

## Ingeniería de requisitos

En este tema pretendemos introducirnos en

- Qué es la ingeniería de requisitos y los problemas que intenta solventar.
- Qué es un requisito, cuales son las características deseables en él y una posible clasificación.
- Cómo se realiza la educación de requisitos de un proyecto.

### Ingeniería de requisitos

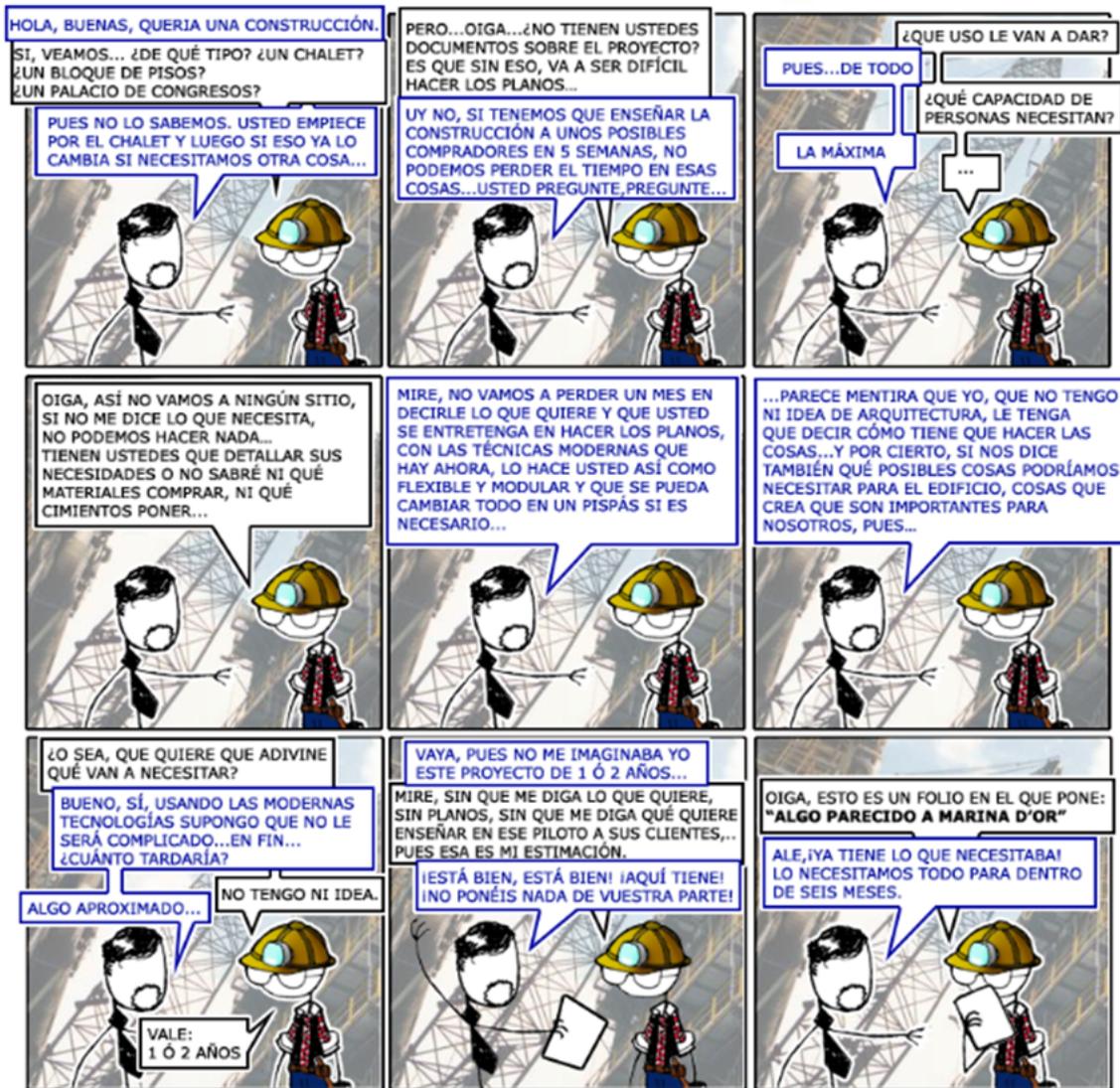
La **Ingeniería de Requisitos** se encarga de determinar las necesidades o condiciones a cumplir por un producto/entregable/servicio único.

Las actividades a realizar en la ingeniería de requisitos se resumen en:

- **Educación de los requisitos de usuario:** Se realiza con entrevistas con los *stakeholders* para saber cuáles son sus expectativas y se hace uso de la documentación de procesos de negocio para completarla.
- **Análisis de requisitos:** Determinar si los requisitos actuales cumplen las características de "un requisito bien especificado" y resolver posibles conflictos entre ellos.
- **Documentación de requisitos:** Estos pueden ser documentados en diversas formas, por lo general se define una lista resumida y se pueden incluir documentos en lenguaje natural, casos de uso, historias de usuario...
- **Verificar los requisitos:** consiste en comprobar que los requisitos han sido desarrollados acorde a lo especificado.
- **Validar los requisitos:** comprobar que los requisitos implementados cumplan para lo que inicialmente se construyó el producto.

Parte de la dificultad añadida a esta etapa reside en que los *stakeholders* no necesariamente saben transmitir que quieren, pueden tener puntos de vista contradictorios y no suelen tener una visión de conjunto.

A continuación vamos a describir el concepto de requisito, características deseables en ellos y tipos que existen.

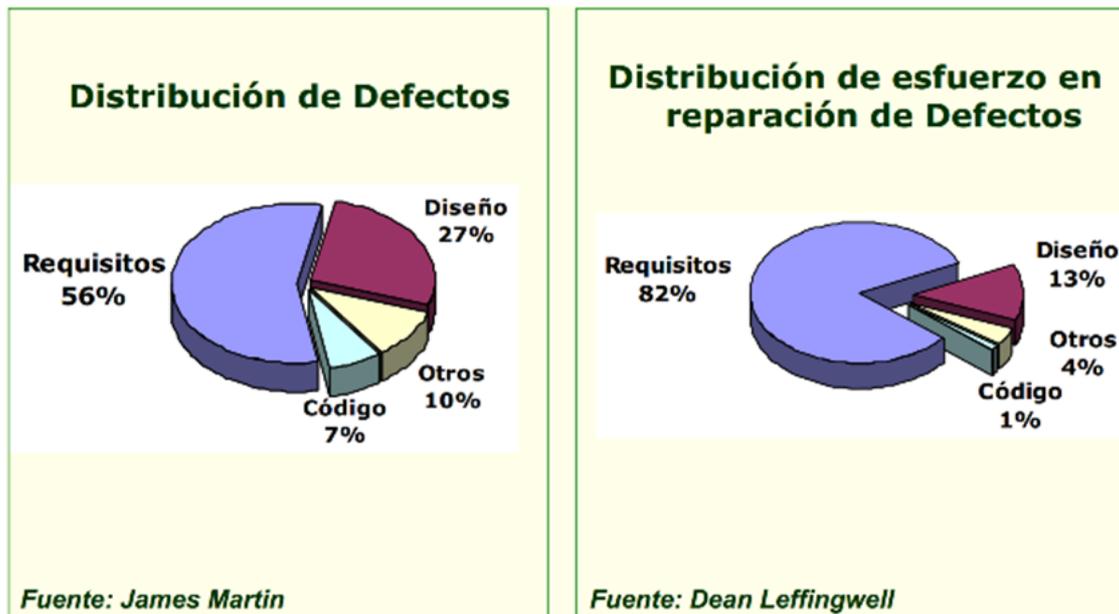


(BASADO EN HECHOS REALES LITERALES)



## ¿Por qué hablar de requisitos?

Los requisitos son los pilares sobre los que se sustentan las fases siguientes a la planificación. Errores en su especificación suelen provocar desecho de trabajo ya realizado, aumentando enormemente el tiempo o coste del proyecto (si queremos mantener la calidad).



## Requisitos

En IEEE Std 610.12-1990 se define un requisito como:

1. Una condición o capacidad requerida por un usuario para resolver un problema o alcanzar un objetivo.
2. Una condición o capacidad que debe ser poseída por un sistema o componente del sistema para satisfacer un contrato, estándar, especificación, u otro documento formalmente impuesto.
3. Una representación documentada de una condición o capacidad como las descritas anteriormente.

De manera más informal, podemos definir un **requisito** como una declaración que identifica una condición o capacidad de un producto/entregable/servicio único para que tenga valor y utilidad para un *stakeholder*.

## Características que debe tener un requisito bien especificado

Un requisito bien especificado es:

- **Completo:** Esta declarado íntegramente en su descripción y no remite a otras fuentes externas que lo explique con más detalle.
- **Consistente:** El requisito no se contradice con cualquier otro requisito y es compatible con toda la documentación externa.
- **Atómico:** No contiene conjunciones. Por ejemplo: "El sistema debe medir la temperatura, el pulso y el nivel de oxígeno en la sangre y transmitir la información vía BLE". Esto son claramente cuatro requisitos: "El sistema debe medir la temperatura ", "El sistema debe medir el pulso", "El sistema debe medir el nivel de oxígeno en sangre" y "El sistema debe transmitir toda la información capturada vía BLE"

- **Rastreadable:** El requisito satisface la totalidad o parte de la necesidades de negocio impuestas por los stakeholders.
- **Actual:** El requisito no se ha quedado obsoleto por el paso del tiempo.
- **No ambiguo:** El texto debe ser claro, preciso y tener una única interpretación posible. No se debe recurrir a jerga técnica o siglas, si estas no han sido previamente definidas. No debe expresar opiniones, sólo hechos objetivos.
- **Verificable:** Se debe poder verificar con absoluta certeza, si el requisito fue satisfecho o no. Esta verificación puede lograrse mediante inspección, análisis, demostración o pruebas.



## Tipos de requisitos

La siguiente clasificación de requisitos es una de las más extendidas ([BABOK Guide](#)):

- **Requisitos de negocio:** Especificaciones a alto nivel de los objetivos o necesidades de la empresa. Describen porque el proyecto se ha iniciado, que objetivos alcanzará y que medidas se usarán para medir el éxito del mismo. Describen las necesidades de la empresa en su conjunto, no se incluyen necesidades de *stakeholder*.
- **Requisitos de stakeholder:** Especificaciones de medio nivel de un *stakeholder* particular o un grupo de ellos. Describen las necesidades que tienen y cómo van a interactuar con el producto/entregable/servicio único. A menudo sirve como punto intermedio entre una especificación más exhaustiva y los requisitos de negocio.
- **Requisitos de solución:** Describen las características de un producto/entregable/servicio único que cumplen con los requisitos de negocio y de *stakeholder*. Se dividen en:
  - **Requisitos funcionales:** Los requisitos funcionales establecen el comportamiento del producto/entregable/servicio único, describen los servicios (funciones) que se esperan de él.
  - **Requisitos no funcionales:** Los requisitos no funcionales son restricciones que se imponen al producto/entregable/servicio único (rendimiento, seguridad, costo, portabilidad, escalabilidad, etc).
- **Requisitos de transición:** Requisitos que especifican la transición de la situación actual de la empresa a un estado deseado futuro, pero que una vez cumplidos no serán necesarios nunca más (contratación, cambios de roles, migración de datos de un sistema a otro, etc). Se diferencian de otros tipos de requisitos por su naturaleza temporal, debido a que no pueden desarrollarse hasta que ambos productos/entregables/servicios únicos, existente y nuevo coexisten.

## Ejemplos

### Funcionales

- El sistema medirá el pulso en la muñeca de la persona usando PPG
- El sistema mostrará el pulso en un display incorporado al sistema
- El sistema transmitirá la información de pulso vía Bluetooth Low Energy (BLE)
- El sistema avisará por vibración cuando el pulso supere las 120 bpm

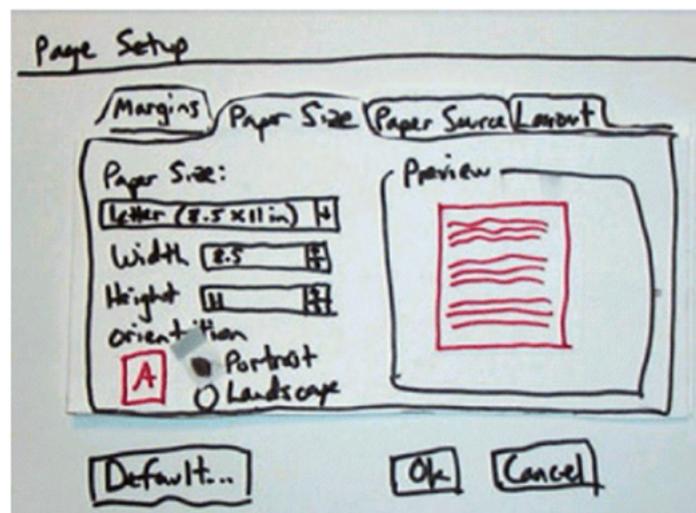
### No funcionales

- El sistema debe permitir pulso de hasta 200 bpm
- El sistema lee los datos del sensor cada 50 ms
- Cada sistema costará menos de 60 EUR
- El sistema estará listo para julio del presente año
- ...

## Edución de los requisitos de usuario

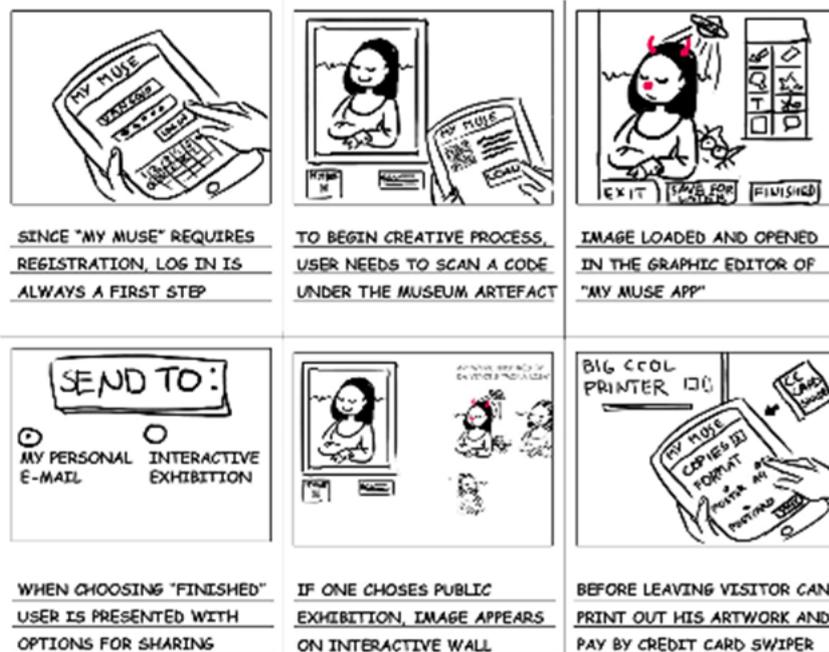
La educación de los requisitos se ha realizado tradicionalmente con entrevistas a los *stakeholder*, para saber cuáles son sus expectativas sobre el producto/entregable/servicio único. Otros métodos que ayudan a la educación de requisitos son:

- Prototipado: Esta técnica es una de las más comunes de apoyo a las entrevistas. Consiste en realizar un prototipo (interfaz, diseño, arquitectura, etc) de como va a ser el producto/entregable/servicio único y discutir sobre el mismo.



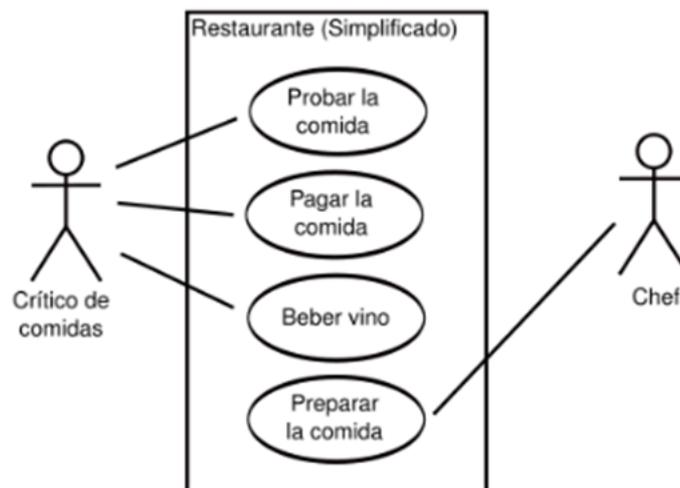
<http://www.snyderconsulting.net/>

- Tormenta de ideas (*Brainstorming*): "...La principal regla del método es aplazar el juicio, ya que en un principio toda idea es válida y ninguna debe ser rechazada... Cualquier persona del grupo puede aportar cualquier idea de cualquier índole, la cual crea conveniente para el caso tratado... Un análisis posterior explota estratégicamente la validez cualitativa de lo producido con esta técnica..." Wikipedia.
- Guiones gráficos (*Story-boards*): La construcción de un guión gráfico significa pensar en la funcionalidad propuesta como una historia separada en una serie de pasos o acciones discretas. La componente visual hace que sea fácil de entender para obtener feedback de los *stakeholder*.



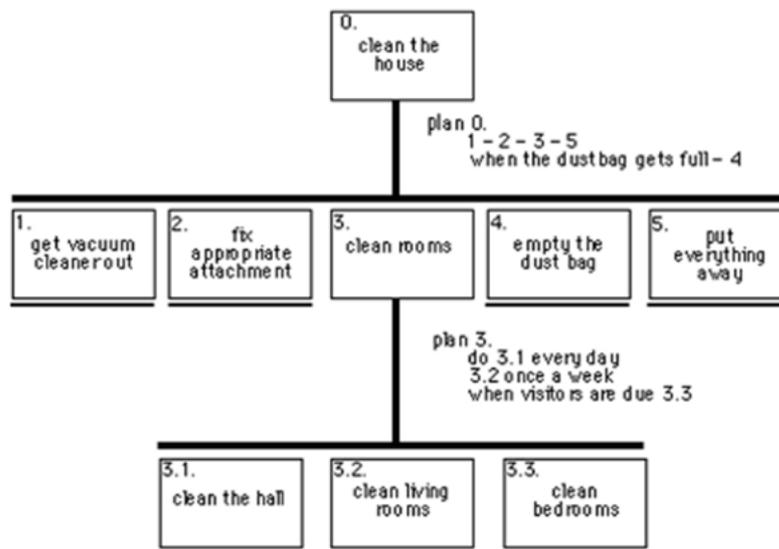
<https://darjaimke.wordpress.com>

- Casos de uso: Los casos de uso son básicamente guiones gráficos representados como diagramas que describen cómo funcionan los procesos discretos. Estos incluyen los actores (entidad externa al sistema que guarda una relación con éste y que le demanda una funcionalidad) y describen cómo funciona la solución desde su perspectiva. Los casos de uso pueden ser más fácil de expresar para los usuarios, aunque pueden necesitar ser refinamiento posterior para obtener requisitos más específicos.



UML\_diagrama\_caso\_de\_uso [GFDL or CC-BY-SA-3.0], via [Wikimedia Commons](#)

- Hierarchical task analysis (HTA), es decir, análisis jerárquico de tareas: Es un enfoque estructurado para describir como se realiza una tarea compleja. Entre las ventajas de este método está la facilidad para comparar enfoques que solucionan una misma tarea.



Modelado jerárquico de tareas de la limpieza con una aspiradora de una casa (Fuente: [HCI Book](#))