

UNIDAD 2: INGENIERÍA DE REQUISITOS

ESPECIFICACION, VALIDACION y GESTIÓN DE REQUISITOS

Agenda

- Especificación de Requisitos
- Validación y Verificación de Requisitos
- Gestión de Requisitos

ESPECIFICACIÓN DE REQUISITOS

Especificación de Requisitos

- En esta fase se documentan los **requerimientos acordados** con el cliente, en un nivel apropiado de detalle.
- Se documenta la descripción completa de las necesidades y funcionalidades del sistema que será desarrollado; describe el alcance del sistema y la forma como hará sus funciones, definiendo los requerimientos funcionales y no funcionales.
- En la práctica, esta etapa se va realizando *conjuntamente con el análisis*, se puede decir que la especificación es el "pasar en limpio" el análisis realizado previamente aplicando técnicas y estándares de documentación.



Especificación de Requisitos

- Definir los requisitos teniendo en cuenta la perspectiva del usuario. Esto implica producir un documento que puede ser revisado, evaluado y aprobado.
- Reutilizar requisitos, revisando proyectos ya finalizados para ver si contienen material potencialmente reutilizable.
- Documentar los requisitos de la forma correcta. Aunque escribir los requisitos puede parecer una tarea tediosa, es la única manera de asegurar que la esencia de los requisitos ha sido capturada correctamente, y que esto pueda ser probado.
- Un mal uso del lenguaje puede llevar a un mal entendimiento, horas de trabajo perdido, una mala comunicación entre miembros del equipo, etc.
- El término especificación de requisitos hace referencia a la colección de requisitos especificados y definidos previamente.
- La especificación no tiene que estar en un determinado formato, puede ser una especificación sobre papel, o un blog, o similares.

Que actividades principales requiere:

- Seleccionar el estándar de documentación
- Documentar los requisitos del cliente
 - Elaborar el Documento de Definición de Requisitos
- Documentar los requisitos del desarrollador
 - Elaborar el Documento de Especificación de Requisitos (SRS / ERS)

Técnicas de Especificación

Se pueden especificar los requisitos usando:

Lenguaje Natural

- Comprensible para el Cliente/Usuario.
- Ambiguo (glosario).
- Poca legibilidad (plantilla, formateo del texto).
- Difícil de tratar (Verificar correctitud, consistencia, completitud).

Notaciones Especiales :

- Son de expresión mas formal
- Poca o ninguna ambigüedad.
- Facilita tratamiento.
- Necesidad de entrenamiento en la notación.
- Dificultades de comprensión por Cliente/Usuario

Técnicas de Especificación

Las **Notaciones Especiales** según su naturaleza pueden ser:

- Gráficas vs. Basadas en texto
- Estáticas vs. Dinámicas

Descripciones Estáticas: Se especifican entidades y sus atributos, los requerimientos se pueden ver como las relaciones entre las entidades. No describe como cambian las relaciones con el tiempo

Descripciones Dinámicas : Especifican estados y las transiciones entre estados en el tiempo.

Técnicas de Especificación

Técnicas se utilizan para especificar los requisitos:

- **Uso de estándares de documentación de requisitos:**

- IEEE P1233/D3 – Guide for Developing System Requirements Specification
- ISO/IEC 12119-1994 – Quality Requirements and Testing
- IEEE 1362-1998 – Guide for concepts of operations document
- IEEE 830-1998 – Recommended Practice for Software Requirements specifications

- **Indicadores de calidad :**

- Modelos de calidad del software

- **Lenguajes y notaciones**

- Lenguajes de modelado gráfico
 - 🌐 Lenguajes OO: UML
 - 🌐 Lenguajes estructurados: DFD, SADT, IDEF
 - 🌐 Lenguajes dinámicos: Redes de Petri y Diagramas de Estado

- Lenguajes formales de especificación

- **Notación Z**



IEEE 830

Creado por Software Engineering Standards Committee, del IEEE Computer Society IEEE (Institute of Electric and Electronic Engineers, USA)

- Fecha : 1998
- Su uso no es Obligatorio

Plantilla de especificación de requisitos de software

La especificación de requisitos de software (ERS) tiene como función la de cubrir la necesidad de describir con mayor detalle los requisitos y todos los aspectos del software a desarrollar por la empresa proveedora.

La especificación de requisitos de software (ERS) está destinada a ser usada por diferentes usuarios, según Sommerville (2005) **"...el documento de requerimientos tiene que presentar un equilibrio entre la comunicación de los requisitos a los clientes, la definición de los requisitos en detalle para los desarrolladores y probadores, y la inclusión de información sobre la posible evolución del sistema."**

IEEE 830

Quién la puede usar:

- **Un cliente/usuario** que vaya a definir requerimientos (características) de un software que necesite
- **Un desarrollador** (interno/externo) que haga software “a la medida” mediante proyecto
- **Un desarrollador que haga software** “de paquete” que se venda masivamente

IEEE 830

Se usa para que:

- Un cliente **describa** claramente lo que quiere
- Un proveedor **entienda** claramente lo que el cliente quiere
- Se establezcan bases para un **contrato** de desarrollo (o de compra-venta)
- Se **reduzca el esfuerzo** de análisis, diseño, y programación (evitando re-trabajos)
- Se tenga una base o **referencia para validar** o probar el software solicitado
- Se **facilite el traspaso** del software a otros clientes/usuarios
- Se le puedan **hacer mejoras** (o innovaciones) a ese software

IEEE 830 - SRS

Generalmente no necesita escribirse en el :

- **Cómo se deben organizar** los módulos del software
- **Cómo se deben asignar funcionalidades** específicas a cada módulo
- **Cómo será el flujo de datos** o el control de ejecución de los módulos
- **Elección de estructuras de datos** que se usarán dentro del código

Sin embargo en algunas situaciones especiales pueden inducir requerimientos estrictos de diseño

Características de una SRS

- **Corrección:** Cada requerimiento representa precisamente alguna característica deseada en el sistema final.
- **Compleitud:** Todas las características deseadas están descritas. La característica más difícil de lograr Corrección y completitud están fuertemente relacionadas.
- **No ambigua:** Cada requerimiento tiene exactamente un significado Si es ambigua los errores se colarán fácilmente. Particular atención si se usa lenguaje natural.
- **Consistente:** Ningún requerimiento contradice a otro. Ej: conflictos lógicos, temporales, de dependencias.

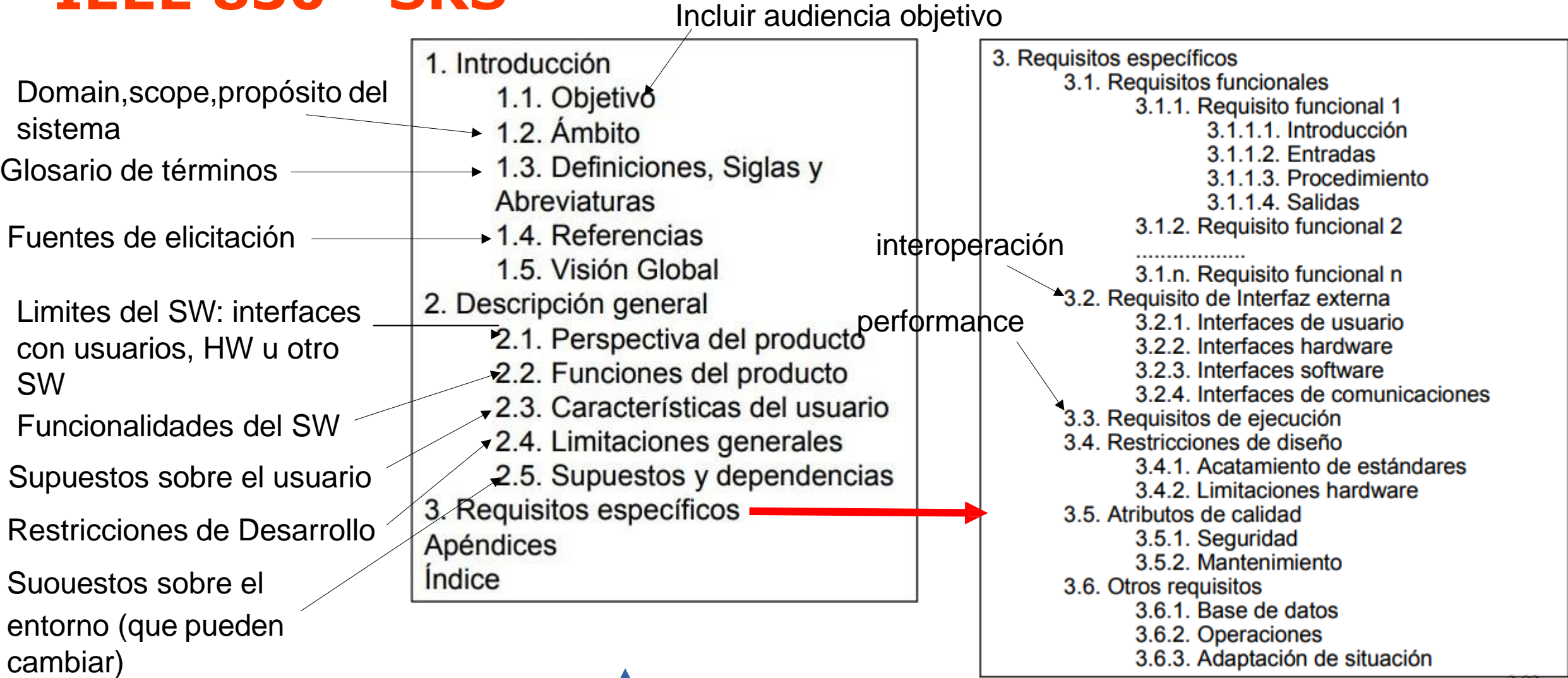
Verificable: Si cada requerimiento es verificable. si existe algún proceso efectivo que puede verificar que el sw final satisface el requerimiento.

Rastreable: Se debe poder determinar el origen de cada requerimiento y cómo éste se relaciona a los elementos del software. hacia adelante y hacia atrás

Modificable: si la estructura y estilo de la SRS es tal que permite incorporar cambios fácilmente preservando completitud y consistencia.

Ordenada en aspectos de importancia y estabilidad: Los requerimientos pueden ser críticos, importantes pero no críticos, deseables pero no importantes.

IEEE 830 - SRS



Ver en la sgtes láminas

Detalle del IEEE 830-SRS

A continuación se detalla el contenido que la SRS de acuerdo a cada sección:

1. Introducción

Debe describir el contenido de la ERSy su organización.

Propósito y Alcance

Debe describir el propósito de la ERSy a quienes va dirigida (audiencia), identificar el o los productos de software, que hará y si es necesario que no hará, que beneficios se tendrán y qué metas se alcanzarán.

Definiciones, acrónimos y abreviaturas

Debe incluir todos los requeridos para interpretar adecuadamente la ERS. Pueden ser por referencia a anexos u otros procedimientos.

Referencias

Debe contemplar la relación de documentos referenciados en la ERS.

2. Descripción General

En esta sección deben describirse los factores y características generales que afectan tanto al software como a sus requerimientos específicos.

- Perspectiva del software.

Debe especificarse si el software mantendrá relaciones con otros productos de software, si forma parte de un sistema mayor o si es completamente independiente.

SRS = ERS

- Software Requirements Specification
- Especificación de Requerimientos de Software

Plantilla de especificación de requisitos de software

- Funciones del software.

Resumen de las funciones principales que deberá realizar el software sin mencionar los detalles al respecto ya que éstos serán definidos en los requerimientos específicos. Las funciones deberán organizarse de tal manera que sean sencillas de entender por los involucrados con el proyecto.

- Características de usuarios.

Deben mencionarse las características generales que deben cumplir los usuarios que harán uso del software considerando, nivel académico, experiencia y características técnicas.

- Restricciones y Dependencias.

En ésta sección se presentará una descripción general de otros factores que limitarán o que pueden afectar el desarrollo o implementación del software (a los requerimientos específicos de la ERS) tales como: Políticas regulatorias, limitaciones de hardware, interfaces hacia otras aplicaciones, operaciones en paralelo, funciones de auditoria y control, requerimientos de lenguajes de mas alto nivel, protocolos de señalización, requerimientos de confiabilidad, criticidad de la aplicación y consideraciones de seguridad.

- Asignación de Prioridad de Requerimientos.

En esta sección se identificarán los requerimientos que pueden ser aplicados en versiones posteriores del software.

3. Requerimientos específicos.

Plantilla de especificación de requisitos de software

3. Requerimientos específicos.

Esta sección debe contener todos los requerimientos con suficiente detalle de tal manera que permitirán a los responsables efectuar lo necesario para satisfacerlos y probarlos.

☐ Interfaces externas.

Deben contener una descripción detallada de todas las entradas y salidas aplicables al software.

☐ Funciones.

Los requerimientos funcionales definen las acciones principales que se llevan acabo en el Software considerado las entradas y generando las salidas

☐ Requerimientos de desempeño.

Estos requerimientos establece límites numéricos relacionados con el desempeño del sistema .

☐ Requerimientos de Lógica de Base de Datos

Deben definir los requerimientos lógicos para la información que se manejará en la base de datos.

☐ Consideraciones de Diseño

Restricciones impuestas por limitaciones de hardware, estándares y regulaciones.

☐ Atributos del sistema

Existen ciertos atributos de software que pueden servir como requerimientos; es importante que esos atributos sean especificados para que puedan ser verificados objetivamente, los atributos esperados incluyen confiabilidad, seguridad, facilidad de mantenimiento, portabilidad y desempeño.

4. Anexos

En ésta sección se incluyen ejemplos de formatos de entrada/salida, información de soporte que ayude a la lectura y entendimiento de la ERS.

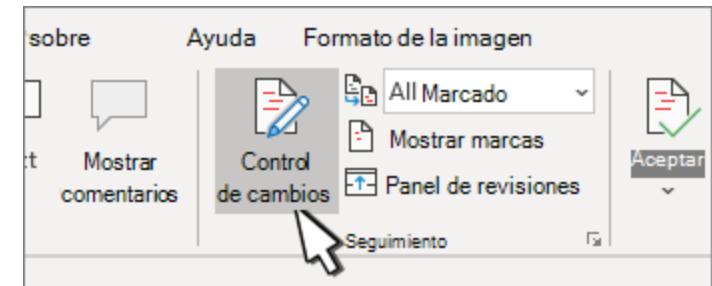


UNIVERSIDAD
DE LIMA



Finalmente ...

- Generalmente, la ERS deberá ser actualizada a medida que progresa el producto software ya que es casi imposible especificar algunos detalles en el momento en el que se inicia el proyecto: habrán deficiencias, defectos e inexactitudes que se descubren y solucionan a medida que el proyecto avanza.
- Debe considerarse también un proceso formal de cambio para identificar, controlar, seguir e informar de cambios tan pronto como sean identificados



VERIFICACIÓN Y VALIDACIÓN DE REQUISITOS

Validación de Requisitos

Objetivo: chequear que los requisitos obtenidos **sean los que definen al sistema que** el cliente desea.

- Comprende un **bosquejo completo** del documento de requisitos, a diferencia del análisis que implica trabajar con requisitos incompletos.
- **Es necesaria**, pues un cambio en los requisitos tiende a traducirse en cambios en el diseño e implementación del sistema, y nuevas pruebas...lo que es muy costoso.

Validación de Requisitos

Los requisitos deben ser validados para

- **Asegurar** que el ingeniero software los ha **comprendido**
- El documento de **ERS** es **conforme a los estándares establecidos**, comprensible, consistente y completo.
- Se trata de **asegurar** que el documento ERS **define el software adecuado**, es decir, el que espera el usuario.

Verificación y Validación

- **Validación:** ¿Estamos construyendo el producto correcto?

La validación es un proceso mas general. Se debe asegurar que el software cumple las expectativas del cliente. Va mas allá de comprobar si el sistema está acorde con su especificación, para probar que el software hace lo que el usuario espera a diferencia de lo que se ha especificado.

- **Verificación:** ¿Estamos construyendo correctamente el producto?

El papel de la verificación comprende comprobar que el software está de acuerdo con su especificación. Se comprueba que el sistema cumple los requerimientos funcionales y no funcionales que se le han especificado.

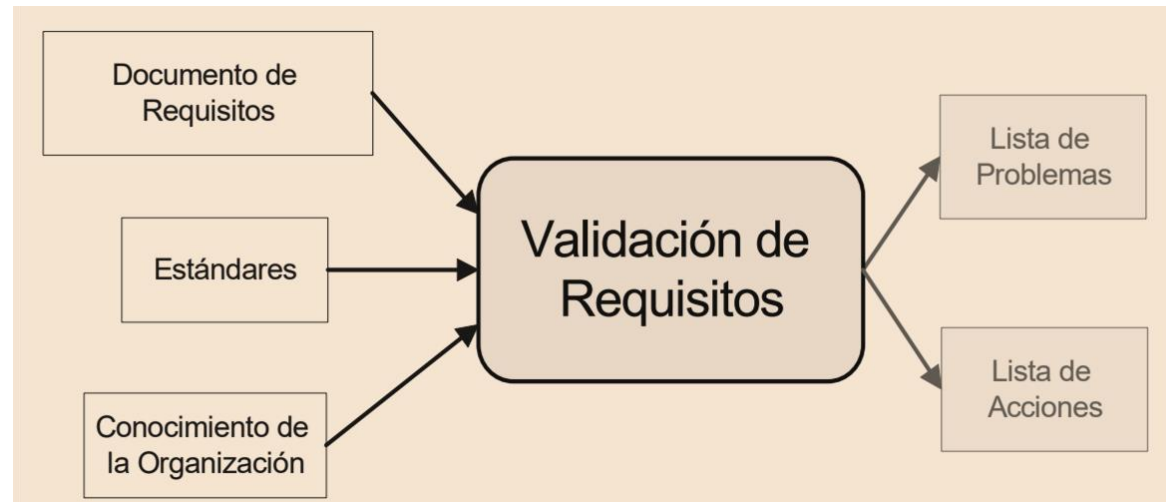
Validación de Requisitos

Las principales técnicas de validación:

- Revisiones
- Prototipado
- Validación de Modelos
- Pruebas de Aceptación

Validación de Requisitos

La actividad de validación tiene como **entrada el documento de requisitos**, los **estándares** relacionados y el **conocimiento** de la organización, y como salida se obtiene **una lista de problemas y una lista de acciones** recomendadas.



¿Tenemos los requisitos descritos correctamente?

Validación de Requisitos

Debemos comprobar:

- **Validez.** El sistema provee las funciones que soportan mejor las necesidades del cliente.
- **Consistencia.** No existen conflictos entre los requisitos.
- **Compleitud.** Están incluidas todas las funciones requeridas por el usuario.
- **Realismo.** Los requisitos pueden ser implementados con el presupuesto y tecnologías disponibles.
- **Verificable.** Los requisitos pueden ser verificados.

1. Validación por Revisiones



- Un grupo de personas lee, ***analiza, discute el documento de ERS*** y las formas de solucionar los problemas detectados.
- Deben participar representantes del cliente y los usuarios.
- Norma a usar : IEEE Std 1028-1997, IEEE Standard for Software Reviews

Pueden ser de tipo:

Informal: discutiendo requisitos con los stakeholders.

Formal: donde el equipo de desarrollo debe “**conducir**” al cliente a través de los requisitos del sistema, explicándole las implicancias de cada uno de ellos.

Validación por Revisiones

La revisión de requisitos es **uno de los mejores** métodos de validación de requisitos. Generalmente hablando, las revisiones de requisitos permiten:



- Descubrir una gran cantidad de defectos en los requisitos.
- Reducir los costos de **desarrollo** entre un **25% y un 30%**.
- Reducir el tiempo de **pruebas** entre un **50% y un 90%**.

Validación por Revisiones

Se pueden usar Listas de Comprobación:

Unicidad

¿Está cada requisito identificado de forma unívoca?

Comprensibilidad

Se describen en el glosario los términos especializados o técnicos?

¿Es cada requisito autocontenido o se tienen que examinar otros requisitos para comprenderlo?

¿Hay contradicciones?

¿Están agrupados los requisitos relacionados?

Trazabilidad

¿Está establecido adecuadamente el origen de cada requisito?

Adaptabilidad

¿Puede cambiarse un requisito sin impactar demasiado a los otros?

Validación por Revisiones

Para llevar a cabo tanto la preparación de la reunión de revisión, como la reunión en sí misma, se utilizan **checklists** de validación.

Example Requirements Review Checklist

Intended use of this checklist	When examining a set of requirements (which may be in a Statement of Work), the reviewer might use the following questions to see that the document is adequate, perhaps while doing some type of review.
---------------------------------------	---

ID	Defect Type	Items to Examine
Organization and Structure of the Documentation		
1	Standards	Have appropriate requirements documentation standards been followed?
2	Standards	Are all figures, tables, and diagrams labeled and referenced?
3	Standards	Are all terms and units of measure defined?
4	Standards	Are all requirements written at a consistent and appropriate level of detail?
5	Standards	Are individual requirements rated (or ranked), with descriptions of priority provided?
6	Unverifiable	Do the requirements provide an adequate basis for design and system test?
Completeness and Correctness		
7	Correctness	Are all internal cross-references to other requirements correct? [For

Validación por Revisiones

Un posible formato para la lista de errores y acciones recomendadas

Nº de requisito	Defectos detectados	Acciones recomendadas
1	Error de estilo, que lleva a ambigüedad	Modificar el texto del requisito de tal forma que diga algo como “El sistema deberá permitir el registro de los fondos bibliográficos”
2	Idem	Idem
11	Ambigüedad	Precisar la duración de las reservas
12	Concisión	No se han identificado diferencias entre profesores y alumnos a todo lo largo de la lista de requisitos. Este requisito se debería eliminar, ya que proporciona ningún tipo de información relevante.
15	Realizabilidad	El sistema no puede realizar automáticamente los préstamos. Quizás se refiere el requisito a que debe proporcionar el máximo de automatización? Precisar, en este caso, o eliminar por irreal.
17	Concisión	Separar lo referido a libros prestados de lo referido a libros reservados.

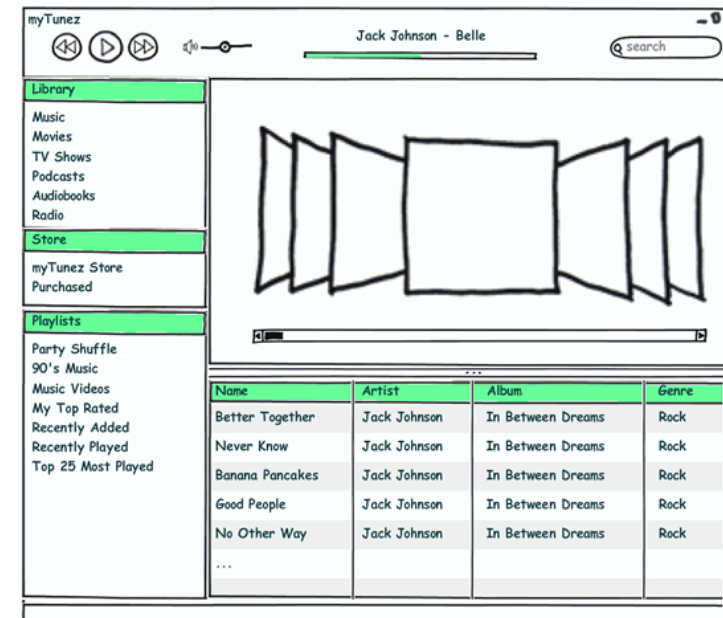
2- Validación por Prototipos

Los prototipos son un método de validación ampliamente utilizado en muchas disciplinas, y en todos los casos, los principios subyacentes son los mismos: el prototipado **consiste en la creación de una maqueta** o versión del producto final. Se utilizan, fundamentalmente, **para comprobar la corrección y completitud de la especificación de requisitos.**

Existen varios tipos de prototipos, cada uno de los cuales permite la realización de un tipo determinado de pruebas y con un determinado nivel de realismo

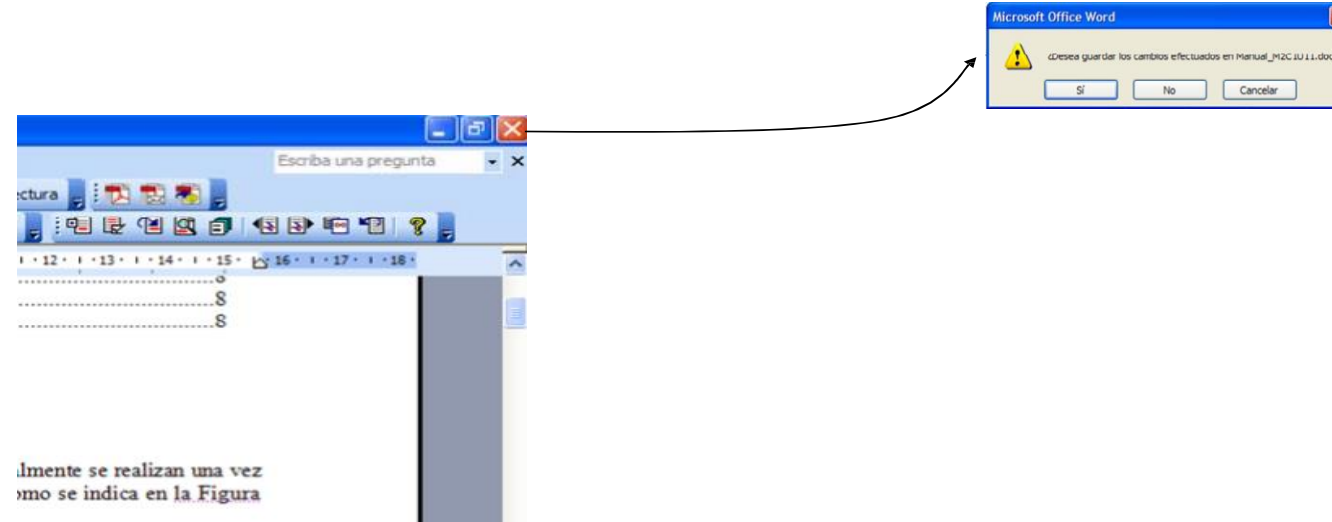
Validación por Prototipos

Mock-ups. Se trata de pantallas, típicamente dibujadas a mano en papel, que representan un aspecto concreto del sistema. El soporte que proporcionan a la validación es muy limitado, con la excepción, quizás, de aclarar el interfaz gráfico deseado en casos complejos.



Validación por Prototipos

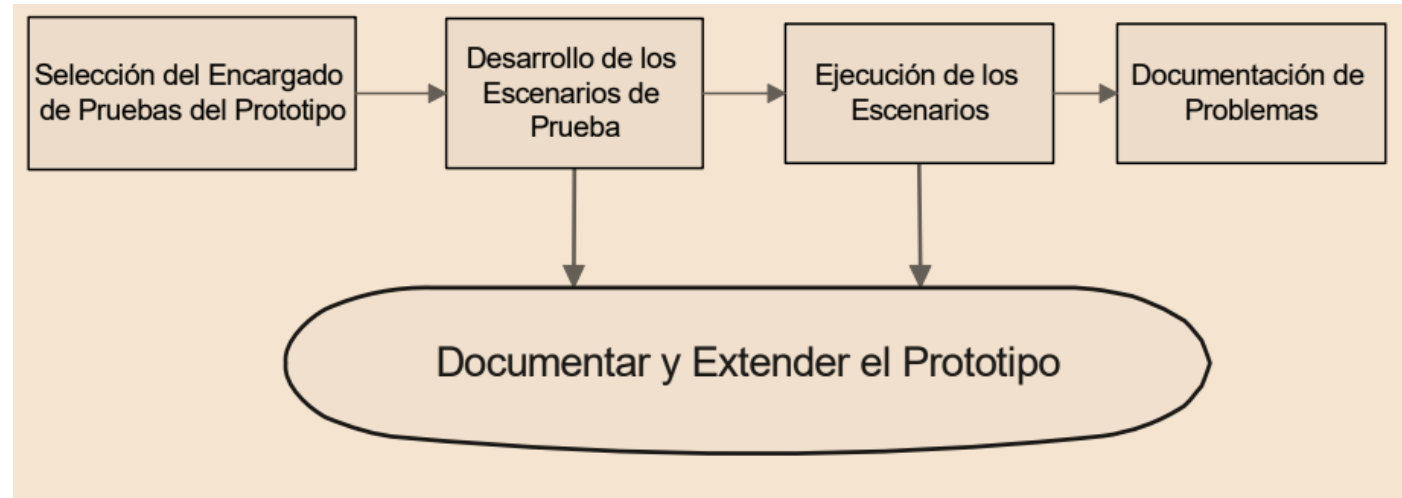
Storyboards. Son una evolución de los mock-ups, ya que además del interfaz, se muestra la secuencia de acciones, o escenarios, que se deben realizar con el programa.



Maquetas. Es una versión simplificada del sistema software deseado. Típicamente, una maqueta representa únicamente el interfaz del sistema y, opcionalmente, las conexiones entre pantallas mediante la utilización de elementos activos como los botones.

Validación por Prototipos

Prototipo en el proceso de Validación de Requisitos:



3. Validación por Casos de Pruebas

Un caso de prueba es una serie de pruebas de entrada, condiciones de ejecución y resultados esperados desarrollados para un objetivo en particular, tal como ejecutar una ruta particular de un programa o verificar el cumplimiento con un requerimiento en específico.

Plantilla para Casos de Prueba

Proyecto No.: Nombre del Proyecto:	Página No.:
Caso No.: Nombre del Caso:	Ejecución No.: Nombre: Estado de la prueba:
Marca/Subsistema/Módulo/Nivel/Función/Código de la Unidad bajo prueba:	Requisito No.: Nombre:
Escrito por: Fecha:	Ejecutado por: Fecha:
Descripción del caso de prueba (propósito y método):	
Configuración de la prueba para (H/W, S/W, N/W, datos, pre-requisitos de prueba, seguridad y tiempo):	

Paso	Acción	Resultados esperados	Pasado/Fallido
1			
2			
...			

Validación por Casos de Pruebas

Durante la validación de requisitos, los casos de prueba se utilizan del mismo modo que durante la prueba del software: **es necesario poder describir, como se ha indicado anteriormente, tanto los datos de entrada como las tareas a realizar y los resultados esperados, para lo cual es necesario que estén perfectamente descritos los requisitos.**

Por ejemplo; dado el requisito:

X. El sistema deberá generar facturas



Para el ejemplo anterior:

- Es posible describir la tarea a realizar,
- pero no los datos de entrada ni los resultados a obtener.
- Esto ocurre porque el requisito está incompleto y no están bien definidos.
 - No se indica cómo se deben generar las facturas
 - No se indica qué se factura

Los defectos anteriores no están presentes en la siguiente definición del mismo requisito:

X. El sistema deberá generar facturas

- 1. Se generará una factura por cliente.
Esta factura incluirá todos los pedidos no facturados del cliente en cuestión.*
- 2. El formato de la factura será el especificado en el anexo 1*

GESTION DE REQUISITOS

Agenda

1. Gestión de Requisitos (Requirements Management, RM)
2. Herramientas y técnicas que se utilizan en la gestión de requisitos
 1. control de cambios
 2. Atributos de los requerimientos
 3. Matrices de seguimiento de requerimientos
3. Ventajas y desventajas del uso de herramientas software en la IR
- 4. Especificación de Requisitos No Funcionales**

Introducción

Por lo general los **requisitos de software son volátiles**, ya que pueden cambiar debido a varios factores tales como:

- Errores u omisiones en la fase de elicitación,
- La naturaleza cambiante,
- La comprensión compleja de sistemas,
- Las tecnologías emergentes,
- Los requisitos cambiantes del negocio o las exigencias reglamentarias,
- y las presiones competitivas.

1. Gestión de Requisitos

Es el **proceso de seguimiento** de la situación de los requisitos y **controlar los cambios** de los requisitos de la **línea de base**.

La gestión de requisitos **es una actividad del ciclo de vida**, comenzando por el desarrollo de los requisitos y continuando en el desarrollo del software.

Las líneas base o más conocidas por su termino en inglés, "*baselines*", 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" IEEE 610.12-1990

La líneas base

Por lo tanto, una **línea base** se define como un producto que acaba de ser aprobado y que define la "base" de ese producto que para ser modificado deberá pasar por un protocolo de control de cambios. También puede verse como un punto de referencia en la configuración de un proyecto que marca un estado estable en algún producto del proyecto.

El uso de las **líneas base en un proyecto** vendrá dado por la definición de las **diferentes líneas base que** se realizarán a lo largo del proyecto (generalmente requisitos de usuario, requisitos software, diseño, código fuente, planes o procedimientos, pruebas, funcionamiento) para controlar cuando se aprueban los distintos productos y asegurar que se realiza el correspondiente control de cambios que ayudará a mantener la coherencia y calidad de todo el proyecto software.

Por que es importante GR?

Para gestionar los requisitos, es necesario establecer procedimientos que permiten al equipo de forma rápida **entender el impacto de los cambios**, decidir **cómo hacer frente** a las nuevas necesidades, y renegociar los acuerdos sobre los requisitos.

En que Consiste esta Etapa IR?

Primordialmente, **en gestionar los cambios a los requisitos**, puesto que este es uno de los atributos de calidad que deben cumplirse y es de suma importancia, puesto que asegura la consistencia ente los requisitos y el sistema que se está desarrollando, se ha de considerar las cantidades de **tiempo y esfuerzo** que se utilizará puesto que abarca todo el ciclo de vida del software

Gestión de Requisitos

La gestión de requisitos implica la definición de ***procedimientos de gestión de cambios***:

- Definen los pasos y los análisis que se realizarán antes de aceptar los cambios propuestos o si es necesario cambiar los atributos de los requisitos afectados, como se realizará el proceso, cuáles serán los responsables y de esta manera contar con una **trazabilidad**: hacia atrás, hacia delante y entre requisitos, con la aplicación de **métodos de versionamiento de los documentos de requisitos**

2. Herramientas y técnicas que se utilizan en la gestión de requisitos

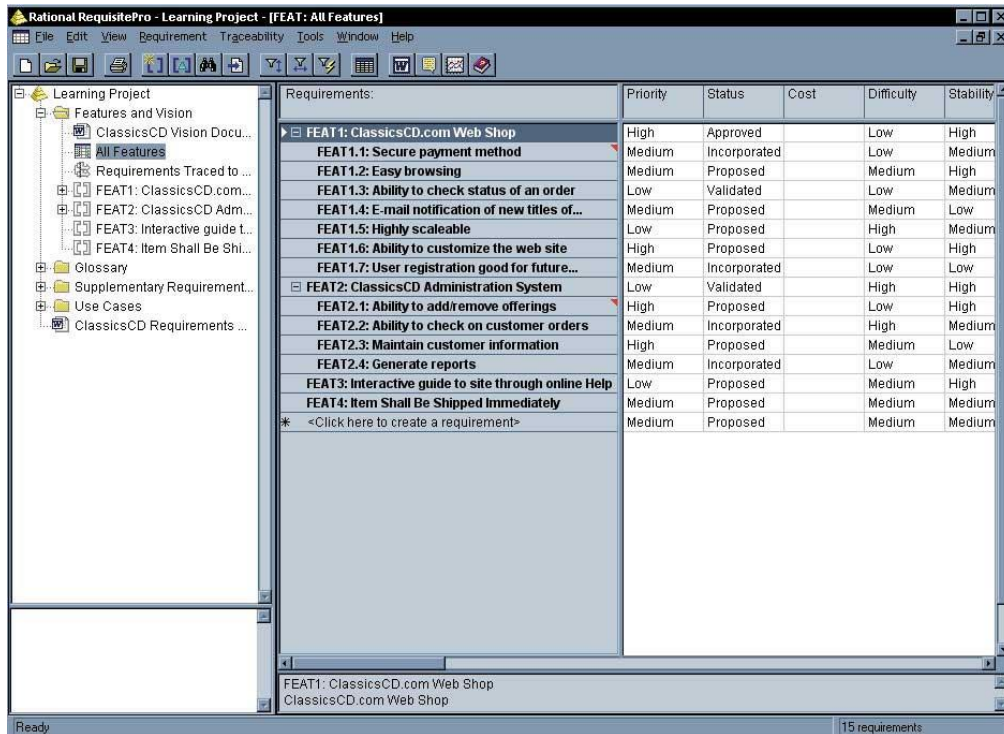
En la actualidad existen herramientas que facilitan las **tareas de la escritura, trazabilidad y gestión de cambios en los requisitos**, además de mantener una adecuada administración de estos en bases de datos u otros repositorios.

Herramientas de la Gestion de Requisitos

Estas herramientas aunque son estándares, nos permiten también modificar atributos de los requisitos como para poder adaptar los mismos a los cambios de cada organización. Dentro de las herramientas utilizadas en la ingeniería de requisitos están:

- Rational Requisite Pro
- IRqA (Analizador integral de requisitos)
- LEAP SE
- Visual Requisite

Herramientas de la Gestion de Requisitos



Rational RequisitePro - Learning Project - [FEAT: All Features]

File Edit View Requirement Traceability Tools Window Help

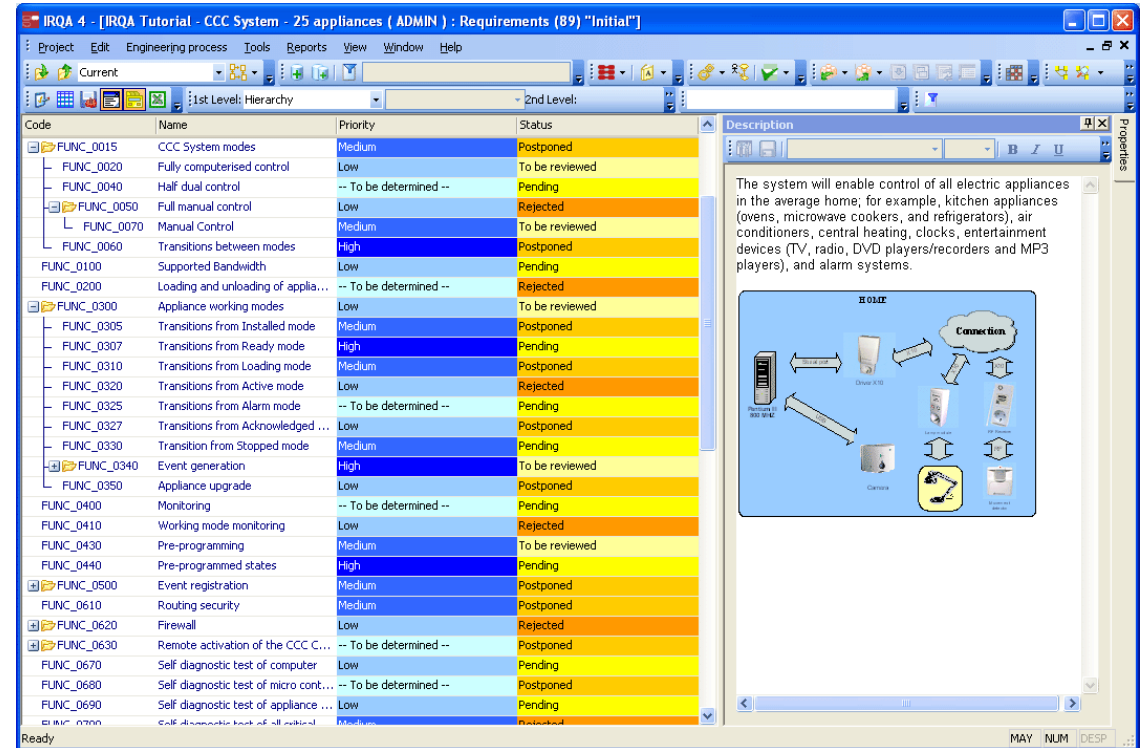
Learning Project

- Features and Vision
 - ClassicsCD Vision Docu...
 - All Features
 - Requirements Traced to ...
 - FEAT1: ClassicsCD.com...
 - FEAT2: ClassicsCD Adm...
 - FEAT3: Interactive guide t...
 - FEAT4: Item Shall Be Shi...
 - Glossary
 - Supplementary Requirement...
 - Use Cases
 - ClassicsCD Requirements ...

Requirements:	Priority	Status	Cost	Difficulty	Stability
FEAT1: ClassicsCD.com Web Shop	High	Approved		Low	High
FEAT1.1: Secure payment method	Medium	Incorporated		Low	Medium
FEAT1.2: Easy browsing	Medium	Proposed		Medium	High
FEAT1.3: Ability to check status of an order	Low	Validated		Low	Medium
FEAT1.4: E-mail notification of new titles of...	Medium	Proposed		Medium	Low
FEAT1.5: Highly scalable	Low	Proposed		High	Medium
FEAT1.6: Ability to customize the web site	High	Proposed		Low	High
FEAT1.7: User registration good for future...	Medium	Incorporated		Low	Low
FEAT2: ClassicsCD Administration System	Low	Validated		High	High
FEAT2.1: Ability to add/remove offerings	High	Proposed		Low	High
FEAT2.2: Ability to check on customer orders	Medium	Incorporated		High	Medium
FEAT2.3: Maintain customer information	High	Proposed		Medium	Low
FEAT2.4: Generate reports	Medium	Incorporated		Low	Medium
FEAT3: Interactive guide to site through online Help	Low	Proposed		Medium	High
FEAT4: Item Shall Be Shipped Immediately	Medium	Proposed		Medium	Medium
* <Click here to create a requirement>	Medium	Proposed		Medium	Medium

FEAT1: ClassicsCD.com Web Shop
ClassicsCD.com Web Shop

Ready 15 requirements



IRQA 4 - [IRQA Tutorial - CCC System - 25 appliances (ADMIN) : Requirements (89) "Initial"]

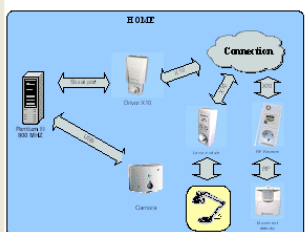
Project Edit Engineering process Tools Reports View Window Help

Current

1st Level: Hierarchy 2nd Level:

Code	Name	Priority	Status
FUNC_0015	CCC System modes	Medium	Postponed
FUNC_0020	Fully computerised control	Low	To be reviewed
FUNC_0040	Half dual control	-- To be determined --	Pending
FUNC_0050	Full manual control	Low	Rejected
FUNC_0070	Manual Control	Medium	To be reviewed
FUNC_0060	Transitions between modes	High	Postponed
FUNC_0100	Supported Bandwidth	Low	Pending
FUNC_0200	Loading and unloading of appla...	-- To be determined --	Rejected
FUNC_0300	Appliance working modes	Low	To be reviewed
FUNC_0305	Transitions from Installed mode	Medium	Postponed
FUNC_0307	Transitions from Ready mode	High	Pending
FUNC_0310	Transitions from Loading mode	Medium	Postponed
FUNC_0320	Transitions from Active mode	Low	Rejected
FUNC_0325	Transitions from Alarm mode	-- To be determined --	Pending
FUNC_0327	Transitions from Acknowledged ...	Low	Postponed
FUNC_0330	Transition from Stopped mode	Medium	Pending
FUNC_0340	Event generation	High	To be reviewed
FUNC_0350	Appliance upgrade	Low	Postponed
FUNC_0400	Monitoring	-- To be determined --	Pending
FUNC_0410	Working mode monitoring	Low	Rejected
FUNC_0430	Pre-programming	Medium	To be reviewed
FUNC_0440	Pre-programmed states	High	Pending
FUNC_0500	Event registration	Medium	Postponed
FUNC_0610	Routing security	Medium	Postponed
FUNC_0620	Firewall	Low	Rejected
FUNC_0630	Remote activation of the CCC C...	-- To be determined --	Postponed
FUNC_0670	Self diagnostic test of computer	Low	Pending
FUNC_0680	Self diagnostic test of micro cont...	-- To be determined --	Postponed
FUNC_0690	Self diagnostic test of appliance ...	Low	Pending
FUNC_0700	Self diagnostic test of all netw...	Medium	Rejected

The system will enable control of all electric appliances in the average home; for example, kitchen appliances (ovens, microwave cookers, and refrigerators), air conditioners, central heating, clocks, entertainment devices (TV, radio, DVD players/recorders and MP3 players), and alarm systems.



Ready MAY NUM DESP

Técnicas de la Gestión de Requisitos

- La elección de las técnicas y herramientas de software, dependen de la **metodología de desarrollo de software** que se vaya a aplicar
- Depende del **tipo de proyecto a construir**, es así que;
 - en un proyecto de gestión académica universitario en el que se esté aplicando RUP como metodología de desarrollo, es posible que el ingeniero de requisitos aplique **Rational Requisite Pro**, ya que es una herramienta propia de esta metodología y, le permite manejar y administrar de forma adecuada la gran cantidad de requisitos que se generan del proyecto.

Características de las herramientas de gestión de Requisitos

Dentro de las principales características de las herramientas de gestión de requisitos están:

- ***Captura de requisitos .***
- ***Análisis***
- ***Negociación .***
- ***Especificación .***
- ***Validación y verificación .***
- ***Gestión de requisitos***
- ***Trazabilidad***
- ***Facilidad de uso***
- ***Redundancia***
- ***Generación de modelos***
- ***Comunicación de requisitos.***
- ***Generar informes***



UNIVERSIDAD
DE LIMA



Características de las herramientas de gestión de Requisitos

Dentro de las principales características de las herramientas de gestión de requisitos están:

- **Captura de requisitos:** Comprende la obtención de los requisitos o necesidades planteados por el cliente.
- **Análisis:** Se realiza la identificación de los requisitos y la clasificación por tipo: funcional, no funcional, de software, de usuario, de interfaz, de negocio.
- **Negociación:** Implica contrastar y dar opiniones respecto a los requisitos que se han obtenido, con el fin de determinar las funciones y las restricciones que tendrá el sistema o proyecto.
- **Especificación:** Consiste en la definición, descripción detallada y completa de cada uno de los requisitos que serán sometidos a validación.
- **Validación y verificación:** Comprobar que los requisitos obtenidos cumplen con los objetivos del cliente y si fueron construidos en base a estándares o criterios desarrollados por el equipo de desarrollo.



Características de las herramientas de gestión de Requisitos

- **Gestión de requisitos:** Se realiza la comprensión y control de los cambios de cada uno de los requisitos.
- **Trazabilidad:** Consiste en determinar el ciclo de vida de los requisitos, además de que cada uno de estos posee su propia identificación, diferenciándolos de los demás.
- **Facilidad de uso:** Quiere decir facilidad en el manejo de herramientas para la administración de requisitos por parte de aquellas personas que realicen el mantenimiento y explotación de los mismos, además de su fácil modificación tanto en su estructura como en su estilo o tipo.
- **Redundancia:** No se debe permitir redundancia, cada uno de los requisitos se debe utilizar en un solo lugar y ser identificados de manera diferente.
- **Generación de modelos:** Algunas de las herramientas utilizadas en la ingeniería de requisitos deben permitir generar modelos como: casos de uso, conceptuales, de estado entre otros; de tal manera que e ayude a validar y verificar cada uno de los requisitos y a la identificación de potenciales errores.

Características de las herramientas de gestión de Requisitos

- **Comunicación de requisitos:** Lo más importante en el desarrollo de todo proyecto es la comunicación que exista en el equipo de desarrollo, toda la información debe ser transmitida de manera oportuna, en el caso de los requisitos, a través de las herramientas se puede lograr esto con la generación de usuarios, estos tendrán acceso a cada uno de los requisitos con el fin de analizarlos, refinarlos de ser necesario y sobre todo obtener coherencia en los requisitos.
- **Generar informes:** Es importante para el respectivo análisis de los requisitos y de un proyecto.

Ejercicio:

Realice un cuadro comparativo de las siguientes herramientas: Rational Requisite Pro, IRqA (Analizador integral de requisitos), LEAP SE, Visual Requisite e incluya al menos otras 3 herramientas.

Técnicas de gestión Requerimientos

En cuanto a las técnicas para manejar requerimientos, ponemos a su disposición el siguiente cuadro resumen:

Cuando necesite:	Entonces crear:
Establecer mecanismos para manejar los cambios de los requerimientos	Cambio de control de políticas y procedimientos.
Identificar información suplementaria de requisitos	Atributos de requerimientos
Entender la procedencia de los requerimientos y sus relaciones	Matrices de trazabilidad de requerimientos

2.1 Control de cambios

Cambiar las políticas y procedimientos de control se refiera a establecer mecanismos y reglas para la identificación, evaluación y decidir la forma de integrar los requisitos de las nuevas y cambiantes necesidades en la línea de base.

Cuando necesite:	Entonces crear:
Establecer mecanismos para manejar los cambios de los requerimientos	Cambio de control de políticas y procedimientos.
Identificar información suplementaria de requisitos	Atributos de requerimientos
Entender la procedencia de los requerimientos y sus relaciones	Matrices de trazabilidad de requerimientos

Control de cambios

Es necesario realizar el control de cambios, para **anticiparse** y responder a las necesidades cambiantes, establecer procedimientos eficaces que permitan los cambios legítimos a los requisitos de un **mínimo de perturbaciones** para los planes del proyecto, y para hacer el uso más eficaz del tiempo de las partes interesadas para evaluar y resolver los cambios.

Control de cambios

Al aplicar el control de cambios se tiene:

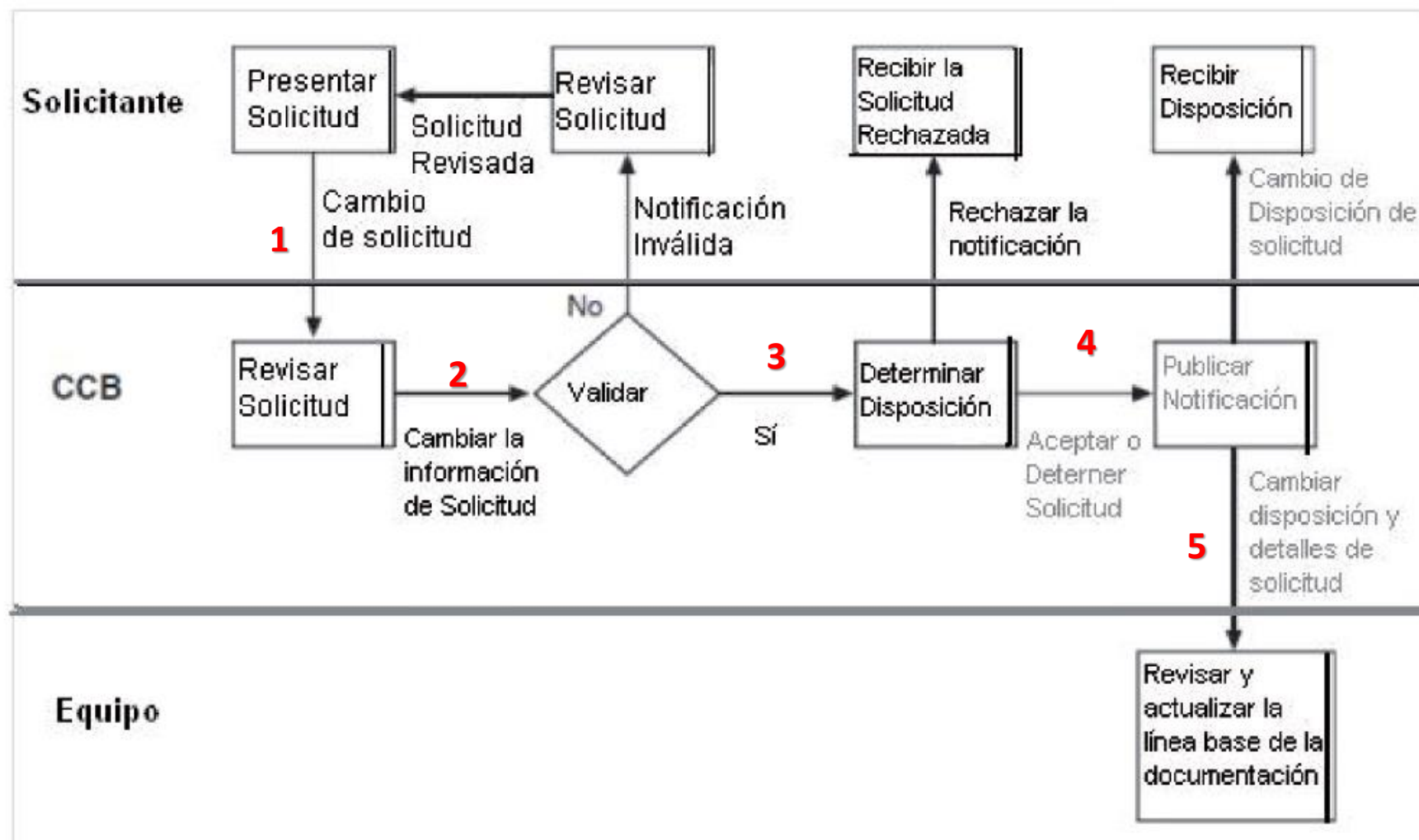
- **Alinear el proyecto** de software con el cambio de necesidades de negocio.
- **Asegurar que los clientes entienden y acepten** los cambios de requisitos.
- **Definir** que y cómo los cambios de requisitos serán registrados.
- **Establecer los procedimientos** para la comprensión de cuáles son los requisitos y los resultados de desarrollo que están asociadas a las nuevas necesidades, para ayudar en el análisis del impacto.
- **Programar la planificación** de la aplicación o aplazamientos de requisitos.

Control de cambios

Para lo cual se debe:

- Identificar los procedimientos de control de cambio y crear **un tablero de control de cambios** (CCB)
- crear la línea base de los requisitos
- y finalmente implemente su proceso de control de cambio una vez que usted ha creado una línea base de requerimientos, que se debe ver reflejado en **un informe y** supervisar cualquier cambio.

Control de cambios



Mapa del Proceso de Requerimientos (Gottesdiener E. , 2005)



UNIVERSIDAD
DE LIMA



Proceso de Control de cambios

Para realizar el proceso de control de cambio, se debe:

1. Identificar el procedimiento para control de cambios.
2. Creación de una línea base para los requisitos
3. Implementación del proceso de control de cambios

Proceso de Control de cambios

1. Identificar el procedimiento para control de cambios. Incluir procedimientos para: proponer requerimientos de cambio, realizar un análisis de impacto, actualizar requerimientos, etc.

Proceso de Control de cambios

2. Creación de una línea base para los requisitos.

Para crear la línea base de los requisitos es necesario, identificar de forma única cada requisito, pero hay que asegurarse de registrarse todos los atributos necesarios de los requisitos; se recomienda utilizar una herramienta de gestión de requisitos para registrar la información de requisitos.

Proceso de Control de cambios

3. Implementación del proceso de control de cambios. Se debe usar procesos informales de control de cambios para pequeños y mínimos riesgos en los proyectos; como por ejemplo un pequeño desarrollo de un par de semanas para entregar un componente de un software.

Control de cambios

- Por cada incremento, el equipo debe explorar y priorizar los requerimientos; además se deben desarrollar pruebas, diseño e implementación de código, y presentar el producto a los clientes y usuarios para su evaluación y aceptación.
- **Pida** el personal de negocio y técnico actuar como un consejo de control de cambio, en la decisión de que requerimientos deben desarrollarse prioritariamente.
- **Típicamente** el control de cambio es manejado durante cada incremento diciendo «No» (por ejemplo no permiten a ningunos cambios a los requisitos), aunque el cambio de requisitos que se solicite debería ser registrado.

Ejemplo

A continuación se muestra un ejemplo de cómo se debe desarrollar un control de cambios :

Cambio en la identificación del requisito	Identificación de requerimientos	Pequeña descripción	Fecha solicitada	Disposición	Fecha de disposición	Justificación de la disposición
CR-3	EST.BID-2.0	El sistema diferido deberá proporcionar a los contratistas la capacidad de ofertar en trabajos sobresalientes	5 de Agosto	Retrasado	8 de Agosto	Depende del apoyo de varias empresas, y añade complejidad técnica y el riesgo. Los objetivos primarios para optimizar las operaciones se deben lograr en primer lugar.



2.2 Atributos de los requerimientos

Los atributos de requisitos son la información adicional asociada con los requisitos.

- Es necesario realizarlo para recopilar información útil para explicar, justificar, el seguimiento y la presentación de informes sobre los requisitos

Cuando necesite:	Entonces crear:
Establecer mecanismos para manejar los cambios de los requerimientos	Cambio de control de políticas y procedimientos.
Identificar información suplementaria de requisitos	Atributos de requerimientos
Entender la procedencia de los requerimientos y sus relaciones	Matrices de trazabilidad de requerimientos

Atributos de los requerimientos

Con los atributos de requisitos se logra:

- **Dar** a los interesados **información útil** para filtrado, selección y análisis de requisitos.
- Proporcionar **información a los responsables** de control de decisión sobre el impacto de los cambios en los requisitos.
- Ayuda a **educar a los nuevos miembros** del equipo acerca de los requisitos.
- Proporciona **información histórica** acerca de los requisitos que ayuda a los equipos a mantener o mejorar el software entregado.

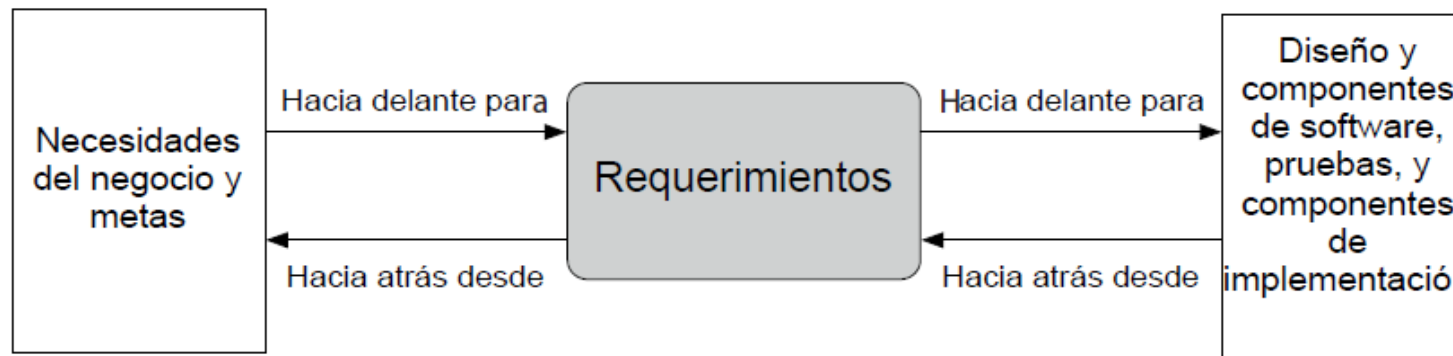
Atributos de los requerimientos

El proceso a realizar es:

- Identificar los atributos que Ud. necesita para hacer un seguimiento.
- Definir y mantener los atributos de todos los requisitos.

2.3 Matrices de seguimiento de requerimientos

Identifica cómo los requisitos están relacionados con resultados de desarrollo de software y otros requisitos



Trazabilidad de requisitos (Gottesdiener E. , 2005)

Cuando necesite:	Entonces crear:
Establecer mecanismos para manejar los cambios de los requerimientos	Cambio de control de políticas y procedimientos.
Identificar información suplementaria de requisitos	Atributos de requerimientos
Entender la procedencia de los requerimientos y sus relaciones	Matrices de trazabilidad de requerimientos

Las matrices de seguimiento de requisitos muestran los requisitos relacionados con el linaje hacia adelante y hacia atrás a los entregables del proyecto.

Matrices de seguimiento de requerimientos

Estas matrices son útiles para ***entender cómo los cambios de requisitos tienen un impacto y en otras prestaciones posteriores de desarrollo de software.***

¿Qué se debe hacer?

- Muestra las interdependencias entre los requisitos.
- Ofrece una visión de gestión mediante la identificación de lo que se debe entregar para satisfacer los requisitos.
- Proporciona informes que sean útiles para la vigilancia del cumplimiento del contrato.
- Demuestra que las necesidades han sido satisfechas por la asociación a los componentes del sistema y las pruebas .

Matrices de seguimiento de requerimientos

El proceso que se debe seguir es:

- Determinar qué resultados de desarrollo de software se deben dar seguimiento.
- Crear las matrices de seguimiento de los requisitos.

Ejemplo:

A continuación se indica un ejemplo de la construcción de una matriz de seguimiento de requisitos.

Matriz de seguimiento de requisitos durante el desarrollo de requisitos, muestra los requisitos funcionales derivados de los Casos de Uso				
Identificación de requerimientos	CASOS DE USO			
	UC1	UC2	UC3	UC4
SCH-3.2	X			
EST-3.2		X		
CLO-2.3			X	X

Matriz de seguimiento de requerimientos en el desarrollo de software						
Requisitos	Diseño de elementos		Código	Pruebas del sistema	Pruebas de aceptación	
Identificación de requerimientos	Identificación del paquete	Número de versión	Módulo	Script	Pruebas de casos	Fecha de pruebas de aceptación
SCH-1.2	DE-436	1.8	CVCS48491	SSCVSC01	ACTSC124	Julio 5
				SSCVC04	ACTSC249	Julio 8
SEC-1.0	DE-887	2.1	LBR309	SSSR9	ACTSR01	Agosto2

3. Ventajas y desventajas del uso de herramientas software en la IR

Para finalizar este tema de gestión de requisitos, se hace una evaluación de las ventajas y desventajas de las herramientas de software que se utilizan en la ingeniería de requisitos:

Ventajas

- a) Algunas de las herramientas permite **generar grandes repositorios de requisitos** que pueden ser utilizados no solo en uno, sino en varios proyectos dependiendo del contexto en el que se manejan.
- b) Permite **tener una visibilidad clara de cada uno de los requisitos**, es decir, la identificación de funcionalidades y restricciones que tendrá un proyecto o sistema.
- c) Permite **dar seguimiento a los requisitos durante todo el desarrollo** de un proyecto, el seguimiento implica trazabilidad, es decir, que los requisitos tendrán vida durante todo el proyecto.
- d) Algunas de las **herramientas permiten el manejo de versiones de los requisitos**, es una característica importante a la hora de analizar los requisitos y verificar la evolución de cambios que se ha desarrollado.
- e) **Permiten generar modelos** como casos de uso, de estado, diagramas de flujo entre otros, lo que facilita el análisis de los requisitos, su especificación y validación.
- f) **Manejo de un grupo de usuarios**, cada usuario registrado en el proyecto tendrá acceso a todos los requisitos capturados, podrá modificarlos, analizarlos, validarlos y notificar los cambios realizados.
- g) **Controlar, administrar los requisitos** y sus cambios para luego reutilizarlos en cualquier actividad del proceso de software que se necesite.
- h) **Reducción de costos y tiempo** en el proceso de requisitos de los proyectos.



Desventajas

- Una de las principales desventajas es **la cultura de uso de las herramientas de ingeniería de requisitos**, dichas herramientas no son muy utilizadas en la administración y gestión de requisitos en el desarrollo de proyectos, en la mayoría de casos este proceso es manual, y en otros casos se usa plantillas elaboradas en Microsoft Word solo para captura de requisitos.
- **La infraestructura necesaria para usar estas herramientas es otro impedimento**, se necesita comunicar los equipos en red, en algunos casos, la utilización de bases de datos potentes para la obtención de grandes repositorios de requisitos es muy necesario.
- **Los costos por licenciamiento de las herramientas** como por ejemplo Rational Requisite Pro e IRqA (que ofrecen grandes beneficios), son altos. Adicional a esto, se suma los costos de las herramientas que se incorporen o interactúen con las de requisitos.
- **No permiten incorporar procesos ni modelos** de ingeniería de requisitos.

Comparación de Herramientas

Características	Requisite Pro	IRqA	LEAP SE	Visual Requisite
Captura	X	X	X	X
Análisis	X	X		
Negociación				
Especificación	X	X		
Validación y verificación		X		
Gestión de requisitos	X	X	X	X
Trazabilidad	X	X		
Facilidad de uso			X	X
No redundancia				
Generación de modelos	X	X		X
Comunicación de requisitos	X	X		
Generación de informes	X	X		X

Contrastación de herramientas frente a características de administración de requisitos



UNIVERSIDAD
DE LIMA





