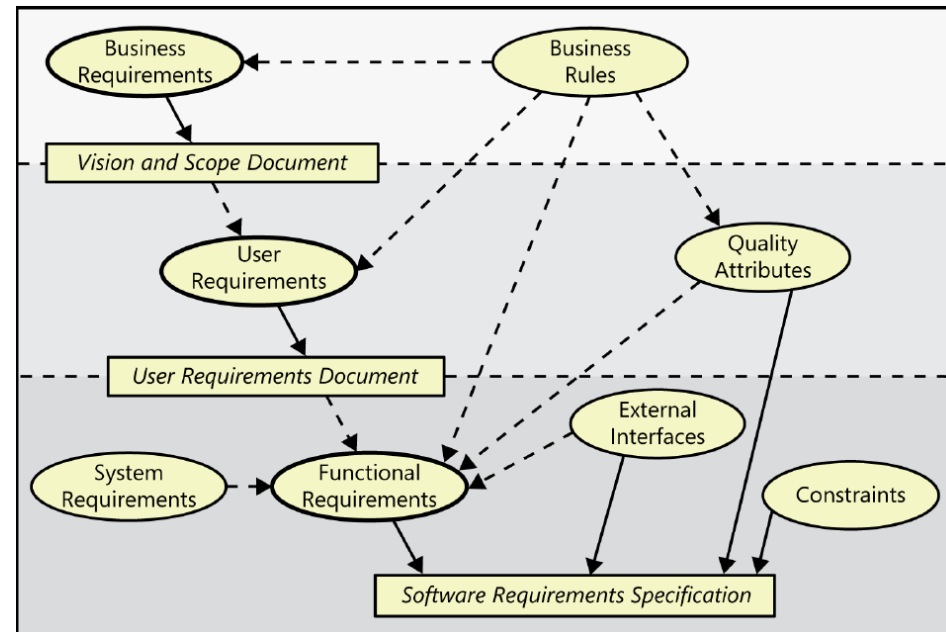


# Ingeniería de Software I

## Requerimientos

Docente: Dr. Pedro E. Colla

# Requerimientos



# Requerimientos



## Definición

Los *requerimientos* de **sistema** son una descripción de las funciones y características del sistema objetivo. Capturan las expectativas del cliente. Pueden ser obvios, conocidos, implícitos o explícitos, inesperados o básicos desde el punto de vista del cliente. El proceso de análisis y diseño determinará que parte se plasma en un aplicativo de software.

## Proceso para Ingeniería de Requerimientos

Compuesto por cuatro etapas:

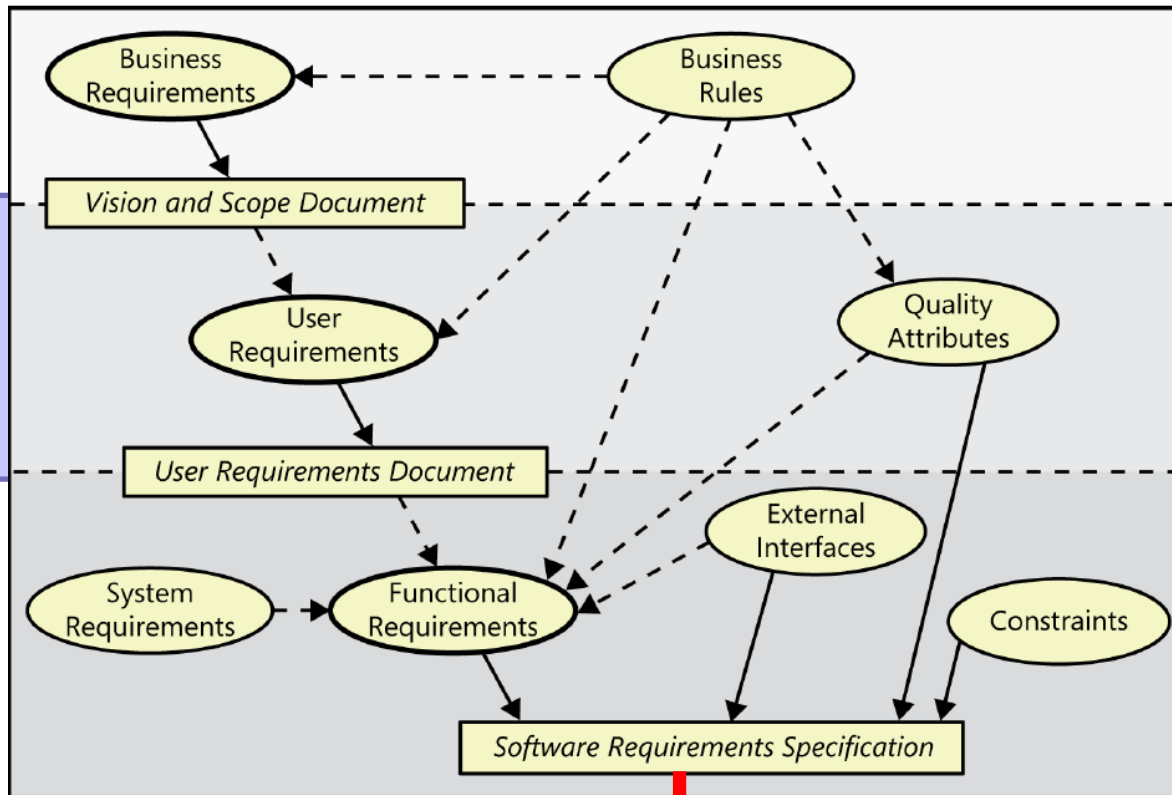
- Estudio de alcance y factibilidad.
- Relevamiento de requerimientos (vistas de usuario).
- Especificación de requerimientos.
- Validación de requerimientos.

Relevar requerimientos es un mecanismo de agregación y consenso.

# Vista Organizacional



## Definición



Procesos de negocios  
(Organizacional)

Requerimientos  
Funcionales y no-  
Funcionales

Análisis y  
especificación de  
requerimientos.

Backlog

# Requerimientos



# Tipos de requerimientos

FUNCIONALES



NO FUNCIONALES



**Requerimientos funcionales** son aquellos **que** definen que debe hacer el sistema. Mayoritariamente capturan las necesidades de los procesos de negocios desde la perspectiva del cliente.

- Se expresan en términos de lo que el cliente espera que el sistema haga. Puede ser parte de un proceso explícito (documentado) o implícito (usos y costumbres, proceso “ad-hoc”).
- Normalmente debe expresar al menos un “**camino feliz**” y todas aquellas circunstancias excepcionales que sea posible. Reciben el nombre de “**trayectorias**” pues son recorridos alternativos en la topología de un proceso (explícito o implícito).
- Debe proveerse información para, posteriormente, realizar el análisis correspondiente de las “trayectorias” para determinar su prioridad relativa de implementación.
  - Cantidad de veces o frecuencia de cada que es la trayectoria de máximo valor para la organización.
  - Magnitud total del negocio gestionado por la función.

# Tipos de requerimientos

FUNCIONALES



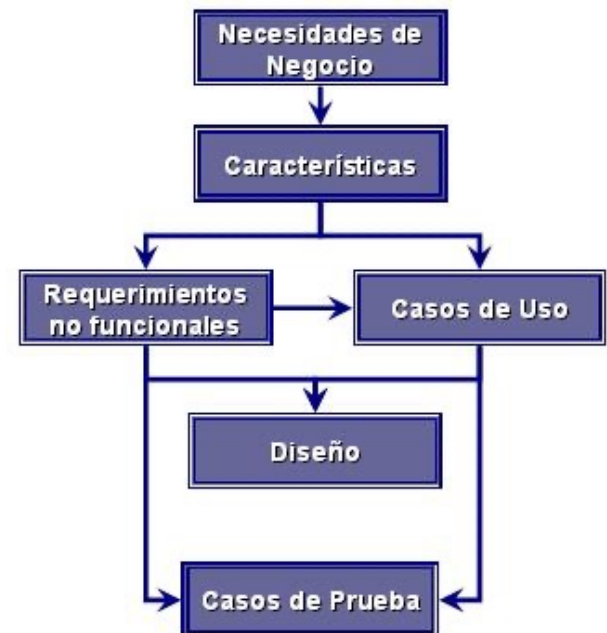
NO FUNCIONALES



**Requerimientos no-funcionales** son aquellos **que** definen pautas de comportamiento que debe mostrar el sistema. Pueden tanto expresar condiciones de negocio, restricciones organizacionales o buenas prácticas de implementación.

- Es común que el cliente los especifique en forma implícita o no los especifique “dándolos por obvios”.
- Pueden constituirse en condiciones de aceptación o rechazo del sistema al final del proyecto, por esa razón es **muy importante** documentarlos.
- Abarcan factores tales como:
  - Plataformas de uso organizacionalmente mandatoria (licencias existentes).
  - Factores de usabilidad.
  - Performance, tiempos de respuesta, ventanas de procesamiento.
  - Mantenibilidad.
  - Condiciones de garantía explícita.
  - Requerimientos de seguridad transaccional y de almacenamiento.
  - Respaldo y capacidades de “back-out”.
  - Condiciones de gestión de desastres.

# Gestión de Requerimientos





# Ciclo de gestión

FUNCIONALES



NO FUNCIONALES



Los requerimientos deben ser **recolectados** o **relevados** mediante una tarea de tipo holística que incluya a todos los estamentos relevantes de la organización cliente.



La Ingeniería de Requerimientos es la disciplina para desarrollar una especificación completa, consistente y no ambigua, la cual servirá como base para acuerdos comunes entre todas las partes involucradas y en dónde se describen las funciones que realizará el sistema. (Boehm)

Condición o capacidad requerida por el usuario para resolver un problema o alcanzar un objetivo (IEEE).

# Etapas



*Los requerimientos se relevan durante todo el proyecto de desarrollo, pero en particular al comienzo.*

- Debe considerar tres elementos fundamentales:
  - Comprender el problema.
  - Describir formalmente el problema.
  - Obtener un acuerdo sobre la naturaleza del problema.
- Las etapas para lograrlo son:
  - Elicitación de requerimientos
  - Especificación.
  - Validación

*Basado en Loucopolus et. al extraído de SEDICI (UNLP)*

# Elicitación



*El propósito de la elicitación de requerimientos es ganar conocimientos relevantes del problema, que se utilizarán para producir una especificación formal del software necesario para resolverlo. "Un problema puede ser definido como la diferencia entre las cosas como se perciben y las cosas como se desean".*

- Debe considerar tres elementos fundamentales:
  - Comprender el problema.
  - Describir formalmente el problema.
  - Obtener un acuerdo sobre la naturaleza del problema.
- El uso del lenguaje propio del usuario mejora considerablemente la comunicación. En el proceso de Ingeniería de Requerimientos la validación de los diferentes productos requiere una fuerte interacción con el usuario [Loucopoulos].
- Debe contener los elementos para que el analista pueda:
  - Comprender el dominio del problema.
  - Obtenga información sobre la importancia para el negocio.
  - Disponga de la perspectiva de prioridades relativas entre requerimientos.

# Elicitación



*El análisis de requerimientos siempre comienza con una comunicación entre dos o más partes. En Ingeniería de Software<sub>[Pressman]</sub> se establece que un cliente tiene un problema al **el cree** que puede encontrar una solución basada en computadora.*

- La principal herramienta para elicitar requerimientos es la entrevista.
  - Estudiar el dominio del problema.
  - Seleccionar entrevistados.
  - Determinar objetivos.
  - Planificar las entrevistas.
  - Elaborar cuestionarios (guías o formales).
- Realizar las entrevistas.
  - Apertura.
  - Desarrollo.
  - Terminación.
- Análisis de entrevistas.
  - Notas formales (compartir con el entrevistado, **siempre**).
  - Contrastar con otras entrevistas.
  - Identificar faltantes.

# Elicitación



La técnica de “*brainstorming*” puede ser usada para coleccionar requerimientos en dominios poco estructurados, sin soluciones aceptadas o con visiones divergentes entre los actores.

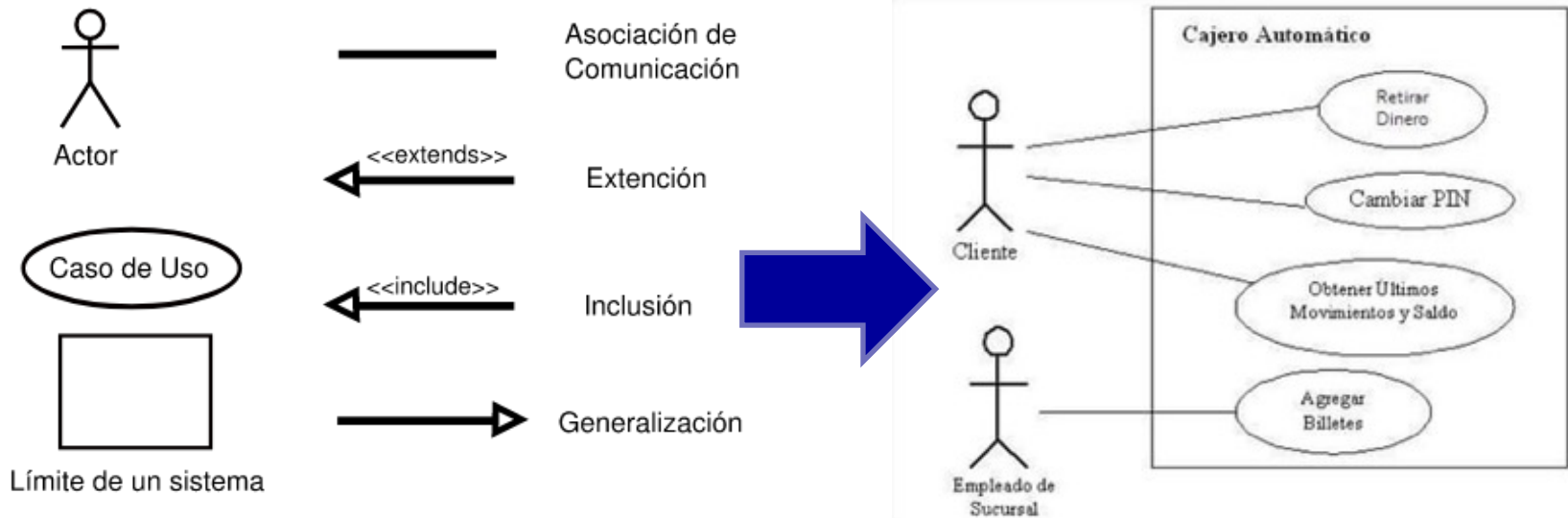
- Preparación, agendar formalmente.
- Generación (libre, sin crítica de ideas).
- Consolidación.
  - Revisar.
  - Descartar.
  - Priorizar.
- Documentar conclusiones.
- Análisis de entrevistas.
  - Notas formales (compartir con los participantes, **siempre**).
  - Contrastar con otras fuentes de requerimientos.
  - Identificar faltantes.

# Elicitación

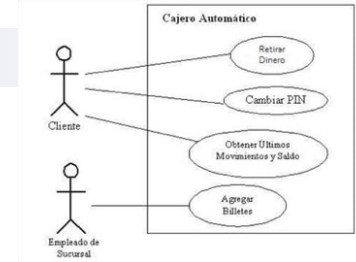


Un **caso de uso** es la descripción de una secuencia de interacciones entre el sistema y uno o más actores en la que se considera al sistema como una caja negra.

- Identifican eventos e interacciones.
- Describen el sistema desde el punto de vista de quien lo usa.
- Es menos vulnerable a las ambigüedades del lenguaje natural.
- Propone que la mejor forma de empezar a entender un sistema es a partir de los servicios o funciones que ofrece a su entorno, independientemente de los objetos que interactúan dentro del sistema para proveerlos.



# Taller Requerimientos



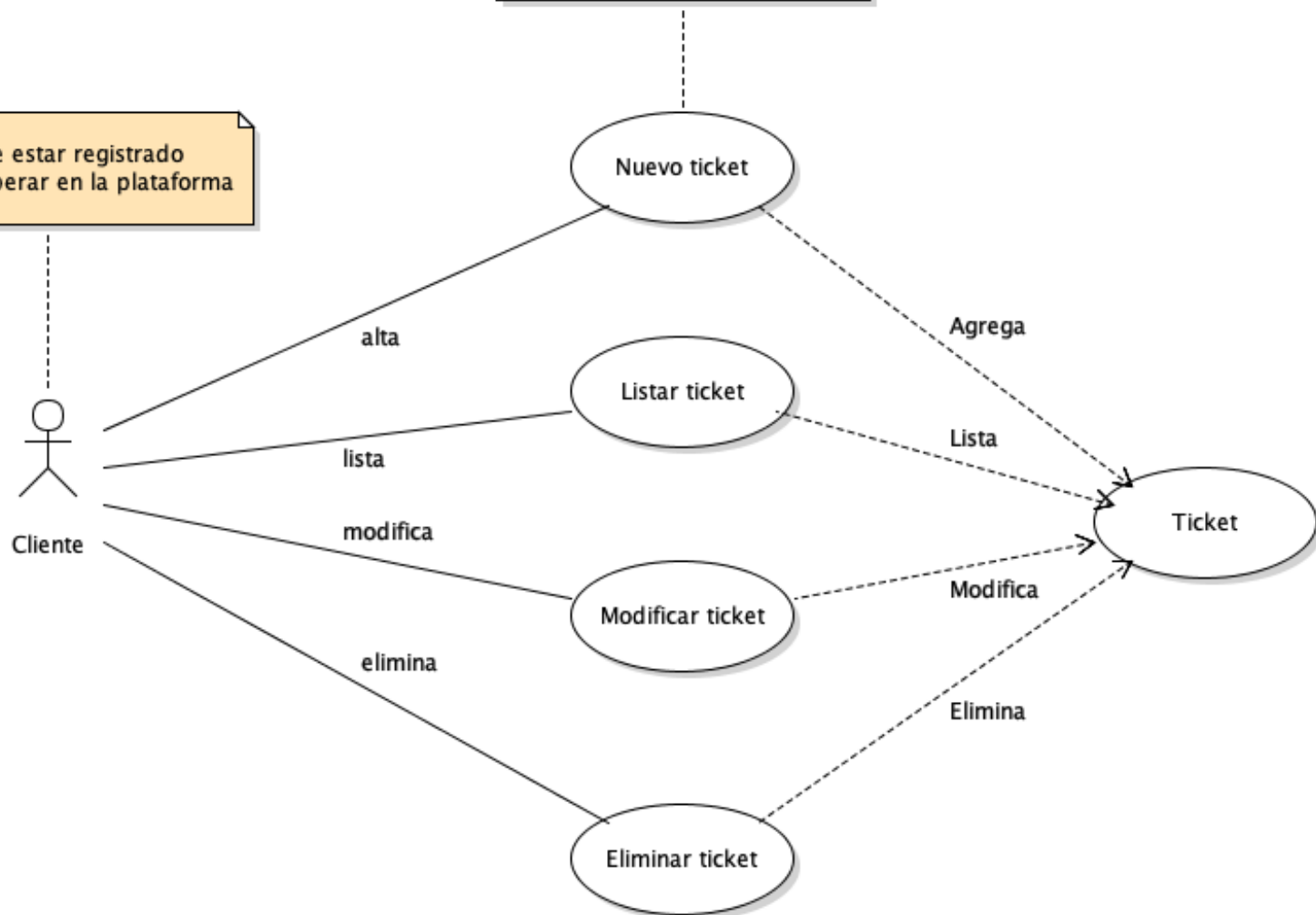
Documentar mediante la técnica de caso de uso el siguiente resultado de una entrevista para relevamiento de requerimientos.

- *Se trata de un sistema transaccional para la gestión de una mesa de ayuda de soporte tecnológico.*
- *El cliente (externo) ingresaría un pedido (ticket) de soporte en un portal.*
- *También podría visualizar todos los pedidos que hubiera hecho previamente.*
- *El cliente puede también cambiar un ticket ingresado previamente por el mismo.*
- *O tal vez borrar algún pedido activo que no desearía seguir reclamando.*

Documentar mediante cualquier editor UML disponible.

El cliente debe estar registrado para poder operar en la plataforma

Se puede trabajar solo sobre tickets generados por el cliente





# Optimización



*Extraído de "Elon Musk" por Walter Isaacson*

# Optimización de requerimientos



El criterio utilizado por *Elon Musk* en sus empresas (PayPal, Tesla, Space X, Twitter/X, The Boring Company, Neuralink) para definir el alcance de sus productos es el siguiente:

- ***Cuestiona cada requerimiento.***
  - ¿por qué el requerimiento existe? ¿qué valor aporta? ¿nombre de quien lo pidió?
- ***Elimine las partes del proceso que pueda, incluso las que no esté del todo seguro.***
  - Tener que reponer hasta un 10% de lo eliminado está bien.
- ***Simplifique y optimice.***
  - No optimice algo que debería ser eliminado.
- ***Acelere el tiempo de ejecución.***
  - Todo puede ser acelerado.
- ***Automatice todo lo posible.***
  - Si llegó hasta aquí tiene valor, y si tiene valor hay que hacerlo al menor costo posible.

**¿Preguntas?**

