

Crisis del SW

Causas principales de la crisis del software

Según el “Chaos Report”, entre las principales causas que encontramos detrás de la crisis del software destacamos los siguientes puntos:

- Reinicios: 94 de cada 100 proyectos se reiniciaban, alguno de ellos más de una vez.
- Exceso en costo: De media, entre todas las compañías tenían un sobrecosto del 189% del precio estimado inicial.
- Exceso en tiempo: Un tercio de los proyectos experimentaban entre un 200% y un 300% de retraso, con una media del 222% sobre el tiempo estimado.
- Requisitos incompletos: De media, solo el 61% de los requisitos y funcionalidades originales estaban disponibles finalmente para el uso del cliente.

En este mismo informe de The Standish Group se comentan algunos factores que propician el fracaso en la entrega final del proyecto. Destacan principalmente los requisitos incompletos, con un 13%; la falta de participación del usuario, con un 12%; la falta de recursos, con un 11%; y unas expectativas poco realistas, con un 10%.

Esto hizo que en 1995 las empresas y el gobierno norteamericano invirtieran 81 billones de dólares en proyectos software cancelados y 59 billones para completar otros proyectos, los cuales acabaron extendiéndose en el tiempo. Con todo, vemos que en esa época el 31% de proyectos eran cancelados, el 53% excedían costes y plazos y solo el 16% se completaron con éxito, unas estadísticas que reflejaban la baja calidad del proceso software seguido en esa época.

Ejemplos relevantes

Un ejemplo a destacar es la gestión de Hitachi para solucionar la crisis del software. Comentan varios puntos de las crisis del software y proponen una serie de soluciones a dichos problemas. Uno de estos puntos son los sobrecostes, para evitarlos utilizan un modelo de estimación llamado COCOMO con el principal objetivo de ajustar los presupuestos. Otro punto a destacar es la falta de programadores, sugieren subcontratar ciertas partes del software a empresas externas para evitar este problema. Hablan también de la calidad, abordándolo con un departamento que revise la calidad durante cada fase del proyecto. Y, entre otras cosas, proponen: registrar y analizar errores para poder gestionarlos, aplicar metodologías para monitorizar el avance del proyecto y usar prototipado para reducir posibles cambios futuros durante el desarrollo.

Otro ejemplo que podemos usar como comparación es la creación de software con la creación de puentes. En la ingeniería civil la especificación de requisitos, así como el presupuesto y el tiempo estimado es muy preciso, ya que al no haber una necesidad de cambiar dichos requisitos, no desencadena en excesos de tiempo y dinero, como sí sucede en la ingeniería software. Además, cuando un puente se cae se reporta y se estudian las causas, sin embargo, cuando un software falla se soluciona el problema y no se documenta, volviendo a cometer el mismo fallo en el futuro. Por último, se enfatiza en el uso de estándares y especificaciones para garantizar tanto seguridad como funcionalidad; y aunque en el desarrollo software disponemos de estándares, seguirlos a la perfección no nos asegura desarrollar un producto final con un alto nivel de calidad. Teniendo en cuenta lo anterior, podemos destacar la necesidad de un enfoque estructurado para el desarrollo de software.

Preguntas

- ¿Qué faltaba en los proyectos de SW en ese momento?

Existen gran cantidad de factores que influyeron en los proyectos SW de esa época. Destacan entre otros la falta de programadores, incorrecta estimación y organización del trabajo, mala o incompleta comprensión de los requisitos y falta de un sistema de calidad a lo largo del proceso.

- ¿Qué buenas prácticas de ingeniería soléis incluir en vuestros trabajos que ayudarían a todos estos problemas?

Una práctica que suelo llevar a cabo desde el inicio del proyecto es fragmentar el problema en partes más pequeñas y hacerlas lo mejor posible desde el principio para evitar errores o cambios futuros que llevarían más tiempo solucionar en fases avanzadas.

Otra buena práctica sería intentar que todo el código creado pueda ser reutilizable en otros proyectos futuros con pequeñas y/o simples variaciones ahorrando tiempo de desarrollo.

Por último, lo que hago al inicio de todo es un esquema mental de lo que se pide para comprender bien los requisitos, si hay algo ambiguo intentar darle forma y posteriormente documentarlo todo para que no se olvide.

- ¿Qué desafíos de los proyectos modernos aún recuerdan a los problemas históricos?

Algunos desafíos que nos recuerdan a problemas anteriores del software son la mala planificación, las pocas pruebas, la mala escalabilidad, la incorrecta comprensión de necesidades del cliente, el seguimiento equivocado del proyecto y el uso de nuevas tecnologías sin experiencia o poco probadas por el equipo.

Todos estos problemas se pueden reducir con un buen sistema de calidad durante todo el proceso y con una buena documentación que reporte los errores para no cometerlos en el futuro.

OGCIO

Principales medidas del sistema de calidad OGCIO

Una de las medidas más destacables del sistema de calidad de OGCIO es el control de calidad a lo largo de todo el proyecto, en cada una de sus fases, siguiendo unos estándares bien definidos como puede ser la ISO 9001 y otra serie de buenas prácticas. Además, la calidad es responsabilidad de todo el equipo y tienen una jerarquía organizacional encargada de diferentes tipos de tareas como es el QualCom, el jefe de proyecto o el equipo profesional de desarrollo.

Algo que podemos destacar es la obligación de participación del cliente para proveer la información necesaria a tiempo, evitando futuros cambios. Sus responsabilidades están definidas en el ISPM.

Dividen el proyecto en diferentes fases y se encargan de llevar un control de calidad en cada una de ellas para llevar un control de la misma a lo largo de todo el proyecto. Además, cuentan con procedimientos muy bien definidos para el aseguramiento de la calidad así como revisiones de personas externas al proyecto para una mayor precisión. Por último, cabe destacar la documentación organizada y bien definida para facilitar la aplicación de este sistema de calidad.

Cómo estas medidas abordan los problemas del Chaos Report

Para evitar los problemas del Chaos Report, OGCIO usa su QMS para acotar todos los posibles factores que puedan desviar el proyecto. Comienzan con una buena planificación y hacen un estudio de viabilidad del proyecto para evitar malas estimaciones, buscando establecer los requerimientos de calidad tanto del producto como de las fases del proyecto. Tras ello, durante el proyecto hacen revisiones periódicas para evitar o corregir desviaciones del plan de proyecto original, en el que se incluye el plan de calidad.

De esta forma, se puede conseguir reducir los problemas clásicos del software, como pueden ser los sobrecostos o retrasos, así como un producto final de baja calidad.

Ejemplo relevantes

Uno de los ejemplos que podemos tener en cuenta es la plantilla del plan de calidad, la cual debe ser creada por el jefe de proyecto al inicio del mismo. Vemos que se deben rellenar diferentes campos como son el nombre del proyecto o el nombre del jefe del proyecto, así como la fase del proyecto a la que referencia el documento. Relacionado con la calidad vemos una serie de campos a destacar, como son las responsabilidades y roles de cada miembro del proyecto, las responsabilidades de cada grupo de trabajo y la evaluación de riesgos o los objetivos de calidad.

Además, cabe destacar la importancia de los apartados de criterio sobre las entradas y salidas de cada fase, con una columna para la fase del proyecto en la que estamos, las entradas esperadas, las salidas necesarias para que una fase sea considerada completada satisfactoriamente y los números de referencia de las guías o estándares aplicados a dicha fase. Por último, tenemos una hoja para las revisiones formales de cada uno de los grupos que revisarán de forma independiente este documento.

Opinión sobre qué herramientas de OGCIO son más útiles y por qué. ¿Consideráis incluir alguno en vuestro TFG?

Bajo mi punto de vista, definir los objetivos de calidad de cada fase del proyecto así como revisiones periódicas para revisar que se cumplan dichos objetivos me parece una buena práctica, muy útil para obtener un buen resultado y que sin duda pienso incluir en mi TFG. Entre otras cosas, también destacó la importancia de realizar todo el trabajo desde el inicio siguiendo unos objetivos de calidad, evitando así futuros cambios que puedan consumir gran cantidad de tiempo. Por último considero que la documentación realizada por OGCIO es una muy buena base para realizar mi propia documentación en mi TFG.

Preguntas

- ¿Cuál es el alcance del QMS? ¿A qué fases de desarrollo afecta y a cuáles no?

Afecta a todas las partes del desarrollo, desde el inicio hasta la entrega del software, como se indica en el punto 6.1 del manual de calidad. Sin embargo, durante el ciclo de vida del software, deja fuera las partes de mantenimiento y de explotación del mismo.

- Viendo los documentos... ¿Cómo se identifica al Plan de Calidad dentro de la organización?

El plan de calidad es un documento formal dentro del proyecto y que se aprueba junto al plan general del proyecto. Se identifica por su nombre, "Quality Plan", y se diferencian por el nombre del proyecto y la fase del mismo a la que va dirigido, ya que cada fase del proyecto puede tener diferentes responsabilidades y/o roles.

- ¿Cómo podríamos definir el concepto de calidad en Sistemas de Información?

Para definir el concepto de calidad en un sistema de información podemos basarnos en diferentes métricas. Algunos ejemplos son el número de fallos del código, el tiempo que transcurre entre dos fallos diferentes o el porcentaje de requisitos cumplidos del cliente.

OGCIO lo define mediante un Quality Planning, definiendo al comienzo del proyecto las tareas relacionadas con la calidad para conocer los requisitos específicos de calidad.

- ¿Qué documentos son básicos para el QMS? ¿Cuáles echarías en falta en los documentos proporcionados?

Entre los documentos básicos para el QMS encontramos el manual de calidad, el ISPM y tras ellos los estándares, guías y procedimientos a seguir. Los documentos proporcionados son el manual de calidad, el procedimiento de planificación de calidad, el procedimiento de revisión de aseguramiento de la calidad y una plantilla del plan de calidad. Podríamos echar en falta el ISPM ya que es un documento que se nombra varias veces y está en la parte alta de la jerarquía, también podríamos valorar poder revisar los históricos en el “Quality Records”. Además sería interesante tener el procedimiento formal de cambio o las desviaciones al plan de calidad en caso de ser necesarias.

- ¿Quién es el responsable de crear el Plan de Calidad?

El responsable de crear el plan de calidad es el jefe de proyecto, encargado también de asignar tareas del proyecto y de la calidad del mismo, así como la implementación del plan. En el plan de calidad debe definir como el producto va a ser creado y como se incorporará la calidad esperada del mismo.

- ¿En qué puntos se realiza una revisión de calidad y dónde se especifican esos puntos?

La revisión de calidad del proyecto se hace en las etapas definidas en el punto 6.1.3 del manual de calidad. Los checkpoints son los siguientes: Al final de la fase de estudio de viabilidad, al final de la fase de análisis y diseño del sistema y en dos puntos de la implementación, uno al final del diseño físico del sistema y otro en preproducción.

- Viendo los documentos... ¿Quiénes son los principales responsables del plan de aseguramiento de la calidad, y cuáles son sus responsabilidades principales?

Los principales responsables del plan de calidad son el jefe de proyecto, que es responsable de los entregables del proyecto y de que se ejecuten las revisiones de calidad definidas en el plan de proyecto; el revisor, que no tendrá relación con el proyecto y se responsabilizará de revisar los procesos del proyecto; el equipo de desarrollo profesional, que debe asesorar al equipo del proyecto y a los revisores sobre los procedimientos; y por último, fuera de las tareas de revisión, se nombra a los equipos de soporte, responsables de producir guías, estándares, procedimientos y manuales.