

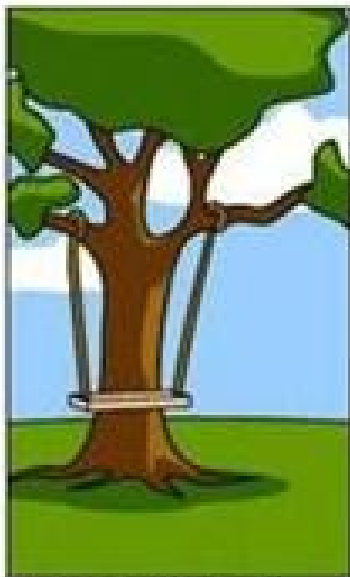
Ingeniería de Requisitos

Temario

- Definiciones
- Requisitos Funcionales y No Funcionales
- Tipos de Requisitos
- Ingeniería de Requisitos
 - Proceso de los Requisitos
 - Obtención de Requisitos - Técnicas
 - Modelado del Sistema - Técnicas
 - Especificación de Requisitos - Documentos de Requisitos
 - Validación y Verificación– Técnicas
 - Administración de los Requisitos
 - Planificación
 - Trazabilidad
 - Administración del cambio
 - Medición
- Metodologías de Desarrollo
- Especificación Formal - Z



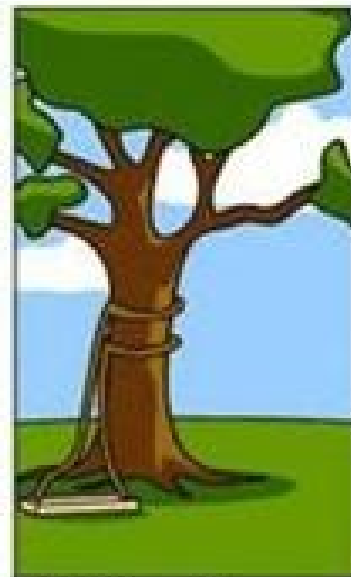
La solicitud del usuario



Lo que entendió el líder del proyecto



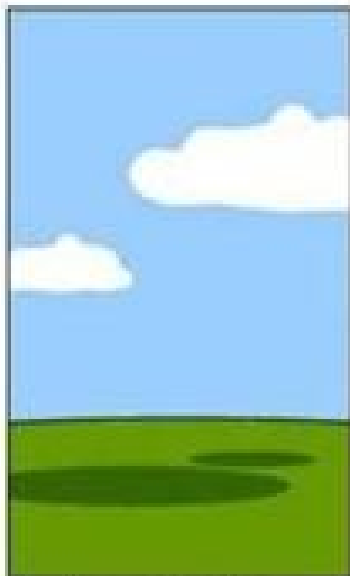
El diseño del analista de sistemas



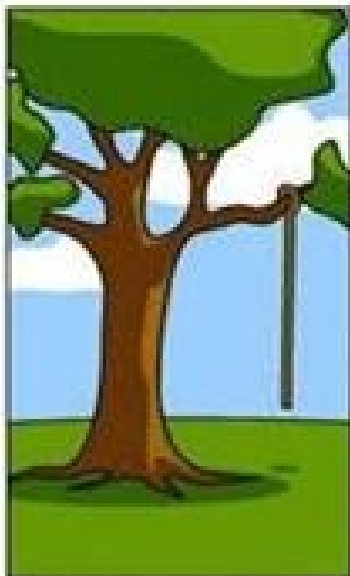
El enfoque del programador



La recomendación del consultor externo



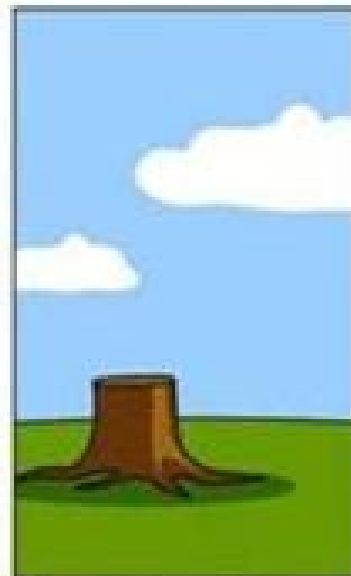
La documentación del proyecto



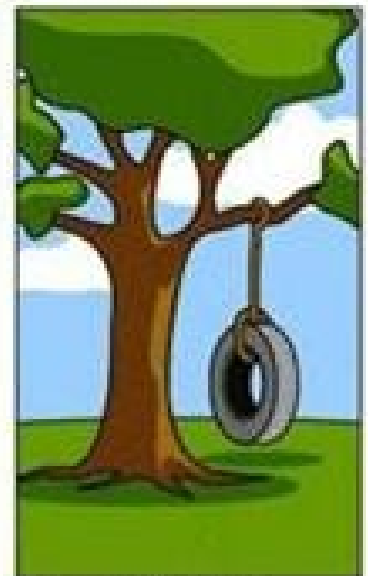
La implantación en producción



El presupuesto del proyecto



El soporte operativo



Lo que el usuario realmente necesitaba

Definiciones

- Requisitos:
 - Descripción de los servicios que debe brindar un sistema y sus restricciones.
- Ingeniería de Requisitos
 - Proceso de descubrir, analizar, documentar y verificar esos servicios y restricciones.

Definiciones

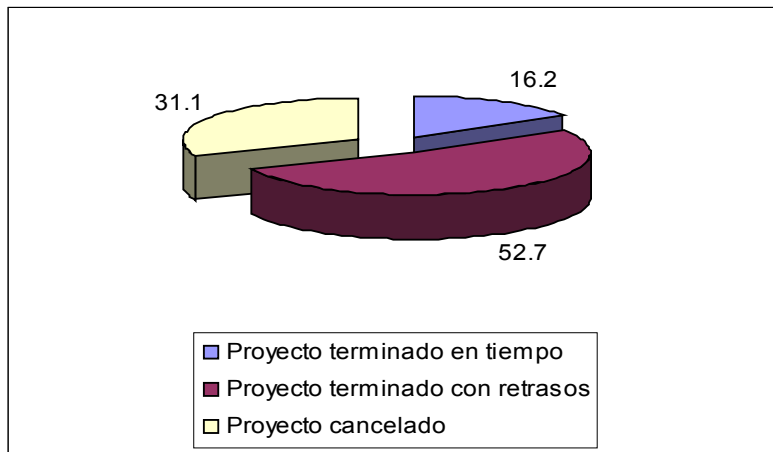
- Sistema
 - Incluye *hardware, software, firmware, personas, información, técnicas, servicios, y otros elementos de soporte*
- Requisitos del Sistema
 - Son los requisitos para el sistema entero
- Requisitos del Software
 - Se refieren solo al SW

Requisitos vs. Diseño

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

Reporte CHAOS de Standish Group `94

- 350 orgs., 8000 proyectos (Standish Gr.1994)

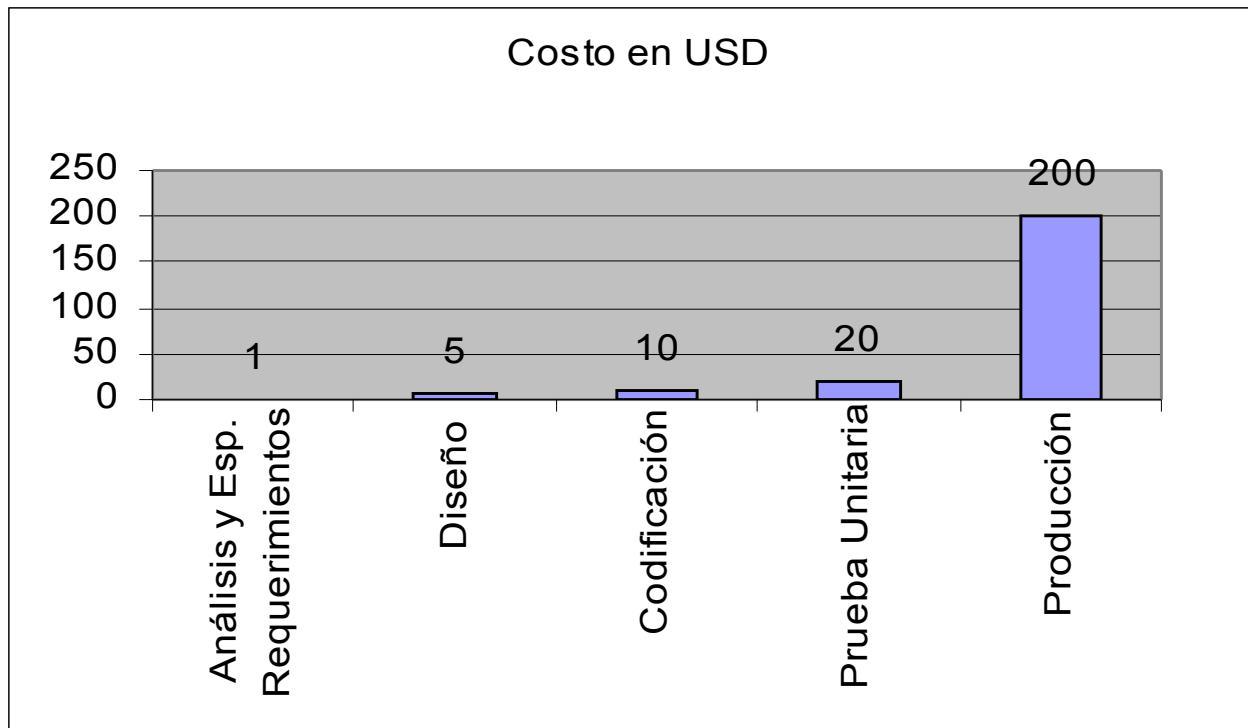


- Causas

Causas	% Respuestas
Requisitos incompletos	13.10%
Falta de involucramiento de usuarios	12.40%
Falta de Recursos	10.60%
Expectativas no realistas	9.90%
Falta de Soporte de Ejecutivos	9.30%
Requisitos y Especificaciones cambiantes	8.70%
Falta de planificación	8.10%
Sistema no se precisaba más	7.50%
	39.2 %

Costos de Errores en los Requisitos

- Costo de corregir un error en los requisitos (Boehm-Papaccio, 1988)



Requisitos Funcionales y No Funcionales

- Funcionales:
 - Servicios o funciones que proveerá el sistema
 - Describen la interacción entre el sistema y su entorno
 - **Ejemplos:**
 - Se deben ingresar cédula, nombre y teléfono de cada cliente
 - Se quiere un listado de los clientes por zona
- No-funcional:
 - Restricciones a los servicios o funciones ofrecidos por el sistema
 - Describen restricciones que limitan las elecciones para construir una solución
 - **Ejemplos:**
 - Las consultas deben resolverse en menos de 3 segundos
 - El lenguaje de programación debe ser Java

Requisitos No Funcionales

- Del Producto: Especifican restricciones al comportamiento del producto
 - Ejemplos: desempeño, confiabilidad, portabilidad, usabilidad
- De la Organización: Se derivan de las políticas y procedimientos existentes en la organización del cliente y en la del desarrollador
 - Ejemplos: estándares, lenguajes de programación, método de diseño
- Externos: Se derivan de factores externos, como:
 - Interoperabilidad: con otros sistemas
 - Legislativos: privacidad, seguridad
 - Éticos: dependen del contexto, las personas, etc

Requisitos - Tipos (1)

Al describir requisitos se deben tener en cuenta los siguientes aspectos:

- Ubicación y Entorno Físicos
 - dónde, uno o varios, restricciones ambientales
- Interfaces
 - Entrada de 1 o + sistemas, Salida a 1 o + sistemas, restricciones de formato, soporte
- Usuarios y Factores Humanos
 - capacidad de cada tipo de usuario, tipo de entrenamiento, facilidad de uso, posibilidad de mal uso
- Funcionalidad y Restricciones asociadas
 - qué debe hacer, cuándo, modos de operación, cómo y cuándo se puede modificar el sistema, restricciones de velocidad, tiempo de respuesta, capacidad de proceso

Requisitos - Tipos (2)

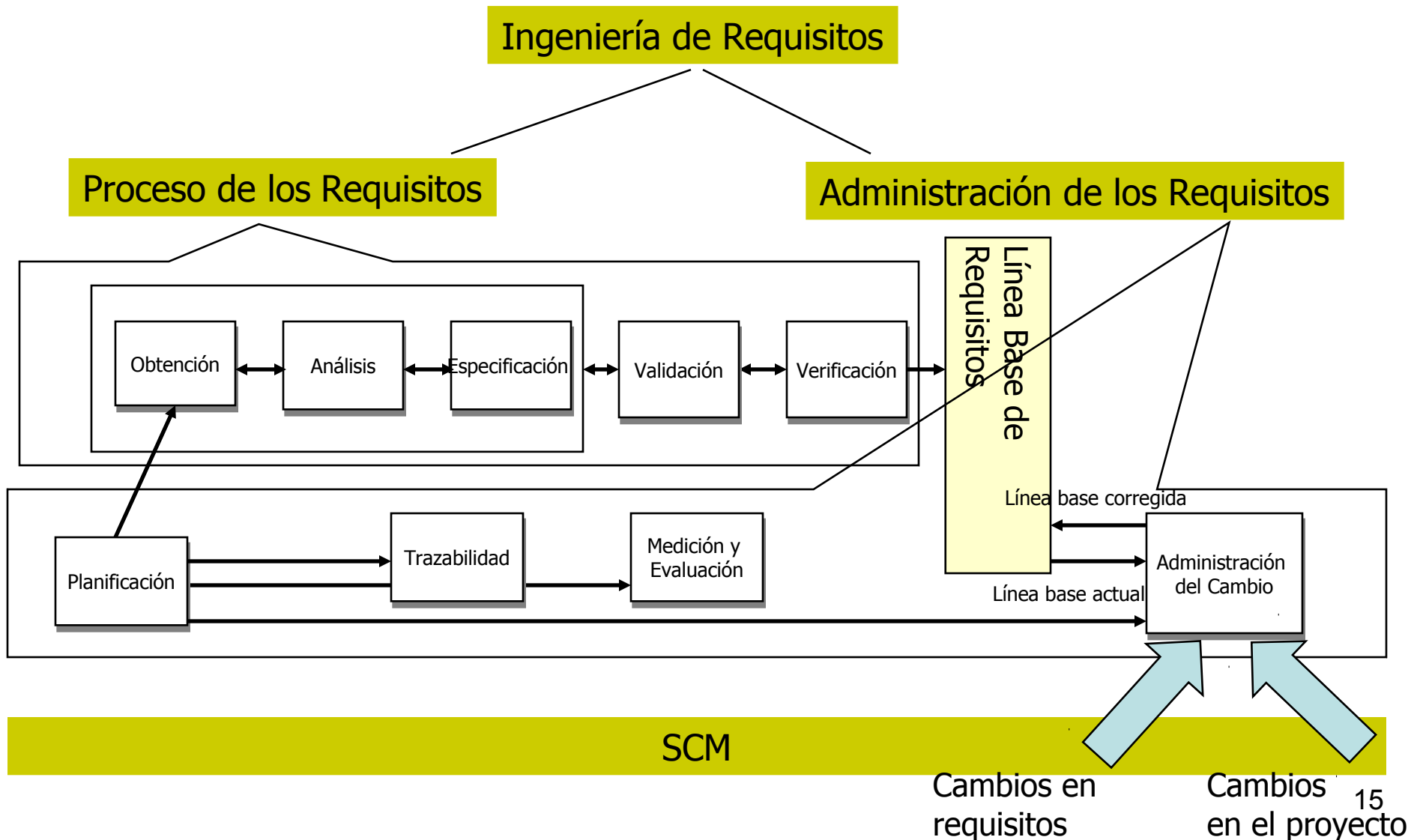
- Documentación
 - cuánta, formato, para quién
- Datos
 - formatos E/S, frecuencia, fuentes, destinos, calidad requerida, precisión en cálculos, flujo en el sistema
- Recursos
 - materiales, personal y otros para construir, usar y mantener el sistema, habilidades de los desarrolladores, necesidades de espacio y ambientales, calendario prescrito, limitaciones en presupuesto

Requisitos - Tipos (3)

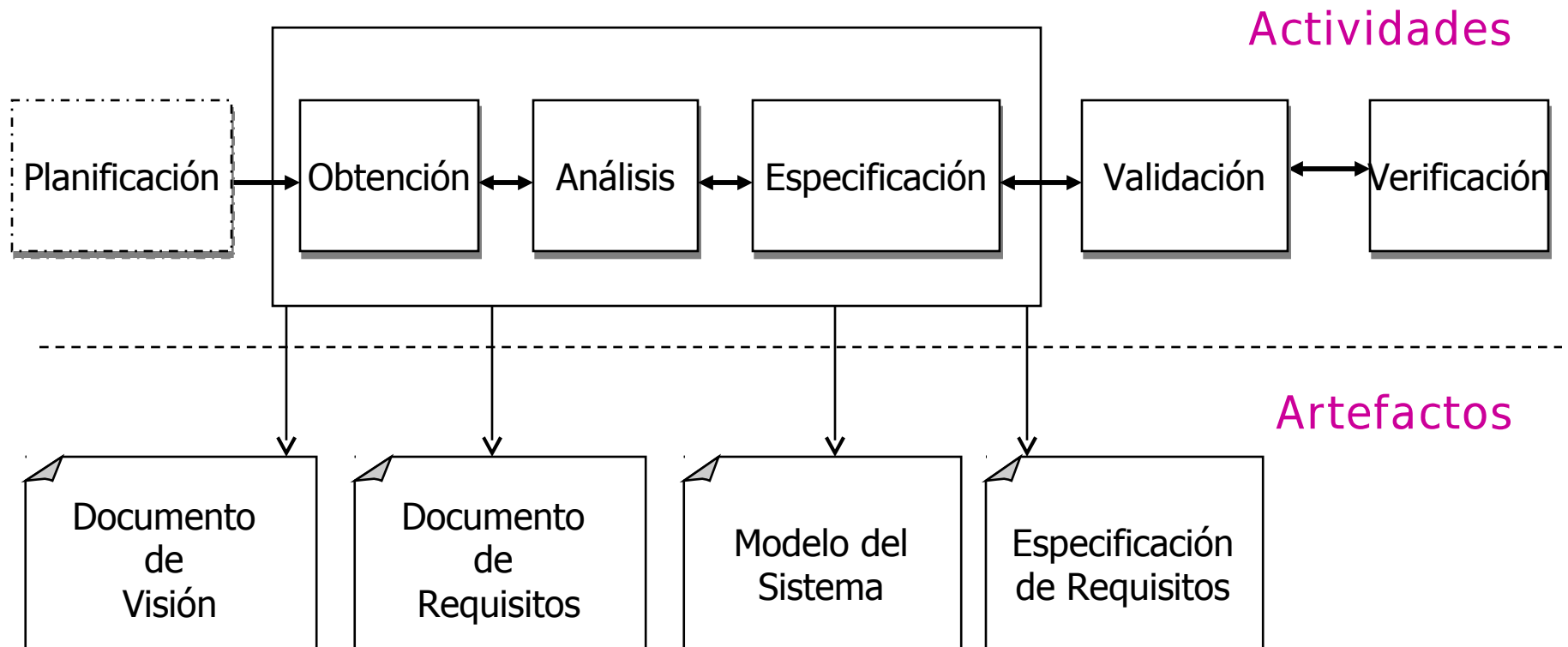
- Seguridad
 - control de acceso a las funciones/datos, aislamiento de los programas, respaldos-frecuencia, disponibilidad-, seguridad física
- Aseguramiento de la Calidad
 - Confiabilidad – tiempo medio entre fallas, robustez, tolerancia a fallas
 - Disponibilidad - tiempo para estar operativo luego de falla-mantenimiento estando activo- tiempo máximo de no disponibilidad
 - Mantenibilidad
 - Seguridad
 - Portabilidad

Ingeniería de Requisitos

Ingeniería de Requisitos



Proceso de Requisitos

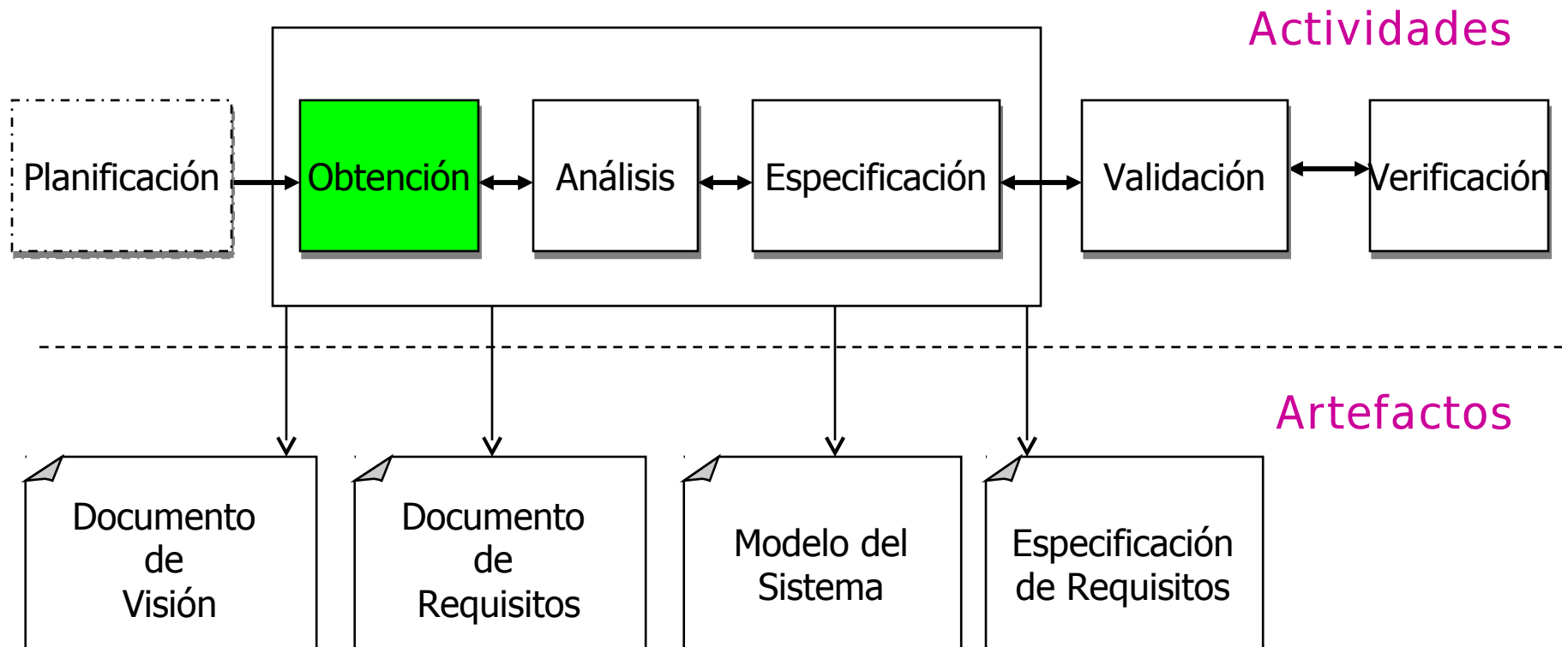


Participantes en el Proceso de Requisitos

- Cliente y Usuarios
 - Requisitos adecuados a sus necesidades
- Diseñadores
 - Comprenderlos para lograr diseño que los satisfaga
- Supervisores del Contrato, sugieren:
 - Hitos de Control, cronogramas
- Gerentes del Negocio, entienden:
 - Impacto en la Organización
- Verificadores
 - Comprenderlos para poder verificar si el sistema los satisface

Obtención de Requisitos

Proceso de Requisitos



Requisitos - Fuentes

- Necesidades del cliente, usuario, otros interesados
- Modelos del dominio
- Revisar la situación actual
- Organización actual y sistemas
- Versión actual del sistema
- Desarrolladores de versión anterior
- Documentos existentes (antecedentes)
- Sistemas análogos ya existentes (antecedentes)

Obtención & Análisis de Requisitos

- Se trabaja en conjunto con los usuarios y clientes
- Problemas comunes:
 - No saben lo que quieren del sistema, sólo en términos generales, no conocen el costo de sus peticiones
 - Los requisitos están en sus términos y con conocimiento implícito de su propio trabajo
 - Distintos usuarios tienen distintos requisitos, se deben encontrar todas las fuentes
 - Influyen factores políticos
 - La prioridad que se da a los requisitos varía con el tiempo
 - Aparecen nuevos requisitos

Brecha en la Comunicación (Scharer '90)

Según desarrolladores, los usuarios...

no saben lo que quieren

no pueden articular lo que quieren

muchas necesidades por motivos políticos
quieren todo ya

son incapaces de definir prioridades entre
sus necesidades

rehúsan asumir responsabilidades por el
sistema

incapaces de dar un enunciado utilizable de
sus necesidades

no están comprometidos con los proyectos
de desarrollo

no aceptan soluciones de compromiso

no pueden mantener el cronograma

Según usuarios, los desarrolladores...

no captan las necesidades operativas

ponen excesivo énfasis en aspectos meramente
técnicos

pretenden indicarnos cómo hacer nuestro trabajo

no son capaces de traducir necesidades claramente
establecidas en un sistema

siempre dicen que no

siempre están pasados del presupuesto

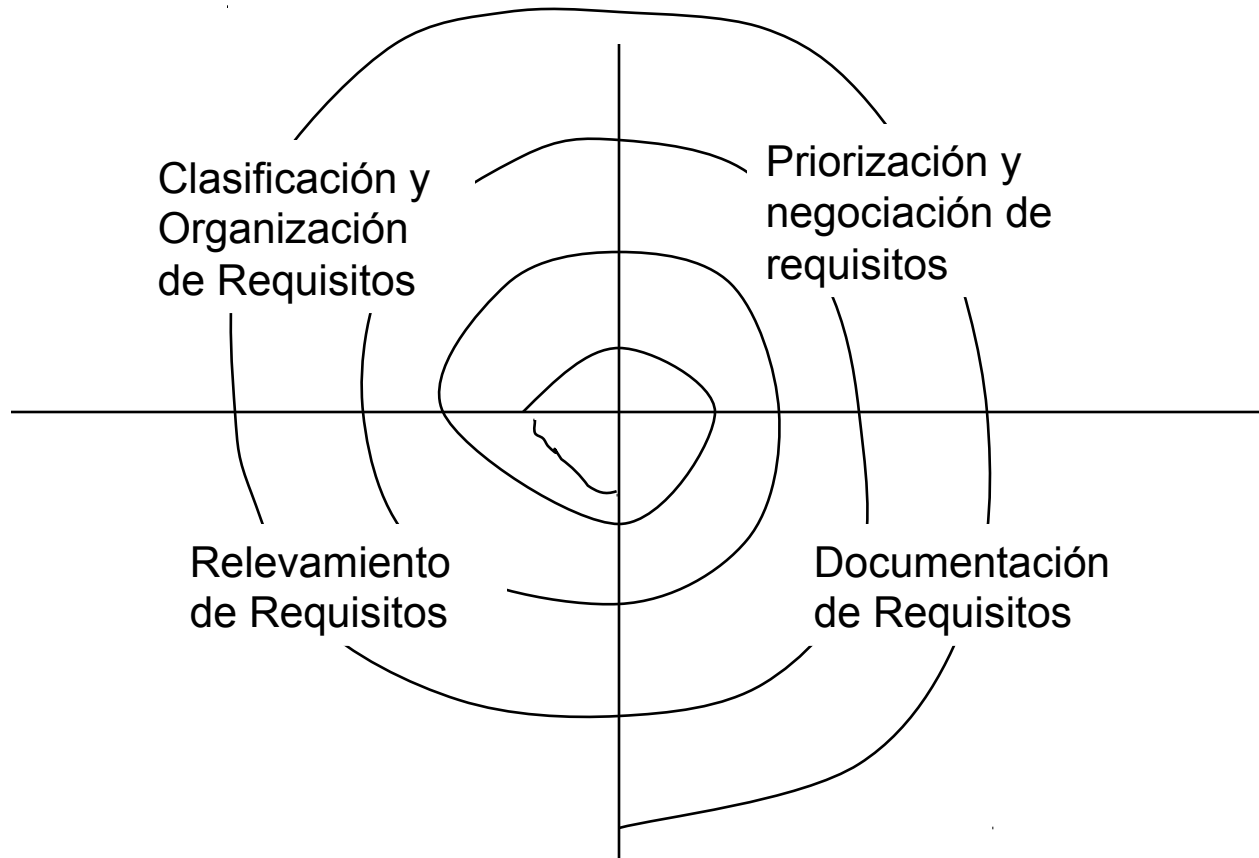
siempre están atrasados

nos exigen tiempo y esfuerzo aún a costa de las
obligaciones esenciales

establecen estándares no realistas para la definición
de requisitos

son incapaces de responder rápidamente a cambios
en las necesidades

Obtención & Análisis de Requisitos (Modelo Genérico)



Qué relevar

- Requerimientos del negocio:
 - Objetivos del negocio de alto nivel. Responden a la pregunta: ¿cómo será el mundo mejor para una determinada comunidad?
 - Categorías:
 - beneficios financieros
 - mejora de las operaciones de negocio
 - posicionamiento estratégico / competitivo
 - adoptar una nueva ley
 - nivelado / desarrollo tecnológico.
 - product vision statement
 - beneficios para los stakeholders
 - descripción de la funcionalidades en alto nivel
 - prioridades del proyecto
 - limitaciones del producto
- **Documento de visión.** Sirve para establecer una visión común con el cliente y para difusión del proyecto.

Qué relevar

- Reglas de negocio. - Existen independientemente del software. Aplican aunque se haga manualmente.
 - políticas de la organización
 - estándares
 - algoritmos
 - leyes y regulaciones
- Requisitos funcionales
- Requisitos no funcionales:
 - atributos de Q
 - interfaces externas
 - restricciones
- Criterios de éxito. Pe.:
 - que puedan desarrollar bien tareas significativas
 - manejo de errores
 - que satisfaga expectativas de calidad

Obtención de Requisitos – Técnicas

- Investigar antecedentes
- Entrevistas individuales/grupales
- Encuestas/Cuestionarios
- Tormenta de ideas
- Workshop
- Casos de Uso
- Observación/Participación
- Prototipado

Investigar antecedentes

- Estudio, muestreo, visitas,...
- Buena forma de comenzar un proyecto
- Interna: estructura de la organización, políticas y procedimientos, formularios e informes, documentación de sistemas
- Externa: publicaciones de la industria y comercio, Encuentros profesionales, visitas, literatura y presentaciones de vendedores

■ Ventajas

- Ahorra tiempo de otros
- Prepara para otros enfoques
- Puede llevarse a cabo fuera de la organización

■ Desventajas

- Perspectiva limitada
- Desactualizado
- Demasiado genérico

Entrevista Individual / Grupal

- Usar para:
 - Entender el problema de negocio
 - Entender el ambiente de operación
 - Evitar omisión de requisitos
 - Mejorar las relaciones con el cliente
- Ventajas
 - Orientado a las personas
 - Interactivo/flexible
 - Rico
- Desventajas
 - Costoso
 - Depende de las habilidades interpersonales
- Pasos para las Entrevistas
 - Seleccionar participantes
 - Aprender tanto como sea posible de antemano
 - Preparar la entrevista
 - Utilizar un patrón de estructura
 - Conducirla
 - Apertura, desarrollo, conclusión
 - Enviar un memo con resultado
 - Seguimiento

Entrevista – Patrón para conducirla

- Datos de las Personas: usuarios, interesados, disparador del proyecto
 - ¿Qué trabajo realizan? ¿Para quién?
 - ¿Qué interfiere con su trabajo?
 - ¿Qué cosas hacen su trabajo mas fácil o mas difícil?
- Datos: entradas y salidas clave, datos ya existentes
 - Listar las entradas y salidas
 - ¿Cuál es el problema? ¿Cómo se resuelve ahora? ¿Como le gustaría que se resolviera?
- Procesos: propósito, objetivos y metas
 - ¿Quién necesita la aplicación?
 - ¿Cuántos usuarios la van a usar y de qué tipo?
- Ubicaciones: lugares involucrados, contexto de los usuarios
 - Entorno de los usuarios, computadoras, plataformas
 - Aplicaciones relevantes existentes
 - Experiencia de los usuarios con este tipo de aplicación, expectativas de tiempo de entrenamiento

Entrevista – Patrón para conducirla (2)

- Evaluar confiabilidad, desempeño y soporte necesario:
 - ¿Cuáles son las expectativas respecto a la confiabilidad?
 - ¿Y respecto a la performance?
 - ¿Qué tipo de mantenimiento se espera?
 - ¿Qué nivel de control y seguridad?
 - ¿Qué requisitos de instalación existen?, ¿cómo se distribuye el software?, ¿debe ser empaquetado?
- Otros
 - ¿Existen requisitos legales, regulatorios u otros estándares que deban ser tenidos en cuenta?
- Factores críticos de éxito:
 - ¿Qué se considera una buena solución?
- Tener en cuenta:
 - Si el entrevistado comienza a hablar sobre los problemas existentes, no cortarlo con una próxima pregunta
 - Luego de la entrevista y mientras los datos aún están en mente, resumir los principales req. (aprox. 3) de este entrevistado

Encuesta / Cuestionario

- No substituye la entrevista
 - Antes de usar el enfoque:
 - Determinar la información que se precisa
 - Determinar el enfoque más adecuado:
 - Abierto, cerrado, combinado
 - Múltiple opción, valor en escala, orden relativo
 - Desarrollar cuestionario
 - Probarlo con perfil típico
 - Analizar resultado de las pruebas
 - Su principal uso es para validar asunciones y obtener datos estadísticos sobre preferencias
- | | |
|---|--|
| <ul style="list-style-type: none">■ Ventajas<ul style="list-style-type: none">▪ Economía de escala▪ Conveniente para quien contesta▪ Respuestas anónimas | <ul style="list-style-type: none">■ Desventajas<ul style="list-style-type: none">▪ Menos rico▪ Problemas por no-respuesta▪ Esfuerzo de desarrollo |
|---|--|

Observación / Participación

- Poco utilizado...
- Antes de usarlo
 - Determinar información necesaria
 - Comunicar a los involucrados
 - Considerar períodos normales y atípicos
 - Planificar las anotaciones
- Ventajas
 - Confiable
 - Muy rico
 - Desarrolla empatía
- Desventajas
 - Efecto Hawthorne
 - Cuidado con generalizar demasiado (sesgo particular/local)

Tormenta de Ideas (Brainstorming)

- Objetivo: Lograr consenso sobre los requisitos
- Ayuda a la participación de todos los involucrados
- Permite pensar en otras ideas
- Un secretario saca notas de todo lo discutido
- Reglas
 - No se permite criticar ni debatir
 - Dejar volar la imaginación
 - Generar tantas ideas como sea posible
 - Mutar y combinar ideas

Tormenta de Ideas – Fase de Generación

- Los principales stakeholders se juntan en un cuarto.
- Se explican las reglas.
- Se establece el objetivo:
 - ¿Qué características esperan en el producto?
 - ¿Qué servicios esperan que provea?Los objetivos permiten decidir cuando terminar.
- Se pide que cada participante escriba sus ideas, luego las ideas son leídas para que otros piensen en ideas relacionadas y de esa forma las ideas mutan y se combinan.

Tormenta de Ideas – Fase de Reducción

- El secretario lee cada idea y pregunta si es válida
 - Si hay cualquier desacuerdo, la idea se queda
- Agrupamiento de ideas
 - Nombrar los grupos
- Definición
 - Se escribe una breve descripción de lo que la idea significa para la persona que la escribió
 - Ayuda a tener un entendimiento común del requerimiento
 - Lleva unos minutos por idea
- Priorización (opcional)
 - Test de los \$100: Cada persona tiene dinero para comprar ideas, se ordena según ideas más compradas
 - Solo se puede hacer una vez
 - Se debe limitar la cantidad a gastar en 1 sola idea

Sesiones de Trabajo (Workshop)

- Ámbito para las tormentas de ideas
- Preparación
 - Venderlo a los posibles miembros de la reunión
 - Asegurarse que asisten los stakeholders correctos
 - Estructurar la invitación, el lugar, etc.
 - Enviar material previo a la reunión
 - Doc de requisitos
 - Entrevistas, defectos de los sistemas existentes, etc.
 - Asegurarse de enviar lo necesario, sin exagerar
 - Organizar la Agenda
 - Introducción
 - Tormenta de ideas – generación
 - Tormenta de ideas – reducción
 - Priorización
 - Resumen

Casos de Uso

- Técnica para entender y describir requisitos
- Los casos de uso son requisitos, describen requisitos funcionales
- Pone el acento en el uso del producto
- Describen como el sistema debe comportarse desde el punto de vista del usuario
- Casos de Uso como caja negra: Especifican qué es lo que el sistema debe hacer, sin especificar cómo debe hacerlo
- Se describen mediante documentos de texto
- Introducido por Ivar Jacobson (1992)

Casos de Uso

- Formato simple y estructurado donde los usuarios y desarrolladores pueden trabajar juntos
 - No son de gran ayuda para identificar aspectos no funcionales
 - Mientras se definen los casos de uso, puede ser un buen momento para definir pantallas u otros objetos con los que el usuario interactúa
 - Pueden ser usados en el diseño y en el testing del sistema
-
- Usarlo
 - Cuando el sistema está orientado a la funcionalidad, con varios tipos de usuarios
 - Cuando la implementación se va a hacer OO y con UML
 - No son la mejor elección:
 - Sistemas sin usuarios y con pocas interfaces
 - Sistemas dominados primariamente por requisitos no funcionales y restricciones de diseño

Prototipado

- Implementación parcial, permite a los desarrolladores y usuarios:
 - entender mejor los requisitos
 - cuáles son necesarios, deseables
 - acotar riesgos
- **Prototipo desechable**: El propósito es solo establecer que algo se puede hacer, luego se parte de cero en la construcción, quedando el conocimiento aprendido
- **Prototipo evolutivo**: Es implementado sobre la arquitectura del producto final, el sistema final se obtiene de evolucionar el prototipo
- Aspectos para los que es frecuente construir prototipos:
 - Apariencia y percepción de la interfaz de usuario
 - Arquitectura (riesgos tecnológicos, tiempos de respuesta)
 - Otros aspectos riesgosos

Mismos datos, pero...

Ingrese

año: _____

mes: _____

día: _____

Julio 1998

			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

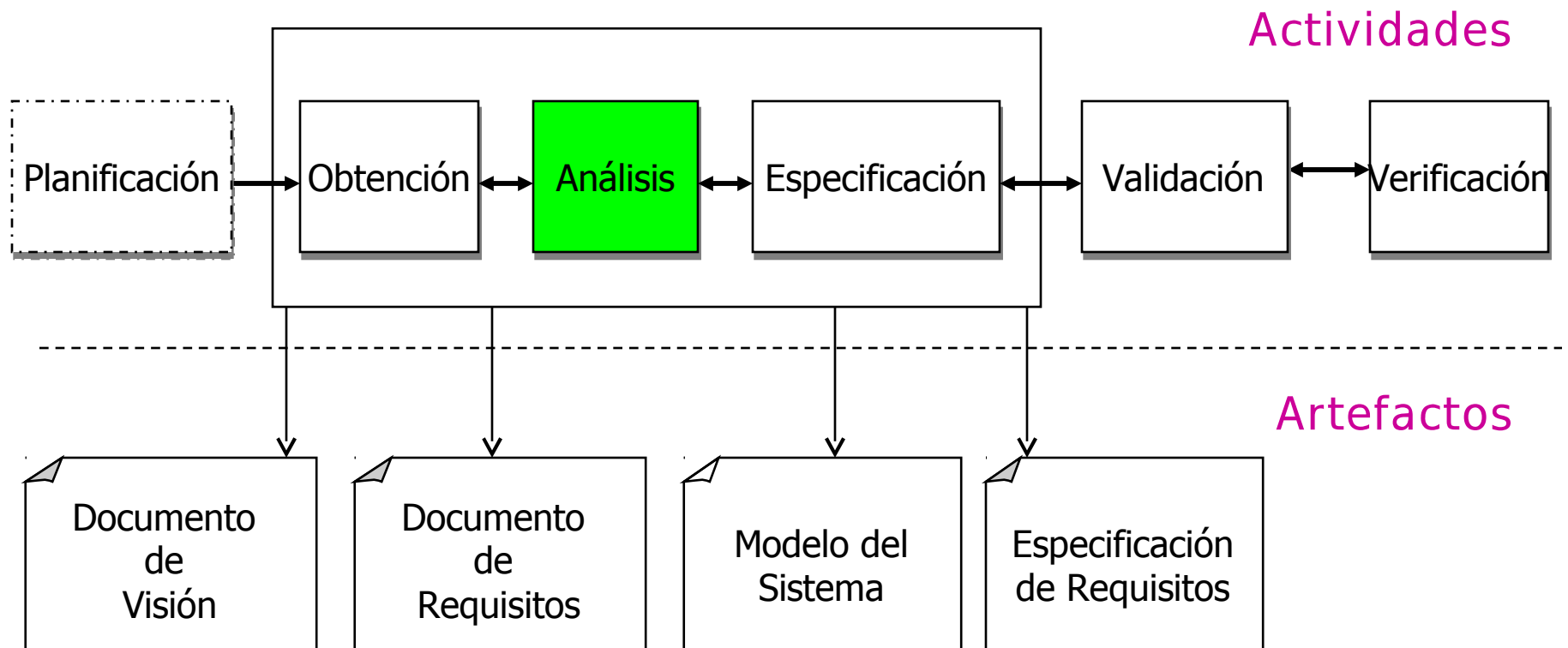
1998 2025

1 31

Ene Dic
Martes 16 Oct. 2002

Análisis de Requisitos

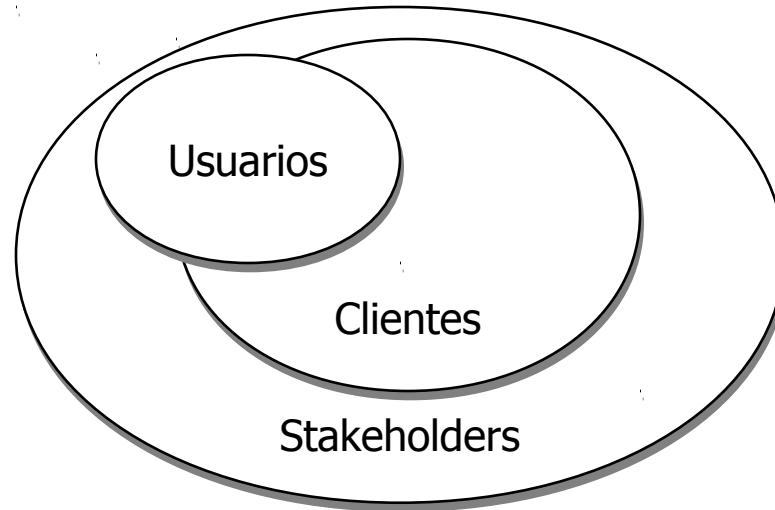
Proceso de Requisitos



Análisis de Requisitos

- Analizar stakeholders / clientes / usuarios
- Crear vistas
- Detallar
- Negociar prioridades
- Buscar reqs que faltan
- Evaluar factibilidad técnica - Prototipos
- Evaluar riesgos de requerimientos – En el Plan

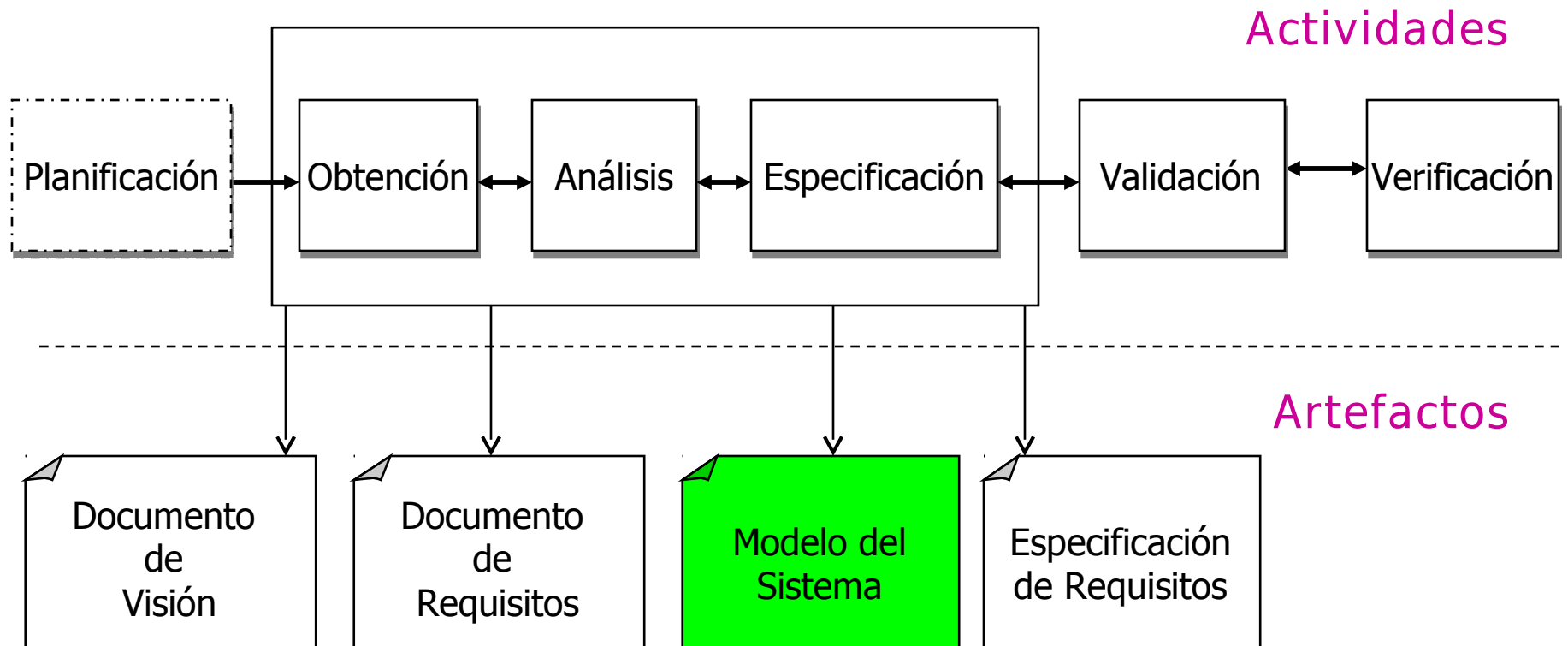
Stakeholders / Clientes / Usuarios



- Clientes:
 - Definir responsable de:
 - resolución de conflictos
 - validación
 - Planificar reuniones de revisión de avance con el responsable.
 - Definir proceso de resolución de conflictos pe. en alcance.

- Usuarios:
 - dividirlos en clases
 - definir representantes
 - definir prototipos
 - acordar responsabilidades y estrategias de colaboración con representantes

Proceso de Requisitos



Modelos o Vistas del Sistema

- Glosario
- Modelos gráficos:
 - Modelo conceptual
 - Diagramas de estado – para entidades complejas que pasen por distintos estados.
- Prototipos de interfaz gráfica. – Definir docs del prototipo (reqs, diseño, CP)
- Casos de Prueba
- Tablas de Decisión
- Redes de Petri
- Casos de Uso

Diagramas UML

- UML
 - Diagramas de Casos de Uso
 - Diagramas de Actividad
 - Diagramas de Máquinas de Estado
 - Diagrama de Clases
 - Modelo Conceptual

Tablas de Decisión

Tablas de Decisión

- Descripción dinámica
- Conjunto de **condiciones posibles** en un cierto instante
- **Estados** donde se verifica una combinación determinada de las condiciones
- **Acciones** a tomar

		Estados				
		1	2	3	4	5
Condiciones	Importe > 1000	F	F	V	V	V
	Buenos Antecedentes	V	F	V	V	F
	Ya operó antes	-	-	V	F	-
Acciones	Autorizar Crédito	X		X		
	Analizar antecedentes		X		X	X

- $\text{Nro estados} = 2^{\text{nro condiciones}} \Rightarrow$ tablas extensas