

# UNIDAD 2: INGENIERÍA DE REQUISITOS

## ELICITACIÓN Y ANÁLISIS DE REQUISITOS

# Agenda

- Problemas de la obtención de los requisitos
- Técnicas de captura de requisitos
  - JAD
  - Cuestionarios
  - Entrevistas
  - Delphi
  - Etnografía
  - Prototipado

# ¿Qué es la Ingeniería de Requerimientos



Dentro de la Ingeniería de Software la IRQ consiste en un uso sistemático y respetivo de técnicas que abarcan las actividades de **captura, análisis, especificación, validación y gestión** de un conjunto de requisitos para el software, con el fin de que estos cumplan con **los objetivos del negocio** y sean **de calidad**

# ¿Qué es la Ingeniería de Requerimientos

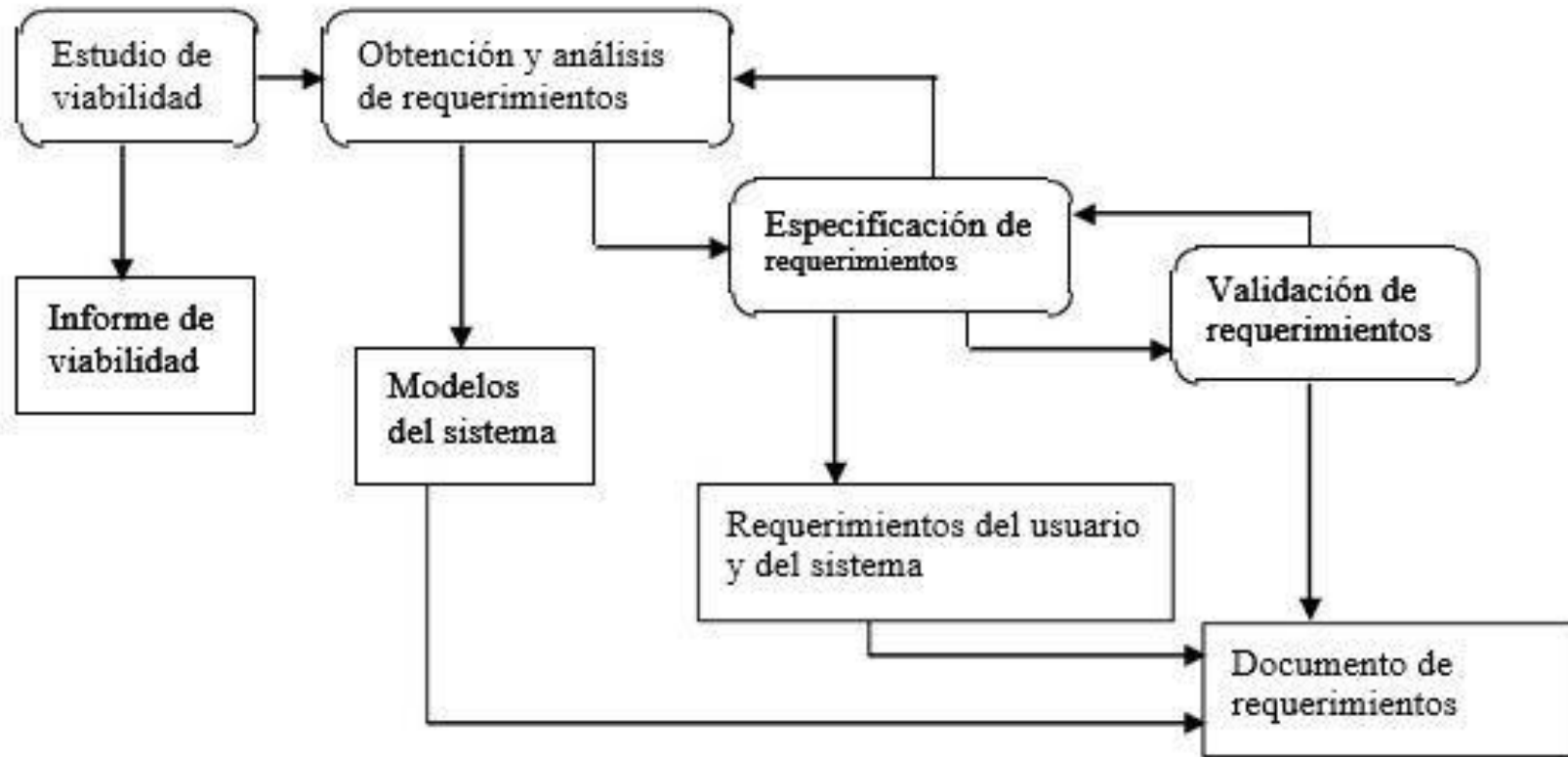
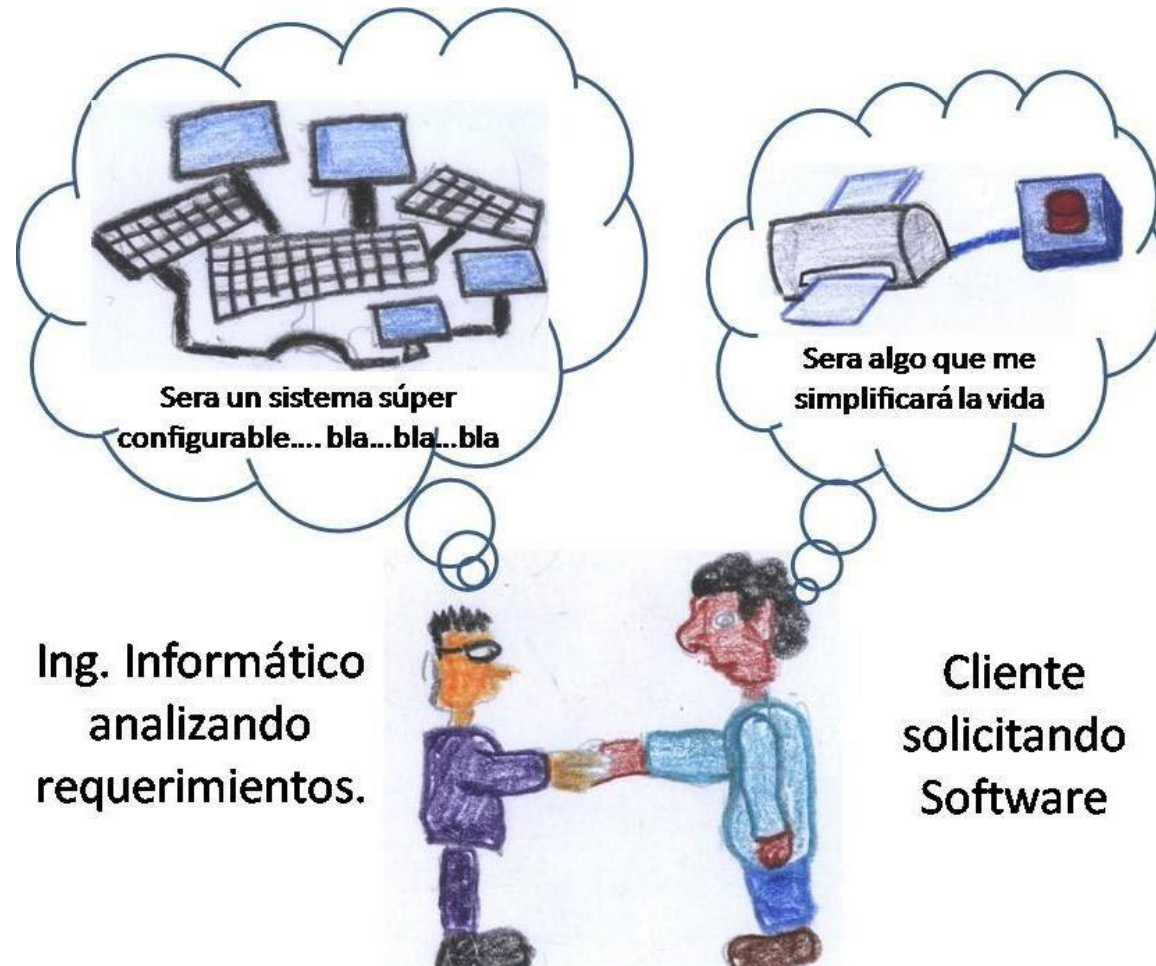
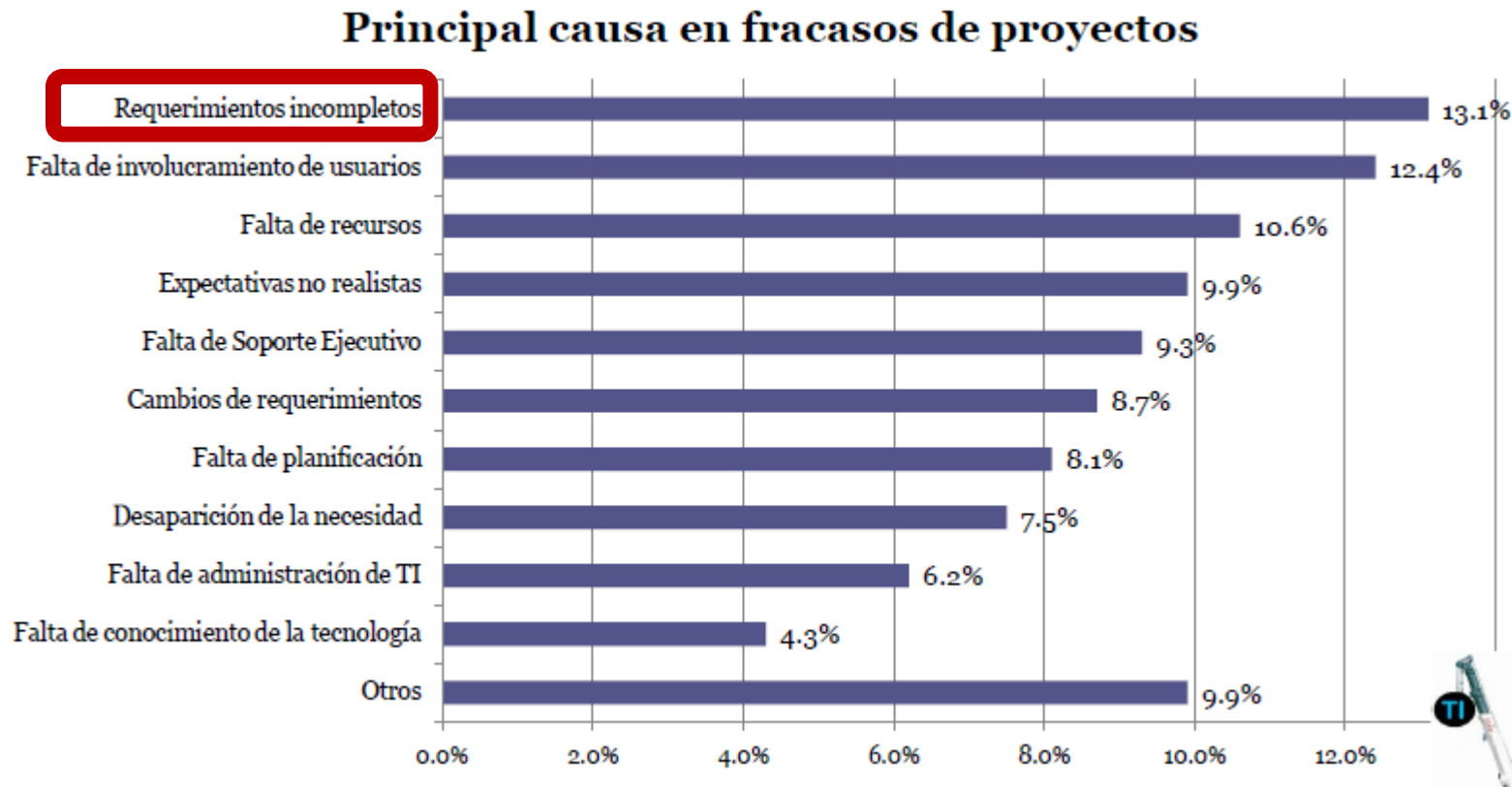


Figura 2.1: Proceso de ingeniería de requerimientos (Sommerville 2005).

# ¿Qué es la Ingeniería de Requerimientos



# Los requisitos ...



- La mayor parte de los defectos encontrados en el software entregado se originan en la fase de análisis de requisitos
- El análisis de los requisitos permite detectar problemas en los requisitos al comienzo del proyecto, evitando que aparezcan después, cuando es mucho más caro resolverlos.



UNIVERSIDAD  
DE LIMA



# ¿ Qué es un requisito ?



# Definiciones

## Requisito:

- Propiedad que debe ser exhibida por un software para resolver un problema particular (SWEBOK).
- Condición o capacidad que necesita el usuario para resolver un problema o conseguir un objetivo determinado.

## Especificación de Requisitos Software (ERS, SRS):

- Documento formal de los Requisitos del Sistema
- SRS – Software Requirements Specification

## Ingeniería de Requisitos:

- “Conjunto de actividades para descubrir, documentar y mantener un conjunto de requisitos”
- Es la ciencia y la disciplina que se ocupa de establecer y documentar los requisitos del software
- Establecer los servicios que el cliente requiere de un sistema y las restricciones bajo las cuales opera y es desarrollado.

## Gestión de Requisitos:

- Actividad para gestionar los cambios en los requisitos de un sistema



# Definiciones

## Sistema

- Incluye *hardware, software, firmware, personas, información, técnicas, servicios, y otros elementos de soporte*

## Requisitos del Sistema

- Son los requisitos para el sistema entero

## Requisitos del Software

- Se refieren solo al SW

# Tipos de requisitos



## Del negocio

Dan una descripción a alto nivel de lo que el sistema debe hacer. Representan los objetivos, estrategias, visión, alcance

NEEDS



## Del usuario

Describen la funcionalidad necesaria para satisfacer tareas específicas, necesidades operacionales y grupos de usuarios

FEATURES



## Del sistema/Software

Base para llevar a cabo la arquitectura, diseño y planes de pruebas del sistema



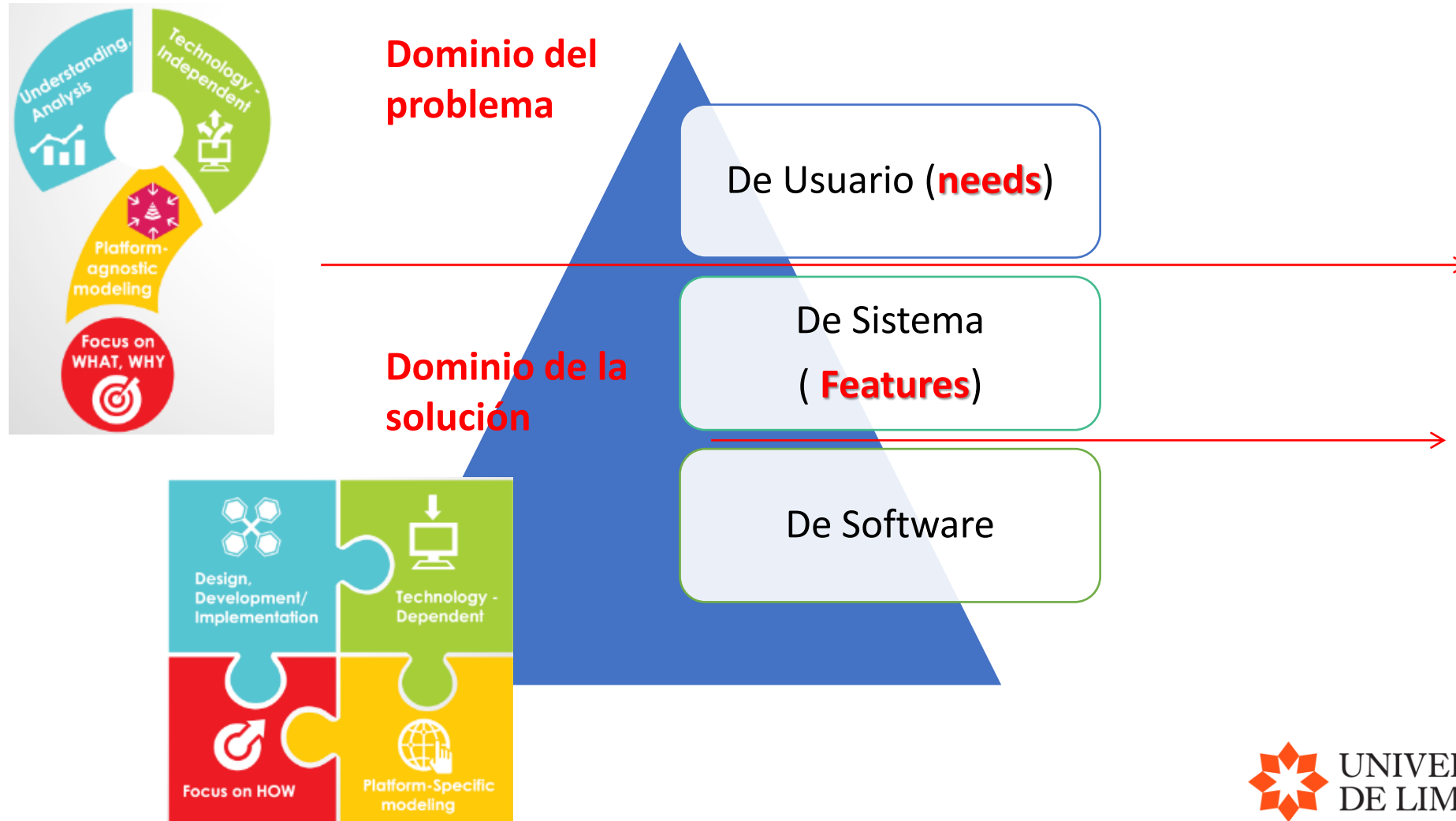
## Restricciones

Son condiciones que limitan las elecciones disponibles al diseñador o programador.

# Requisitos vs. Diseño

- Los Requisitos definen el **Qué** (el problema) del sistema
- El Diseño define el **Cómo** (la solución)

# Niveles abstracción



# Niveles de Requisitos

## De usuario

- Declaraciones en lenguaje natural, tablas y diagramas de los servicios que el sistema provee y sus restricciones operacionales.
- **Ejemplo:** El software debe proporcionar un medio para representar y acceder a los archivos creados por otras herramientas.

## De Sistema

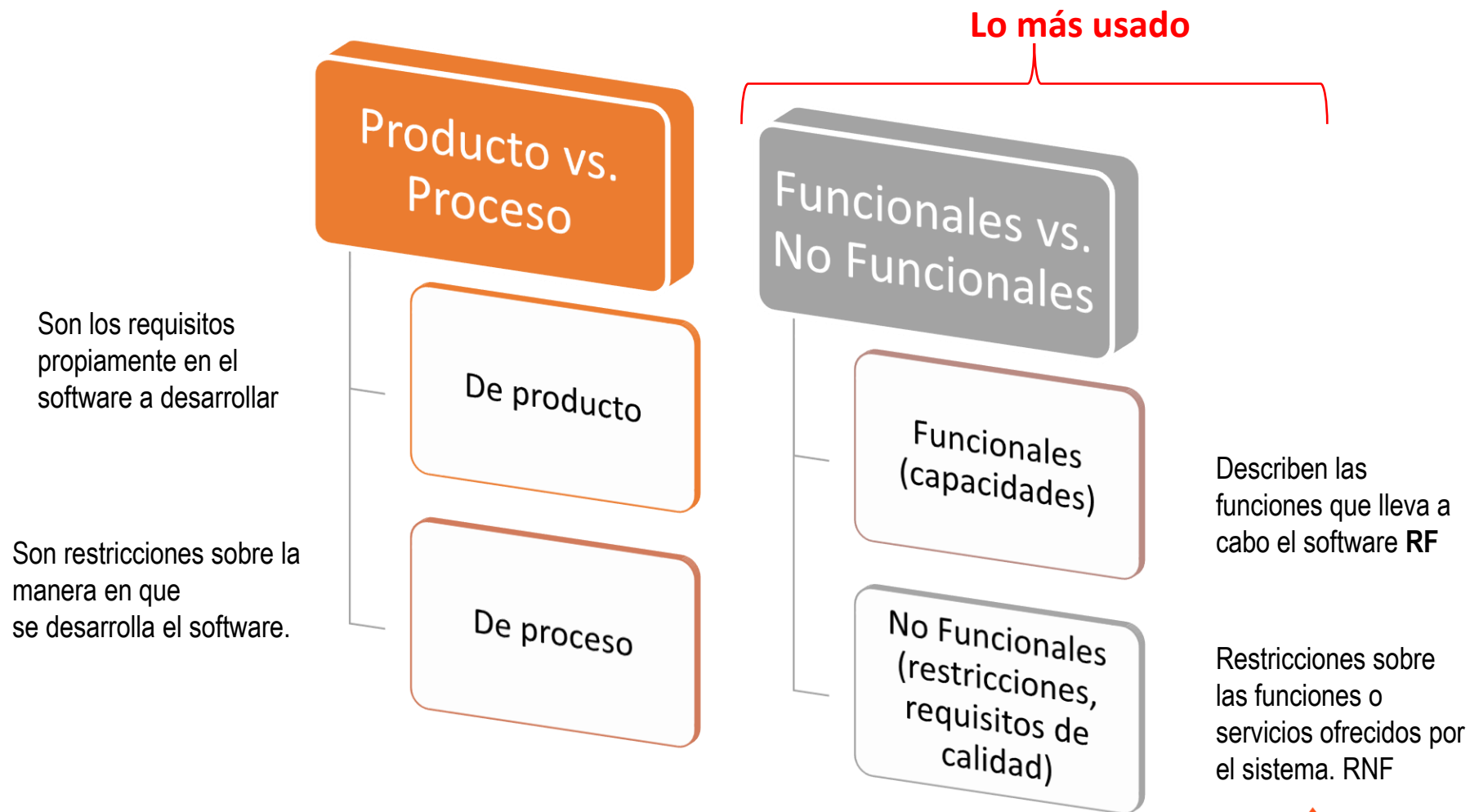
- Un documento estructurado con las descripciones detalladas de las funciones, servicios y restricciones operacionales del sistema.
- **Define lo que deberá ser implementado.**
- **Ejemplo:**
  - El usuario debe poder elegir el tipo de archivo
  - Cada tipo de archivo externo debe poder tener asociada una herramienta para poder abrirlo
  - Cada tipo de archivo externo tiene un icono asociado.

## De Software

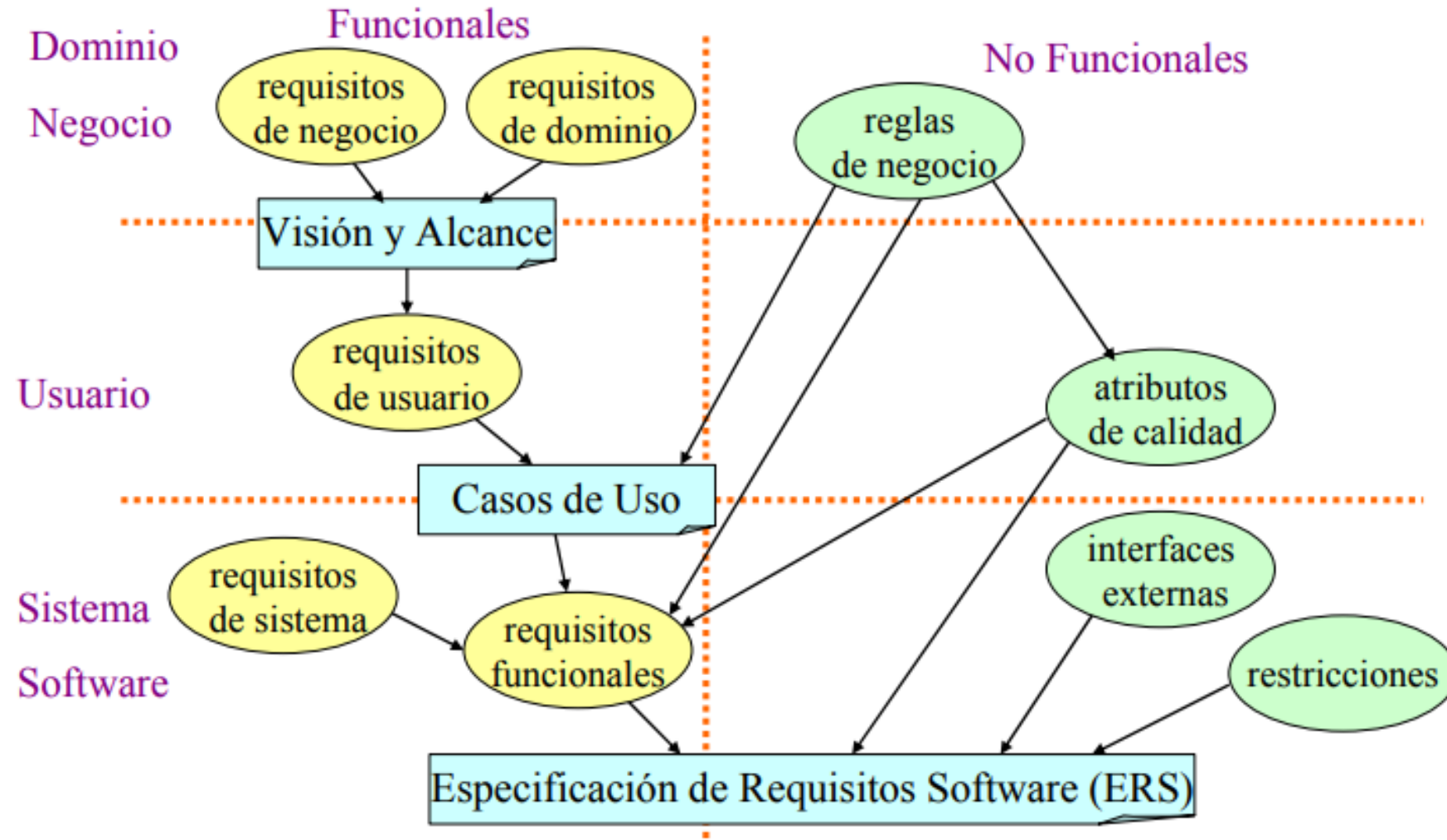
- Definiciones detalladas de diseño e implementación del software
- **Ejemplo:** Los iconos de los tipos de archivo tiene formato JPG.

Cada nivel está dirigido a una clase diferente de “lector”

# Niveles de requisitos: según naturaleza



# Niveles vs. Tipo



# ¿ cómo procedemos ?

- Requisito funcional: tiene que ver con todas las funcionalidades o s



varía en función de la metodología seguida, el dominio de aplicación, las personas participantes y la organización



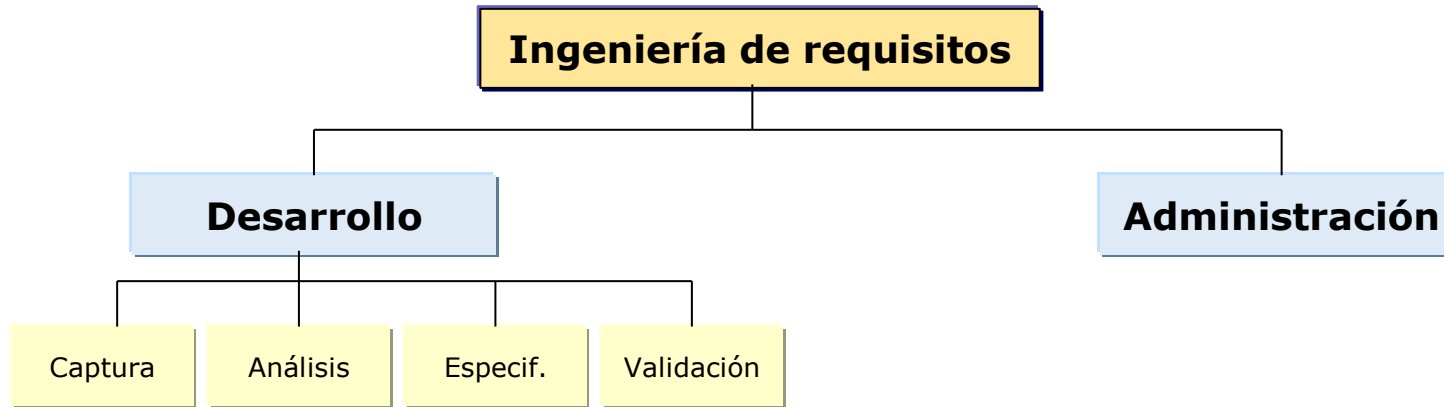
# Aparecen los stakeholders

- **Stakeholder** es una palabra del inglés que, en el ámbito empresarial, significa '**interesado**' o '**parte interesada**', y que se refiere a todas aquellas **personas** u **organizaciones** afectadas por las actividades y las decisiones de una empresa.



- El Proceso de Ingeniería de Requisitos está dominado por factores humanos, sociales y organizacionales.
- Implica diferentes **stakeholders** de diferentes procedencias y perfiles (técnicos, no técnicos de distintas disciplinas) y con distintos objetivos

# Ingeniería de requisitos - ciclo



“Ingeniería de requisitos es un enfoque sistémico para recolectar, organizar y documentar los requisitos del sistema; es también el proceso que establece y mantiene acuerdos sobre los cambios de requisitos, entre los clientes y el equipo del proyecto”

# Dos áreas según CMMI


Capability  
Maturity Model  
Integration-  
Development  
CMMI



UNIVERSIDAD  
DE LIMA



# Ingeniería de requisitos - Etapas



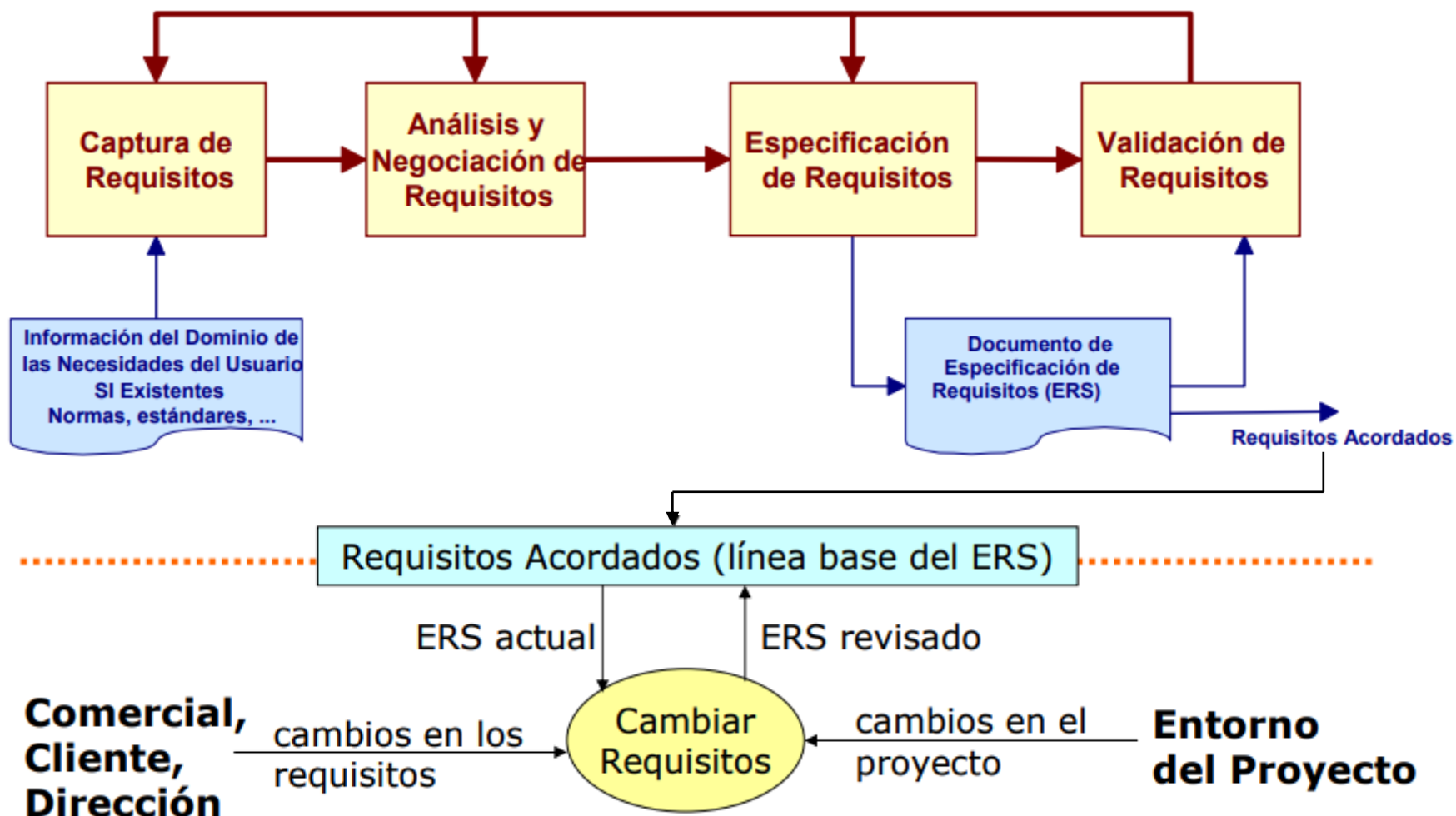
**Captura :** Son las actividades involucradas en el **descubrimiento de los requisitos** del sistema. Aquí los analistas de requisitos deben trabajar junto al cliente para descubrir el problema que el sistema debe resolver, los diferentes **servicios** que el sistema debe **prestar** y las **restricciones** que se pueden presentar.

**Análisis:** Se enfoca en **descubrir problemas con los requisitos** del sistema identificados hasta el momento. Usualmente se hace un análisis luego de haber producido un bosquejo inicial del documento de requisitos; en esta etapa se leen los requisitos, se conceptúan, se investigan, se intercambian ideas con el resto del equipo, se resaltan los problemas, se buscan alternativas y soluciones, y luego se van fijando reuniones con el cliente para discutir los requisitos.

**Especificación:** En esta fase se **documentan los requisitos** acordados con el cliente, en un nivel apropiado de detalle. 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/o estándares de documentación, como la notación UML.

**Validación:** Es la etapa final de la Ingeniería de Requisitos. Su objetivo es **verificar** todos los requisitos que aparecen en el documento especificado para **asegurarse que representan una descripción, por lo menos, aceptable del sistema** que se debe implementar.

# Desarrollo vs. Gestión



# Captura de Requisitos de Software

# Elicitación de Requisitos

1

- Se define como el **proceso de identificar las necesidades del negocio**, solucionando las posibles divergencias entre las personas involucradas en el mismo, con el propósito de definir y depurar los requisitos para cumplir las restricciones impuestas por las distintas partes.
- Un buen proceso de obtención de requisitos soporta el desarrollo de la especificación de los requisitos, de tal forma que tengan los siguientes atributos:
  - Deben ser **completos, consistentes** y han de estar dentro del alcance del proyecto
  - Deben tener un único identificador
  - Cumplen con los objetivos de los clientes
  - Son viables y apropiados para el desarrollo
  - Los requisitos deben ser “testeables” (debe poderse probar.)



# Elicitación de Requisitos

- El término Elicitación lo utiliza la comunidad para resaltar el hecho de que los buenos requisitos no sólo se obtienen desde los clientes, como se indica cuando se utiliza la frase “recopilación de requisitos”
- También llamada ***Captura / Descubrimiento de Requisitos***
- Se trata, esencialmente, de una **actividad humana** donde se identifican las partes interesadas y se establecen las relaciones entre el cliente, los usuarios y el equipo de desarrollo.



# ¿ De donde se originan los requisitos ?

Objetivos de negocio

Conocimiento del Dominio de problema

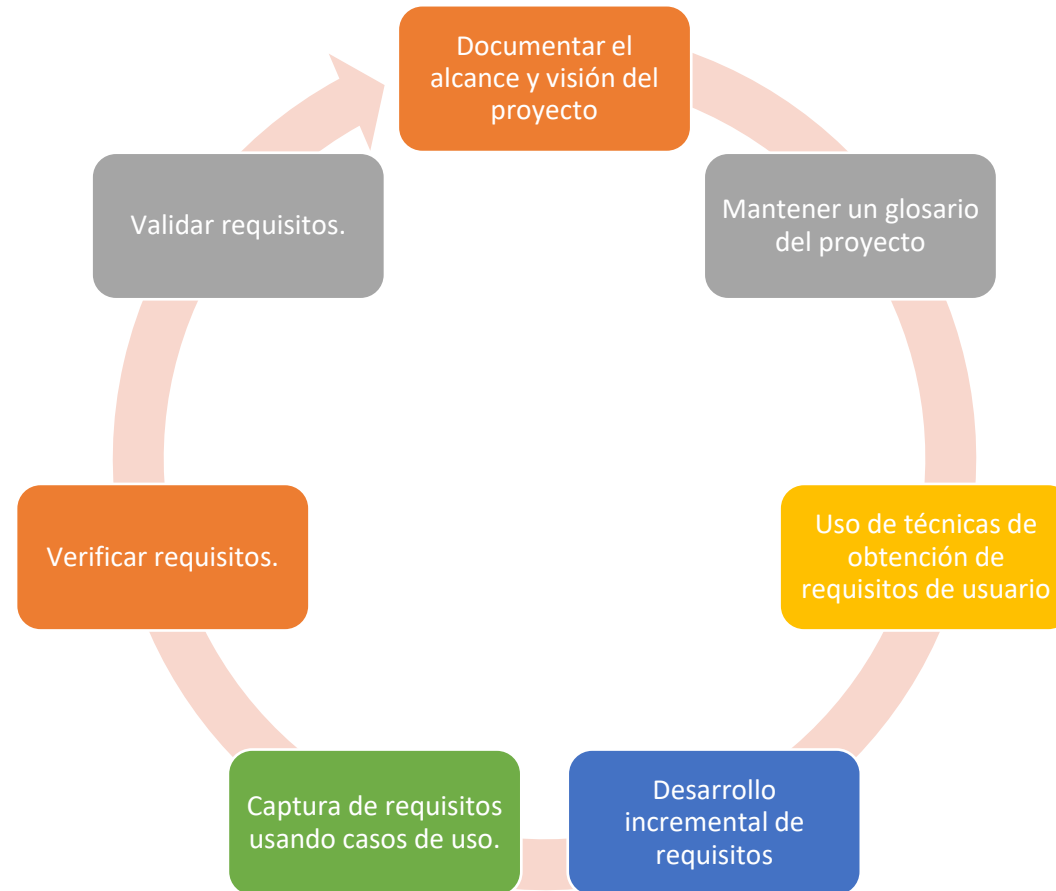
Stakeholders

Entorno operacional

Entorno Organizacional



# Mejores prácticas para los requisitos



# Problemas para obtener requisitos

Problemas de Alcance

Problemas comprensión

Problemas de Volatilidad

- La calidad de la **elicitación** determina la exactitud de la retroalimentación al cliente acerca de la integridad y validez de los requisitos.
- Debido a que esta fase es crítica y de alto impacto en el proyecto, es muy importante que la labor de elicitar se realice lo más cercano posible a la **perfección**.

## En consecuencia ...

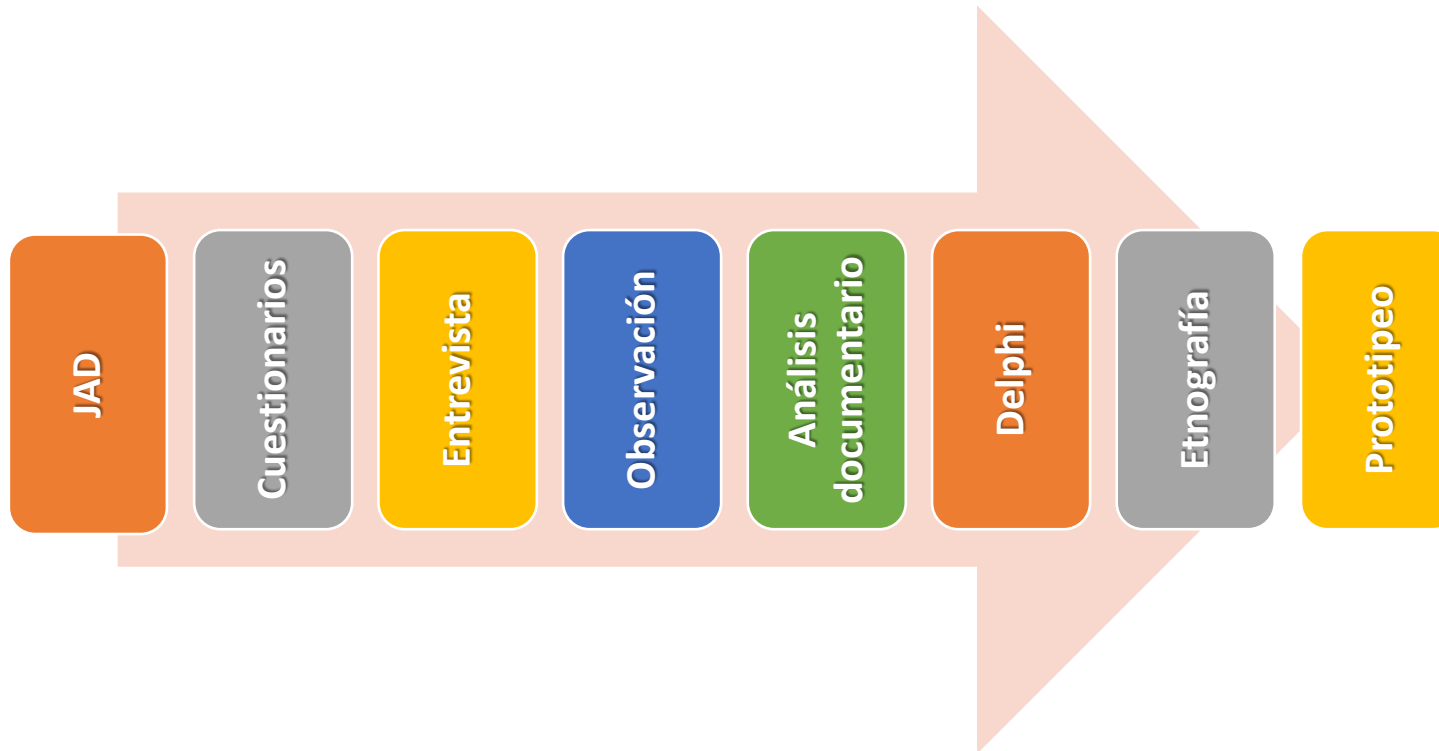




**Si existen tantos  
problemas,  
entonces ¿ cómo  
obtenemos los  
requisitos ?**

# Técnicas - Captura de Requisitos

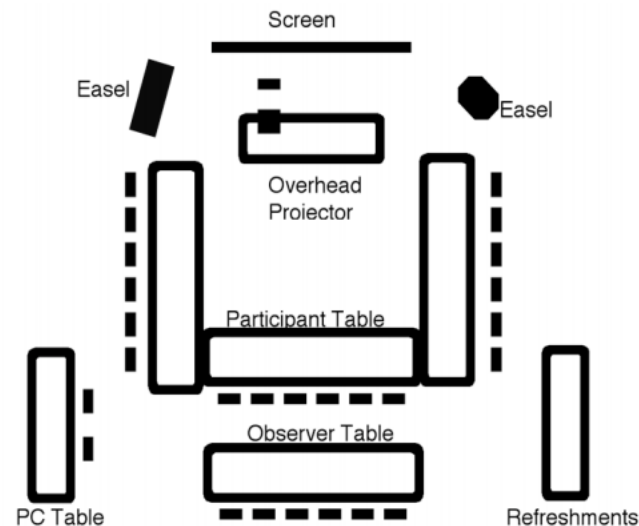
- Las Técnicas de Captura para Requisitos sirven para:
  - Conseguir que los stakeholders articulen sus requisitos
  - Recopilar el conocimiento sobre los requisitos del sistema



# Técnica: JAD: Join Application Development

1

- Es una técnica que se utiliza para promover la cooperación y el trabajo en equipo entre usuarios y analistas.
- Consiste en realizar sesiones en las que participan usuarios expertos del dominio junto a analistas de software.
- No es muy utilizado, debido a que requiere una mayor organización que las entrevistas y porque el ambiente o los métodos de trabajo convencionales en las empresas no facilitan este tipo de actividades



*Sample JAD Room Layout*

# JAD: ventajas

- Las entrevistas requieren mucho tiempo, no solo en prepararlas y hacerlas sino también en redactar un conjunto de requisitos coherente a partir de opiniones diferentes de los distintos entrevistados.
- En el **JAD todo el grupo puede actuar como revisor y detectar defectos.**
- Impulsa una participación más activa de los usuarios en el proyecto, hasta el punto que los usuarios que participan adquieren un cierto sentido de propiedad en el sistema que se construye.
- **Los 4 pasos principales en el JAD:**
  1. Definir los objetivos de la sesión
  2. Preparar la sesión
  3. Conducir la sesión
  4. Producir los documentos



# JAD: pasos



Los objetivos deben ser **SMART**

Specific	states what will be done
Measurable	you know when they have been achieved
Achievable	can be accomplished in the time frame required
Realistic	achievable with adequate resources
Time-framed	states when they will be achieved

# Técnica: Cuestionarios

- El uso de cuestionarios permite a los analistas **reunir información proveniente de un grupo grande de personas**.
- El empleo de formatos estandarizados para las preguntas puede proporcionar datos más confiables que otras técnicas;
- Su amplio alcance al distribuirse asegura el anonimato de los encuestados, lo que puede conducir a respuestas más honestas.



- El inconveniente es que la respuesta puede ser limitada, ya que es posible que no tenga mucha importancia para los encuestados.
- Al igual que con las entrevistas, se debe seleccionar a los encuestados (en base al conocimiento y experiencia de éstos).

# Técnica: Entrevistas

- Es muy útil para obtener información cualitativa como opiniones o descripciones subjetivas de actividades.
- Es una técnica muy utilizada y requiere una mayor preparación y experiencia por parte del analista.
- Se define como un “**intento sistemático de recoger información de otra persona**” a través de una comunicación interpersonal que se lleva a cabo por medio de una conversación estructurada.



- No basta con hacer preguntas para obtener toda la información necesaria: es importante la forma en que se plantea la conversación y la relación que se establece con el entrevistado.

# Entrevistas

- Formales o Informales
- Cerradas (limitadas a un tema) o abiertas (sin agenda previa)
- Se debe tener en cuenta:
  - **Preparación.** documentarse e investigar la situación de la organización analizando los documentos disponibles , de tal forma que la entrevista se enfoque en aquellos aspectos que están solamente en la mente del entrevistado y que no son accesibles por otros medios como la observación o el análisis de documentos.
  - **Entrevistar al personal adecuado.** La mayoría de los analistas adoptan un enfoque top-down, comenzando a entrevistar a directivos para que brinden un panorama general de hacia donde deberían ir las cosas, y terminando por hablar con los empleados que aportan detalles importantes de la operación.
  - **Duración.** No más de un par de horas.
  - **Formato.** Utilizar preguntas abiertas, donde los entrevistados puedan dar detalles

# Entrevistas : Problemas

- Los especialistas en la aplicación usan terminología y jerga específica de un dominio: para los que toman requerimientos es fácil de malinterpretar.
- Cierta conocimiento del dominio es tan familiar a los participantes que encuentran difícil de explicarlo, o bien, creen que es tan fundamental que no vale la pena mencionarlo.
- Las entrevistas tampoco son una técnica efectiva para adquirir conocimiento sobre los requerimientos y las restricciones de la organización, porque existen relaciones sutiles de poder entre los diferentes miembros en la organización.

**debe usarse junto con otras técnicas de adquisición de requerimientos.**

# Técnica: Observación

4

- Por medio de esta técnica el analista **obtiene información de primera mano** sobre la forma en que se efectúan las actividades.
- Este método permite observar la forma en que se llevan a cabo los procesos y, por otro, **verificar que realmente se sigan todos los pasos especificados** ( a veces lo que dice el papel difiere de la práctica)
- Los observadores experimentados saben qué buscar y cómo evaluar la relevancia de lo que observan.
- **No se plantearán preguntas: sólo se debe observar y tomar notas o dibujos.**
- Las personas observadas **no deben saber que se les está evaluando**. En caso contrario, harán su trabajo de manera más eficaz que lo normal.



## Técnica: Análisis documentario

- Varios tipos de documentación, como manuales y reportes, pueden proporcionar al analista información valiosa con respecto a las organizaciones y a sus operaciones.
- La documentación difícilmente refleja la forma en que realmente se desarrollan las actividades, o donde se encuentra el poder de la toma de decisiones. Sin embargo, puede ser de gran importancia para introducir al analista al dominio de operación y el vocabulario que utiliza.
- Esto evita quitarle tiempo a la gente.
- Permite estar más preparados para emplear otras técnicas como entrevistas, cuestionarios o JAD.

## Técnica: Delphi

- El método Delphi, (basado en el oráculo de Delphos), fue ideado originalmente a **inicios de los 50** en el Centro de Investigación estadounidense RAND Corporation (Olaf Helmer y Theodore J. Gordon).
- **Inicialmente creado como un instrumento para realizar predicciones** sobre un caso de catástrofe nuclear. Se ha utilizado frecuentemente como sistema para obtener información sobre el futuro.
- Se define la técnica Delphi como un **método de estructuración de un proceso de comunicación grupal** que es efectivo a la hora de permitir a un grupo de individuos, como un todo, tratar un problema complejo.

*es un método de pronóstico sistemático que implica la interacción estructurada entre un grupo de expertos en un tema.*

La calidad de los resultados depende, sobre todo, del cuidado que se ponga en la elaboración del cuestionario y en la elección de los expertos consultados.



UNIVERSIDAD  
DE LIMA





# Delphi

- Una sesión Delphi consiste en la **selección de un grupo de expertos** a los que se les pregunta su opinión sobre cuestiones referidas a acontecimientos del futuro.
- La Técnica Delphi incluye **típicamente al menos dos rondas de expertos** respondiendo preguntas y dando justificación de sus respuestas, proporcionando la oportunidad entre las rondas de cambios y revisiones.
- En esta técnica los expertos se les permiten **ajustar sus respuestas** en las siguientes rondas.
- Debido a múltiples rondas de preguntas se hacen y porque cada miembro del panel se le dice lo que el grupo piensa en su conjunto, el Método Delphi **busca llegar a la respuesta “correcta” a través del consenso.**



# PASOS DEL MÉTODO DELPHI



# Etnografía

- Los sistemas de software no existen de forma aislada; se utilizan en un **contexto social y organizacional**, y los requerimientos de sistemas de software se derivan y se restringen acorde a ese contexto.
- Una razón de por qué muchos sistemas de software se entregan pero nunca se utilizan es porque **no se toma en cuenta la importancia de este tipo de requerimientos**.



- La **etnografía** es una técnica de observación que se puede utilizar para **entender los requerimientos sociales y organizacionales**.
- Un analista se sumerge por sí solo en el entorno laboral donde el sistema se utilizará. El trabajo diario se observa y se hacen notas de las tareas reales en las que los participantes están involucrados.



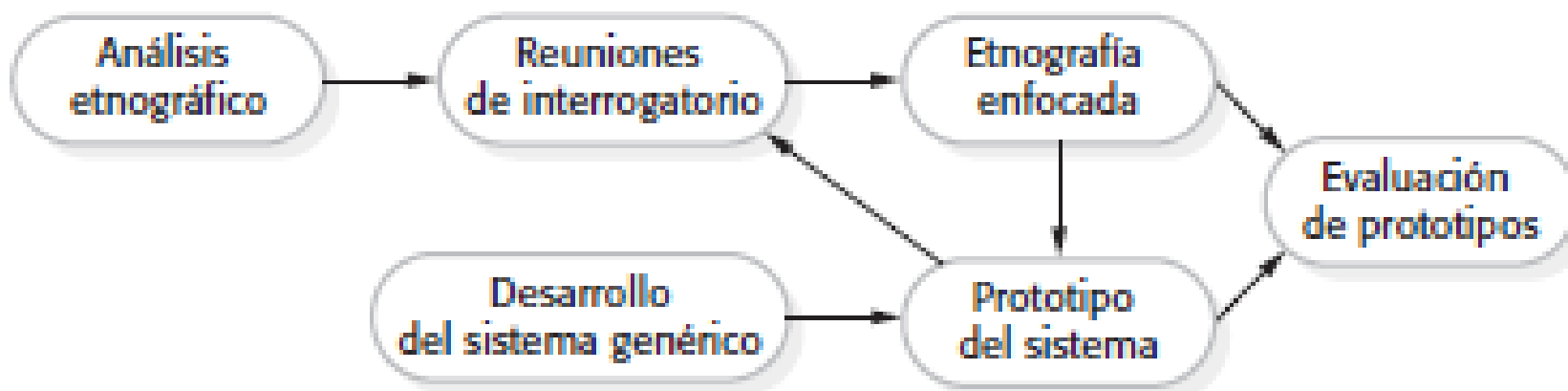
# Técnica: Etnografía

**La etnografía** es especialmente efectiva para descubrir dos tipos de requerimientos:

1. Los requerimientos que se derivan de la forma en la que la gente trabaja **realmente** más que de la forma en la que las definiciones de los procesos establecen que debería trabajar.
  2. Los requerimientos que se derivan de la **cooperación** y conocimiento de las actividades de la gente.
- Los estudios etnográficos **pueden revelar los detalles de los procesos críticos** que otras técnicas de obtención de requerimientos a menudo olvidan.
  - Sin embargo, sólo centran en el usuario final, por lo que este enfoque **no es apropiado para descubrir los requerimientos organizacionales o del dominio.**
  - La etnografía tampoco está diseñada para identificar nuevas propiedades a agregar al sistema.

# Etnografía

- Por lo tanto, la etnografía no es un enfoque completo para la obtención de requerimientos y **debe utilizarse en conjunto con otras técnicas**, como el análisis de casos de uso.
- La etnografía puede combinarse con la creación de prototipos. La etnografía informa del desarrollo del prototipo, de modo que se requieren menos ciclos de refinamiento del prototipo.



# Técnica: Prototipo

8

- Los prototipos son **versiones reducidas**, demos o conjuntos de pantallas (parcialmente operativos) de la aplicación pedida.
- Esta técnica es útil cuando:
  1. El dominio de la aplicación **no está bien definida** (posiblemente por ser algo muy novedoso).
  2. El **costo del rechazo** de la aplicación por los usuarios es **muy alto**.
  3. Es necesario **evaluar previamente el impacto** del sistema en los usuarios y en la organización
- permite solventar objeciones del usuario del tipo: “No sé exactamente lo que quiero, pero lo sabré cuando lo vea”.
- Incrementa los costos en las etapas iniciales de un proyecto, pero esto se recupera en etapas posteriores.







**Ya sabemos como  
obtener los  
requisitos**

**¿ cómo los clasificamos ?**

# Análisis de Requisitos de Software



# Análisis de Requisitos

2

- **El Análisis de Requisitos se refiere al esfuerzo para:**
  - **Detectar y resolver** conflictos entre requisitos
  - **Descubrir los límites** del software y cómo debe interactuar con su entorno
  - **Elaborar los requisitos** de sistema de cara a la posterior obtención de los requisitos de software
- Asegurarse de que los requisitos tengan **consistencia**, y en caso contrario, que cualquier conflicto entre los requisitos ha sido resuelto. **Dos requisitos están en conflicto si no pueden implementarse juntos**, es decir, si la solución a un requisito impide la implementación de otro.
- Cuando las expectativas del cliente son altas, la fecha de entrega y los recursos son limitados, es importante **asegurarse de que las funciones más importantes del producto sean entregadas** y además tan pronto como sea posible.
- Otro problema que se puede plantear es que haya demasiados requisitos.
- La solución a los problemas anteriores es la **priorización** de los requisitos.



# Análisis de Requisitos

El **Análisis de Requisitos** se refiere al esfuerzo de analizar los requisitos para :

- **Detectar** y resolver conflictos entre requisitos
- **Descubrir** los límites del software y cómo debe interactuar con su entorno
- **Elaborar los requisitos** de sistema de cara a la posterior obtención de los requisitos software.

Para esta etapa se recomienda realizar **4 tareas** a realizar con los requisitos :

- **Clasificación**
- **Modelado Conceptual**
- **Localización**
- **Negociación**



# Actividades para el análisis

## Clasificación

- RF vs. RNF
- Producto vs. Proceso
- Prioridad
- alcance

## Modelado Conceptual

- Busca la comprensión del problema
- Diagrama de Flujo de datos
- Máquinas de estado
- Diagramas de Eventos
- Diagramas de interacción de usuario
- Modelo de objetos y de datos

## Localización

- Asignar a componentes la responsabilidad de satisfacer requisitos
- Se requiere el diseño arquitectural previo
- Se puede hacer una análisis más detallado de cada componente

## Negociación

- También llamado resolución de conflictos.
  - ? Entre interesados que requieren características mutuamente incompatibles,
  - ? Entre requisitos y recursos
  - Entre requisitos funcionales y no funcionales
  - También puede realizarse durante la validación.
- Se busca consenso

# Análisis de Requisitos



En esta fase el cuidado se debe tomar para **describir requisitos con exactitud**, suficientemente como **para permitir que los requisitos sean validados**, su implementación **sea verificada**, y sus costes estimados. (IEEE Computer Society, 2004).

# 1- Clasificación

Según la naturaleza del proyecto, los requisitos se pueden clasificar por varios criterios:

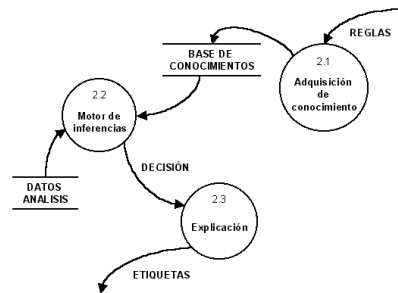
- **Funcionales vs No Funcionales**
- **Origen**
  - Requisito de alto nivel vs establecido directamente sobre el software
- **De producto vs de Proceso.**
- **Prioridad**
- **Alcance**
  - Componentes o piezas software afectados.
  - Algunos no funcionales tienen alcance global.
- **Volátiles vs Estables.**



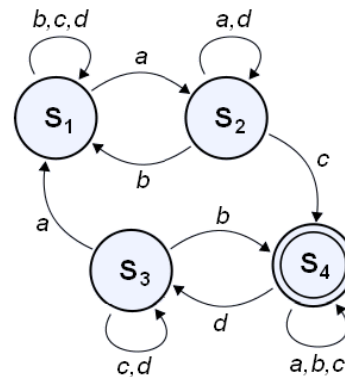
## 2- Modelado Conceptual

En Ingeniería del software, el **modelado conceptual** de los requisitos busca la comprensión del problema antes de iniciar el diseño de la solución.

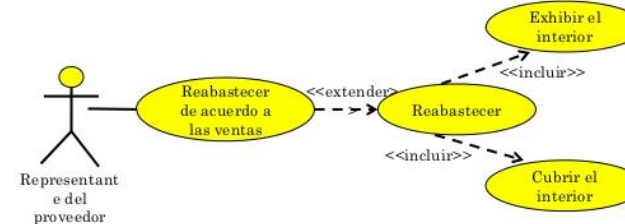
Principales Técnicas de Modelado de Requisitos:



Flujos de Datos y de Control (DFDs)



Modelos de Estados (máquinas de estados)



Modelos de Objetos y de Dato

Otros.

- Diagramas de Eventos
- Diagramas de Interacción de usuarios

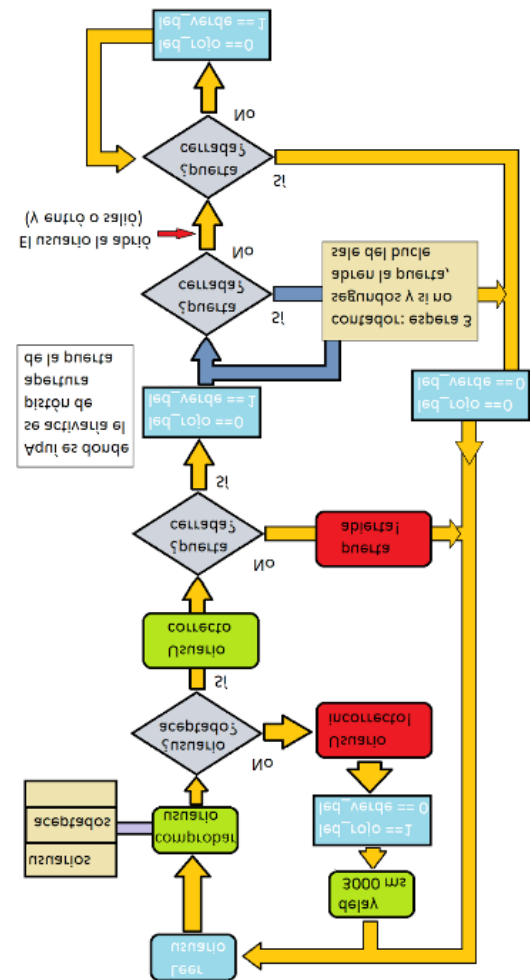
### 3- Localización

**La localización de requisitos** consiste en asignar a componentes la responsabilidad de satisfacer los requisitos.

- Para esto es necesario realizar antes el **diseño arquitectural** del software, ya que de el se derivan los componentes que tendrá el sistema.
- Una vez asignados requisitos a componentes es posible realizar un análisis más detallado de cada componente para descubrir nuevos requisitos sobre cómo debe interactuar con los demás componentes

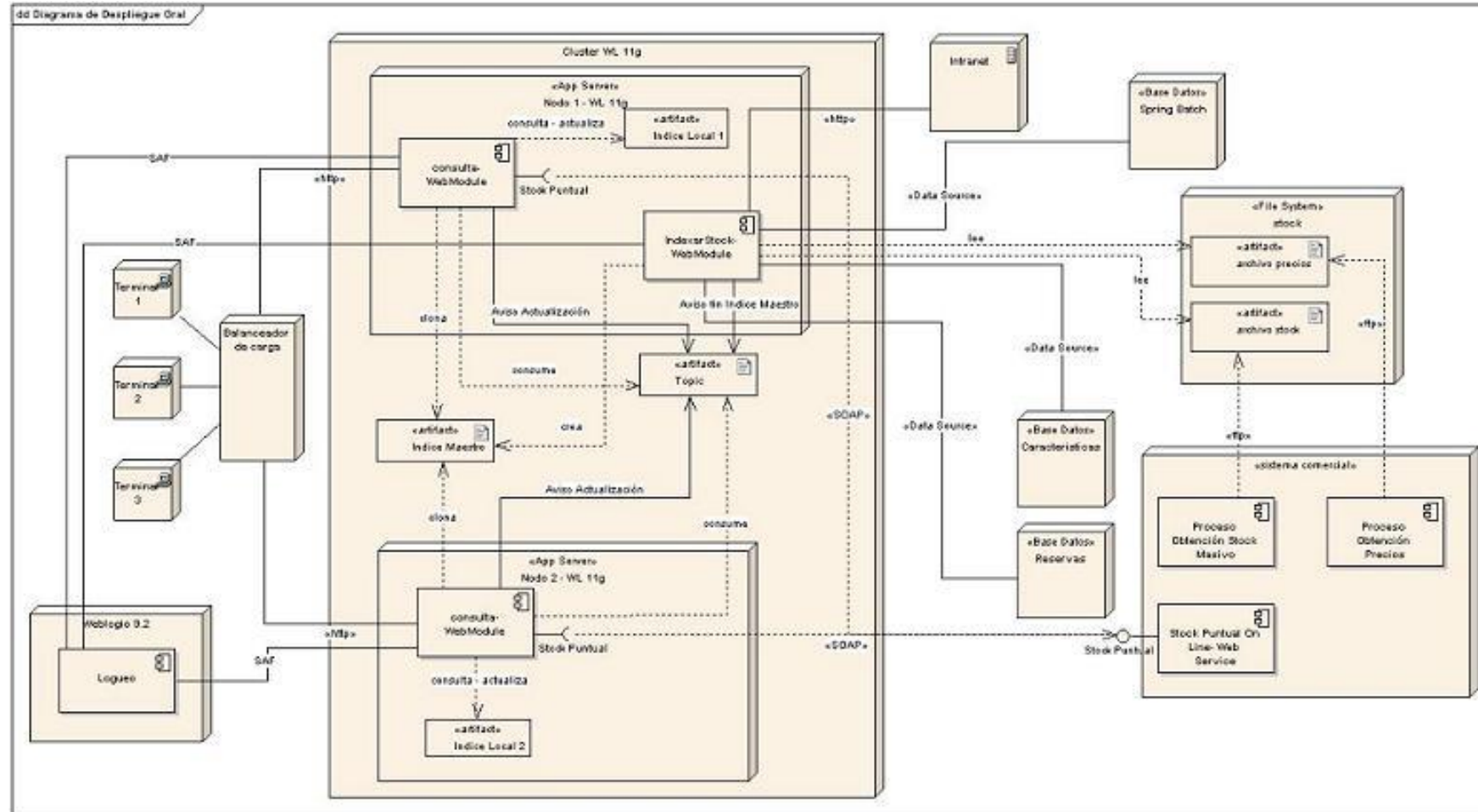


# Ejemplo





# Ejemplo



## 4. Negociación

También llamado **resolución de conflictos**. Consiste en resolver los problemas :

- Cuando dos interesados ***requieren características mutuamente incompatibles,***
  - Entre ***requisitos y recursos,***
  - Entre requisitos ***funcionales y no funcionales.***
- 
- Posibilita que , las decisiones deben estar basadas en el consenso.
  - También puede tener que llevarse a cabo durante la actividad de validación.



# Negociación de Requisitos

Los conflictos son prácticamente inevitables cuando intervienen diversos interesados ( Conflicto  $\neq$  Fallo )

**Para resolver un conflicto se recomienda seguir tres pasos:**

**1. Información.** Se explica la naturaleza de los problemas asociados con un requisito

**2. Debate.** Cada beneficiario expone cómo se podría resolver el problema

**3. Resolución.** Se llega a un acuerdo sobre las acciones que afectan a cada requisito (borrado, modificación o captura de más detalles)



***La negociación es el paso final en el análisis de los requisitos innecesarios, conflictivos, incompletos o inviables***



UNIVERSIDAD  
DE LIMA



# Técnicas para Análisis de Requisitos

# 1. Listas de Comprobación (CheckList)



La lista de comprobación (Checklist) consiste en una serie de preguntas o revisiones que se realizan sobre los requerimientos de software, que nos sean presentados de forma escrita.

- |                           |  |
|---------------------------|--|
| - Diseño Prematuro        | → Conformidad con Objetivos de Negocio |
| - Requisitos Combinados   | → Ambigüedad                           |
| - Requisitos Innecesarios | → Realismo (viabilidad)                |
| - Uso de HW no estándar   | → Facilidad de Prueba                  |

La lista de comprobación sirve de marco de trabajo y procedimiento para revisar el requerimiento, facilitando su análisis de forma estructurada.

Los requerimientos se pueden revisar sobre la matriz de trazabilidad de requerimientos o sobre la definición del alcance.



UNIVERSIDAD  
DE LIMA



# Listas de Comprobación (CheckList)



Una lista de chequeo puede realizar preguntas como:

- ¿Se han especificado los requisitos de hardware y software?
- ¿Se han realizado consideraciones de seguridad?
- ¿El nivel de granularidad del requerimiento se ha incluido?
- ¿Se ha incluido el código de referencia en para identificar el requisito en el desglose de requerimientos?
- ¿Está escrito el requerimiento en un lenguaje claro y conciso?
- ¿El requerimiento es único? no existe duplicidad con otro requerimiento.
- Y otras preguntas más .

## 2. Matriz de Interacción

Sirve para determinar si entre los requerimientos se **Repite** información o existe **Conflictos**

Requirement	R1	R2	R3	R4	R5	R6
R1	0	0	1000	0	1	1
R2	0	0	0	0	0	0
R3	1000	0	0	1000	0	1000
R4	0	0	1000	0	1	1
R5	1	0	0	1	0	0
R6	1	0	1000	1	0	0

### 3. Inspección

Revisión **no destructiva** de los requerimientos de software.

Por ejemplo:

- Examinar un software visualmente para constatar que las pantallas solicitadas se encuentran incluidas.
- Verificar la inclusión de los campos necesarios para el ingreso de datos.
- Verificar la existencia de los botones necesarios para iniciar la funcionalidad que ha sido requerida.
- Verificar que el requerimiento se apega a los estándares definidos para la aplicación. Por ejemplo estándares de navegación entre pantallas y estándares de interfaz gráfica.

De forma similar al uso de la lista de chequeo, **la inspección consiste en tomar el requerimiento definido en la matriz de trazabilidad** o definición de alcance, leerlo y producir un resultado para su corrección.



DE LIMA

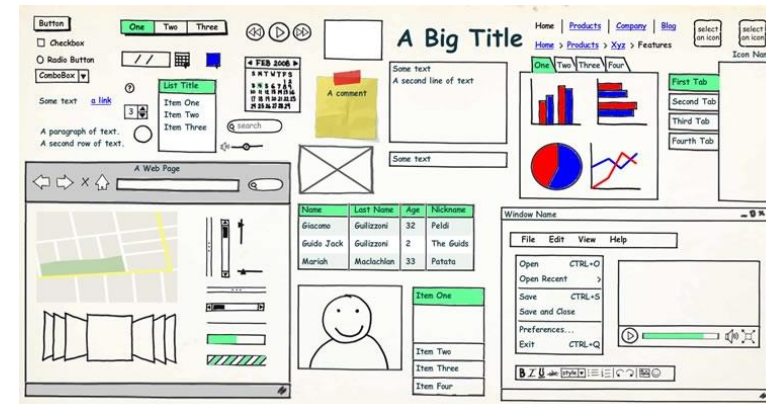




## 4. Prototipos

Consiste en elaborar **representaciones visuales** de los requerimientos de software.

- Permite a desarrolladores y usuarios entender mejor los requerimientos, determinar cuáles son **indispensables y cuales deseables**, e identificar riesgos de forma temprana.
- Puede enfocarse en toda la solución o sólo en áreas específicas.
- Puede extenderse innecesariamente en el tiempo si las discusiones se realizan en torno **al como** en lugar de en torno **al que**.
- **La elaboración de prototipos con lleva iteraciones entre desarrolladores y usuarios**, en los cuales se van elaborando varios prototipos y sometidos a evaluación del cliente.



## 5. Descomposición funcional

Se refiere al **proceso de identificar y resolver las relaciones funcionales en sus partes constituyentes**, de tal forma que la función global pueda ser reconstruida a partir de sus partes.

- En este proceso, es vital identificar las interacciones entre componentes.
- Aplicado a la Ingeniería de requisitos, consiste en tomar los requerimientos de software, dividirlos en partes y analizarlos individualmente. De ser necesario, se pueden descomponer en más partes hasta lograr un nivel adecuado de detalle.
- En cierto sentido, el proceso es similar al de la elaboración de una estructura de desglose de trabajo de un proyecto.
- La intención es dividir un sistema de tal forma que cada componente se pueda describir sin necesidad de referir a otro componente.
- De esta forma, cada parte del sistema tendrá funciones independientes, que pueden reusarse y reemplazarse.

