

# Estandares de Programacion

**Docente: Ing. Ruben Escobar Ortegon**

**Correo: [ruben.escobar@mail.utec.edu.sv](mailto:ruben.escobar@mail.utec.edu.sv)**

# Unidad 3: Ingeniería de Pruebas del Software

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# Estrategias Finales: Calidad y Aceptación

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# OBJETIVOS

- Aseguramiento de calidad (QA): Garantizar calidad en todo el proceso.
- Aceptación de usuarios: Validar el software con el cliente final.
- Cerrar la Unidad 3 entendiendo cómo asegurar y validar la calidad del software.

**Es como inspeccionar un edificio durante la construcción (QA) y luego dejar que los dueños lo aprueben (aceptación)**

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# ¿Qué es el Aseguramiento de Calidad (QA)?

- El aseguramiento de calidad (QA, por sus siglas en inglés) es un proceso proactivo para prevenir defectos y garantizar que el software cumpla estándares de calidad durante todo su desarrollo.
- No es solo probar, sino construir calidad desde el inicio.

Diferencia con pruebas:

- Pruebas: Detectan errores (reactivo).
- QA: Previene errores (proactivo).

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# ¿Qué es el Aseguramiento de Calidad (QA)?

Ejemplo:

- Pruebas: Encontrar que un login falla.
- QA: Revisar requisitos y código para que el login nunca falle.

Según **ISO 9000**, QA mejora la satisfacción del cliente en un 20%.

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# ISO 9000

Son un conjunto de normas internacionales creadas por la Organización Internacional de Normalización (ISO) que establecen los principios para un Sistema de Gestión de la Calidad (SGC).

- Su objetivo principal es ayudar a las organizaciones a asegurar que sus productos y servicios cumplan con los requisitos del cliente y mejoren continuamente.
- Entre ellas, la más conocida es la ISO 9001, que contiene los requisitos específicos que una empresa debe cumplir para certificarse en calidad.

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# Enfoques del Aseguramiento de Calidad

## Enfoques clave:

- **Prevención:** Establecer estándares (ejemplo: guías de codificación).
- **Revisión:** Inspecciones de código, diseño y requisitos.
- **Procesos:** Usar metodologías como CMMI (Capability Maturity Model Integration) para estructurar el desarrollo.



# Enfoques del Aseguramiento de Calidad

## Actividades:

- Auditorías: Revisar si el equipo sigue las reglas.
- Capacitación: Enseñar buenas prácticas a desarrolladores.
- Métricas: Medir defectos por línea de código (ejemplo: <1 error por 1000 líneas).

## Ejemplo:

**En una calculadora, QA revisa que el diseño de `sum()` sea claro antes de codificarla.**

# QA en el Ciclo de Desarrollo

## Integración en fases:

- **Requisitos:** Verificar que sean claros y completos (ejemplo: "La calculadora suma números enteros").
- **Diseño:** Revisar diagramas antes de codificar.
- **Codificación:** Hacer revisiones de código entre pares (peer review).
- **Pruebas:** Asegurar que las pruebas cubran todo.

# QA en el Ciclo de Desarrollo

Ejemplo práctico:

Proyecto de tienda en línea: QA revisa que el requisito "carrito suma items" sea testable.

Inspecciona el código de calcularTotal antes de integrarlo.

Ventaja: Reduce errores en un 30% antes de las pruebas.

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# Ejemplo Detallado

Escenario: Desarrollo de un sistema de reservas de vuelos.

Aplicación de QA:

- Requisitos: Auditoría asegura que "reservar vuelo" incluya fechas válidas.
- Diseño: Revisión confirma que la base de datos soporta 1000 reservas diarias.
- Código: Peer review detecta un error en la lógica de precios.

# Ejemplo Detallado

- **Métricas:** Tasa de defectos baja de 2% a 0.5% tras revisiones.
- **Resultado:** El sistema es más robusto antes de las pruebas formales.

**QA es un escudo preventivo, no solo un corrector.**

# ¿Qué es la Aceptación de Usuarios?

La aceptación de usuarios (UAT, User Acceptance Testing) es la estrategia final donde los usuarios validan si el software cumple sus expectativas y requisitos.

Es la "prueba del cliente" antes del lanzamiento.

Propósito:

- **Confirmar usabilidad y funcionalidad desde la perspectiva real.**

# ¿Qué es la Aceptación de Usuarios?

Asegurar que el producto es útil, no solo técnico.  
Ejemplo:

- Calculadora: Usuarios prueban si es fácil sumar y si los resultados son claros.

**El 40% de los proyectos fracasan sin UAT adecuada.**

# Tipos de Pruebas de Aceptación

## Pruebas Alfa:

- Hechas en el entorno del desarrollador por usuarios internos (empleados). Buscan defectos finales antes de la entrega.

## Ejemplo:

- **Equipo prueba la calculadora en la oficina.**



# Tipos de Pruebas de Aceptación

## Pruebas Beta:

- Hechas por usuarios reales en un entorno real (beta testers). Recogen retroalimentación para mejoras.

## Ejemplo:

- **100 usuarios prueban la calculadora en sus dispositivos.**

## Aceptación contractual:

- Validación formal contra requisitos (ejemplo: cliente firma aprobación).

Ventaja: Conecta el software con el usuario final.

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# Cómo Ejecutar la Aceptación de Usuarios

## Pasos:

1. Definir criterios: Qué debe cumplir (ejemplo: "suma correcta en <1 seg").
2. Preparar entorno: Sistema listo y datos reales/simulados.
3. Involucrar usuarios: Entrenar o guiar a los testers.
4. Recoger resultados: Retroalimentación y defectos reportados.
5. Ajustar: Corregir antes del lanzamiento.

# Cómo Ejecutar la Aceptación de Usuarios

Ejemplo:

- UAT para tienda en línea:
  1. Criterio: "Pagar un ítem en 3 clics".
  2. Usuarios: 10 clientes prueban el carrito.
  3. Resultado: "Botón 'Pagar' confuso", se rediseña.

**Gestionar expectativas del usuario.**

# Ejemplo Práctico

Escenario: Sistema de reservas de hotel.

Aplicación de UAT:

- Criterios: Reservar en <2 min, interfaz clara.
- Entorno: App instalada en móviles de 20 usuarios beta.

Ejecución:

- Alfa: Empleados reservan 5 hoteles, encuentran error en fechas.
- Beta: Usuarios externos reportan "botón pequeño".

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# Ejemplo Práctico

- **Resultado: Correcciones antes del lanzamiento.**

**UAT asegura que el software no solo funcione, sino que guste.**

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# Relación entre QA y Aceptación

## Conexión:

- QA prepara el software (calidad interna); UAT lo valida (calidad externa).

## Ejemplo:

- QA revisa el código de **calcularTotal** en una tienda.
- UAT confirma que los usuarios pagan sin problemas.

# Relación entre QA y Aceptación

Ciclo:

- QA durante desarrollo reduce defectos.
- UAT al final asegura aceptación.

**QA es el cimiento; UAT es la aprobación final del edificio.**

**Transforma tu realidad.**

Viví la experiencia **UTEC**

# RESUMEN

Aseguramiento de calidad (QA):

- Previene defectos en todo el proceso.

Aceptación de usuarios (UAT):

- Valida el software con el cliente.



# PRACTICA

## Instrucciones

Diseñar una actividad de QA y un plan de UAT para un sistema.

Escenario: Probar una calculadora web (**DIVISIONES**).

# PRACTICA

## Aseguramiento de calidad:

- Propongan una actividad de QA (ejemplo: revisión, métrica).
- Expliquen cómo la aplicarían y qué esperan lograr.

# PRACTICA

**Aceptación de usuarios: Diseñen un plan de UAT:**

- **Criterio de aceptación (qué debe cumplir).**
- **Tipo (alfa o beta).Pasos para ejecutarlo.**
- **Describan un posible resultado (éxito o ajuste).**

# PRACTICA

## Ejemplo:

- QA: Revisión de código de `sum()`, esperar  $<1$  error.
- UAT: Criterio: "Sumar en  $<2$  seg".
- Tipo: Beta con 5 usuarios.
- Pasos: "Probar  $5+5$ , reportar".
- Resultado: "Números pequeños, ajustar fuente".

**Consejo: Dividan: uno QA, otro UAT, otro resultado.**

# MODELO

QA:

- Auditoría de requisitos, "suma debe ser clara", reducir ambigüedad.

UAT:

- Criterio: "Resta correcta".
- Tipo: Alfa con empleados.
- Pasos: "Probar 5-3, verificar 2".
- Resultado: "Éxito, listo para lanzar".

**Transforma tu realidad.**

Viví la experiencia **UTEC**