

2.1 Análisis y especificación de requisitos

Indice

- Introducción
- Ingeniería de requisitos
 - Obtención de requisitos
 - Definición de requisitos
 - Verificación de requisitos
 - Gestión de requisitos
- Tipos de requisitos



LA FASE DE ANÁLISIS

¿Qué vamos a hacer?

~~¿Cómo lo vamos a hacer?~~

Introducción

- Características del proyecto según el tamaño
 - Gastos a considerar
 - Ingresos a obtener
 - Fuentes de gasto e ingreso pensar a quién va dirigido, si pagarían por el producto
 - Redes de difusión que la gente se entere para que lo compren, cómo difundirlo
 - Plataformas para las que vamos a desarrollar el proyecto

Introducción

- Proyectos de tamaño pequeño
 - Inversión mínima
 - Muchos ingresos pequeños
 - Gastos de contratación y licencias
 - Ingresos directos por descarga
 - Caché en la store de cada plataforma, boca a boca...
 - App Store, Android Market, WiiWare, Playstation store, Xbox Live
- Concepto habitual de minijuego

Introducción

- Ejemplos de proyecto pequeño (Paper Toss, Crush the Castle...)



Introducción

- Proyectos de tamaño mediano
 - Inversión considerable (No afrontable como primer proyecto)
 - Muchos ingresos de importe medio
 - Añadir gastos de servidores, administración, difusión
 - Ingresos por descarga y/o mejoras de contenido por actualizar la versión
 - Añadir banners y redes sociales para difundirlo
 - Principalmente PC, también navegador, algunas videoconsolas...
- Concepto habitual de “primer proyecto exitoso”

Introducción

- Ejemplos de proyecto mediano (Minecraft, Cut the Rope...)



Introducción

- Proyectos de tamaño grande
 - Inversión astronómica
 - Muchos ingresos de importe grande
 - Añadir gastos de marketing, publicidad, atención al cliente...
 - Ingresos por venta física y/o descarga y/o cuota mensual
 - Añadir publicidad a gran escala
 - En general PC y videoconsolas de última generación
- Grandes videojuegos creados por grandes compañías

Introducción

- Ejemplo de proyecto de tamaño grande (World of Warcraft, GTA)



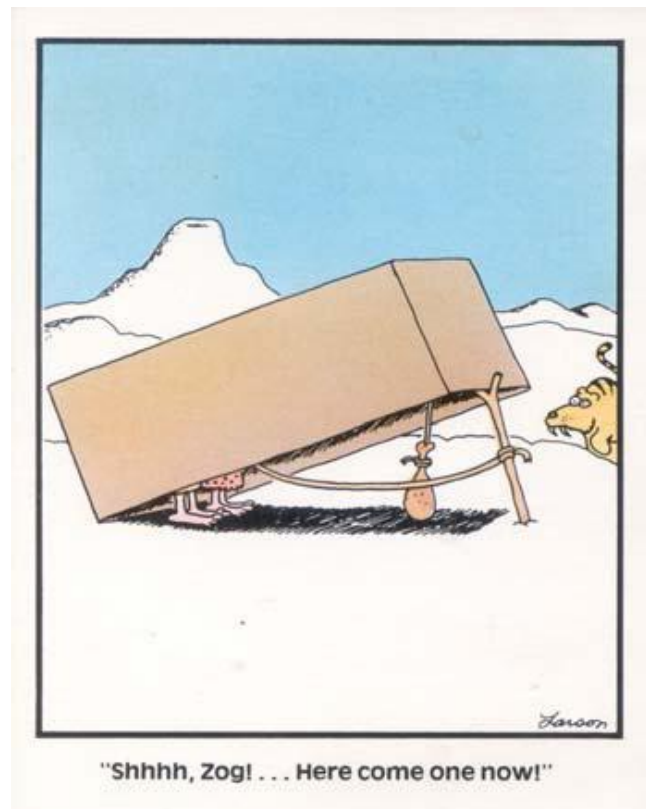
Introducción

- Para conseguir un buen producto final es necesario:
 - Entender las necesidades del usuario
 - Saber cómo afectará el producto a la forma de trabajar/interactuar con el usuario tiene q darle algo a cambio al usuario, beneficiarle en algún sentido
 - Adecuar el producto a los objetivos de la organización/negocio
 - Estudiar los beneficios/ventajas que se van a obtener frente a otras del mismo sector que la nuestra

Escribir el código para una funcionalidad (objetivo) no es la tarea más difícil. Lo complicado es entender cuál es la funcionalidad (objetivo) que el código debe satisfacer

Introducción

- **¿Qué es un requisito?** HAY QUE TENER LOS REQUISITOS CLAROS
 - Es algo que el producto debe hacer o una característica que debe tener



Introducción

- **¿Qué describe un requisito?**
 - Una utilidad para el usuario *una función*
 - “La página web debe tener una opción que permita elegir el idioma”
 - Una propiedad general del sistema
 - *“El sistema ha de garantizar que la información personal solamente sea accesible mediante autorización explícita”*
 - Una restricción general del sistema
 - “La imagen de la cámara de seguridad debe actualizarse cada 3 segundos”

Introducción

- **¿Qué describe un requisito?**
 - Cómo llevar a cabo cierto cálculo
 - “Calificación final = $\frac{1}{2}$ nota examen teoría + $\frac{1}{2}$ nota prácticas”
 - Una restricción sobre el desarrollo del sistema
 - “El sistema se tiene que implementar en C#”

Introducción

- ¿Por qué son necesarios los requisitos?

- **Coste**

Si los requisitos no son correctos, como el proyecto se basa en ellos, todo el proyecto está mal

- Requisitos pobres o mal formulados dan como resultado un producto deficiente o de poca calidad
 - El coste de reparar un producto deficiente o cancelar un proyecto es mucho mayor que el coste de una buena recogida de requisitos y análisis del sistema
 - El éxito o fallo de un sistema depende de cómo de bien se hayan capturado, entendido y usado los requisitos

PROCESO:

Análisis —> Diseño —> Implementación —> Pruebas —> Mantenimiento

Introducción

Si fallamos en el análisis, arrastramos el error en todas las demás fases (efecto bola de nieve).

- Los requisitos son una etapa clave en el ciclo de vida de desarrollo de software:
 - Su coste es alrededor de 10-15% del coste total del proyecto.
 - Un error en los requisitos puede ser hasta 100 veces más costoso que un error en el código.
 - Una equivocación en la etapa de requisitos se arrastra en las demás fases del ciclo de vida
 - Los procesos/sistemas complejos implican miles de requisitos
 - Necesidad de gestión y soporte automatizado

Introducción

- **Problemas con los requisitos**

Es complicado de entender si el cliente no te explica claramente lo que quiere que hagamos, y si no lo entendemos problema.

- Los requisitos no reflejan las necesidades reales del cliente
- Los requisitos pueden ser inconsistentes o incompletos
- Introducir un cambio en los requisitos una vez aceptados, puede ser muy costoso
- Existen problemas de comunicación
 - Malentendidos entre los clientes, los que desarrollan los requisitos y los ingenieros del software que desarrollan y mantienen el sistema

Introducción

- **Características de los requisitos:**
 - **Volátiles:** inconstantes
 - **Mutantes:** presentan alteraciones que se transmiten a otros requisitos
 - **Emergentes:** surgen al ir analizando en sistema en profundidad
 - **Colaterales:** surgen como efecto de la inclusión de otros requisitos
 - **Por compatibilidad:** se añaden para adaptar el sistema a su entorno, debido a que el entorno cambia.

Introducción

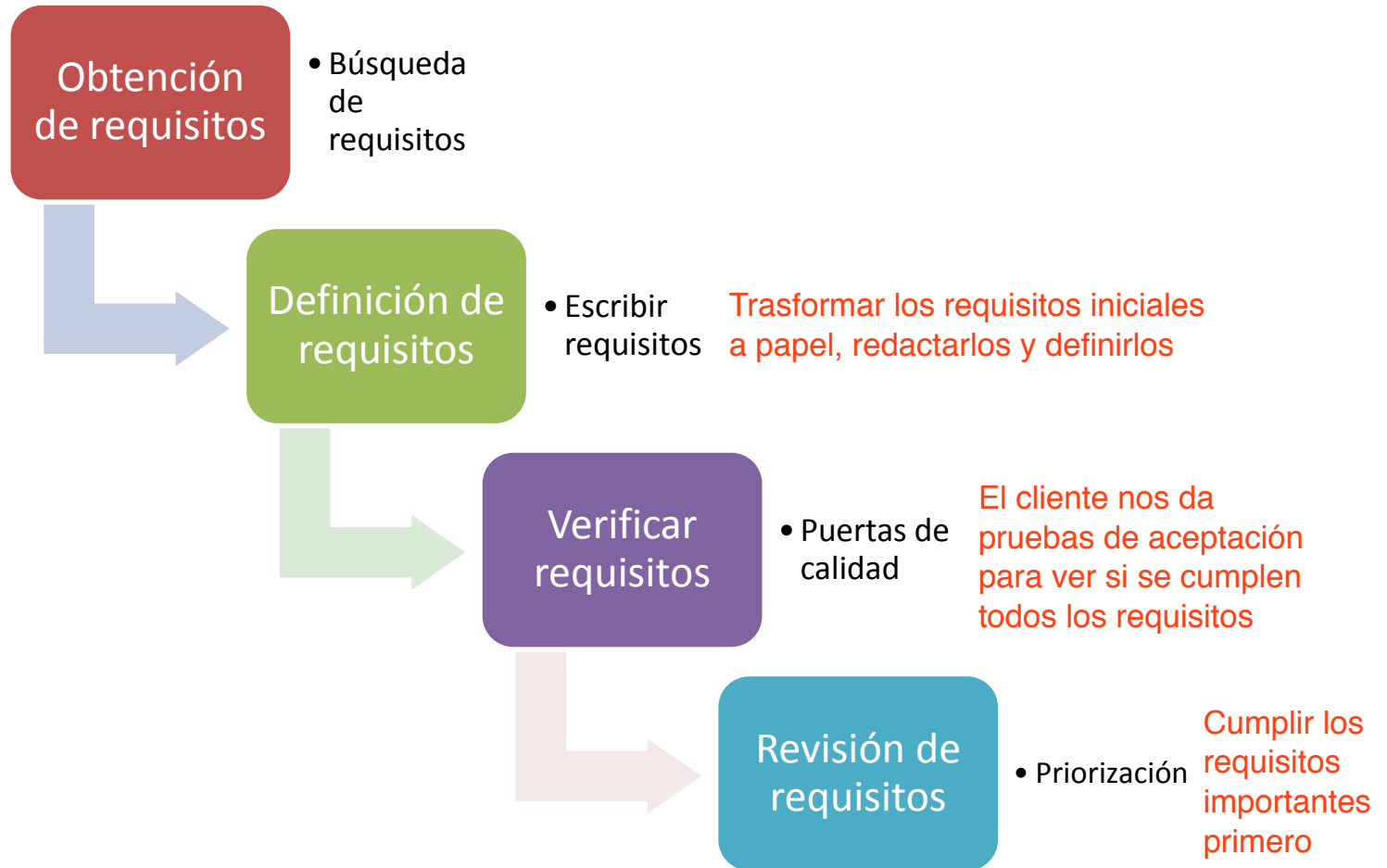
- **¿Por qué cambian los requisitos?**
 - Porque las **necesidades** de los usuarios **varían** en el transcurso del proyecto
 - Porque se producen **cambios tecnológicos**
 - Porque las **restricciones** del Sistema **cambian**
 - Porque el **entorno y reglas de negocio** evolucionan
 - Porque al analizar el problema, **no se hacen las preguntas correctas a las personas correctas**
 - Porque **cambia el problema** que se está resolviendo
 - Porque **cambia el mercado** en el cual se desenvuelve el negocio

Introducción

- Proceso de **Ingeniería de requisitos**
 - Fase de la Ingeniería del Software donde se definen las propiedades y la estructura del software
 - Procesos:
 - **Desarrollo de requisitos:** Entender los requisitos de negocio, identificar requisitos de usuario y trasladar los requisitos de usuario y de negocio a requisitos de sistema/software
 - **Gestión de requisitos:** Gestionar los cambios de los requisitos y mantener la consistencia entre los requisitos y otros productos de trabajo del proyecto Cualquier cambio puede afectar a otros requisitos, por lo tanto hay q controlar ese aspecto

Ingeniería de requisitos

Hablar con el cliente y buscar requisitos



Ingeniería de requisitos

- La comunicación es uno de los aspectos más destacables en la ingeniería de requisitos



- Esta característica hace de la ingeniería de requisitos una disciplina especialmente compleja al intervenir el factor humano

Obtención de requisitos

- Proceso de **identificar las necesidades** del negocio
 - Cumplir las necesidades de las distintas partes
- Los requisitos deben tener las siguientes características:
 - Deben ser **completos, consistentes** y deben 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
 - Han de ser **testeables**

Técnicas de recogida de requisitos

- Existen diferentes técnicas de recogida de requisitos. Las más comunes:
 - Entrevistas, reuniones, cuestionarios...
- Otras técnicas no tan comunes serían:
 - Utilización de casos de uso
 - Escenarios
 - Prototipos

Técnicas de recogida de requisitos

- **Entrevistas:** LO MAS FACIL Y ACCESIBLE
 - Se llevan a cabo con **clientes, usuarios finales, expertos** o cualquier persona interesada en el proyecto (*stakeholder*)
 - Generalmente, la información recopilada en una entrevista se debería verificar o cotejar con otras fuentes, reduciendo las posibles omisiones
 - Las entrevistas son muy flexibles y permiten la participación de los “*stakeholders*” aunque se requiere mucha experiencia y práctica para obtener buenos resultados
 - Puede ocurrir que los **datos recopilados** sean demasiado **vagos** o que el entrevistador **deje de lado algo** que no considere interesante

Técnicas de recogida de requisitos

- Hay dos tipos básicos de entrevistas: formales e informales
- Una **entrevista formal** se prepara con antelación, planificando la agenda y las preguntas a realizar. Algunas **preguntas** pueden ser **abiertas** (no se puede predecir la respuesta) y otras pueden ser **cerradas** (la respuesta se escoge de una serie de posibles respuestas)
- Las entrevistas formales necesitan ser complementadas por entrevistas informales

Técnicas de recogida de requisitos

- Las **entrevistas informales** son similares a una reunión informal donde no existen cuestiones predeterminadas u objetivos anticipados
- El propósito de una entrevista informal es animar al cliente a que se exprese con libertad y extraer requisitos que de otra forma podrían no tenerse en cuenta
- Ambos tipos de entrevistas deben proporcionar un punto de partida y un contexto para la discusión
 - Esto se puede conseguir escribiendo un pequeño documento o enviando un e-mail a los participantes explicando el objetivo

Técnicas de recogida de requisitos

- Se deben evitar algunos tipos de preguntas:
 - Preguntas donde se exprese directa o indirectamente la opinión del entrevistador:
 - ¿Tenemos que hacer las cosas según las estamos haciendo ahora?
 - Preguntas donde el entrevistador de por hecho alguna cosa:
 - No vas a hacer esto, ¿verdad?
 - Preguntas donde la respuesta venga impuesta en la pregunta:
 - Piensas como yo, ¿verdad?

Técnicas de recogida de requisitos

- Hay cierto tipo de preguntas que se deberían hacer:
 - Preguntas sobre detalles específicos del problema. **QUE, COMO, CUANDO, DONDE, POR QUE**
 - Preguntas sobre visión de futuro. Determinar si lo que se pretende hacer es posible o imposible
 - Preguntas sobre ideas alternativas. Tanto del cliente hacia el entrevistador, como sugerencias del entrevistador hacia el cliente
 - Preguntas sobre una solución mínimamente aceptable. Encontrar una solución lo más sencilla posible es el punto de partida para determinar el **alcance del sistema**

Técnicas de recogida de requisitos

- Preguntas sobre otras fuentes de información. Pueden proporcionar documentación importante y otro tipo de información hasta la fecha desconocida por el entrevistador
- Preguntas solicitando diagramas o modelos. Los gráficos o modelos realizados por un cliente para explicar los procesos de negocio son de mucha ayuda

Técnicas de recogida de requisitos

- Problemas de las Entrevistas:
 - Discrepancia de objetivos
 - Barreras de comunicación
 - Oír lo que queremos
 - Pasar por alto ideas contrarias
 - Prejuicios sobre el emisor
 - Diferente significado de las palabras
 - Comunicación no verbal
 - Emociones *intentar controlar las emociones, no mostrar si estás enfadado, contento, etc*

Técnicas de recogida de requisitos

- **Cuestionarios:**

- Los cuestionarios tienen la ventaja de proporcionar una gran cantidad de información rápidamente
- Se pueden realizar con gran cantidad de personas y no se tarda mucho tiempo en realizarlos
- Además, permiten el anonimato
- El **problema** de los cuestionarios es que **diseñarlos** suele ser bastante **difícil**, las **cuestiones** pueden ser **ambiguas** y la falta de contacto directo puede ocasionar la **pérdida de información**
- Otro **problema** es la **predisposición** de los encuestados, pueden responder muy cuidadosamente o tomarlo como una carga y responder rápido sin prestar mucha atención

Técnicas de recogida de requisitos

- En los cuestionarios se deben evitar preguntas abiertas. La mayoría de las preguntas deberían ser cerradas
- Hay 3 tipos de preguntas cerradas:
 - Preguntas de elección múltiple
 - Preguntas de selección rango. Muy bueno, bueno, neutral, malo, muy malo.
 - Preguntas de ordenación. Donde las respuestas se deben ordenar según números, porcentajes, etc

Técnicas de recogida de requisitos

- **Observación directa**

- Consiste en observar la forma de actuar de los usuarios con un sistema previo o en cierto tipo de tareas para encontrar la mejor forma de adecuar el nuevo sistema a sus necesidades
- La observación directa se considera esencial para entender la situación que ha generado la necesidad de un cambio
- De esta forma se puede entender lo que realmente se está haciendo sin tener que recurrir a manuales o documentación
- Esta técnica puede provocar recelo en los usuarios que se sienten observados

Técnicas de recogida de requisitos

- La observación directa puede tener 3 formas:
 - **Observación pasiva.** El analista observa las actividades sin interrumpir a los usuarios
 - **Observación activa.** El analista participa en las actividades y forma parte del equipo
 - **Observación explicativa.** Los usuarios explican lo que hacen al analista directamente
- Para obtener buenos resultados la observación directa debería realizarse durante un período de tiempo prolongado, en diferentes intervalos de tiempo y con diferente volumen de trabajo

Técnicas de recogida de requisitos

- **Estudio de documentación**

- El estudio de la documentación y de los sistemas software es una técnica muy útil para encontrar los requisitos de un sistema. Se suele estudiar ciertos aspectos del sistema
- Se utilizan los **formularios del negocio** (rellenados si es posible), descripciones de puestos de trabajo, **manuales** sobre normativas, planes de negocio, quejas de usuarios, etc
- **Captura de pantallas**, informes, documentación técnica, etc

Técnicas de recogida de requisitos

- **Prototipado**

- Es uno de los métodos más utilizados para la captura de requisitos
- Los prototipos se construyen para visualizar el sistema completo o parte de él. Los clientes pueden proporcionar su opinión al respecto
- Un prototipo es un **modelo de la solución** que proporciona una interfaz gráfica y simula el comportamiento del sistema para distintos eventos

Técnicas de recogida de requisitos

- Existen dos clases de prototipado:
 - **Prototipado de usar y tirar.** Se descarta cuando la obtención de los requisitos se completa. El objetivo es entender los requisitos del sistema se usa para q el cliente vea como va a quedar y luego se desecha
 - **Prototipado evolutivo.** El prototipo se mantiene tras la obtención de los requisitos y se utiliza para producir el producto final. Se centra en obtener rápidamente una primera versión del producto.

Técnicas de recogida de requisitos

- **Sesiones de grupo:**

- Las sesiones de grupo son **reuniones informales** de un grupo reducido de personas (entre 5 y 10) que representan uno o varios aspectos de los requisitos. Por ejemplo, se puede hacer una sesión de grupo para el diseño de la interfaz de usuario donde estén presentes una selección representativa de personas que van a utilizar el sistema
- Las sesiones de grupo son excelentes para proporcionar retroalimentación
- La atmósfera del grupo puede servir para extraer ideas y ayudar a detectar cuestiones relativas a los requisitos
- Sin embargo, esta técnica no suele profundizar demasiado en los detalles de los requisitos

Técnicas de recogida de requisitos

- **Tormenta de ideas**

- Mediante **Brainstorming** se pueden generar gran cantidad de ideas en un corto período de tiempo
- Funciona muy bien si el grupo de personas involucrado se encuentra en un ambiente relajado e informal
- Se enuncia el problema para el cual se necesitan ideas y la gente comienza a lanzar alternativas hasta agotar el tiempo
- Todas las ideas se tienen en cuenta y al finalizar la sesión se seleccionan aquellas que mejor se adapten o solucionen el problema
- Este tipo de reuniones es apropiada para afrontar un **reto nuevo**

Técnicas de recogida de requisitos

- Preguntas a realizar en una sesión de Brainstorming:
 - ¿Qué funcionalidades debería soportar el sistema?
 - ¿Cuáles son las entradas y las salidas del sistema?
 - ¿Qué clases se necesitan en el modelo de negocio?
 - ¿Qué preguntas se deberían realizar en los cuestionarios o entrevistas?
 - ¿Qué partes del sistema necesitan una mejora?
 - ¿Cuáles son los principales riesgos del proyecto?

Técnicas de recogida de requisitos

- **Casos de uso:**

- Describen las **interacciones entre el usuario y el sistema**, centrándose en qué hace el sistema para el usuario

Qué tiene que hacer el sistema para que el usuario lo pueda utilizar sin problema

- Los pasos a seguir son:

- Identificar actores
- Identificar escenarios
- Identificar casos de uso

Técnicas de recogida de requisitos

- **Casos de uso:**

- Un **actor** puede ser una agrupación uniforme de personas, dispositivos hardware u otros sistemas que interactúen con nuestro sistema
- Los actores son externos al sistema que vamos a desarrollar
- Un **caso de uso** es una descripción de un proceso de principio a fin relativamente amplia y suele abarcar muchos pasos o transacciones
- La regla general es: *“una función del sistema es un caso de uso si se indica explícitamente al sistema que uno quiere acceder a esa función”*.

Técnicas de recogida de requisitos

- **Escenarios:**

- Son una descripción paso a paso de la funcionalidad de un caso de uso del producto o del negocio
 - No tienen demasiado detalle. El objetivo es hacer entender el funcionamiento del caso de uso
- Se utiliza el lenguaje de las personas involucradas en el negocio

Técnicas de recogida de requisitos

- **Escenarios:**

- Cada caso de uso es una colección de **escenarios**
- Cada escenario es una **secuencia de pasos**

Caso de Uso: Ingresando Pedido	
Actor: Empleado de ventas	
Curso Normal	Alternativas
1) El cliente se comunica con la oficina de ventas, e informa su número de cliente	
2) El oficial de ventas ingresa el número de cliente en el sistema	
3) El sistema obtiene la información básica sobre el cliente	3.1 Si no está registrado, se le informa que debe registrarse en la oficina de clientes
4) El cliente informa el producto que quiere comprar, indicando la cantidad	
5) El sistema obtiene la información sobre el producto solicitado, y confirma su disponibilidad.	5.1 Si no hay disponibilidad del producto, el sistema informa la fecha de reposición
6) Se repite el paso 4) hasta que el cliente no informa más productos	
...	

Definición de requisitos

- Los requisitos especifican lo que hace el producto y no qué tecnología se usará para crearlo
- La especificación no debe ser ambigua
- Se deben utilizar siempre los sujetos en lugar de pronombres
- Evitar usar palabras que lleven a confusión: “debería” da entender que es opcional
- Usar una convención de nombres y definiciones común dentro de la organización

Definición de requisitos

- Ejemplo de requisito para el guiado de un avión:
 - **Guiado 18.1.2.** Cuando el piloto mueva la palanca hacia la derecha, el alerón derecho se elevará adecuadamente y el alerón izquierdo bajará adecuadamente. Igualmente, cuando el piloto mueva la palanca hacia la izquierda, el alerón derecho bajará adecuadamente y el alerón izquierdo se elevará adecuadamente.
- ¿Es correcta esta definición del requisito?

Definición de requisitos

- Ejemplo de requisito para el guiado de un avión:
 - **Guiado 18.1.2.** Cuando el piloto mueva la palanca hacia la derecha, el alerón derecho se elevará **adecuadamente** y el alerón izquierdo bajará **adecuadamente**. Igualmente, cuando el piloto mueva la palanca hacia la izquierda, el alerón derecho bajará **adecuadamente** y el alerón izquierdo se elevará **adecuadamente**.
- ¿Es correcta esta definición del requisito?

Definición de requisitos

- **Refinamiento I**

- **18.1.2 Guiado**

- **18.1.2.1 Dirección:** Cuando el piloto mueva la palanca hacia la derecha, el alerón derecho se elevará y el alerón izquierdo bajará. Igualmente, cuando el piloto mueva la palanca hacia la izquierda, el alerón derecho bajará y el alerón izquierdo se elevará

- **18.1.2.2 Graduación:** Movimientos incrementales de la palanca corresponderán a movimientos incrementales de los alerones

- ¿Es correcta esta nueva definición del requisito?

Definición de requisitos

- **Refinamiento II**
- Resulta que hoy, gracias al software se puede proteger la integridad estructural del avión:
 - **18.1.2.2 Graduación**: Movimientos incrementales de la palanca corresponderán a movimientos incrementales de los alerones, sin superar nunca la integridad estructural del avión
- ¿Podemos refinar todavía más los requisitos?

Definición de requisitos

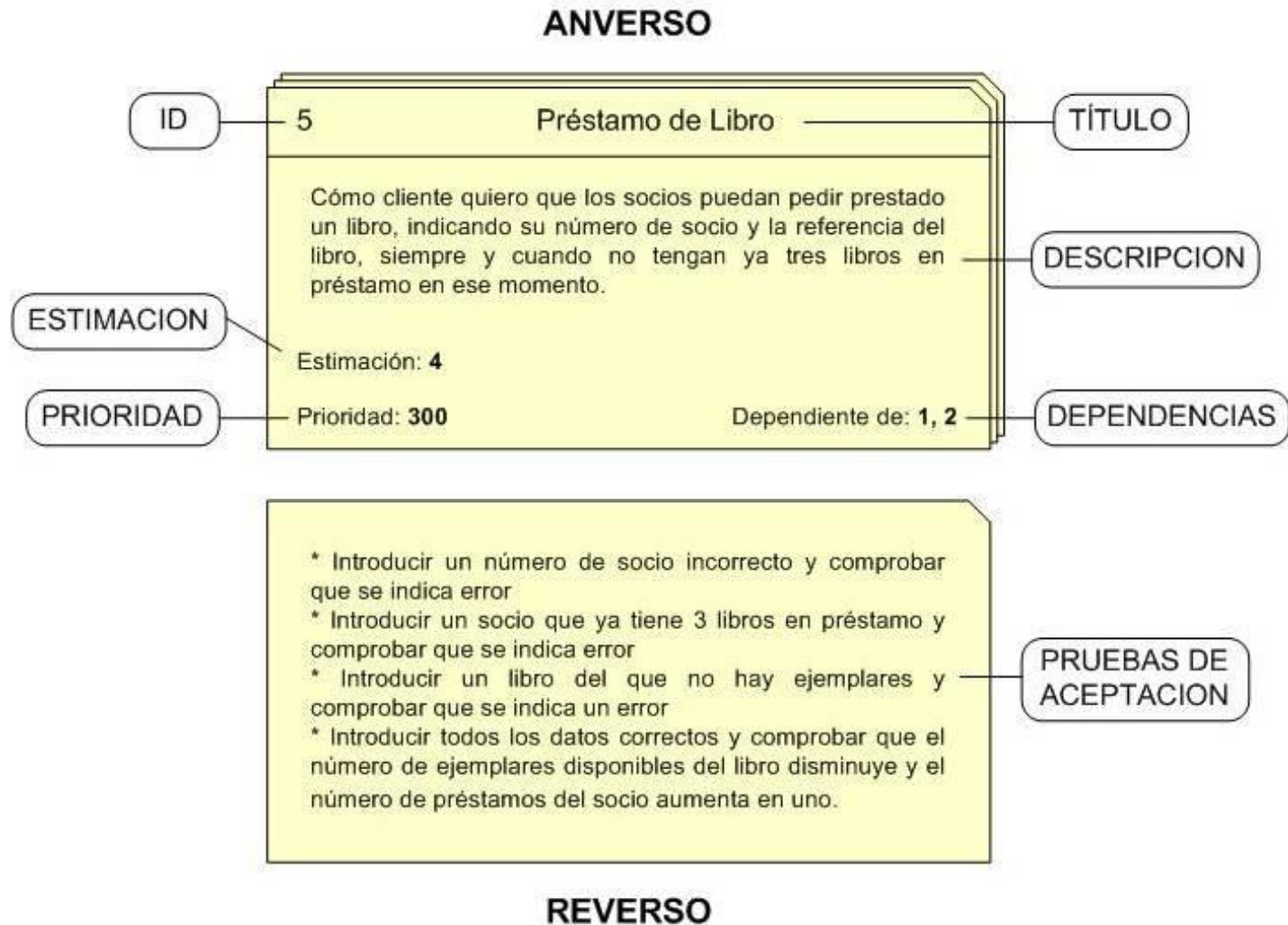
- **Refinamiento III**

- **18.1.2.2.1 Seguridad**: Si el piloto mueve bruscamente la palanca, el avión ejecutará el mayor giro posible que no exceda la tolerancia estructural del avión
- **18.1.2.2.1.1** El sistema monitorizará la velocidad del aire
- **18.1.2.2.1.2** El sistema monitorizará...
- ...
- **18.1.2.2.1.n** El sistema calculará el mayor ángulo posible compatible con las restricciones estructurales del avión

Verificación de requisitos

- El usuario final añade criterios de aceptación a los requisitos
- Todos los requisitos deben cumplir los criterios de aceptación
- La **especificación de requisitos** hace referencia a la colección de requisitos especificados y definidos previamente
 - Se debe revisar para comprobar que no falta ningún requisito

Verificación de requisitos



Gestión de requisitos

- La **gestión de requisitos** es el conjunto de actividades que ayudan al equipo de trabajo a identificar, controlar y seguir los requisitos y sus cambios en cualquier momento
- Los objetivos principales son:
 - Gestionar la recogida de requisitos
 - Obtener la aprobación de los participantes del proyecto
 - Gestionar los cambios (trazabilidad)
- La gestión de requisitos es un proceso que se desarrolla a lo largo de toda la vida del producto

Gestión de requisitos

- Los requisitos cambian a lo largo del ciclo de vida de desarrollo del producto
- Los **cambios** deben controlarse y documentarse
- Hay que **evaluar el impacto**
 - Estudiar cómo afectan los cambios al resto de requisitos
- **Aceptación** de los cambios
 - Si se acepta el cambio hay que adaptar los requisitos al cambio. En caso contrario, negociar con el cliente otra solución

Gestión de requisitos

- Trazabilidad
 - Los requisitos deben ser rastreables
- Es importante conocer:
 - Su origen (QUIEN los propuso)
 - Necesidad (POR QUE existen)
 - Relación con otros requisitos (DEPENDENCIAS)
 - Relación con otros elementos (DEPENDENCIAS)
- Suelen utilizarse matrices de trazabilidad

Gestión de requisitos

- Matriz de trazabilidad

Requisitos				Diseño Alto nivel	Diseño detallado	Código	Pruebas unitarias	Pruebas integración	Petición cambio
Req negocio	Req usuario	Req Sistema/SW	C.U						

- En la matriz se irán registrando los requisitos de negocio
- Por cada requisito de negocio se identificarán los requisitos de usuario correspondientes
- De cada requisito de usuario se identificarán cuáles son los requisitos del sistema

Gestión de requisitos

- Matriz de dependencias

Requisito	R1	R2	R3	R4
R1				
R2	Conflicto			
R3				
R4		Solapamiento	Solapamiento	

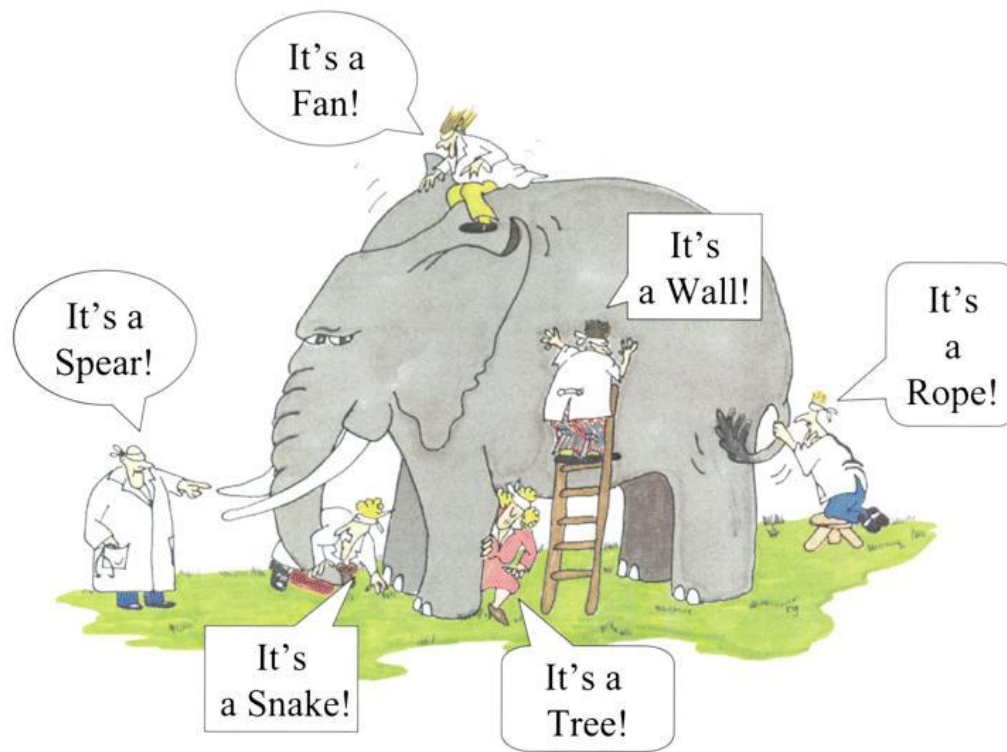
- La parte de arriba de la matriz incluyendo la diagonal no se utiliza
- El resto de celdas indican si existe o no solapamiento entre requisitos, si están en conflicto o son independientes (celdas vacías)

Tipos de requisitos

Tipos de requisitos	Descripción
Requisitos de negocio	Dan una descripción a alto nivel de lo que el sistema debe hacer. Representan: los objetivos, la base del negocio, estrategias, visión, alcance y el valor esperado del desarrollo del software
Requisitos de usuario	Son una descripción de las tareas que el sistema ha de ejecutar cuando el usuario opera con él. Describen la funcionalidad necesaria para satisfacer tareas específicas, necesidades operacionales y grupos de usuarios.
Requisitos del sistema/software	Definen las funcionalidades y características que debe tener el sistema para satisfacer tanto los requisitos de negocio como los de usuario. Van a servir como 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. Pueden ser restricciones del propio proyecto o del diseño del producto.

Especificación de requisitos

- Un documento de requisitos ayuda a todo el equipo a tener una visión general del producto.



Bibliografía

- Requirements Analysis and System Design. Leszek A. Maciaszek
- Ingeniería del Software. Ian Sommerville. Séptima edición
- Agile Software Requirements. Dean Leffingwell
- Ingeniería del Software. Un enfoque desde la guía SWEBOK. Salvador Sánchez et al.