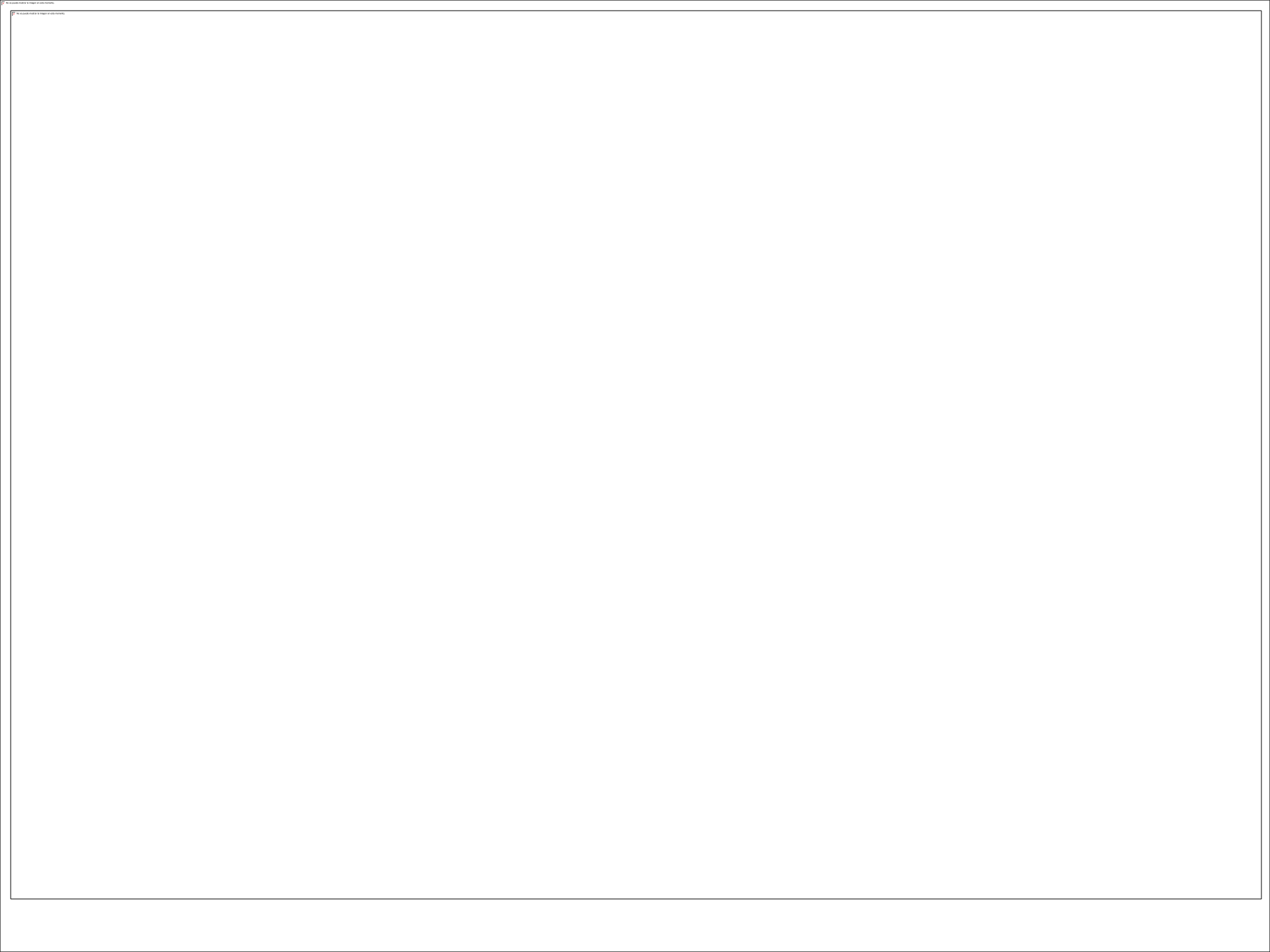


# Modelos de la Calidad del Software

La calidad está determinada por una composición de muchas características (atributos de calidad)

Un modelo de calidad describe entonces estas características y sus relaciones.

El ***modelo de McCall*** fue el primero en ser presentado en 1977, se focaliza en el producto final, identificando atributos claves desde el punto de vista del usuario estos atributos se denominan ***factores de calidad*** y son normalmente atributos externos pero también se incluyen algunos atributos internos



# Factores de Calidad

## Operación del Producto

### *Correctitud ¿Hace lo que quiero?*

- ✓ Lista de las salidas (típicamente MULTIDIMENSIONALES) requeridas del sistema de software.
- ✓ Dimensiones:
  - Misión
  - La precisión requerida
  - La completitud de la información
  - Lo actual de la información
  - La disponibilidad de la información
  - El estándar para codificación y documentación

# Factores de Calidad

## Operación del Producto

### *Correctitud ¿Hace lo que quiero?*

#### Sistema de Membresía

##### Dimensiones:

- **Misión** :Lista definida de 11 tipos de reportes (4 tipos de cartas estándares ...)
- **La precisión requerida:** Probabilidad de error para salida de 1 reporte no excederá el 1%
- **La completitud de la información:** La probabilidad de perder datos acerca de un miembro – sus asistencias y pagos – no excederá el 1%
- **Lo actual de la información:** No mas de dos días hábiles acerca de participación en eventos y no más de uno acerca de pago o datos personales
- **La disponibilidad de la información:** Tiempo de reacción será menos de dos segundos – tiempo de reportes menos que cuatro horas
- **El estándar para codificación y documentación:** Guía del cliente

# Factores de Calidad

## Operación del Producto

***Eficiencia* ¿Se ejecuta en mi HW lo mejor que se puede?**

Lidia con los recursos de HW necesarios para realizar todas las funciones del sistema acorde a los requisitos.

Tipicos

- Capacidades de procesamiento – MIPS – millones de instrucciones por segundo, MHz – millones de ciclos por segundo.
- Capacidades de almacenamiento – GBs
- Capacidades de líneas de comunicación – MBPS
- Tiempo en recargar unidades portables del sistema ...

Ejemplo:

- Una cadena de tiendas analizando a quien adjudicar la construcción de su SW:
  - Difieren en el volumen de almacenamiento: 20GB por computador y 100 GB en la oficina principal
  - Difieren en el número de líneas de comunicación necesarias (A: 3 de 28.8 KBPS, B: 2 de igual capacidad)

# Factores de Calidad

## Operación del Producto

### Integridad ¿Es seguro?

Lidia con la seguridad del sistema de software – requerimiento para prevenir el acceso a personas no autorizadas para distinguir entre la mayoría del personal permitido para ver la información y un grupo limitado con permisos de escritura

Ejemplo:

- Departamento de Ingeniería de una municipalidad local opera un GIS
- El departamento está planificando permitir a los ciudadanos acceder al GIS a través de Internet.
- Los requerimientos de Software incluyen la posibilidad de ver y copiar pero no cambiar.

# Factores de Calidad

## Operación del Producto

### Usabilidad ¿Esta diseñado para ser usado?

Lidia con el alcance de los recursos de personal necesarios para entrenar a un nuevo empleado y operar el sistema

Ejemplo – help desk:

- Personal debería ser capaz de manejar al menos 60 llamadas al día
- Entrenar un nuevo empleado no debería tomar más de dos días – luego de lo cual el entrenado debería ser capaz de manejar al menos 45 llamadas al día

# Factores de Calidad

## Revisión del Producto

### Mantenibilidad ¿Puedo corregirlo facilmente?

- Determinar el esfuerzo que necesitarán usuarios y personal de mantención en identificar las razones de las fallas del software, y verificar su éxito en las correcciones.
- Refieren a la estructura modular del software, a la documentación interna de los programas, a los manuales del programador ....
- Ejemplo:
  - Tamaño del módulo del software no deberá exceder de 30 sentencias
  - La programación se adicionará a las guías y estándares de la compañía



# Factores de Calidad

## Revisión del Producto

**Mantenibilidad ¿Puedo corregirlo facilmente?**

### **Tipos de mantención?**

Ghezzi la divide en tres categorías:

**Correctiva** concerniente a remover pequeñas fallas remanentes después del testeo.

**Adaptativa** concerniente al cambio del producto necesario por el cambio de sus requerimientos

**Perfectiva** busca solo mejorar los algoritmos usados para hacerlos más eficientes

# Factores de Calidad

## Revisión del Producto

### Flexibilidad ¿Puedo Cambiarlo?

Las capacidades y esfuerzos requeridos para apoyar las actividades de mantención adaptativas.

- Incluyen los recursos (persona-día) requerida para adaptar un paquete de software a una variedad de clientes de la misma línea, un rango de productos ...

#### Las perfectivas

- tales como cambios y adiciones al SW para mejorar su servicio y adaptarlo a cambios
- Ejemplo – SW de apoyo al profesor – lidia con la documentación de los avances del alumno, cálculo de notas, impresión de documentos, cartas de aviso a padres ...:
  - SW debería ser adecuado para todos los profesores de todas las materias de todos los niveles de los colegios

# Factores de Calidad

## Revisión del Producto

### Capacidad de ser Probado ¿Puedo probarlo?

- Lidia con las pruebas de un sistema como de su operación.
- Relacionados a las funcionalidades especiales que ayudan a probar como resultados intermedios o archivos, diagnóstico automático, monitoreo de fallas, etc
- Ejemplo – unidad de control industrial computarizada que mide estado de producto, reporta niveles de rendimiento y opera en situaciones predefinidas
  - Desarrollar un conjunto de datos de prueba estándar con un conjunto de reacciones de corrección esperadas en cada etapa – esto se ejecutará cada día

# Factores de Calidad

## Transición del Producto

### **Portabilidad ¿Podre usarlo en otra maquina?**

Lidian con la adaptación de un sistema a otro ambiente consistente de diferente HW, SO

Ejemplo – se requiere que un paquete diseñado y programado para operar en un Windows 2000 se transfiera a Linux y Windows NT

# Factores de Calidad

## Transición del Producto

### Reusabilidad ¿Podré reutilizar una parte del SW?

- Lidia con el uso de módulos de software originalmente diseñados para un proyecto en un nuevo proyecto.
- Se espera ahorrar recursos, acotar tiempos de desarrollo y proveer módulos de alta calidad.
  - Supuesto: la mayoría de los defectos se han detectados por las actividades de SQA previas, las fallas capturadas por los usuarios originales del SW ...
- Ejemplo – sistema de operación y control de piscina de un hotel para clientes y miembros
  - El equipo desarrollador decidió adicionar este requerimiento pues ellos desarrollaron el sistema de golden splash

# Factores de Calidad

## Revisión del Producto

### **Interoperabilidad ¿Podre hacerlo actuar con otros sistemas??**

- Se enfoca en crear interfaces con otros sistemas o con otros equipos
- Ejemplo – el firmware de un equipo de lab médico es requerido para procesar sus resultados acorde a una estructura de datos estándar que sirve como entrada para un número de SIs de laboratorios apegados al estándar

# Modelo de Calidad de McCall

Fin clase 1

# Preguntas

