Mantención de Sistemas

Mantención de Sistemas

Calidad QA

¿Qué es la calidad?

¿Entregar productos de baja calidad y luego reparar problemas y deficiencias?

¿Lograr altos niveles de calidad de productos y servicios?

Software es un producto como cualquier otro producto fabricado (Autos, televisores, celulares etc…)

Para componer un software necesitamos una herramienta capaz de trabajar con programación, como Visual Studio Code.

Calidad en Sw (Software)

¿Cuándo empieza el proceso de calidad?

* ¿Después de terminado el proceso de programación? NO
* ¿En la mantención de los sistemas? NO
* ¿Durante la construcción del software? SI

Para poder diseñar un producto de calidad se debe primero saber el propósito, requisitos del software, y exigencias del cliente.

Pero, ¿qué es la calidad del Sw?

“El producto desarrollado debe cumplir con las especificaciones.” (Crosby, 1979)

pero…

¿Qué hacer si no están especificados los requerimientos de mantención?

¿Las especificaciones son ambiguas o no?

Los requerimientos pueden no ser completos o definir un producto de calidad.

Administrador de calidad (QA manager):

* Encargado de la calidad de los productos Sw en la organización.
* Debe asegurar que los niveles de calidad requeridos de un producto sean alcanzados.
* Definir políticas de calidad y asegurar que sean cumplidas.
* Buscar una “Cultura de calidad”.

Políticas de calidad

* Definición de procedimientos y estándares que deben ser seguridad durante el proceso de desarrollo.
* Asegurarse que son seguidas por los desarrolladores, en cada proyecto de software.
* Desarrollar estrategias para mejorar el proceso y, en consecuencia, la calidad del producto final.

Cultura de la Calidad

Todos son responsables por el desarrollo de los productos con altos niveles de calidad.

* Si puedo hacerlo bien a la primera, ¿Para qué hacerlo de nuevo?
* Reducir la cantidad de re-trabajo
* Mejora aspectos de costos y tiempo al mercado

Seguir aspectos intangibles, que no están en los estándares:

* Elegancia, buena lectura, etc.

Actividades de calidad

Aseguramiento de la calidad (I)

* Conjunto de acciones planificadas y sistemáticas: Sistema de Calidad.
* Deben ser demostradas: Generan Confianza
* Orientado a la prevención. Por tanto es:
  + Proactiva
  + Es parte de los proyectos
  + Centrada en el desarrollo y testing.

Aseguramiento de la calidad (II)

* El proceso debe estar documentado:
  + Manual de calidad
  + Instrucciones Técnicas
  + Auditorias
* Contempla aquellos aspectos que inciden en la calidad final del producto o servicio.

Actividades de Calidad

* Control de calidad
  + Mide la calidad real
  + Compara las normas y actúa por sobre las diferencias.
* Ejecuta métodos de ensayo para determinar características de calidad de materias primas, materiales, productos finales: Inspecciones (revisiones técnicas, formales, pruebas, etc…)

Aseguramiento vs Control:

|  | Aseguramiento de calidad | Control de calidad |
| --- | --- | --- |
| Concepto | “Asegura” la calidad de la producción. | “Controla” el producto después de la producción. |
| Costo | Espera cero rechazos. | Determina una tasa % de falla. |
| Proceso | Calidad incluye distribución y servicios post-venta. | Calidad termina con el trabajo. |
| Personas | La calidad depende de un equipo: “Cultura de calidad”. | Calidad depende de un departamento de control. |

Ambiente de trabajo

* Ambiente de Desarrollo
  + Entorno de desarrollo
  + Se escribe el proyecto
* Ambiente de Testing
  + Se prueba todo el proyecto
  + Entorno de pruebas integrales (Departamento alterno pero de misma empresa)
  + Reside, idealmente, en un servidor compartido por los equipos de desarrollo. Independiente del ambiente de desarrollo.
  + Conjunto de herramientas para pruebas (Gestión de casos de prueba, incidencias, control de versiones).
  + Responsabilidad del jefe de Testing/Jefe de Integración.
* Ambiente de Producción
  + Entorno en donde se ejecuta el software por parte de los usuarios finales.
  + Conjunto de herramientas que permiten la ejecución (clientes, base de datos, aplicaciones, etc).
  + Ambiente de producción determina/homologa los ambientes de testing y desarrollo.
  + A cargo de jefe de Proyecto/DBA/Oficial de Seguridad.

Unidad de mantención

* Roles y responsabilidades por ambientes:
* Desarrollo
  + Jefe de proyecto, administrador ambiente desarrollo, Desarrolladores.
* Testing
  + Jefe de integración (Implantación)
  + Jefe de Testing (Pruebas).
* Producción
  + Jefe de proyecto, DBA, Oficial de seguridad, Usuarios, Helpdesk.

Ambiente de desarrollo

* Conjunto de herramientas para el desarrollo:
  + Suite de Desarrollo (IDE):
    - Eclipse, Visual Studio, Netbeans.
  + Suite de datos (Base de datos):
    - Sql Server, Oracle, MySql, Postgres.
  + Control de versiones:
    - SVN, Git, Team Foundation.
  + Herramientas de monitoreo y desempeño (debugging).
  + Herramientas de Testing individual.
* Homologación de Ambientes.
  + Disponer de las mismas versiones y releases qué ambiente de producción.
  + Disponer de una rápida provisión de los ambientes de desarrollo, en forma personal (appliances, VM).
  + Homologar lo mejor posible:
    - Cluster de servidores en producción
    - Certificados de seguridad.
    - Usuarios y permisos.
  + Ejemplo: Si en desarrollo se usa una base de datos MySQL 5.7 y en producción hay MySQL 8.0, podrían surgir errores al desplegar el código. Homologar los ambientes significaría usar la misma versión en ambos lados o al menos validar la compatibilidad.
* Buenas prácticas
  + Uso de nombres de dominio diferentes en los distintos ambientes (igual para base de Datos)
    - ProyectoX\_desarrollo (ambiente Desarrollo)
    - ProyectoX\_testing (ambiente Testing
    - ProyectoX (ambiente Producción)
  + Pruebas unitarias exclusivas en ambiente de Desarrollo.
  + Pruebas de integración exclusiva en ambiente de Testing
* Administración.
  + Definición de ambiente de desarrollo.
    - De acuerdo a ambientes/plataforma de producción
    - Realizar con antelación y tiempo, para entregar calidad de servicio.
  + Determinar:
    - Requisitos de usuario y privilegios de acceso.
    - Parámetros de configuración del sw y sistemas.
    - Requisitos de estaciones de trabajo: sw, componentes, plugins.
  + Homologar cuentas de usuario en los diversos ambientes.
  + Definir mecanismos de actualización
    - Plan de actualización de aplicaciones
    - Plan de actualización de Datos de desarrollo.
      * Datos de prueba no criticos
      * Datos de pruebas blanqueados
  + Obtención de código fuente actualizado (Control de cambios y control de versiones)
  + Homologación a ambiente de producción en forma periódica.