

# Clase 18/4 Quality Gateway

- QG intenta asegurarse de que cada requisito sea lo más perfecto posible antes de incluirlo en la especificación.
- Deben de pasar pruebas de aceptación; para asegurarse de que los devs sepan exáctamente lo que deben hacer.

## Alcance

- Los requisitos fuera de alcance **deben de evitarse**.
- **Modelo de contexto** es una **herramienta** para determinar si un requerimiento está dentro o fuera de alcance.

## Relevancia

- Evaluar si el requerimiento es relevante para:
  - Usuarios
  - Restricciones
  - Hechos relevantes
  - Supuestos

## Test de Completitud

- El requerimiento dentro de la plantilla definida debería tener **todos los atributos indicados**
  - **Si no los tiene, definir el por qué:**
    - El texto describe **perfectamente** la **razón fundamental**, por ende no se reitera.
    - No hay conflictos
    - Puede estar aún en investigación.

## Test de Criterios de Aceptación

- Todo requerimiento debe tener un criterio de aceptación capaz de hacerlo **medible**.

- **Ejemplo:**

- **Requerimiento:** El producto debe ser fácil de aprender
- **Criterio de Aceptación:** Un usuario debe poder aprender a procesar un reclamo dentro de 30 minutos de empezar a usar el producto por primera vez.

## **Estándares:**

Cuando posible, citarlos a la hora de utilizarlos como criterios de aceptación.

- Internos: Organización, área, marca
- Exterior: IEEE, Google (pq solo relativos de Ernesto trabajan ahí), etc
- Seguridad y UI basados en estándares:
  - UI: Requisitos de apariencia.
    - En el caso de que la apariencia sea similar a la de una organización, debería ser certificado por alguien que tenga poder dentro de la misma.

## **UX Test**

- Para escribir el criterio de aceptación para c/u de estos requerimientos, debes encontrar una escala de medición que cuantifique el objetivo del requisito.
  - “El producto debe ser intuitivo”, “intuitivo” se mide en base al usuario promedio que usará la app.
  - A veces, intuitivo implica “fácil de aprender”, por lo cual se mide en el tiempo que puede demorar en aprender a usar la app.

## **Performance Test**

Son **cuantificables**, por lo cual son **fáciles de medir**. Describimos en estos tests **velocidad, precisión, capacidad, disponibilidad, confiabilidad, estabilidad...**

## **Operativos**

Se evalúa en base a una cuantificación de un uso exitoso en el entorno requerido.

## **Mantenibilidad**

Se utilizan estándares.

## **Legal**

## Gold Plating

- Proviene de canillas de baño: a algunas personas les gusta tener canillas de oro. El agua no fluye mejor de la canilla si es de oro o de cromo. La diferencia es que la canilla de oro cuesta más y podría verse mejor.
- Refiere a características o requisitos innecesarios que contribuyen más al costo de un producto que a su funcionalidad o utilidad
  - Si nadie puede justificar realmente su inclusión, entonces puede considerarse gold plating
  - Mirar las calificaciones de satisfacción/desatisfacción del cliente asociadas con el requisito.
    - Una baja calificación de satisfacción puede hacerlo candidato de gold plating.

## Requirements Creep

- Se agregan nuevos requisitos a la especificación después de que se consideran completos. Este fenómeno puede tener un impacto serio en el presupuesto del proyecto, y algunos comentaristas lo sitúan en alrededor del 30% del costo total de cumplir con todos los requisitos.
- La QG tiene un papel en el control del “requirements creep”. Es importante asegurar de que cada requisito tenga calificaciones válidas de satisfacción/desatisfacción del cliente. Además, es importante que los requisitos sean relevantes tanto para el propósito del producto como dentro del alcance del trabajo.
- El requirements creep a menudo ocurre porque:
  - No se recopilaron adecuadamente desde el principio
  - El presupuesto original fue establecido de forma irreal
  - Cambian debido a cambios en el negocio o avances tecnológicos
- En cualquier caso, es importante que la QG haga al equipo de gestión del proyecto consciente del “requirements creep” y trate de determinar la causa subyacente.

