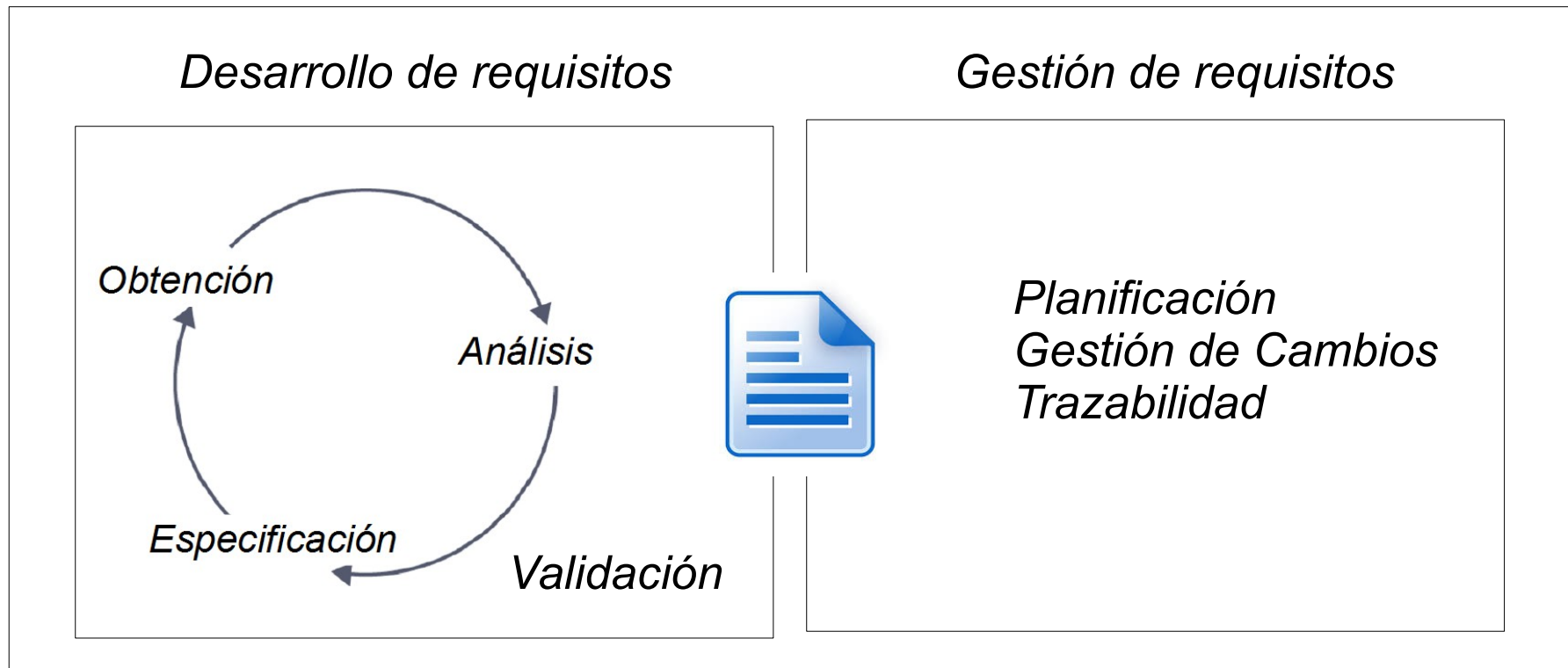


Ingeniería de Software

Ingeniería de Requisitos Clase 4

Actividades de la ingeniería de requisitos



Stakeholders

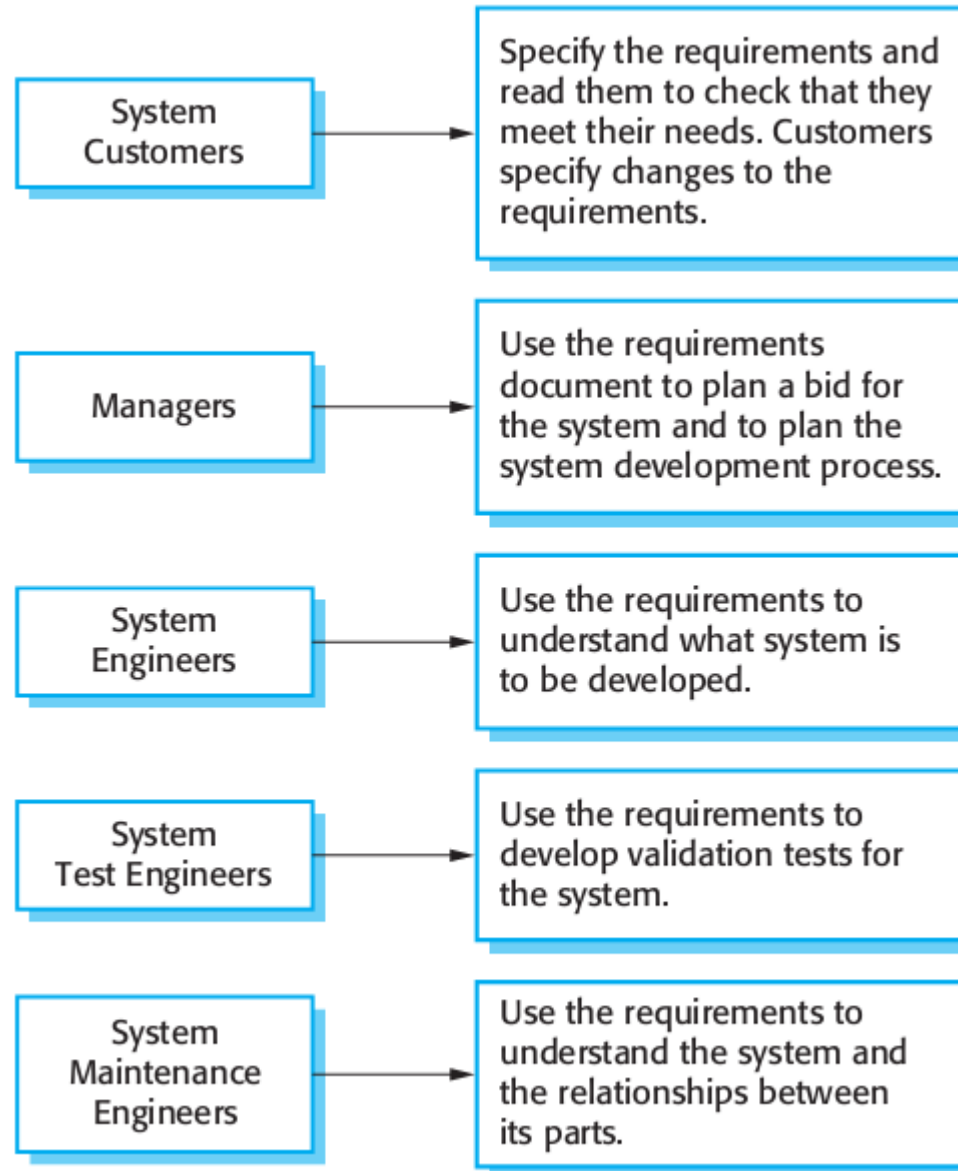
Temario

- Documento de requisitos
- Priorización de requisitos
- Validación de requisitos
- Gestión de requisitos

Documento de requisitos de software

- El documento de requisitos de software (SRS) es una declaración oficial de lo que se requiere de los desarrolladores del sistema.
- Debe incluir una definición de los requisitos del usuario y una especificación de los requisitos del sistema.
- No es un documento de diseño. Se debe indicar lo que el sistema debe hacer y no la forma en la que lo debe hacer.
- Metodologías ágiles prefieren, en lugar de construir un documento formal, recolectar requisitos de forma separada. Por ejemplo: las historias de usuario en Extreme Programming (XP).

Usuarios de un documento de requisitos



Variabilidad del documento de requisitos

- La información del documento de requisitos depende del tipo de sistema y el enfoque de desarrollo usado.
- Los sistemas desarrollados de forma incremental tienen típicamente menos detalles en el documento de requisitos.
- Se han diseñado estándares para documentos de requisitos ej: estándar de IEEE 29148:2011 (antes el IEEE 830:1998).

Estructura de un documento de requisitos

Capítulo	Descripción
Prefacio	Definir los lectores esperados y describir el historial de versiones, incluyendo una justificación para la creación de una nueva versión y un resumen de los cambios realizados en cada versión.
Introducción	Describir las necesidades y las funciones del sistema y explicar como va a funcionar con otros sistemas. También debe describir como el sistema se ajusta a los objetivos del negocio o a los planes estratégicos de la organización.
Glosario	Definir los términos técnicos usados en el documento. No se deben hacer suposiciones acerca de la experiencia o de los conocimientos del lector.
Definición de requisitos de usuario	Aquí, se describen los servicios prestados para el usuario. En esta sección también deben ser descritos los requisitos no funcionales del sistema. Esta descripción puede utilizar el lenguaje natural, diagramas u otra notación entendible por los clientes. Las normas de producto y proceso que deben seguir deben ser especificados.
Arquitectura del sistema	Este capítulo debe presentar una visión de alto nivel de la arquitectura del sistema, mostrando la distribución de las funciones sobre los módulos del sistema.

Estructura de un documento de requisitos

Capítulo	Descripción
Especificación de requisitos del sistema	Describir con más detalle los requisitos funcionales y no funcionales, así como interfaces con otros sistemas.
Los modelos del sistema	Aquí se puede incluir modelos gráficos del sistema que muestren las relaciones entre los componentes del sistema y la relación con su entorno.
Evolución del sistema	Describir los supuestos fundamentales en los cuales está basado el sistema y cualquier cambio previsto debido a la evolución del hardware, cambios en las necesidades de los usuarios, etc. Esta sección es útil para los diseñadores del sistema ya que puede ayudar a evitar que las decisiones de diseño restrinjan posibles cambios futuros.
Apéndices	Debería proporcionar información detallada y específica relacionada con la aplicación que se está desarrollando, por ejemplo hardware y descripción de la base de datos.
Índice	Se pueden incluir varios índices. Entre ellos puede ser un índice alfabético, índice de diagramas, índice de funciones, etc.

Ejemplo de un SRS – IEEE 29148:2011

1. Introduction
1.1 Purpose
1.2 Scope
1.3 Product overview
1.3.1 Product perspective
1.3.2 Product functions
1.3.3 User characteristics
1.3.4 Limitations
1.4 Definitions
2. References
3. Specific requirements
3.1 External interfaces
3.2 Functions
3.3 Usability Requirements
3.4 Performance requirements
3.5 Logical database requirements
3.6 Design constraints
3.7 Software system attributes
3.8 Supporting information
4. Verification
(parallel to subsections in Section 3)
5. Appendices
5.1 Assumptions and dependencies
5.2 Acronyms and abbreviations

Figure 8 — Example SRS Outline

Documento de requisitos - Consideraciones

- Registrar los requisitos en términos del cliente.
- Características de una buena especificación:
 - Correcta / Válida
 - No ambigua
 - Completa
 - Consistente internamente
 - Agrupados y ordenados por importancia y/o estabilidad
 - Verificable
 - Modificable
 - Trazable
 - Factibles
 - Entendible

Priorización de requisitos

- Todo proyecto que tenga recursos limitados necesita definir prioridades en los requisitos que va a incluir en el producto de software.
- La priorización ayuda a entregar el mayor valor de negocio tan rápido como sea posible con las restricciones existentes.
- A veces los clientes no quieren priorizar requisitos pensando que no se hará lo que sea de prioridad baja.
- Los programadores por su lado no suelen priorizar porque eso daría la impresión de que no son capaces de realizar todos los requisitos.

Priorización de requisitos

- Se debe elegir un nivel de abstracción apropiado (casos de uso, historias de usuario, requisitos funcionales o quizás flujos dentro de un caso de uso).
- Se deben tener en cuenta:
 - Las necesidades de los clientes.
 - La importancia relativa de los requisitos para los clientes.
 - Las relaciones de precedencia entre los requisitos.
 - Requisitos que deban ser implementados en grupo.
 - El costo de satisfacer cada requisito.

Priorización de requisitos

- Un estudio muestra que cerca de dos tercios de las características de los sistemas de software se usan raramente o nunca (The Standish Group 2009).
- Importante: si terminamos el proceso de priorización con todos los requisitos con la misma prioridad entonces no priorizamos nada.
- Hay que evitar dos casos de priorización:
 - Por decibeles — los que gritan más definen las prioridades.
 - Por amenazas — los que tienen más poder definen las prioridades.

Priorización de requisitos

- Algunas técnicas posibles:
 - Adentro y afuera — se nombra cada requisito y se decide sí queda en el alcance o no.
 - Comparación por pares y ranking — se hace un ranking general de todos los requisitos mediante comparación de a pares.
 - Escala de tres niveles — se utilizan dos dimensiones.

	<i>Important</i>	<i>Not So Important</i>
<i>Urgent</i>	High Priority	Don't Do These!
<i>Not So Urgent</i>	Medium Priority	Low Priority

Priorización de requisitos

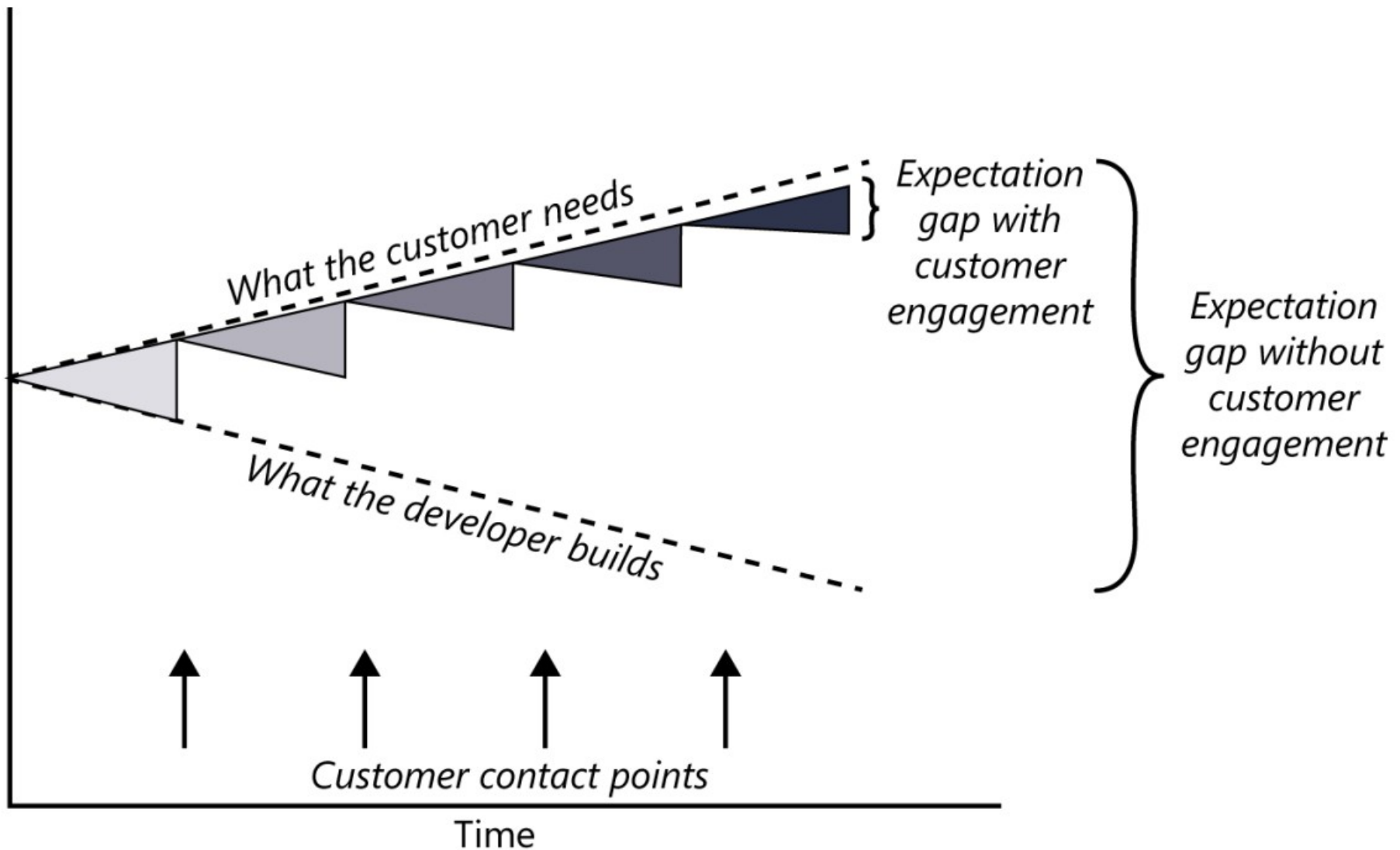
- Algunas técnicas posibles:
 - 4 niveles (MoSCoW) — debe, debería, podría, no.
 - \$100 — asignar dinero disponible y luego “invertirlo” en los requisitos a priorizar.
 - Priorización basado en valores, costos y riesgos.

Relative weights		2	1			1		0.5		
Feature		Relative benefit	Relative penalty	Total value	Value %	Relative cost	Cost %	Relative risk	Risk %	Priority
1.	Print a material safety data sheet.	2	4	8	5.2	1	2.7	1	3.0	1.22
10.	Import chemical structures from structure drawing tools.	7	4	18	11.6	9	24.3	7	21.2	0.33
Totals		53	49	155	100.0	37	100.0	33	100.0	

Validación de requisitos

- Proceso por el cual se determina si los requisitos relevados son consistentes con las necesidades del cliente.
- *Verificación* — determina si el producto de alguna actividad de desarrollo cumple los requisitos (*hacer las cosas bien*).
- *Validación* — evalúa si un producto satisface las necesidades del cliente (*hacer la cosa correcta*).
- Proceso:
 - Planificar quién (qué stakeholder) va a validar qué (artefacto) cómo (técnica).
 - Ejecutar
 - Registrar – Reporte de validación / Firma

Expectativas del cliente



Chequeo de requisitos

- **Validez**— ¿El sistema provee las funciones que mejor soportan las necesidades del cliente?
- **Consistencia**— ¿Hay algún requisito en conflicto?
- **Compleitud**— ¿Están incluidas todas las funciones requeridas por el cliente?
- **Realismo**— ¿Los requisitos pueden ser implementados con el presupuesto y la tecnología disponible?
- **Verificabilidad**— ¿Los requisitos pueden ser chequeados?

Validación de requisitos no funcionales

- Son difíciles de validar.
- Se deben expresar de manera cuantitativa utilizando métricas que se puedan probar de forma objetiva (esto es ideal).

Propiedad	Medida
Rapidez	Transacciones por seg
Tamaño	KB
Fiabilidad	Tiempo promedio entre fallas
Portabilidad	Número de sistemas, especificar
Facilidad de uso	Tiempo de capacitación

- Para los usuarios es difícil especificarlos en forma cuantitativa.

Técnicas de validación de requisitos

- **Revisión de requisitos**
 - Análisis manual y sistemático de los requisitos.
- **Prototipado**
 - Uso de un modelo ejecutable del sistema para chequear los requisitos.
- **Generación de casos de prueba**
 - Desarrollo de pruebas para los requisitos que pueden ser testeados.

¿Cómo validar requisitos?

- **Manuales**
 - **Lectura por parte del cliente.**
 - **Recorridas.** Útiles con muchos stakeholders que no lo leerían de otra manera.
 - **Entrevistas.**
 - **Chequeo manual de referencias cruzadas.**
 - **Instancias de validación formal:**
 - **Revisiones - Stakeholders revisan por separado y se reúnen para discutir problemas.**
 - **Inspecciones formales – roles y reglas.**
 - **Listas de comprobación.**
 - **Escenarios.**
 - **Generación de Casos de Prueba.**
- **Automatizadas**
 - **Chequeo automático de referencias cruzadas,**
 - **Ejecución de Modelos para verificar funciones y relaciones.**
 - **Construcción de Prototipos.**
 - **Simulaciones.**

Revisión de requisitos

- Proceso manual. Se revisa el documento de requisitos buscando anomalías y omisiones:
 - Revisiones informales: discusión informal.
 - Revisiones formales: se hace una “recorrida” del doc de req con el cliente, explicando implicancias de cada requisito.
- Participan representantes:
 - del cliente: operadores, quienes realicen entradas, utilicen salidas, y sus gerentes.
 - del equipo de desarrollo: analistas de requisitos, diseñadores, encargados de pruebas y gestión de configuración.
- Incluye:
 - revisar objetivos del sistema.
 - evaluar alineamiento de requisitos con los objetivos (necesidad).
 - revisar el ambiente de operación y las interfaces con otros sistemas.
 - funciones completas, restricciones realistas.
 - evaluar riesgos.
 - considerar cómo se harán:
 - pruebas del sistema.
 - cambios en los requisitos en el proyecto, su verificación y validación.

Revisión de requisitos

- Revisiones periódicas deben mantenerse mientras se formulan los requisitos.
- El cliente y el personal contratado deben estar involucrados en las revisiones.
- Las revisiones pueden ser formales (con documentos completos) o informales. Una buena comunicación entre desarrolladores, clientes y usuarios puede resolver problemas en fases tempranas.

Chequeos en la revisión

- **Verificabilidad**
 - ¿Es posible encontrar alguna forma realista de probar el requisito?
- **Comprensibilidad**
 - ¿El requisito se entiende de forma apropiada?
- **Trazabilidad**
 - ¿Se indica claramente el origen del requisito?
- **Adaptabilidad**
 - ¿Se pueden cambiar los requisitos sin un gran impacto en otros requisitos?

Ejemplo de checklist de defectos

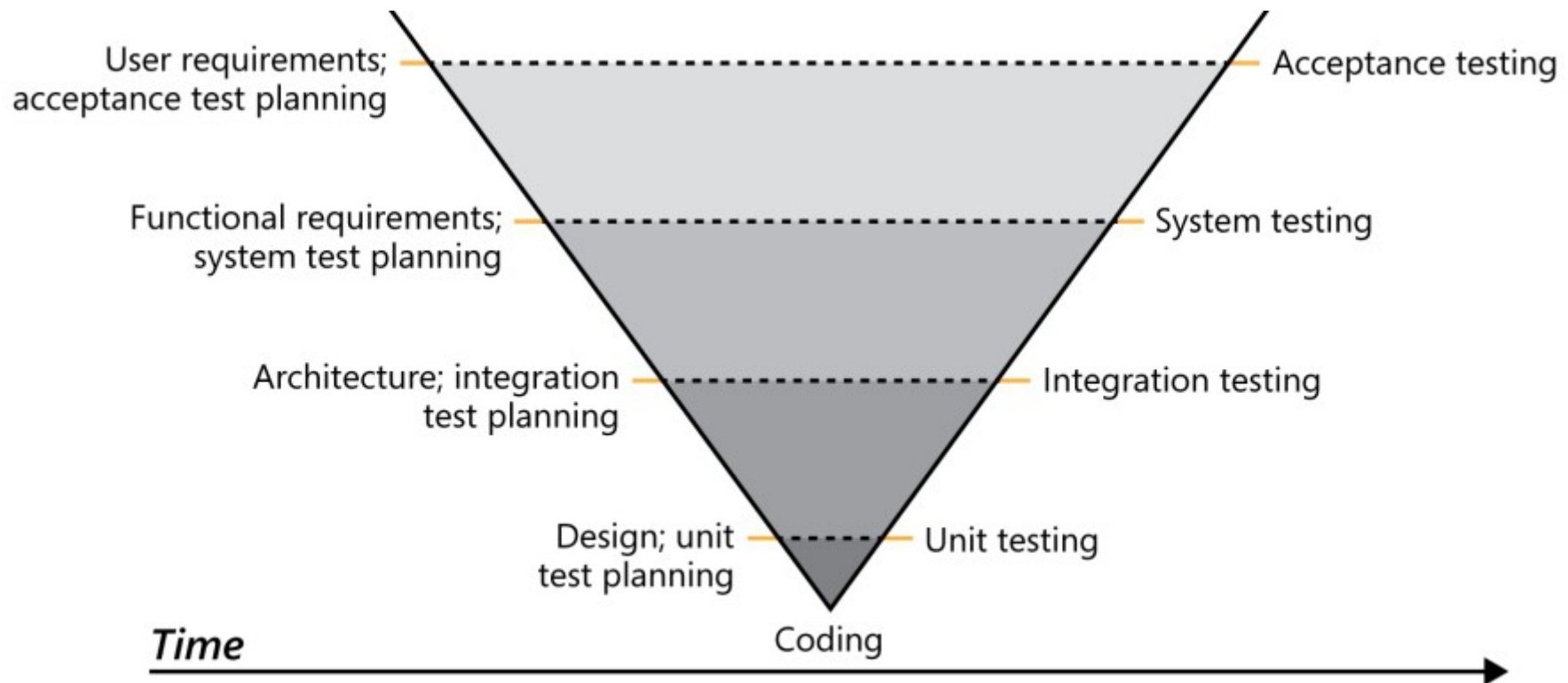
Completeness

- Do the requirements address all known customer or system needs?
- Is any needed information missing? If so, is it identified as TBD?
- Have algorithms intrinsic to the functional requirements been defined?
- Are all external hardware, software, and communication interfaces defined?
- Is the expected behavior documented for all anticipated error conditions?
- Do the requirements provide an adequate basis for design and test?
- Is the implementation priority of each requirement included?
- Is each requirement in scope for the project, release, or iteration?

Correctness

- Do any requirements conflict with or duplicate other requirements?
- Is each requirement written in clear, concise, unambiguous, grammatically correct language?
- Is each requirement verifiable by testing, demonstration, review, or analysis?
- Are any specified error messages clear and meaningful?
- Are all requirements actually requirements, not solutions or constraints?
- Are the requirements technically feasible and implementable within known constraints?

Probando los requisitos



Validar utilizando criterios de aceptación

- El cliente es el que tendrá la opinión final sobre un producto. La idea es entonces establecer, de forma complementaria a la definición de los requisitos, los criterios con los cuales se aceptarán.
- Los criterios de aceptación no son pruebas funcionales o unitarias; sino condiciones que deben cumplir el sistema. Las pruebas son más completas y prueba todos los flujos funcionales, excepciones, límites, etc.

Criterios de aceptación - Ejemplos

Requerimientos	Criterios de Aceptación
Registrar Oferta Laboral	Permite que una empresa pueda registrar una oferta laboral basándose en las capacidades y competencias que maneja la carrera de computación.
Toma de tests por alumnos de psicología a personas de otras instituciones	Permite convocar a un alumno que, luego de ser evaluado por la empresa, cumple con todos los requisitos que la oferta laboral establece.

Gestión de requisitos

- Es el proceso de gestionar los cambios de los requisitos durante el proceso de ingeniería de requisitos y el desarrollo del sistema.
- Es común que nuevos requisitos surjan cuando un sistema está siendo desarrollado y después de haber entrado en uso.
- Es necesario mantener un rastreo individual de los requisitos y mantener sus referencias a otros requisitos de forma de facilitar el análisis de impacto ante posibles cambios.
- Se recomienda establecer un proceso formal de control de cambios.

Cambios en los requisitos

- El entorno empresarial y técnico del sistema siempre cambia después de la instalación.
 - Nuevo hardware, nuevas interfaces con otros sistemas, cambian las prioridades del negocio, nuevas legislaciones, etc.
- Las personas que pagan por un sistema y los usuarios de ese sistema casi nunca son las mismas personas.
- Generalmente los grandes sistemas tienen una diversa comunidad de usuarios, algunos de los cuales tienen diferentes requisitos y pueden existir conflictos en las prioridades.

Planificación de la gestión de requisitos

- Establece el detalle del nivel de gestión requisitos que es requerido.
- Decisiones de gestión de requisitos:
 - **Identificación de requisitos**— Cada requisito debe ser identificado de modo que pueda hacerse una referencia cruzada con otros.
 - **Proceso de gestión de cambios**— Es el conjunto de actividades que evalúan el impacto y el costo de los cambios.
 - **Políticas de trazabilidad**— Estas políticas definen cómo registrar las relaciones entre los requisitos y el sistema diseñado.
 - **Soporte de herramientas**— Las herramientas que utilizarán.

Gestión de cambios en los requisitos

- Decidir sí un cambio en los requisitos debe ser aceptado.
- Problemas de análisis y cambios en la especificación
 - Durante esta fase, el problema o cambio propuesto es analizado para chequear si es valido.
- Cambiar el análisis y calculo de costos
 - El efecto del cambio propuesto se evalua a través de la información de trazabilidad y el conocimiento general de los requisitos del sistema. Luego se toma la decisión de sí se debe proceder o no con el cambio en el requisito.
- Implementación del cambio
 - El documento de requisitos y en su caso el diseño del sistema y la aplicación se modifican. Idealmente, el documento debe estar organizado para que los cambios puedan ser fáciles de implementar.

Gestión de cambios en los requisitos

