

Procedimiento de control en cambios en sistemas

ITAPP

Hoja de control

Organismo	Equipo de desarrollo del producto ITAPP		
Proyecto	ITAPP		
Entregable	Procedimiento de control en cambios en sistemas		
Autor	Haritz Saiz, Xabier Gandiaga, Álvaro Huarte, Xabier Etxezarreta, Onintza Ugarte y Ander Bolumburu		
Versión/Edición	0001	Fecha Versión	30/05/20
Aprobado por		Fecha Aprobación	08/06/20
		Nº Total de Páginas	13

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
0100	Versión inicial	Ander Bolumburu	25/05/20

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos
Ander Bolumburu Casado

Índice

1.	Objetivo	3
2.	Alcance.....	4
3.	Definiciones y Roles	5
3.1	Definiciones	5
3.2	Roles.....	5
4.	Modo de operación.....	6
4.1	Presentación de solicitudes de cambio formal.....	6
4.2	Envío y revisión.....	7
4.3	Definición final de opciones y documentación de respuesta	7
4.4	Toma de decisiones y aprobación	8
5.	Resumen y reflexión	10

1. Objetivo

El principal objetivo de la Gestión de Cambios es la evaluación y planificación del proceso de cambio para asegurar que, si este se lleva a cabo, se haga de la forma más eficiente, siguiendo los procedimientos establecidos y asegurando en todo momento la calidad y continuidad del servicio.

El proceso de Control de Cambios se aplica de igual forma tanto en el proceso de desarrollo del software como en la etapa de garantía; esto con el propósito de asegurar que los cambios son controlados por un método unificado y garantiza el desempeño de los procesos y correcta administración del alcance.

Los cambios pueden ocurrir durante las fases de elaboración, construcción y transición de la solución del proyecto, es decir en cualquier tiempo antes o después de su implantación y estabilización en producción.

Debido a que pocos planes de proyecto funcionan exactamente como se planificaron en la práctica, es esencial contar con un proceso de gestión del cambio que permite que los cambios se realicen sin problemas y obteniendo mejores resultados de los que se obtendrían si un plan o proceso defectuoso sigue vigente.

Y es por ello por lo que es necesario un procedimiento de control de cambios.

2. Alcance

Esta Política debe ser conocida y aplicada por todos los miembros del equipo. En caso de que los miembros del equipo de desarrollo no tengan la capacidad de hacerlo se les formará con el fin de que sean capaces.

El hecho de ser capaces consiste en tener las habilidades necesarias para gestionar e implantar los diferentes cambios en el sistema a desarrollar.

3. Definiciones y Roles

Mediante este apartado se procederá a definir tanto aspectos necesarios para la comprensión del control como los diferentes roles y responsables identificados:

3.1 Definiciones

- **Desarrollo Seguro:** Requisito para generar un servicio, arquitectura, software y sistema seguro, desde la perspectiva del resguardo de la información.
- **Vulnerabilidades:** Se refiere a alguna condición de debilidad o fragilidad que se encuentra presente en el activo identificado. Usualmente se traduce en una debilidad o ausencia de control, que posibilita la ocurrencia de un incidente y que pueden afectar a uno o más activos de información.
- **Equipo de Desarrollo:** Corresponde a los funcionarios institucionales o personal externo que forman parte del grupo a cargo de analizar y programar las funcionalidades de proyectos de desarrollo de software y sistemas.

3.2 Roles

- **Desarrolladores:** Cumplen cabalmente con las disposiciones y requerimientos establecidos en la presente política. Cada miembro del equipo deberá velar por la correcta implementación de las normas de desarrollo seguro de software, así como del cumplimiento por parte de su equipo de trabajo.
- **Jefe de desarrollo:** Encargado de verificar que el desarrollo marcha correctamente tal y como se define en las políticas.

4. Modo de operación

Cuando uno de los miembros del equipo de desarrollo (o a veces el propio cliente) cree necesario un cambio en el sistema desarrollado/a desarrollar se procede de la siguiente forma:

4.1 Presentación de solicitudes de cambio formal

La persona o grupo que desee sugerir un cambio debe hacerlo de una manera que sea absolutamente clara e inequívoca. La documentación detallada de la solicitud de cambio permite una definición clara del cambio deseado. Al prever el procedimiento de control de cambios, las empresas deben decidir si se trata de un envío en un formato específico o si un simple correo electrónico es suficiente.

En el caso de ITAPP se ha decidido anotar en un documento cada uno de los cambios a realizar/realizado con el fin de tenerlos controlados. Se ha creído adecuado este formato ya que, si en un futuro algunas de las versiones de las herramientas utilizadas entran en conflicto con otras, es sencillo detectar en que versión se realizó el cambio y partir de esa versión concreta para solucionar el conflicto.

Los cambios que se lleven adelante generarán una cantidad de issues/tareas en el Gitlab del proyecto, para más detalle referirse al documento de Política de desarrollo seguro.

El formato acordado es el siguiente:

1. Definir la necesidad del cambio: ¿Qué cambio se cree que es necesario?
2. ¿Por qué es necesario el cambio? ¿Qué pasará si no se hacen cambios? ¿Tendrá un impacto significativo en el sistema (problemas de rendimiento, de disponibilidad...)?
3. ¿Qué se necesita para que el cambio tenga éxito? ¿Qué resultado desea ver la persona o entidad que solicita el cambio? ¿De qué depende? ¿Cuántas horas nos llevaría realizar el cambio?
4. ¿Para cuándo es necesario realizar el cambio? ¿Urge? ¿Es viable en el tiempo estimado?
5. ¿Agrega un valor diferencial el cambio? ¿El cambio propuesto mejorará la experiencia del cliente? ¿Traerá mayor precisión?

4.2 Envío y revisión

Una vez que se envía la solicitud de cambio, comienza el proceso de revisión. El equipo de desarrollo debe discutir las opciones en base a la información proporcionada en la solicitud.

Lo ideal sería convocar una reunión formal con todas las partes afectadas. Esto resultará en una mejor comunicación. El equipo de desarrollo puede intercambiar ideas, plantear inquietudes y hacer cualquier pregunta que necesiten hacer.

Primero, la solicitud de cambio se presenta en detalle. A continuación, el equipo discutirá los posibles impactos del cambio, generará opciones, valorará los riesgos mediante un análisis y tomará una decisión.

Este análisis de riesgo es de importancia ya que nos permite evaluar qué riesgos puede suponer para nuestro software en desarrollo/producción, comprobando si merece la pena la actualización del componente. En caso de que así sea se debe realizar un control de impacto, es decir, comprobar si el hecho de actualizar el software va a suponer un tiempo de mantenimiento en el que el servicio deje de ser ofrecido. De ser así puede suponer pérdidas que el equipo no está dispuesto a asumir ya que la vulnerabilidad que tapa/nueva funcionalidad no es lo suficientemente grande como para justificar un mantenimiento/cambio del servicio.

Las decisiones tomadas y los argumentos sobre los cuales se basan se recogerán en issues/tareas de Gitlab con comentarios detallando el porqué de que no se llevaran a cabo.

Dado que el cliente/solicitante del cambio espera recibir comentarios dentro de un cierto período de tiempo, el equipo del proyecto necesita saber cuánto tiempo tienen para tomar una decisión o al menos proporcionar comentarios.

La persona que se relaciona con el cliente debe aclarar las expectativas basadas en el tiempo al manejar la solicitud de cambio.

4.3 Definición final de opciones y documentación de respuesta

Normalmente, las solicitudes de cambio tienen más de una solución posible. El equipo de desarrollo debe determinar qué opciones son viables y presentar al menos dos opciones.

Al igual que con la solicitud de cambio, se ha decidido que la respuesta debe estar argumentada.

Se ha acordado el siguiente formato para las respuestas:

1. Referencia de opción: Puede ser un número o un nombre
2. Propuesta de solución: Se deben presentarse tanto la solución sugerida para el problema/cambio como el razonamiento detrás de ella. Cuando se ofrecen soluciones técnicas como opciones potenciales, se debe explicar en detalle lo que implicarían. Si la conclusión resulta en rechazar la solicitud del cliente, también se deben dar argumentos.
3. Plazo previsto para la implementación: El cliente necesita saber cuánto tiempo llevará implementar las opciones, ya que esto puede afectar la elección final.
4. ¿Cómo impactarán los cambios al proyecto? Hay momentos en que la implementación de un cambio propuesto no tiene un impacto significativo en el proyecto en su conjunto. Sin embargo, la mayoría de los cambios afectan a la planificación del desarrollo. El cambio podría modificar el alcance, los plazos, el presupuesto del proyecto o la calidad del producto final. Naturalmente, los clientes querrán saber cuáles son las implicaciones de los costes, si se deben modificar los hitos y los objetivos, y cómo esto afectará el alcance y la calidad general del producto o servicio.
5. ¿Cuándo debe el cliente comunicar una decisión? Establecer fechas límite es bueno para todos los interesados. La falta de certeza y claridad puede representar un obstáculo importante para el progreso, por lo que, para hacer avanzar el proceso, los plazos son una necesidad. Por ejemplo, si es probable que se reduzcan el alcance del proyecto y los recursos de asistencia necesarios, el equipo debe saber cuándo debe dejar de invertir recursos en el diseño del proyecto anterior. Por esta razón, los clientes que superen su fecha límite serán reevaluados sus solicitudes de cambio en caso de que comuniquen su decisión después de la fecha de vencimiento.

4.4 Toma de decisiones y aprobación

El acuerdo oficial y la aprobación finalizan el proceso de control de cambios. El equipo responsable debe saber a quién informar y quién en la organización del cliente/solicitante tiene derecho a aprobar formalmente los cambios.

Este es un paso absolutamente esencial. Así como los tomadores de decisiones aprueban los proyectos y contratos iniciales, también deben ser ellos los que aprueben cualquier cambio.

En el caso de que la evaluación y el control de riesgos dé un resultado favorable a la actualización, antes de llegar a implementar nada, es recomendable hacer pruebas con la nueva versión de la herramienta en un entorno controlado. De este modo evitaremos que la implementación no afecte de forma inesperada al software en producción.

Por otro lado, se debería implementar una versión funcional con la nueva versión de la herramienta con el fin de pasar el conjunto de validación para poder comprobar que la implementación de la nueva versión del software no rompe por completo ninguna funcionalidad del software.

En caso de que la nueva herramienta sea un desarrollo interno, este se testeará en un entorno de testeo y en un entorno de staging. Referirse al documento de Política de Desarrollo Seguro.

Una vez tenemos aprobados todos los pasos previos, es el momento de avisar al encargado principal del desarrollo que aplique el cambio en el repositorio principal y notifique al resto de desarrolladores del cambio con el fin de asegurar que no existen conflictos de versiones.

Cabe destacar que el cambio solo lo puede aplicar el jefe de desarrollo ya que en caso contrario (que cada uno pueda hacer los cambios referentes a versiones que quieran) la probabilidad de existir un conflicto de versiones tanto en el repositorio como en las copias locales aumentaría.

Por último, resulta recomendable destacar que es interesante limitar la cantidad de actualizaciones al mes, ya que, si se hacen muchas actualizaciones de forma simultánea, en caso de que en un futuro ocurra un error (por una posible colisión de versiones) resultaría más complicado encontrar que herramienta es la causante del problema.

5. Resumen y reflexión

Finalmente, todo el proceso se resume en el siguiente diagrama, en el que podemos observar los pasos a seguir en el momento de querer hacer un cambio:

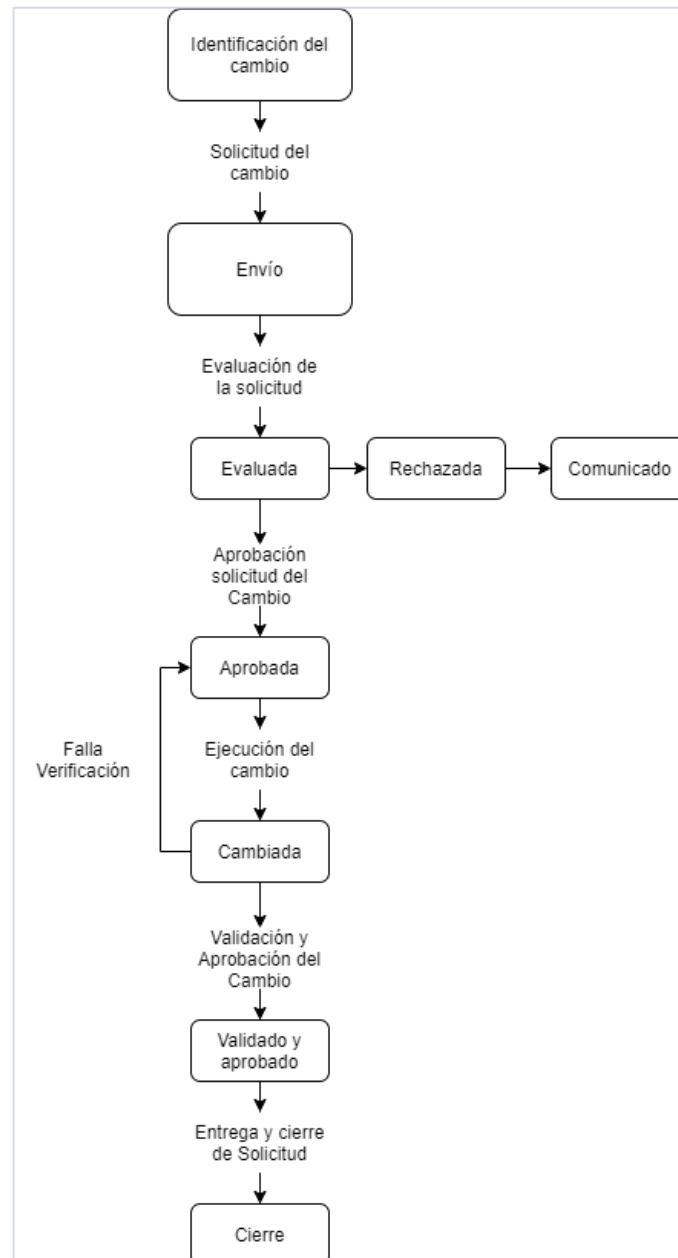


Ilustración 1. Diagrama del proceso

Finalmente, a modo de reflexión, es interesante indicar que debido a que se está tratando con clientes/integrantes del equipo de desarrollo, es posible que encuentre cierta resistencia cuando se intente implementar un proceso de gestión de cambios.

Aunque con el tiempo, se comprueba que este control de cambios es mutuamente beneficioso ya que aclara qué pueden esperar los clientes y cuáles serán las implicaciones

del cambio propuesto. Dado que esto incluye el coste y la calidad final, el proceso de control de cambios se realiza no solo en interés de su negocio sino también del cliente.

Los clientes pueden ser reacios a preparar la solicitud inicial para la documentación de cambios, pero es posible prepararlos en base a una conversación y luego enviarlos al cliente para determinar si ha entendido correctamente lo que se requiere. Una vez que el cliente está dispuesto a aceptar formalmente que la solicitud de cambio se ha capturado con precisión, el proceso puede avanzar sin problema.

