# QA

\*\* Es un atributo de calidad que define una característica o propiedad de calidad sistémica o de un componente, servicio o función.

Un atributo de calidad (QA) es una propiedad medible o comprobable de un sistema que se utiliza para indicar cuán bien el sistema satisface las necesidades de sus partes interesadas.

Requisitos de atributos de calidad. Estos requisitos son la calificación de los requisitos funcionales o del producto en general. La calificación de un requisito funcional es un elemento como la rapidez con la que se debe realizar la función, o cuán resistente debe ser a la entrada errónea. La calificación del producto global es un elemento como el momento de desplegar el producto o la limitación de los costos operativos.

Para expresar un requisito de atributo de calidad, usamos un escenario de atributo de calidad.

Las partes del escenario son las siguientes:

1. Fuente de estímulo. Esta es una entidad (un ser humano, un sistema informático, o cualquier otro actuador) que generó el estímulo.

2. Estímulo. El estímulo es una condición que requiere una respuesta cuando llega a un sistema.

3. Medio ambiente. El estímulo se produce bajo ciertas condiciones. El sistema puede estar en una condición de sobrecarga o en funcionamiento normal, o algún otro estado relevante. Para muchos sistemas, la operación "normal" puede referirse a uno de varios modos. Para este tipo de sistemas, el entorno debe especificar en qué modo está ejecutando el sistema.

4. Artefacto. Se estimula algún artefacto. Esto puede ser una colección de sistemas, todo el sistema, o alguna pieza o piezas de la misma.

5. Respuesta. La respuesta es la actividad emprendida como resultado de la llegada del estímulo.

6. Medida de respuesta. Cuando se produce la respuesta, debe ser medible de alguna manera para que el requisito pueda ser probado.

Desempeño

\* En condiciones normales el sistema permite al usuario acceder a la aplicación en menos de 2 segundos al entrar en su usuario y acceder correctamente

Ejemplo mal

El sistema debe ser seguro.

Ejemplos QA

Escenario de flexibilidad:

“El contratista desea hacer unas modificaciones en el sistema antes de su versión final”

1. Fuente: Contratista

2. Estímulo: Modificaciones en el proyecto

3. Entorno: Sistema casi finalizado (aun en tiempo de diseño)

4. Artefacto: Código

5. Respuesta: Cambios realizados sin excesiva dificultad y sin efectos secundarios

6. Medida de la Respuesta: Los cambios deben hacerse en el menor tiempo posible

7. Atributo de calidad afectado: AT01

Escenario de Modificabilidad

El sistema debe estar abierto a la hora de realizar cambios en los algoritmos y métodos de los que se compone, sin perjudicar la funcionalidad que este ya posee.

1. Fuente: agregación o actualización de un algoritmo

2. Estímulo: cambio de versión del algoritmo

3. Entorno: Explotación

4. Artefacto: Sistema

5. Respuesta: Actualización del algoritmo del sistema

6. Medida de la Respuesta: numero de paquetes procesados con el nuevo algoritmo

7. Atributo de calidad afectado: AT02

Escenario de Rendimiento

El sistema deberá facilitar un mínimo de información al mes, con la finalidad de que los usuarios no dejen de utilizar el portal por inanición.

1. Fuente: Actualización de la información

2. Estímulo: Actividad en la pagina web

3. Entorno: Explotación

4. Artefacto: Pagina web

5. Respuesta: Se incrementa el numero de visitas e interacción en la página

6. Medida de la Respuesta: numero de visitas a la pagina

7. Atributo de calidad afectado: AT03

Escenario de Testabilidad

El sistema deberá facilitar la realización de pruebas sobre si los cálculos de los datos estadísticos realizados por los algoritmos son correctos, o de si la información obtenida es veraz.

1. Fuente: Operaciones de pruebas

2. Estímulo: Operaciones de comunicación de datos

3. Entorno: Explotación

4. Artefacto: Pagina web y servidor

5. Respuesta: log con los cálculos realizados sobre los datos

6. Medida de la Respuesta: es posible trazar las operaciones relacionadas con los cálculos

7. Atributo de calidad afectado: AT04

Escenario de Seguridad

El sistema deberá impedir, en la medida de todo lo posible , los fallos de seguridad como intrusos que puedan acceder a la pagina web haciéndose pasar por usuarios con determinados permisos dentro de esta, o el acceso a los datos, pudiendo modificarlos, borrarlos o extraerlos.

1. Fuente: Acceso a la información privada o intrusión

2. Estímulo: Restringir acceso a datos

3. Entorno: Explotación

4. Artefacto: Pagina web, servidor y base de datos

5. Respuesta: Acceso mediante claves seguras

6. Medida de la Respuesta: numero de intrusiones y acceso a datos de forma ilícita:0

7. Atributo de calidad afectado: AT05

Escenario de Confiabilidad

La pagina web deberá proteger que los equipos de los usuarios queden infectados con virus al acceder a la pagina web, protegiéndolos de cross site scripting o que los datos descargables contengan algún tipo de software malicioso.

1. Fuente: Equipos de los usuarios

2. Estímulo: Acceso a la pagina o descarga de datos

3. Entorno: Explotación

4. Artefacto: Pagina web

5. Respuesta: Aumento de la seguridad de la pagina web

6. Medida de la Respuesta: numero de reportes sobre malware:0

7. Atributo de calidad afectado: AT06

Escenario Time to market

Los desarrolladores deben implementar el sistema en el menor tiempo posible

1. Fuente: Desarrolladores

2. Estímulo: Implementación rápida del sistema

3. Entorno: Sistema en desarrollo

4. Artefacto: Código

5. Respuesta: El sistema estará disponible en un margen de tiempo razonable

6. Medida de la Respuesta: Cumplimiento del plazo establecido para sacar el producto

7. Atributo de calidad afectado: AT07

# Design, tactics and Patterns

## Diseño

Es una colección de toma de decisiones

Flujo del diseño de arquitectura

Identificar el problema a resolver

Identificar las opciones de solución

Tomar decisiones de diseño

Administrar decisiones de diseño

## Tacticas:

Cada táctica es una opción de diseño para la arquitectura que promueve algo

Es una decisión de diseño que afecta un solo atributo de calidad

Cada táctica es un a hipótesis

### Disponibilidad

Detectar fallas, recuperación de fallas, prevenir fallas

### Rendimiento

Control de demanda de recursos, administración de recursos, recursos arbitrarios

### Modificabilidad

Reducir costos de modificar una responsabilidad, incrementar cohesión, reducir acoplamiento

### Testeabilidad

Administrar entradas y sálidas, monitoreo interno

### Usabilidad

Separar interfaz de usuario, apoyar la iniciativa del usuario

### Seguridad

Evitar la falla, detectarla, falla de contención

### Interoperabilidad

Reducir acoplamiento, incrementar cohesión, administrar interfaces.

## Patrones

Son paquetes de tácticas

# SW Architecture Patterns

Documento a parte.

# Choose your weapon wisely

Extreme Programming (XP)

Scrum

Rational Unified Process (RUP)

Microsoft’s Synch-and-Stabilize Process (MSS)

Team Software Process (TSP)

## Extreme Programming (XP)

Dos programadores en una sola maquina, pequeños despliegues en iteraciones cortas, Metaforas, diseño simple, refactorin, integración continua, estándar de código, 40 horas por semana

Roles : 7 roles que puede ser desempeñados por la misma persona excepto customer.

Programmer, customer,tester, tracker,coach, consultant, big boss

Artefacto :El artefacto primario es el código

10 personas

## Scrum

No produce software, organiza como el software es desarollado, sprint,

Roles: Scrum Master

Artefacto : master backlog

Equipos de 7 o menos

## Rational Unified Process (RUP)

Es un proceso de producto

Dos dimensiones : tiempo, aspectos del ciclo de vida, vertical 🡪 disciplinas procesos core. 30 roles

Tipicamente los artefactos no son documento, pero hay 60 artefactos

## Microsoft’s Synch-and-Stabilize Process (MSS)

Hay 3 fases principales : planeación, desarrollo y estabilización

Roles : Project manager, developer y tester

Artifacts : Tiene dos piezas principales de documentación : la visión del documento y la especificación del documento.

## Team Software Process (TSP)

Fue creado por Humphrey en el SEI CMU. SCRIPT , el proceso de desarrollo es iterativo.

Tiene dos sabores TSP : Un objetivo sobre un ambiente académico (TSPi) y otro hacia el uso industrial.

Roles : 5 roles team leader, development manager, planning manager, quality manager, support manager.

Artifact : documentación basada en formas de planes, script, documentos, reportes

# Diapos

## Niveles de diseño:

Impementación

Diseño detallado

Diseño de alto nivel

Diseño conceptual

Requerimientos

## Conceptos de lenguajes OO

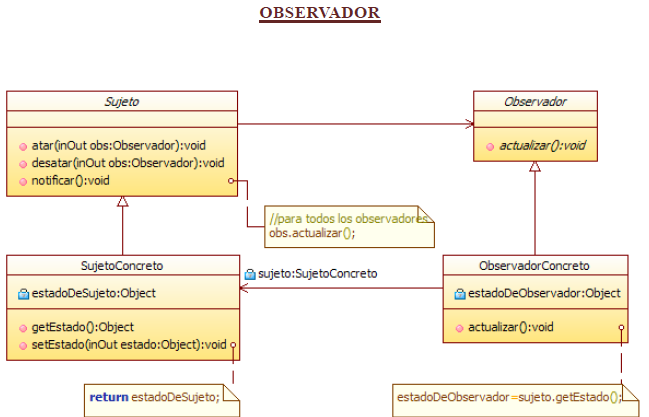
Cuatro conceptos clave : Localización dinámica, abstracción, subtipado, herencia

### Patrónes

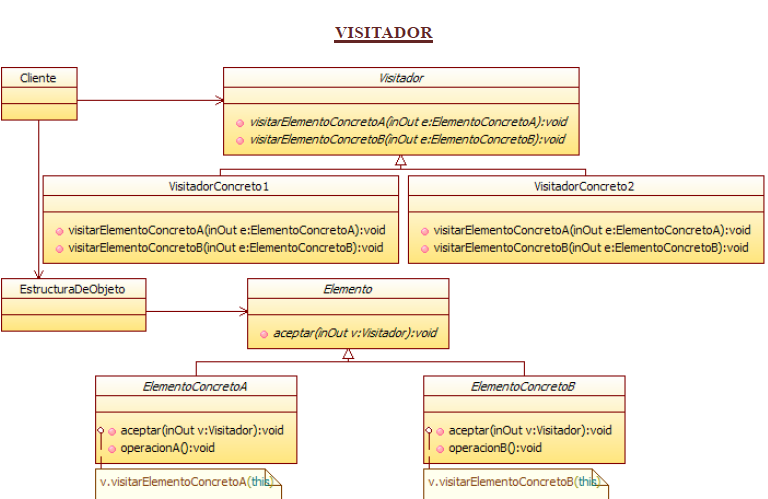
Singleton : una sola instancia

Facade : Interfaz de un grupo de objetos

Observador : Define una dependencia de uno a muchos objetos tal que cuando un objeto cambia de estado todos sus dependientes son notificados y actualizados automáticamente. (Para manejar dependencias)



Visitador : Representa una operación a ser ejecutada en elementos de una estructura de objetos. Permitiendo definir nuevas operaciones sin cambiar las clases de los elementos sobre los que se opera.



## Introducción y fundamentos de requerimientos

Tipos de requerimientos: Funcionales, atributos de calidad, requerimientos.

Objetivos de la Ing de requerimientos: **Establecer una visión de contexto**

Cuatro facetas del contexto:

Tema, Uso, Sistemas TI, Desarrollo.

**Actividades principales**

Documentación, educción, negociación

**Actividades transversales**

Validación, administración

Tipos de artefactos requerimientos:

Objetivos

Escenarios

Requerimientos orientados al problema / solución.

## Diseño de sistemas de SW de gran escala

Sistema: conjunto de componentes interconectados que trabajan en conjunto npara crear un todo mas complejo.

Arquitectura de un sistema : organiaci9ón fundamental de un sistema concebida a partir de sus componenetes.

## Patrones de diseño arquitectónico

Documentación de patrones. Capas, microservicios …

## Procesos de desarrollo y personalización

Presión – se toman atajos

Las fuentes de presión : jcronograma, recursos insuficientes, compromisos irreales

TSPi propone que se maneje con

Proceso de estrategia y planeación

Equipos auto-gestionados

TPSi considera equipo entre 4 y 8 personas y está hecho para escalar hasta 20.

Test driven developer : Escriba la prueba primero

Lean development : Es un método sistematico para eliminar el desperdicio en un proceso de manufactura.

# PSP

10,11,12, 13,14

## Cap 10

The design Framework

Obtener datos sobre los requerimientos del usuario

Analizar los datos del requerimiento

Realizar el diseño de alto nivel

Validar el diseño contra los requerimientos

Refinar y documentar el diseño

## Cap 11

En PSP el diseño es descrito por 4 formatos : Especificación operacional 🡪 describe el comportamiento operacional del programa por uno o más escenarios. Especificación funcional 🡪 Describe las funciones desempeñadas por el programa. Especificación lógica 🡪 use pseudocodigo para describir la lógica.

## Cap 12

Verificación del diseño

Estándar de diseño

Verificación tabla ejecución

Verificación tabla de trazas

Verificación máquina de estados

Verificación analítica

## Cap 13

Elementos de un proceso

Scripts

Form

Standards

Proceso de mejora

Ítems principales para cualquier proceso

* Objetivo
* Criterios de entrada
* Guía general
* Pasos del proceso
* Medida del proceso
* Criterios de calidad
* Condiciones de salida

Los 5 procesos : Percibir, actual, inicial, oficial, objetivo