

## UNIDAD 1. INTRODUCCIÓN A LA INGENIERÍA DE SOFTWARE

### 1.1 Introducción a la Ingeniería de Software

#### 1. ¿El Software se desarrolla o se fabrica? Justifique.

- El software se desarrolla, no se fabrica.
  - El enfoque del desarrollo de Hw y Sw se adquiere la calidad mediante un buen diseño
  - Ambas actividades:
    - Dependen de personas, pero la relación entre las personas dedicadas y el trabajo realizado es diferente.
    - Requieren la construcción de un producto pero los enfoques son diferentes.
  - Los costos de sw se encuentran en la ingeniería, por lo cual los proyectos de sw no se pueden encarar como si fueran de fabricación.

#### 2. ¿Cuáles son los mitos del desarrollo de sw? Mencione ejemplos de c/u

- *Mitos de la gestión:*
  - Los gestores con responsabilidad sobre el sw, están normalmente bajo la presión de cumplir con las propuestas, hacer que no se retrase el proyecto y mejorar la calidad. Un gestor se agarra frecuentemente a un mito de gestión. *Si se falla en la planificación, se puede añadir más programadores y adelantar el tiempo perdido.*  
  
Ej: Mi gente dispone de las herramientas de desarrollo de sw más avanzadas, después de todo les compramos el hw más moderno. (Se necesita mucho más que hw para hacer desarrollo de sw de calidad).
- *Mitos del cliente:*
  - Los mitos conducen a que el cliente se cree una falsa expectativa y, finalmente, quede insatisfecho con el desarrollo de sw.  
  
Ej: Si los requisitos del proyecto cambian continuamente, los cambios pueden acomodarse fácilmente, ya que el sw es flexible. (Los requisitos cambian, pero el impacto del cambio varía según el momento en que se introduzca, creciendo más el costo a medida que se acerca el final del proyecto)
- *Mitos de los desarrolladores:*
  - Los mitos en los que aún creen muchos desarrolladores se han ido fomentando durante 50 años de cultura informática. Durante los primeros días del desarrollo del sw, la programación se veía como un arte. Las viejas formas y actitudes tardan en morir.  
  
Ej: Una vez que escribimos el programa y hacemos que funcione, nuestro trabajo ha terminado. (Entre el 60 y el 80% de todo el esfuerzo dedicado al programa se realiza una vez que se haya entregado por primera vez para ser testeado)

### 3. ¿Cuáles son la características de del sw?

- El sw se desarrolla, no se fabrica en un sentido clásico (Ídem pregunta 1)
- El sw no se estropea
  - El hw exhibe relativamente muchos fallos al principio de su vida por defectos de fabricación, luego se reduce y cuando empieza a desgastarse, la tasa de fallos se incrementa. Si una pieza se rompe, esta re reemplaza
  - El Sw no es susceptible a los males del entorno que hacen que el Hw se estropee. A medida que se deteriora se introduce nuevos defectos y aparecen picos por que se deteriora el Sw debido a los cambios. Si un componente de Sw tiene errores en el diseño o procesos no se puede reemplazar. El mantenimiento del Sw tiene una complejidad considerablemente mayor que la del mantenimiento del Hw.
- La mayoría del sw se construye a medida, en vez de ensamblar componentes existentes
  - No existen catálogos de componentes de sw. Se puede comprar sw ya desarrollado, pero solo como una unidad completa, no como componentes que pueden re-ensamblarse en nuevos programas
  - La reutilización es una característica que debe implementarse para que pueda volver a ser reutilizado en muchos programas diferentes, ya sea algoritmos o estructuras de datos.
  - Los componentes reutilizables encapsulan tanto datos como procesos, que se aplica a los datos, como por ej. interfaces graficas de usuario.

### 4. ¿Cuáles son las aplicaciones de sw?

- *Software de sistemas*
  - EL sw de sistemas es un conjunto de programas que han sido escritos para servir a otros programas
  - El área del Sw de sistemas se caracteriza por:
    - Fuerte interacción con el hw del sistema
    - Gran utilización por múltiples usuarios
    - Operación concurrente que requiere una planificación
    - Compartición de recursos y una sofisticada gestión de procesos
    - Estructuras de datos complejas y múltiples interfaces externas
- *Software de tiempo real*
  - El sw que mide/analiza/controla sucesos del mundo real conforme a lo que ocurre, se denomina de tiempo real.
  - Entre los elementos del sw de tiempo real se incluye un componente de:
    - Adquisición de datos que recolecta y da formato a la información recibida de entorno externo
    - Análisis que transforma la información recibida del entorno externo
    - Análisis que transforma la información según lo que requiera la aplicación

- Control/salida que responda al entorno externo
- Monitorización que coordina todos los demás componentes, de forma tal que puedan mantenerse la respuesta en tiempo real
- *Software de gestión*
  - El procesamiento de la información comercial constituye la mayor de las áreas de aplicación de sw
  - Los sistemas discretos (ej. Sueldos, contabilidad, etc.) han evolucionado hacia el sw de sistemas de gestión (SIG), que accede a una o más bases de datos grandes que contienen información comercial. Las aplicaciones en esta área reestructuran los datos existentes para facilitar las operaciones comerciales o gestionar la toma de decisiones.
  - Además de las tareas convencionales de procesamiento de datos, las aplicaciones de sw de gestión también realizan cálculo interactivo (ej.: procesamiento de transacciones en puntos de ventas).
- *Software de ingeniería y científico*
  - Se caracteriza por algoritmos de manejo de números. Las aplicaciones van desde la astronomía a la vulcanología, desde el análisis de la presión de los automotores a la dinámica orbital, etc.
- *Software empotrado*
  - Reside en memoria de solo lectura y se utiliza para controlar productos y sistemas de los mercados industriales y de consumo.
  - Puede ejecutar funciones muy limitadas y curiosas o suministrar una función significativa y con capacidad de control
- *Software de computadores personales*
  - El mercado del sw de computadoras personales ha germinado en la pasada década. El procesamiento de texto, la hoja de cálculo, los gráficos, multimedia y entretenimiento son alguna de los cientos de aplicaciones.
- *Software de inteligencia artificial*
  - El sw de inteligencia artificial hace uso de algoritmos no numéricos para resolver problemas complejos para los que no son adecuados el cálculo o el análisis directo.
- *Software basado en web*
  - Las páginas web buscadas por un explorador son sw que incorpora instrucciones ejecutables (HTML, Perl o Java), y datos. La red tiende a ser una gran computadora que proporciona recursos casi ilimitado que puede ser accedido por cualquiera

## 5. Definición de ingeniería de sw

- Definición: Es una disciplina que comprende todos los aspectos de la producción de sw desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después de que se utilizara.

## 6. ¿Cuál es el objetivo de la Ingeniería de sw?

- El objetivo de la ingeniería de sw es lograr productos de sw de calidad (tanto en su forma final como durante su elaboración), mediante un proceso apoyado por métodos y herramientas.

## 7. ¿Qué diferencias hay entre Ing. de sw, Ciencias de la computación e Ingeniería en Sistemas?

- *Ingeniería de sw:*
  - Se refiere a problemas prácticos de producir el sw y liberarlo
- *Ciencia de la computación:*
  - Teorías y métodos subyacentes a las computadoras y sistemas de sw
- *Ingeniería en sistemas:*
  - Se refiere a todos los aspectos del desarrollo de sistemas basados en computadoras, incluyendo Hw y Sw e ingeniería de procesos.
  - Lleva más de 100 años.
  - Se involucra en las especificaciones del sistema, en la definición de su arquitectura y en la integración para crear el sistema final.

## 8. Definición de proceso, proceso de sw y sus características (ver tb en material 1.3)

- *Proceso*
  - Conjunto ordenado de tareas, una serie de pasos que involucran actividades, restricciones y recursos, y que producen una salida determinada.
- *Proceso de sw*
  - Conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema de sw
- *Características del proceso de sw*
  - Tiene una serie de actividades principales
  - Utiliza recursos, está sujeto a restricciones y genera productos intermedios y finales.
  - Compuesto por subprocesos que se encadenan de alguna forma
  - Cada actividad tiene sus criterios de entrada y salida que permite conocer cuando comienza y termina dicha actividad
  - Existen principios orientadores que explican las metas de cada actividad

## 9. ¿Qué debe especificar un proceso de sw?

- La secuencia de actividades a realizar por el equipo de desarrollo
  - Flujo de actividades
- Los productos que deben crearse
  - Resultados de trabajo (modelos, documentos, datos, informes, etc)
  - Qué y cuándo
- La asignación de tareas a cada miembro del equipo y al equipo
  - Como un todo
- Los criterios para controlar el proceso
  - Se establece el control de gestión de los proyectos de sw
- Las posibles heurísticas

#### 10. ¿Cómo se llama un proceso cuando implica la construcción de un producto?

- El proceso se llama “ciclo de vida”
  - Aportan consistencia y estructura sobre un conjunto de actividades, lo que permite realizar la misma tarea correctamente de forma repetida
  - Existen diferentes modelos de procesos

#### 11. ¿Qué es una metodología? ¿Puede seguir uno o más modelos de ciclo de vida?

Podemos definir metodología como un conjunto de métodos. La metodología **indica cómo hay que obtener los distintos productos parciales y finales**.

METODOLOGÍA = CICLO DE VIDA + COLECCIÓN DE MÉTODOS

Una metodología puede seguir uno o varios modelos de ciclo de vida, es decir, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto pero no cómo hacerlo.

#### 12. ¿Qué es un ciclo de vida? ¿Es igual a un ciclo de desarrollo?

El **ciclo de vida** es una sucesión de etapas por las que pasa el software en su desarrollo, desde que se concibe la idea hasta que el software deja de utilizarse (obsolescencia).

“Un marco de referencia que contiene los procesos, las actividades y las tareas involucradas en el desarrollo, la explotación y el mantenimiento de un producto de software, abarcando la vida del sistema desde la definición de los requisitos hasta la finalización de su uso” ISO 12207-1

A **diferencia** del ciclo de vida, el ciclo de desarrollo abarca únicamente desde el análisis hasta la entrega al usuario.

#### 13. ¿Cuándo se usan los distintos modelos de ciclo de vida?

- Para sistemas bien comprendidos utiliza el Modelo de Cascada. La fase de análisis de riesgos es relativamente fácil.
- Con requerimientos estables y sistemas de seguridad críticos, utiliza Modelos Formales.
- Con especificaciones incompletas, utiliza el Modelo de Prototipado.
- Pueden utilizarse Modelos Híbridos en distintas partes del desarrollo.

#### 14. ¿Cuáles son los métodos de Ingeniería de Software?

- Desarrollo Convencional (Sin Metodología)
- Desarrollo Estructurado
  - Orientadas a Procesos
  - Orientadas a Datos
    - Jerárquicas
    - No Jerárquicas
- Mixtas
- Desarrollo Orientado a Objetos
- Tradicionales y Ágiles

### 15. Nombre diferencias entre métodos tradicionales y ágiles

Metodología Ágil	Metodología Tradicional
Pocos artefactos. El modelado es prescindible, modelos desechables.	Más artefactos. El modelado es esencial, mantenimiento de modelos.
Pocos roles, más genéricos y flexibles	Más roles, más específicos
No existe un contrato tradicional, debe ser bastante flexible	Existe un contrato prefijado
Cliente es parte del equipo de desarrollo (además in-situ)	El cliente interactúa con el equipo de desarrollo mediante reuniones
Orientada a proyectos pequeños. Corta duración (o entregas frecuentes), equipos pequeños (< 10 integrantes) y trabajando en el mismo sitio	Aplicables a proyectos de cualquier tamaño, pero suelen ser especialmente efectivas/usadas en proyectos grandes y con equipos posiblemente dispersos
La arquitectura se va definiendo y mejorando a lo largo del proyecto	Se promueve que la arquitectura se defina tempranamente en el proyecto
Énfasis en los aspectos humanos: el individuo y el trabajo en equipo	Énfasis en la definición del proceso: roles, actividades y artefactos
Se esperan cambios durante el proyecto	Se espera que no ocurran cambios de gran impacto durante el proyecto

### 16. ¿Cuáles son los retos de la Ingeniería de Software? Explíquelos

Según Sommerville, existen 4 problemas esenciales en los comienzos del siglo XXI:

- *El reto de lo heredado:*
  - Sistemas antiguos que deben ser mantenidos y mejorados
- *El reto de la heterogeneidad:*
  - Sistemas que incluyen una mezcla de software y hardware
- *El reto de la entrega:*
  - Existe una presión incremental por una entrega a tiempo de los productos de SW
- *El reto de la formalidad:*
  - Existe una gran demanda de que exista formalidad en el proceso de desarrollo de SW

### 17. ¿Cuáles son las características de los productos de SW?

- *Mantenibles*
  - Debe ser posible que el SW evolucione y que siga cumpliendo con sus especificaciones
- *Confiabilidad*
  - El SW no debe causar daños físicos o económicos en el caso de fallos
- *Eficiencia*
  - El SW no debe desperdiciar los recursos del sistema
- *Utilización Adecuada*
  - El SW debe contar con una interfaz de usuario adecuada y su documentación

### 18. ¿Qué es la reingeniería de SW? ¿Cuál es el objetivo principal?

La reingeniería de SW es un proceso de examen y modificación de un sistema objeto para reconstruirlo en un formato nuevo.

- La reingeniería puede implicar modificar el formato del sistema:
  - Sin cambiar su funcionalidad
  - Cambiando su funcionalidad

Sistema de SW existente -> Comprensión y transformación -> Sistema con reingeniería

El objetivo principal de la reingeniería es actualizar los sistemas existentes dotándolos con las ventajas de las tecnologías nuevas, y capacitar los nuevos esfuerzos de desarrollo con las ventajas que se obtienen de la reutilización de los sistemas actuales.

### **19. ¿Qué es ingeniería Web? ¿Con qué se relaciona? ¿Es igual a Ingeniería de SW?**

La ingeniería web es la aplicación de metodologías sistemáticas, disciplinadas y cuantificables al desarrollo eficiente, operación y evolución de aplicaciones de alta calidad en WWW.

La ingeniería web está relacionada con el establecimiento y utilización de principios científicos, de ingeniería y gestión, y con enfoques sistemáticos y disciplinados del éxito y desarrollo, empleo y mantenimiento de sistemas y aplicaciones basados en la Web de alta calidad.

La ingeniería Web no es un clon de la ingeniería del SW aunque ambos involucren programación y desarrollo de software. Si existe la posibilidad de utilizar herramientas y técnicas basadas en la ingeniería del SW para poder garantizar el buen funcionamiento y administración de los sitios web.

### **20. ¿Cuál es el concepto de una fábrica de SW? ¿A qué necesidades responden?**

Una fábrica de SW es un conjunto de RRHH, materiales, procesos, metodologías y técnicas estructuradas de manera semejante a la de las industrias tradicionales, emplea las mejores prácticas creadas para procesos, desarrollo, pruebas y mantenimiento del SW.

Responden a 3 necesidades en el campo del desarrollo y mantenimiento de SW:

- La escasez de mano de obra calificada
  - Por cambios tecnológicos
  - Diferentes ciclos de la industria del SW
- Los fracasos en la industria del SW
  - Falta de madurez de la propia industria
  - La complejidad creciente de los sistemas a desarrollar
- La evolución incesante de la tecnología, que obliga a las empresas a cambiar constantemente (de sistemas centralizados y tratamientos batch, a ordenadores centrales con terminales, ordenadores personales, arquitecturas cliente-servidor, servicios web, etc.)

### **21. ¿En qué consiste el desarrollo distribuido y global de SW?**

- *Desarrollo distribuido de SW*
  - Los stakeholders (clientes, usuarios, desarrolladores, líderes, etc) que llevan a cabo los proyectos se encuentran distribuidos entre varios sitios remotos.
  - Son sub-equipos y equipos en lugares geográficos diferentes, haciendo uso de las tecnologías de comunicación para interactuar y llevar a cabo una tarea en conjunto
- *Desarrollo global en SW*

- Los equipos se distribuyen más allá de la frontera que deben hacer frente a desafíos únicos y de gran interés para la comunidad informática
- Los equipos pertenecen a distintos sub-equipos de la misma organización
  - Offshoring: deslocalización, transferencia de empleos a otro país

## 22. ¿Cuáles son los desafíos del Desarrollo Global de sw?

Los desafíos del desarrollo global de sw son:

- Problemas ocasionados por la comunicación inadecuada.
- Problemas ocasionados por la diversidad cultural.
- Problemas en la gestión del conocimiento
- Problemas ocasionados por la diferencia horaria.

## 23. ¿Cuáles son las herramientas que más se utilizan para modelos de provisión y la globalización de servicios?

- Herramientas asíncronas: mail, grupos de noticias y novedades, foros, boletines electrónicos, documentos compartidos, pizarras de dibujos compartidas asíncronas, etc.
- Herramientas sincrónicas: pizarras de dibujos compartidos sincrónicos, chat, mensajería instantánea, conferencias, videoconferencias, etc.

## 24. ¿Cuáles son los modelos de provisión y la globalización de servicios? ¿Cómo se proveen los servicios?

Modelos:

1. Outsourcing: ciertos procesos de negocio se estarán ejecutando por un proveedor de servicios insourcing cuando son generados por la propia compañía
2. Offshore: ejecución remota de los procesos de negocio, que podrán ser entregados in o out sourcing
3. Nearshore: es un refinamiento de offshore, pero los servicios están en los países vecinos con los mismos horarios. Pueden tener variables adicionales tales como acuerdos comerciales con libre movimiento de profesionales (Nafta en América y UE en Europa, por ej) y que existan similares culturas de negocios entre países
4. Onshore: los servicios estarán prestando en el mismo país. Puede ser un refinamiento del nearshore.
5. Onsite: los servicios se están presentando en las instalaciones del cliente
6. Modelo de Global Delivery: Hace referencia a una estrategia de realizar el delivery de los procesos del negocio, que por su naturaleza, se podían estar realizando o usando varias modalidades simultáneamente, tanto offshore, nearshore u onsite

### 1.2 Ingeniería de Requisitos

## 25. ¿Qué no es un requisito?

Un requisito no es un detalle de diseño o implementación o prueba, información relativa a la planificación del proyecto, necesidades del proyecto. A estos elementos se los pueden llamar especificaciones suplementarias y no requisitos.



## 26. ¿Qué es un requisito? Definiciones

1. Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo
2. Una condición o capacidad que debe estar presente en un sistema o componente de sistema para satisfacer un contrato, estándar, especificación u otro documento formal
3. Una representación documentada de una condición o capacidad como en 1 o 2 [IEEE]

“Una especificación de qué se debería implementar. Son descripciones de cómo se debe comportar el sistema, o de un atributo o propiedad del sistema. Puede ser una restricción en el proceso de desarrollo de un sistema”

## 27. ¿Qué son las características en un requerimiento? ¿Cuáles son las más importantes?

Las características de un requerimiento son sus propiedades principales. Un conjunto de requerimientos en estado de madurez, deben presentar una serie de características tanto individualmente como en grupo.

Las más importantes:

- *Consistente*: Un requerimiento es consistente si no es contradictorio con otros requerimientos.
- *No ambiguo*: Un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.
- *Verificable*: Un requerimiento es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación: inspección, análisis, demostración o prueba.
- *Necesario*: Un requerimiento es necesario si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso.
- *Conciso*: Un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- *Completo*: Un requerimiento está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.

## 28. ¿Cuáles son las dificultades para definir los requerimientos?

Los requerimientos no son obvios y vienen de muchas fuentes, es por eso que son difíciles de expresar en palabras (el lenguaje es ambiguo o desconocido), existen muchos tipos de requerimientos y diferentes niveles de detalle, la cantidad de requerimientos en un proyecto pueden ser difícil de manejar, nunca son iguales, algunos son más difíciles, mas riesgosos, mas importantes o más estables que otros.

- Los requerimientos están relacionados unos con otros, y a su vez se relacionan con otras partes del proceso.
- Cada requerimiento tiene propiedades únicas y abarcan áreas funcionales especiales.
- Un requerimiento puede cambiar a lo largo del ciclo de desarrollo.
- Son difíciles de cuantificar, ya que cada conjunto de requerimientos es particular para cada proyecto.

### 29. ¿Cuáles son los tipos de requerimientos ligados con el grado de satisfacción del cliente?

Existen tres tipos:

- Normales: Objetivos y metas para un producto o sistema. Directamente ligados con satisfacción.
- Esperados: Implícitos al producto o sistema, el cliente puede no declarar explícitamente y su ausencia es motivo de insatisfacción.
- Innovadores: Más allá de las expectativas del cliente, suelen ser muy satisfactorias.

### 30. ¿En qué consiste la ingeniería de requerimientos? ¿Qué actividades comprende?

La ingeniería de requerimientos es un proceso que comprende todas las actividades para crear y mantener los requerimientos de un sistema.

Comprende cuatro actividades:

- Estudio de factibilidad.
- Obtención y análisis de requerimientos.
- Validación de requerimientos.
- Administración de requerimientos.

### 31. Explique los pasos para la indagación de requerimientos (Definición y especificación de requerimientos, especificación de software)

Pasos:

- Definición de Requerimientos: Una declaración en un lenguaje natural incluye los diagramas de los servicios del sistema y sus límites operacionales. Escrito para el cliente.
- Especificación de requerimientos: Un documento estructurado con descripción i detalle de los servicios del sistema. Escrito como un contrato entre el cliente y el contratista.
- Especificación de software: Descripción detallada de software, la cual, puede servir como una base para diseño o implementación. Escrito para desarrolladores.

### 32. ¿En qué consiste el proceso de Administración de requerimientos?

El proceso de administración de requerimientos es el proceso de comprender y controlar los cambios en los requerimientos del sistema.

### 33. ¿Cuáles son los requerimientos duraderos o volátiles? De ejemplos de los volátiles

*Requerimientos duraderos:* Los requerimientos duraderos son los que se derivan en la actividad principal de la organización y están relacionados con el dominio del sistema.

*Requerimientos volátiles:* los requerimientos volátiles son los que cambian durante el desarrollo o después de puesto en operación.

Tipos de requerimientos volátiles:

- Mutantes: Cambian por cambios en el ambiente en que opera la organización.
- Emergentes: Emergen al incrementarse la comprensión del cliente en el desarrollo del sistema

- Consecutivos: Son el resultado de la introducción de un sistema, generando nuevas formas de trabajo.
- De compatibilidad: Son requerimientos que dependen de sistemas particulares o procesos de negocio dentro de la organización.

### 34. ¿En qué consiste la técnica general de la gestión de requisitos llamada Etnografía?

La etnografía es una técnica de observación para entender requerimientos sociales y organizacionales. Es especialmente efectiva para descubrir requerimientos derivados de la forma real de trabajar más que de la forma en que debería de trabajar de acuerdo a los procesos establecidos, requerimientos derivados de la cooperación y conocimiento de las actividades de la gente, se puede combinar con prototipos, no es apropiada para descubrir requerimientos organizacionales o del dominio.

### 35. ¿Qué cuestiones deben ser preguntadas y contestadas durante un análisis de requisitos?

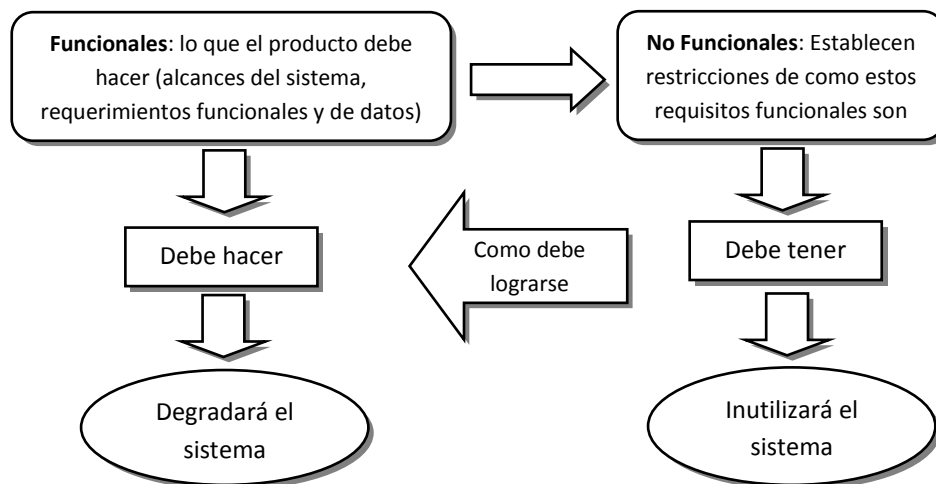
- ¿Cada requisito es consistente con los objetivos generales del sistema/producto?
- ¿Tienen todos los requisitos especificados el nivel de abstracción? (nivel de detalle técnico necesario)
- ¿El requisito es necesario o presenta una característica añadida que puede no ser esencial a la finalidad del sistema?
- ¿Cada requisito está delimitado y sin ambigüedad?
- ¿Cada requisito tiene procedencia, es decir, existe un origen conocido para cada requisito?
- ¿Existen requisitos incompatibles con otros requisitos?
- ¿Es posible lograr cada requisito en el entorno técnico donde se integrará el sistema o producto?
- ¿Se puede probar el requisito una vez implementado?

### 36. ¿Qué son los requerimientos funcionales, no funcionales y los del dominio? ¿Qué son las restricciones del diseño?

- Requerimientos Funcionales: Definen las funciones que el sistema será capaz de realizar. Son los requisitos relativos a las capacidades y servicios. Describen las transformaciones que el sistema realiza sobre las entradas para producir salidas. Describen servicios o funciones.
- Requerimientos no funcionales: Son restricciones o limitaciones de los servicios o funciones ofrecidos por el sistema. Son un límite en el sistema o en el proceso de desarrollo. Incluyen restricciones de tiempo, sobre el proceso de desarrollo, estándares, fiabilidad, respuesta en tiempo, capacidad de almacenamiento, portabilidad, rendimiento, etc. Son relativos a atributos de calidad del sistema mientras desempeña sus funciones.
- Requerimientos del dominio: Son requerimientos que provienen del dominio de aplicación del sistema y que reflejan las características de ese dominio (estos pueden ser funcionales o no funcionales)

Las *restricciones del diseño* son condicionantes existentes para el diseño, sin anticiparlo.

### 37. Analizar el cuadro de incumplimiento en los requerimientos funcionales y no funcionales.



### 38. ¿Qué recomendaciones daría para desarrollar los requerimientos del usuario?

Recomendaciones:

- Inventar un formato estándar y asegurar que los requerimientos se adhieran al formato,
- Utilizar el lenguaje de forma consistente, distinguiendo los requerimientos deseables (en futuro condicional) de los obligatorios (en futuro simple)
- Resaltar el texto (negrita o itálicas) para ver las partes claves del requerimiento.
- Evitar en lo posible, el lenguaje técnico de IT. Aunque será inevitable utilizar términos técnicos detallados provenientes del dominio de la aplicación del sistema.

### 39. ¿En qué consiste en la interfaz usuaria: calidad percibida, calidad externa, calidad en el uso y usabilidad?

- Calidad percibida: Conjunto de características de un producto que aportan una gran satisfacción a unos usuarios específicos.
- Calidad externa: Instante en el cual un producto satisface las necesidades establecidas e implícitas cuando este es utilizado bajo ciertas condiciones específicas.
- Calidad en el uso: La efectividad (precisión y completitud de alcanzar objetivos), eficiencia (usar los recursos necesarios en relación con la precisión y completitud con la que los usuarios alcanzan los objetivos) y satisfacción (aceptación del sistema por parte de los usuarios) con el cual ciertos usuarios pueden alcanzar ciertas metas en entornos concretos.
- Usabilidad: Instante en el cual un producto puede ser utilizado por usuarios específicos para alcanzar metas específicas con efectividad, eficiencia y satisfacción en un contexto de uso específico.

### 40. ¿Cuáles son las características que contiene la usabilidad como concepto multidimensional?

Características:

- Efectividad o relevancia: el sistema atiende a las necesidades de los usuarios.
- Eficiencia: Con que eficacia pueden los usuarios realizar las tareas.
- Actitud del usuario: sentimientos subjetivos en el manejo del sistema.

- Comprensibilidad: nivel de facilidad con el que los usuarios pueden usar por primera vez el sistema
- Seguridad: aportar la posibilidad al usuario de deshacer acciones y no permitir que el sistema realice acciones destructivas.

#### 41. ¿Cuáles son los cuatro puntos sobre los que descansa la usabilidad?

La usabilidad hace referencia a la rapidez y facilidad con la que las personas llevan a cabo sus tareas propias a través del uso del producto objeto de interés, idea que descansa en cuatro puntos:

1. Una aproximación al usuario: Usabilidad significa enfocarse en los usuarios. Se tienen que conocer, entender y trabajar con las personas que representan a los usuarios actuales o potenciales del producto.
2. Un amplio conocimiento del contexto de uso: Las personas utilizan los productos para incrementar su propia productividad. Un producto se considera fácil de aprender y usar en términos del tiempo que toma el usuario para llevar a cabo su objetivo, el número de pasos que tiene que realizar para ello, y el éxito que tiene en predecir la acción apropiada para llevar a cabo.
3. El producto ha de satisfacer las necesidades del usuario: Los usuarios son gente ocupada intentando llevar a cabo una tarea. Se va a relacionar usabilidad con productividad y calidad.
4. Son los usuarios, y no los desarrolladores y los diseñadores, los que determinan cuando un producto es fácil de usar.

#### 42. Para qué sirve una herramienta CARE? Nombre ejemplos de herramientas

- Facilitan el trabajo de los analistas, permitiendo la gestión, organización almacenamiento y gestión de los requisitos.
- A través de la trazabilidad también ayudan a evaluar el impacto de los cambios en los requisitos en los procedimientos, costos y personal.
- CARE debe proporcionar operaciones de creación, edición y recuperación de requisitos y gestión de las trazas, además de:
  - Informes de alto nivel.
  - Plantillas para la documentación del proyecto.
  - Integración con otras herramientas CASE y posibilidad de construir extensiones o add-ins personalizados.

Son ejemplos de herramientas de este tipo:

- REQUISITE PRO (IBM).
- IRQA.
- RTM WORKSHOP
- HP QUALITY CENTER (H.PACKARD).
- NIMBLE (DEASTUB) (Open Source Requirements Management Tool).
- ENTERPRISE ARCHITECT.
- CaliberRM (BORLAND).
- TestLink.
- RationalRequisitePro.

### 1.3 Proceso de ingeniería de software

#### 43. En la Ingeniería de proceso de negocios, que son y en qué consisten Arquitectura de datos, Arquitectura de aplicación y la infraestructura tecnológica?

**Ing. de Proceso de Negocio:** El objetivo es definir arquitecturas que permitan a las empresas emplear la información eficazmente.

- **Arquitectura de datos:** Proporciona una estructura para necesidades de información de un negocio o de una de sus funciones.
- **Arquitectura de aplicación:** Comprende aquellos elementos de un sistema que transforman objetos dentro de la arquitectura de datos por algún propósito del negocio.
- Consideramos que normalmente la arquitectura de aplicación es el sistema de programas o software que realiza esta transformación.
- **Infraestructura tecnológica:** Proporciona el fundamento de la arquitectura de datos y de aplicaciones. Comprende el **Sw** y **Hw** empleados para dar soporte a las aplicaciones y datos. Esto incluye computadoras, redes, enlaces, tecnologías de almacenamiento y la arquitectura (ej. Cliente-servidos) diseñada para implementar estas tecnologías.

#### 44. Para qué se usan las herramientas de procesos?

Se usan para:

- Para mejorar un proceso o **Sw** de negocios.
- Las herramientas de modelado se usan para representar los elementos claves de un proceso de modo que se entienda mejor.
- Se relacionan con descripciones del proceso que ayudan a entenderlo.
- Permiten que un equipo defina elementos de un modelo de procesos único (acciones, tareas, productos de trabajo, puntos de aseguramiento de calidad), dan una guía detallada acerca del contenido o descripción de cada elemento del proceso.
- Incorporan tareas estándares de administración de proyectos, tales como estimación, seguimiento y control.

#### 45. ¿Qué es la agilidad?

**Agilidad:** Capacidad para adaptar el curso del desarrollo a la evolución de los requisitos y las circunstancias del entorno.

- Es la efectiva capacidad de un equipo a responder de manera apropiada (rápida y adaptativa) a los cambios (ya sea de software por nuevos requerimientos o debido a nuevas tecnologías de todas clases que tienen efecto en el producto que se elabora).
- Comunicación eficaz con todos los stakeholders (tecnólogos, ingenieros, gente de negocios, gerentes y clientes).
- Adopta al cliente como parte del equipo de desarrollo.
- Organización de un equipo de modo que esté en el control del trabajo realizado.
- Rendimiento (eliminar todos los productos de trabajo excepto los más esenciales).
- Poner énfasis en una entrega rápida e incremental del software que haga trabajar al software tan rápido como sea posible para beneficio del cliente.

#### 46. ¿Qué es un proceso ágil?

- Es un conducido por descripciones del cliente de qué es requerido (escenarios), produciendo la retroalimentación entre el equipo y el cliente.
- Reconoce que los planes son de breve duración.
- Desarrolla software iterativo incremental con un énfasis especial en actividades de la construcción.
- Entrega múltiples “incrementos de software” (prototipos ejecutables o porciones de un Sistema Operativo, en periodos cortos de tiempo. Permite que el cliente evalúe en forma regular el incremento de software.
- Se adapta cuando ocurren los cambios.

#### 47. ¿De dónde surge la gestión ágil?

Al surgir en los años 80 una forma de gestionar proyectos, se hizo necesario añadir un “apellido” al término “gestión de proyectos” para matizar si se refiere a la nueva o a la de siempre.

La nueva, al autodenominarse ágil, obligó a dar un apellido al modelo de gestión de proyectos que hasta entonces, por único, no lo había necesitado.

#### 48. Sobre qué se formula el resultado una gestión ágil de proyectos?

La *gestión de proyectos* predictiva se ha desarrollado sobre las premisas:

- Todos los proyectos mantienen características y comportamientos regulares.
- El objetivo de la ejecución de un proyecto es lograr el producto previsto en el tiempo planificado, sin desbordar los costes estimados.

La *gestión ágil* al cuestionar la validez de las premisas de la gestión tradicional:

- No hay forma única y válida para gestionar cualquier tipo de proyecto.
- Hay proyectos que tienen como objetivo valor para el producto, y no funcionalidad, fecha y costes.

#### 49. ¿Qué características relevantes del proyecto y de la organización son relevantes para decidir/elegir el estilo de gestión más adecuado?

Las características relevantes del proyecto para decidir el estilo de gestión más adecuado son:

- Prioridad del negocio, estabilidad de los requisitos, rigidez del producto, coste de prototipado, criticidad del sistema y tamaño del sistema.

Las características relevantes dela organización para facilitar la elección del modelo de gestión más adecuado son:

- Nivel profesional, cultura organizativa y entorno de desarrollo.

El resultado en la gestión ágil de proyectos, que no se formula sobre el concepto de anticipación (requisitos, diseño, planificación y seguimiento) sino sobre el de adaptación (visión, exploración y adaptación)

## 50. ¿Cuáles son los objetivos de la gestión ágil? Ver las cuatro demandas y su descripción

La gestión ágil de proyectos tiene como objetivos dar garantías a las cuatro demandas principales de la industria en la que ha generado: Valor. Reducción del tiempo de desarrollo. Agilidad. Fiabilidad.

- **Valor**

La gestión ágil es necesaria en mercados rápidos.

Su objetivo es dar el mayor valor posible al producto; y en el mercado en el que trabaja, este valor es directamente proporcional a la respuesta que pueda ofrecer en:

- **Innovación:** La permanencia de estas empresas depende de su capacidad de innovación continua. Del lanzamiento continuo de novedades, que tienen que competir con los productos de una competencia que a su vez también innova sus productos de forma continua.

- **Flexibilidad**

- En las circunstancias de velocidad del mercado actual, no sólo es importante el valor en el momento del lanzamiento, sino también su capacidad de adaptación y lanzamiento, sino también su capacidad de adaptación y evolución a través de versiones, modificaciones, actualizaciones o ampliaciones.
- Ahora no ocurre como en los años 50 en los que un modelo de auto-radio permanecía años sin desfasarse. Ahora como en Alicia en el país de las maravillas “necesitas correr todo lo que puedas para permanecer en el mismo lugar.

- **Agilidad**

Capacidad para adaptar el curso del desarrollo a la evolución de los requisitos y a las circunstancias del entorno.

- **Reducción del tiempo de desarrollo**

- Las estrategias de la gestión ágil para producir resultados en menos tiempo que la gestión tradicional son:
- **Solapamiento de fases de desarrollo.**
- **Entrega temprana de los primeros incrementos funcionales de producto**, que corresponden con las partes que con mayor urgencia necesita el cliente, de forma que pueda lanzar la primera versión de producto con la mayor rapidez

- **Resultados fiables**

- Los procesos de producción empleados por la gestión de proyectos tradicional tienen como finalidad la repetitividad de los resultados: conseguir el trabajo planificado (y conocido de antemano) en el plazo planificado y por el coste previsto. La gestión ágil no tiene un carácter predictivo o de anticipación. No conoce de antemano el detalle del producto o servicio que va a desarrollar; por eso su objetivo no es la fiabilidad en el cumplimiento de los planes, sino en el valor del resultado y el tiempo de salida al mercado.
- Los procesos de la gestión tradicional son buenos cuando consiguen desarrollar de forma repetible los productos especificados en el tiempo y con los costes previstos.



- Los procesos de la gestión ágil son buenos, cuando consiguen entregar de forma repetible valor innovador.

### 51. ¿Cuáles son las diferencias entre la gestión Tradicional y ágil?

La gestión ágil a diferencia de la tradicional, refleja las preferencias declaradas por el manifiesto ágil:

- La capacidad de respuesta al cambio, sobre el seguimiento de un plan
- Los productos que funcionan frente a especificaciones y documentaciones innecesarias.
- La colaboración con el cliente frente a la negociación contractual.
- A las personas y sus interacción por encima de los procesos y las herramientas

Resumiendo:

Gestión Tradicional	Gestión ágil
Especialización	Equipo Multidisciplinario
Fases	Solapamiento
Requisitos Detallados	Visión de Producto
Seguimiento del Plan	Adaptación a los cambios

## UNIDAD 2. MODELOS DE CALIDAD DE SOFTWARE Y ASEGURAMIENTO DE LA CALIDAD

### 2.1 Introducción a la calidad de Software

### 52. ¿Quién hace la calidad en un proyecto? ¿Por qué es importante?

- ¿Quién lo Hace?
  - Todos los involucrados en el proceso de software son los responsables.
- ¿Por qué es importante?
  - Puede hacerse bien o puede repetirse. Si se le da importancia en un proyecto se reducen las repeticiones. Esto reduce los costos y el tiempo.

### 53. ¿Qué es y qué significa calidad? Que es la calidad para IEEE y para el SEI?

*Calidad de un producto de software* es el cumplimiento de los requerimientos especificados y las expectativas de los usuarios y/o clientes (IEEE)

*Calidad de software* es cumplir con requerimientos del cliente o usuario y con la formalidad del proceso con que es desarrollado (SEI)

### 54. ¿Cuándo se construye la calidad? ¿De qué depende?

La calidad se construye a lo largo de todo el proyecto de desarrollo. Para tener sistemas de calidad hay que tener procesos probados, maduros y formales de desarrollo de software, sustentados en normas y estándares precisos.

La calidad de software **depende** de:

- Calidad del Producto

- Calidad del Proceso
- Calidad de la información

### 55. ¿Cómo se define la Calidad de un producto? (atributos)

En síntesis, la calidad intrínseca de un producto se puede definir de la siguiente forma:

1. Cumplir satisfactoriamente la función que tiene asignada.
2. No presentar fallas o deficiencias.
3. Satisfacer al cliente que lo adquiere o utiliza.
4. Cumplir con especificaciones técnicas establecidas.

Además de estos 4 atributos que le interesan directamente al cliente se deben considerar **los intereses de otras personas/empresas** que pueden ser afectados por el producto o servicio como ser la sociedad, los accionistas, el gobierno, los empleados, etc.

### 56. ¿Cuál es la fórmula de la satisfacción del usuario?

**Satisfacción del Usuario** = Producto que funciona + Buena calidad + entrega dentro del Presupuesto y Plazo

### 57. ¿Qué es la calidad del diseño? ¿Y la calidad de la conformidad?

#### Calidad del Diseño

- Se centra en el grado en que el diseño cumple las funciones y características especificadas en el modelo de requerimientos.

#### Calidad de Conformidad

- Se centra en el grado de implementación se apega al diseño y en el que el sistema resultante cumple sus metas de requerimientos y desempeño.

### 58. Defina la calidad del software

**Calidad de software** es un proceso eficaz de software que se aplica de manera que crea un producto útil que proporciona valor medible a quienes lo producen y quienes lo utilizan.

### 59. ¿Cómo se define la calidad desde el punto de vista del cliente y desde el punto de vista industrial de un producto?

La calidad desde el punto de vista del cliente significa que “el sistema se adecue al usuario”. Aquí el cliente define la calidad del producto o servicio en base a qué tan bien satisface sus necesidades.

Desde el punto de vista industrial de un producto la calidad del software es el grado con el que el sistema componente o proceso cumple los requerimientos especificados, logrando mayor productividad, un menor esfuerzo de mantenimiento bajar los costos de asistencia al cliente, mejorar la trazabilidad, lograr mejores precios en el mercado y alta satisfacción del cliente o usuario.

### 60. En qué se centra el mejoramiento de la calidad.

Se centra en los siguientes puntos:

- Un **proceso eficaz de Sw** establece la infraestructura que da apoyo a cualquier esfuerzo de elaboración de un producto de Sw de alta calidad.
- Un **producto útil** entrega contenido, funciones y características que el usuario final desea, de forma confiable y libre de errores. Un producto útil siempre satisface los requerimientos establecidos por los participantes y satisface el conjunto de requerimientos (por ej. Facilidad de uso) con los que se espera cuenta un Sw de alta calidad.
- Al **agregar valor para el productor y el usuario** de un producto, el Sw de alta calidad proporciona beneficios a la organización que lo produce y a la comunidad de usuarios finales.
- Requiere un menor esfuerzo de mantenimiento.

#### 61. ¿En qué consiste el paso llamado Kaizen en calidad? ¿Qué significa Kaizen?

Kaizen, es el primer paso, el objetivo es desarrollar un proceso que sea visible, repetible y medible, se refiere a un sistema de mejora continua del proceso.

Kaizen significa dar pequeños pasos diarios para la mejora continua.

#### 62. ¿Qué es el aseguramiento de la calidad del software ACS?

El **aseguramiento de calidad del software** es el conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza en que el producto (software) requiere para satisfacer los requerimientos dados de calidad por parte del cliente.

El aseguramiento de calidad se enfoca en identificar y evaluar los defectos que puedan afectar al software

#### 63. ¿Qué entidades intervienen en el ACS?

El Aseguramiento de la Calidad del Software (ACS) está asociado a *dos entidades*:

1. *Ingenieros de Sw*: hacen el trabajo técnico
  - Abordan la calidad y la controlan, aplicando métodos y medidas técnicas sólidas, realizan revisiones técnicas y hacen pruebas de sw planificadas.
2. *Un grupo de ACS*: planean, supervisan, registran, analizan y reportan acerca de la calidad.
  - Dan soporte al equipo para lograr un producto final de alta calidad

#### 64. ¿Cuáles son las tres actividades principales del ACS?

1. Aseguramiento de la calidad: Establecer un marco de trabajo de procedimientos y estándares corporativos que conduzcan a la obtención de software de alta calidad.
2. Planificación de la calidad: Seleccionar procedimientos y estándares adecuados a partir de ese marco de trabajo y adaptación de estos para un proyecto de software específico.
3. Control de la calidad: Definir y aplicar procesos que aseguren que los procedimientos y estándares son seguidos por el equipo de desarrollo.

## 65. ¿Cuáles son los tipos de estándares de ACS? Mencione ejemplos

Hay dos tipos de estándares:

Estándares del producto: se aplican al producto a desarrollar

- Estándares de documentos: por ej. estructura del documento de requerimientos a producir, formato del plan del proyecto.
- Estándares de documentación: por ej. encabezados estándar de comentarios para la definición de una clase
- Estándares de codificación: por ej. cómo utilizar un lenguaje de programación

Estándares del proceso: definen los procesos a seguir durante el desarrollo. Por ejemplo:

- Conducto para la revisión del diseño
- Sometimiento de documentos a revisiones
- Proceso de entrega de las versiones
- Proceso de aprobación del plan de proyecto
- Proceso del control de cambio
- Proceso de registro de las pruebas

## 66. ¿En qué consiste IDEAL?

*IDEAL* es un modelo elaborado por el SEI para guiar el inicio, planificación e implementación de iniciativas de mejora para el proceso de software en las organizaciones.

El modelo consiste de cinco fases, compuestas por actividades:

**Initiating (Iniciar)**: Preparar el terreno para un esfuerzo exitoso de mejora

1. Establecer el contexto (qué objetivos de la empresa serán beneficiados)
2. Desarrollar el sponsoreo (obtener compromisos)

**Diagnosing (Diagnosticar)**: Determinar dónde estamos con relación a dónde queremos estar

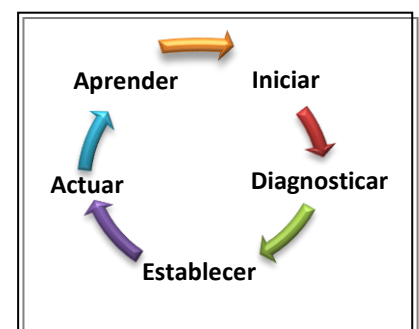
1. Caracterizar los estados actuales y deseados
2. Desarrollar recomendaciones

**Establishing (Establecer)**: Planificar los pasos específicos para alcanzar metas

1. Establecer las estrategias y las prioridades
2. Desarrollar una propuesta
3. Planificar acciones

**Acting (Actuar)**: Realizar las tareas de acuerdo con el plan

1. Crear la solución
2. Ejecutar el proyecto piloto de la solución
3. Refinar la solución
4. Implementar la solución



**Learning (Aprender)**: Aprender de las experiencias y mejorar la capacidad para adoptar nuevas tecnologías en el futuro

1. Analizar y validar
2. Proponer futuras acciones

## 2.2 Modelo de Madurez de la Capacidad

### 67. ¿Qué permite un sistema de certificación de calidad? ¿Cuáles son sus pilares básicos?

Un *sistema de certificación de calidad* permite una valoración independiente que debe demostrar que la organización es capaz de desarrollar productos y servicios de calidad.

Los *pilares básicos* de la certificación de calidad son:

- Una metodología adecuada,
- Un medio de valoración de la metodología,
- La metodología utilizada y el medio de valoración de la metodología deben estar reconocidos ampliamente por la industria.

### 68. ¿Cuáles son los beneficios de la mejora del proceso?

- Predictibilidad de la planificación y el presupuesto
- Tiempo de desarrollo
- Productividad
- Calidad (medida en nº de defectos)
- Satisfacción del Cliente
- Satisfacción de los empleados

### 69. ¿Cuáles son las actividades de QA? (ver gráfica)



### 70. ¿Cuáles son las acciones básicas para lograr la calidad?

Existen tres acciones básicas para lograr la calidad:

- Acciones Preventivas: es el conjunto de actividades y medidas concebidas y ejecutadas para prevenir la ausencia de calidad. (normas, estándares, métodos, procesos, etc.)

- Acciones Detectivas: es el conjunto de actividades y medidas orientadas a detectar tempranamente la falta de calidad. (inspecciones, revisiones, etc.)
- Acciones Correctivas: es el conjunto de actividades y medidas cuyo propósito es descubrir y corregir la falta de calidad. (testing)

**71. ¿Cómo se identifica una empresa inmadura de una madura?**

Empresa Inmadura	Empresa Madura
<ul style="list-style-type: none"><li>• Apaga fuegos</li><li>• Tiene pocos recursos propios</li><li>• Tiene éxito gracias a los héroes</li><li>• Hay altibajos en la productividad por rotación de recursos</li><li>• Planificaciones son poco realistas.</li><li>• Mucho esfuerzo dedicado a “mantenimiento”</li><li>• Los plazos de entrega son impredecibles</li><li>• Los empleados están descontentos</li></ul>	<ul style="list-style-type: none"><li>• Tiene procesos definidos</li><li>• Tiene responsabilidades definidas</li><li>• El conocimiento está en la organización</li><li>• Resultados predecibles</li><li>• Entrega con la calidad esperada</li><li>• Cumple plazos de entrega</li><li>• Incrementa la productividad</li><li>• Reconocer las mejoras</li><li>• Satisface a los clientes</li><li>• Los empleados están a gusto</li></ul>

**72. ¿Cómo surge CMMI? ¿A qué se orienta el modelo?**

El Departamento de Defensa de los Estados Unidos trabajó en cooperación con el Software Engineering Industrial Association (NDIA) para conjuntamente promover el desarrollo del Capability Maturity Model Integration. CMMI surge de la necesidad de integrar varios estándares que las empresas estaban adoptando (SW\_CMM, IPD\_CMM, SE\_CMM entre otros).

En el año 2000 se produjo el primer modelo integrado CMMI. Este modelo consiste en un conjunto de mejores prácticas.

Se orienta a la mejora de procesos e ingeniería y a un diseño de marco de trabajo bien definido para incluir disciplinas adicionales para minimizar el desarrollo de modelos incompatibles en el futuro.

**73. ¿Cuáles son los objetivos iniciales de CMMI?**

- Los objetivos iniciales son:
- Eliminar de inconsistencias
- Reducir duplicaciones
- Incrementar la claridad y la comprensión
- Proporcionar terminología común
- Proporcionar estilos consistentes
- Establecer reglas de construcción uniformes
- Mantener componentes comunes
- Asegurar la consistencia con ISO 15504
- Ser susceptible a la inferencia de esfuerzos legales

#### 74. ¿Qué son los niveles de madurez?

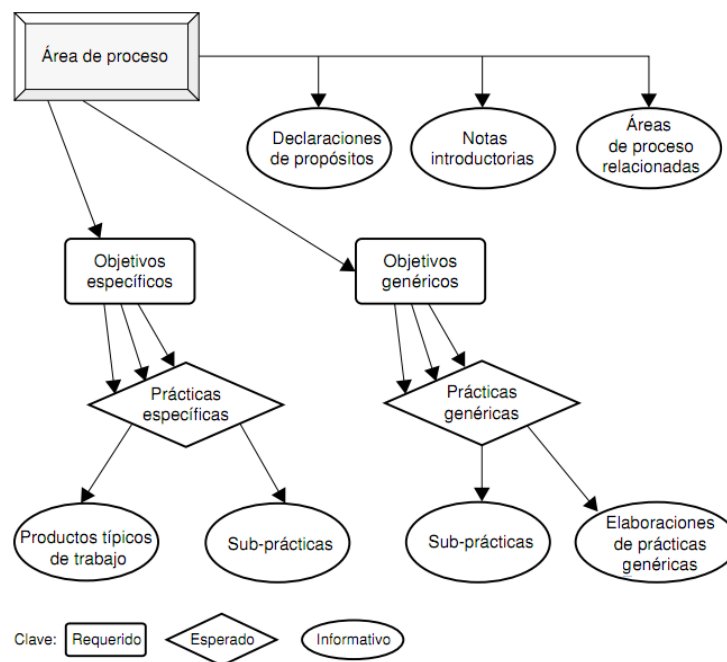
Un nivel de una organización proporciona un camino para predecir el funcionamiento futuro de una organización dentro de una disciplina dada o un conjunto de disciplinas.

#### 75. Nombre los cinco niveles de madurez de CMMI.

- **Nivel 1 (Inicial):** El proceso es impredecible, es reactivo y pobremente controlado.
- **Nivel 2 (Administrado):** El proceso es reactivo y se caracteriza por su aplicación a proyectos.
- **Nivel 3 (Definido):** El proceso es proactivo y se ve a nivel de la organización.
- **Nivel 4 (Administrado Cuantitativamente):** El proceso es medido y controlado.
- **Nivel 5 (Optimizado):** El proceso se enfoca en la mejora continua.

#### 76. ¿Qué son las áreas de proceso de CMMI? ¿Cómo se estructura cada área de proceso (objetivos, metas y prácticas)?

Un **área de proceso** es un conjunto de mejores prácticas relacionadas a un área, que cuando se implementan colectivamente satisfacen un conjunto de objetivos considerados importantes para lograr una mejora significativa en esa área.

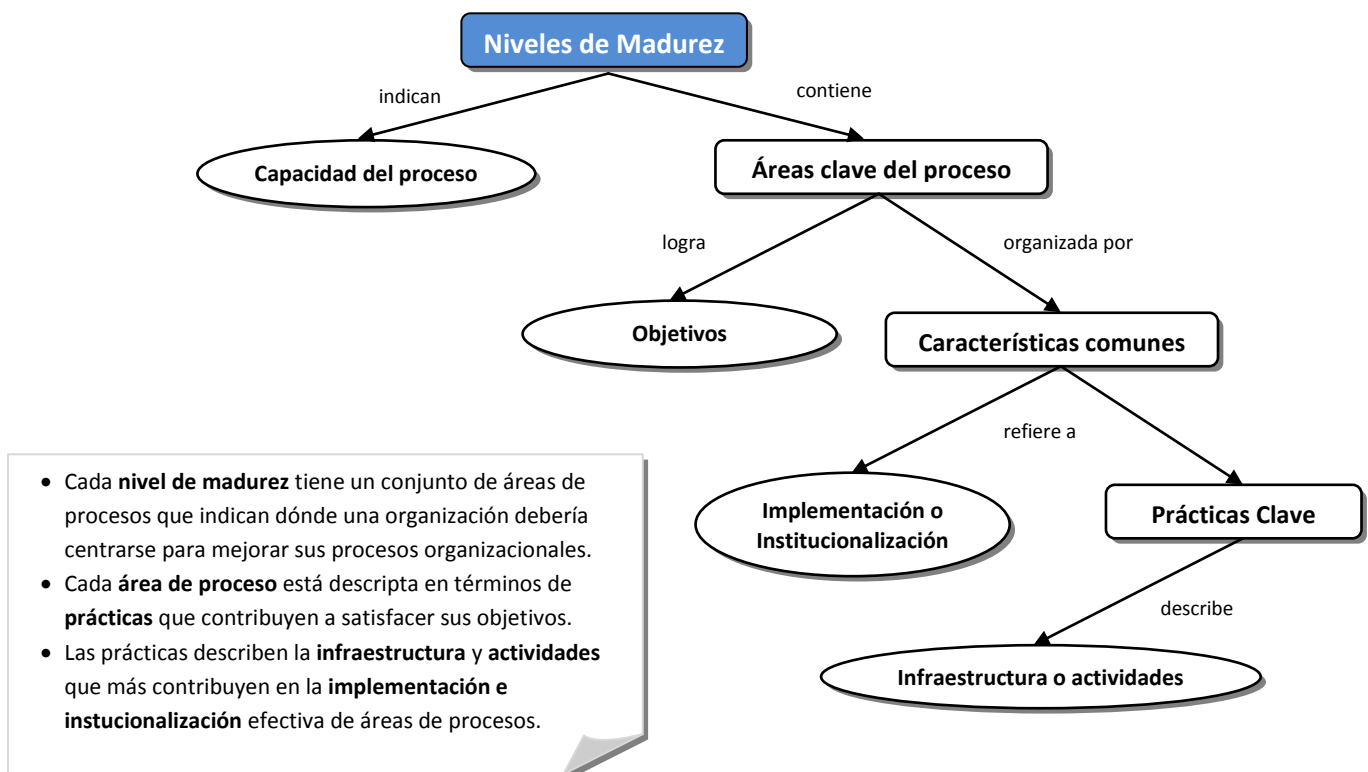


- **Áreas de Proceso:** CMMI define cada área de proceso en término de metas específicas y de prácticas específicas para lograr dichas metas.
  - **Metas Específicas:** es la descripción de una actividad que se considera importante para alcanzar la meta específica asociada.
  - **Prácticas Específicas:** desglosan una meta en un conjunto de actividades relacionadas con el proceso. Implementan el proceso.
    - *Productos de trabajo típicos:* lista muestras de resultados de una práctica específica. Estos ejemplos se denominan productos de trabajo típicos porque a

menudo hay otros productos de trabajo que son igual de eficaces pero no están en la lista

- **Subprácticas:** es una descripción detallada que proporciona una guía para interpretar e implantar una práctica específica.
- **Metas Genéricas:** la misma declaración de la meta se aplica a múltiples áreas de proceso. Una meta genérica describe las características que deben estar presentes para *institucionalizar* los procesos que implementan un área de proceso.
- **Prácticas Genéricas:** la misma práctica se aplica a múltiples áreas de proceso. Una práctica genérica es la descripción de una actividad que se considera importante para el logro de la meta genérica asociada

### 77. Diagramar la estructura del modelo de CMMI (ver gráfica y resumen)



### 78. ¿Cómo se clasifica el material?

Escala que describe la importancia y el papel de los materiales contenidos en CMMI:

- Materiales requeridos
  - El componente requerido es el objetivo.
  - Un objetivo presenta un estado final deseable.
    - Su logro indica que un cierto grado del proyecto y control de procesos ha sido alcanzado.
    - Pueden ser: Objetivos genéricos o específicos.



- Materiales esperados
  - Declaración de una práctica (único material esperado)
  - Una práctica representa un instrumento esperado p objetivo.
    - Pueden ser: prácticas específicas o genéricas.
- Materiales informativos: CMMI tiene 10 componentes informativos
  - Propósito
  - Nota introductoria
  - Referencia
  - Nombres
  - Tabla de relación entre prácticas y objetivos
  - Notas
  - Productos de trabajos típicos
  - Subprácticas
  - Amplificaciones disciplinarias
  - Elaboración de prácticas genéricas

**79. ¿Cuáles son las dos representaciones del modelo de CMMI? ¿Qué factores tomaría en cuenta para elegir una u otra?**

Existen 2 representaciones que describen dos formas distintas de llevar a cabo la mejora de los procesos que aplican las organizaciones:

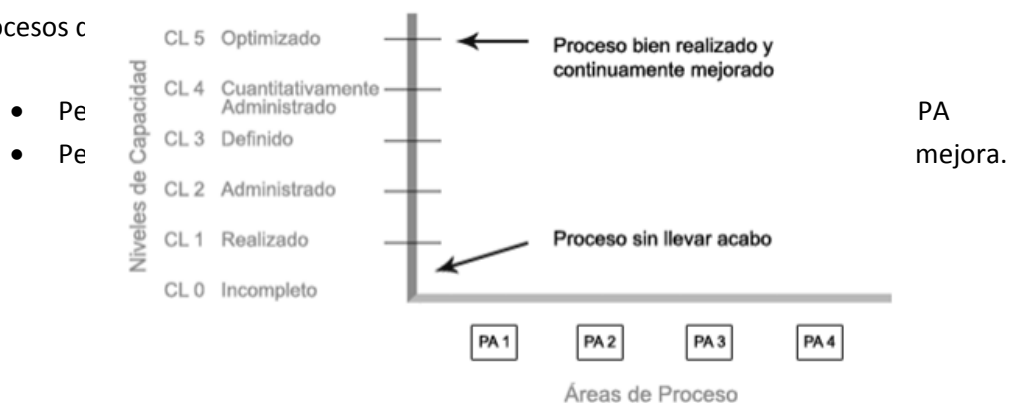
1. Secuencial o por Etapas (staged)

Tiene en cuenta a la organización como un conjunto y propone un camino a seguir que consta de sucesivos niveles para mejorar la habilidad de la organización para comprender y controlar los procesos que en ellas se desarrollan.

- Sumariza los resultados de mejoras de procesos en forma simple, un nivel de madurez.

2. Continua (continuos)

Se centra en actividades individuales y permite a la organización elegir qué proceso o conjunto de procesos c



Para poder elegir entre un modelo o el otro hay que basar la decisión en tres factores:

- Negocios:
  - Continua: evaluar procesos y determinar cuán bien estos procesos soportan y alcanzan los objetivos de negocio.

- Por etapas: la organización está interesada en benchmarking<sup>1</sup> con sus competidores y/o publicar resultados.
- Herencia: si una organización tiene experiencia con una representación (continua o por etapas) sería adecuado que continúe con esa representación.
- Cultura:
  - Continua: organizaciones con fuerte cultura en ingeniería de sistemas
  - Por etapas: organizaciones con fuerte cultura en ingeniería de software

## 80. ¿Qué beneficio trae institucionalizar las actividades repetibles? (con CMMI)

Un proceso está *institucionalizado* cuando está arraigado en la forma de trabajar, de tal manera que se sigue incluso en momentos de crisis.

Los beneficios de la institucionalización de procesos repetibles son:

- Facilidad de rotación de personas entre proyectos
- Medición del impacto en los cambios de requisitos
- Reducción del número de entregables (de 8-10 a 4-5) por proyecto
- Control de incidencias
- Control de versiones y recuperación ante fallos
- Información detallada de seguimiento de proyectos
- Información detallada de áreas de conocimiento
- Métricas. Toma de decisiones
- Cultura de mejora
- Alta receptividad del cambio

## 81. Nombrar los tres pasos del proceso de acreditación

1. *Planificación inicial*
  - Desarrollo de un plan
  - Preparación del equipo
  - Información a los participantes
  - Administrar un cuestionario de evaluación y examinar resultados
  - Documento inicial
2. *Evaluación "on-site"*
  - Preparación
  - Entrevistas
  - Consolidación de información
  - Preparación y redacción de lo encontrado
  - Determinar la clasificación y presentación (opcional)
3. *Exposición de resultados*
  - Presentación del sponsor

---

<sup>1</sup> **Benchmarking:** proceso sistemático y continuo para evaluar comparativamente los productos, servicios y procesos de trabajo en organizaciones.

- Presentación al Director de la empresa (opcional)

## **82. ¿Qué es SCAMPI? ¿En qué consiste? ¿Qué proporciona?**

El *SCAMPI* es un método de evaluación estándar de CMMI para la mejora de procesos (Standard CMMI Appraisal Method for Process Improvement) define el método para identificar fortalezas, debilidades y clasificación con respecto al modelo de referencia.

Consiste en un conjunto estructurado de actividades tales como entrevistar, revisión de documentos, presentaciones y análisis de respuestas a cuestionarios.

Proporciona un método de evaluación en el contexto de mejora interna de procesos, selección de proveedores y seguimiento de procesos.

Hay tres clases de métodos SCAMPI:

- Clase A (corresponde al CBA)
  - Método completo usado para evaluaciones en profundidad.
  - Evalúa el nivel de madurez de la organización.
- Clase B (Corresponde a las pre-evaluaciones)
  - Menos sistemático y profundo, útil para evaluaciones iniciales.
  - No proporciona puntuación sobre el nivel de madurez.
- Clase C
  - Barato (comparado con A y B) y rápido.
  - Evalúa ciertas áreas de riesgos con tomas no extensivas de datos.

## **83. En qué se focaliza el Control de Calidad y el ACS?**

El primero se focaliza en medir y verificar el entregable propiamente dicho. El segundo se focaliza en los procesos que fueron utilizados para crear el entregable.

## **84. Qué se tiene que hacer para tener ACS?**

1. Se tiene que definir lo que es calidad del SW.
2. Crear las actividades para garantizar que todo producto de la ing. de SW tenga alta calidad.
3. Desarrollarse el control de calidad y las actividades para asegurar que está en todo proyecto de SW.
4. Usarse métricas para desarrollar estrategias para mejorar el proceso de SW y la calidad del producto final.

## **85. Qué incluye un proceso de ACS?**

1. Un proceso de ACS.
2. Tareas específicas de ACS y control de calidad (revisiones técnicas y estrategias de prueba).
3. Prácticas eficaces de Ing. SW (métodos y herramientas).
4. Control de todos los productos del trabajo de SW y de los cambios que sufren.
5. Un procedimiento para garantizar el cumplimiento de estándares del desarrollo de SW (cuando sea aplicable).
6. Mecanismos de medición y reporte.

## **86) Cuáles son los elementos de ACS?**

### **1. Estándares**

Se adoptan voluntariamente o se imponen.

ACS asegura que se sigan y los productos de trabajo de apeguen a ellos.

### **2. Revisiones y auditorias**

Revisiones técnicas para detectar errores.

Auditorias para garantizar que se sigan los lineamientos de calidad.

### **3. Pruebas**

Son una función del control de calidad que busca detectar errores.

ACS Garantiza que se planeen de forma apropiada y con eficiencia.

### **4. Colección y análisis de los errores**

ACS reúne y analiza errores y datos acerca de los defectos para entender mejor como se comenten los errores.

ACS asegura que se hayan instituido prácticas adecuadas de Administración de cambios.

### **5. Administración del cambio**

ACS asegura que se hayan instituido prácticas adecuadas de administración de cambios.

### **6. Educación**

La organización de ACS lleva el liderazgo en la mejora del proceso de sw y es clave para proponer y patrocinar programas educativos.

### **7. Administración de proveedores**

ACS garantiza que se obtenga sw de alta calidad a partir de las sugerencias específicas de calidad que el proveedor debe seguir y se incorporen cláusulas de calidad en los contratos.

### **8. Administración de la seguridad**

ACS garantiza que para lograr la seguridad del sw, se utilicen el proceso y la tecnología apropiados.

### **9. Seguridad**

ACS es responsable de evaluar el efecto de las fallas del sw y de dar los pasos que se requieren para disminuir el riesgo.

### **10. Administración de riesgos**

La organización de ACS garantiza que las actividades de administración de riesgos se efectúen en forma apropiada y que se establezcan planes de contingencia relacionados con los riesgos.

## **87. Qué es la confiabilidad del SW? Cómo se mide?**

La confiabilidad o fiabilidad del sw es la “probabilidad que un programa realice su objetivo satisfactoriamente (sin fallos) en un determinado periodo de tiempo y en un entorno concreto (denominado perfil operacional)”.

Puede ser medida o estimada mediante datos históricos o de desarrollo.

## **88. Qué proporciona un plan de ACS? Cómo funciona?**

El plan de ACS proporciona un mapa para instituir el ACS.

Funciona como plantilla para las actividades de ACS que se instituyen para cada proyecto de sw.

## UNIDAD 3: GESTIÓN DE CONFIGURACIÓN DE SOFTWARE

### 3.1 Aseguramiento de la calidad del software

#### 89. ¿Qué es la Gestión de Configuración de SW? ¿Qué busca?

Es el conjunto de actividades y procesos necesarios para identificar y definir los elementos de la configuración de un sistema, controlando la entrega y los cambios de estos elementos a través del ciclo del vida del sistema, almacenando el estado de los elementos de la configuración y de las peticiones de cambio, y verificando que se cumple respecto a los requisitos especificados.

Busca:

- Llevar un control y registro de los cambios del sistema.
- Reducir errores.
- Aumentar la calidad y la productividad.
- Evitar los problemas derivados de una incorrecta sincronización en los cambios.

#### 90. ¿Qué es un elemento de la Configuración del SW (ECS)?

Es cualquier producto (SW – HW) que se puede definir y controlar de forma separada. Se controlan los cambios.

#### 91. ¿En qué consiste la actividad de identificar la configuración?

La actividad de identificación de la configuración identifica los elementos que van a ser controlados, establece esquemas para la identificación de los elementos y sus versiones, establece las herramientas y técnicas a usar para adquirir y gestionar los elementos controlados.

Proporcionan la base para otras actividades de gestión de configuración.

#### 92. Al desarrollar procedimientos para solicitar e implantar los cambios, ¿qué se debe especificar?

- Quién solicita los cambios
- Cómo se notifican los cambios
- Cómo se evalúa el impacto
- Quién evalúa el impacto
- Quién acepta o rechaza el cambio
- Quién modifica los distintos producto (responsable de cada producto)

#### 93. ¿Cómo elegimos los componentes a gestionar?

Se pueden tener en cuenta los siguientes criterios:

1. Productos de trabajo que vayan a ser utilizados por dos o más grupos.
2. Productos de trabajo que puedan cambiar con el tiempo debido a cambios en requisitos o errores.
3. Productos que dependan de otros en el sentido de que un cambio en uno de ellos implique un cambio en los otros.
4. Productos de trabajo que sean críticos para el proyecto.

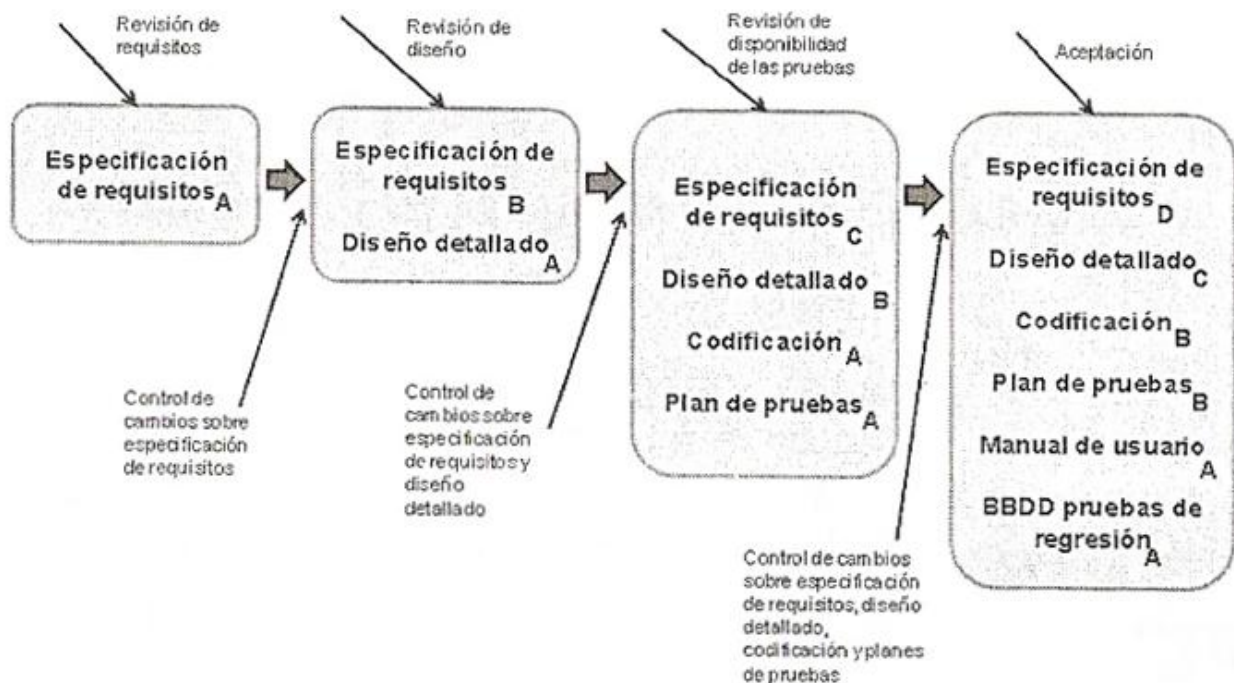
#### 94. ¿Qué es una Línea Base?. Vea ejemplos.

Es una especificación o producto que se ha revisado formalmente y sobre los que se ha llegado a un acuerdo, y que de ahí en adelante sirve como base para un desarrollo posterior y que puede cambiarse solamente a través de procedimientos formales de control de cambios.

Ejemplos:

- LB Funcional: al final de la fase de análisis. Plan de análisis, requisitos del sistema, plan de calidad, plan de gestión de la configuración, plan de pruebas de aceptación.
- LB de Asignación: al final de la fase de diseño de alto nivel. Arquitectura, interfaces de subsistemas, plan de pruebas de integración.
- LB de Diseño: al final de la fase de diseño detallado. Diseño de subsistemas y plan de pruebas de integración.
- LB de Producto: al final de la codificación. Código fuente, objeto y ejecutable, resultados de pruebas de integración y versión preliminar de los manuales.
- LB de Explotación: al final de la implantación. Resultados de pruebas de sistema y documentación de usuario.

#### 95. ¿Cómo incorporo elementos a una LB? Grafica de ejemplo



#### 96. ¿Qué es la Gestión de Cambios?

La Gestión de Cambios se ocupa de seguir la pista de cambios (los sistemas de SW están sujetos a continuas peticiones de cambio de los usuarios, desarrolladores y exigencias del mercado) y asegurar que son implementados de la forma más rentable.

#### 97. ¿Cuáles son los niveles de Gestión de Cambios?

1. Informal: si el ECS no forma parte de la Línea de Base.

2. Semiformal: cuando el ECS en cuestión no forma parte de la LB, pero su cambio afecta a ECS que sí la conforman.
3. Formal: cuando el ECS forma parte de la LB.

**98. ¿Qué es la Gestión de Versiones?**

- Consiste en inventar un esquema de identificación para las versiones de un sistema.
- Planificar cuando debe producirse una nueva versión del sistema.
- Asegurar que los procedimientos y herramientas de gestión de versiones se aplican de manera apropiada.
- Planificar y distribuir las nuevas entregas del sistema.

**99. Definición de versión, variante y entrega/liberación.**

- **Versión:** una instancia del sistema que difiere de alguna manera de otras instancias.
- **Variante:** una instancia de un sistema que es funcionalmente idéntica pero diferentes en su aspecto no funcional.
- **Entrega/Liberación:** una instancia de un sistema que se distribuye a los usuarios externos a equipo de desarrollo.

**100. ¿Qué significa el término liberación?**

El término se utiliza en este contexto para referirse a la distribución del elemento de configuración de SW fuera de la actividad de desarrollo. Esto incluye tanto liberaciones internas como distribuciones al cliente.

**101. ¿Cuáles son las principales actividades que se deben tener en cuenta en la planificación de GC?**

- Identificación de elementos de configuración
- Control de la configuración
- Registro del estado de la configuración
- Auditorías de configuración
- Gestión del despliegue