PLAN DE GESTIÓN DE CONFIGURACIÓN

Javier Santana Delgado
Jose Antonio Santacruz Gallego
Tamara Redondo Soto
Julio Sánchez de las Heras Martín Consuegra
Alejandro Riquelme Castaño
José Antonio Oliver González-Ortega

PLAN DE GESTIÓN DE CONFIGURACIÓN

PLAN DE GESTIÓN DE CONFIGURACIÓN

PROPÓSITO DEL PLAN

En este documento se establecen las actividades de Gestión de configuración que se deben seguir para ser aplicadas a nuestro proyecto Fritura para un restaurante de acuerdo con el estándar **IEEE 828**.

Se controlarán y gestionarán las distintas versiones del proyecto de forma que se consiga **trazabilidad** en todo momento, se discutirá cada nueva posible modificación especificada por el cliente intentando mantener la **consistencia** del proyecto, se establecerán **reuniones** para acordar quién se ocupará de aprobar los requisitos, los recursos que gestionará cada recurso, etc.

En definitiva los componentes estarán **bajo control de configuración** así como los pasos que seguirán los integrantes de cada uno de los equipos de trabajo.

ALCANCE DEL PLAN

Siendo nuestra organización "novatos" en el mundo laboral, este sería el **primer plan de gestión de configuración** que vayamos a desarrollar. Por ello, nuestro objetivo es intentar tener el **máximo control posible** sobre cada uno de los **procesos** que se realizan dentro de las **iteraciones y fases**, por ello el control de configuración debe abarcar la mayor cantidad de elementos posibles, teniendo en cuenta las capacidades y las limitaciones del proyecto.

Si fuera necesario se aplicarán **modificaciones** que haya solicitado el cliente y se actuará de una manera adecuada. Por último, al terminar cada una de las iteraciones, se llevarán a cabo sus **pruebas** correspondientes para comprobar su correcto funcionamiento.

RELACIÓN CON LA ORGANIZACIÓN Y CON OTROS PROYECTOS

En este caso nuestro equipo de trabajo está realizando **diferentes proyectos de manera simultánea**, pero al ser este uno de los más **ostentosos** en cuanto al trabajo en equipo, nos servirá para **mejorar la coordinación** del equipo en los otros.

TÉRMINOS CLAVES

- **Baseline (Línea Base):** Especificación o producto que ha sido formalmente revisado y sobre el que ha sido alcanzado un acuerdo, y que por tanto sirve de base para futuros desarrollos.
- **Branching (Ramificación):** La creación de una nueva línea base a partir de una versión existente.
- **Build (Construcción o Producto o Versión):** Versión operativa del sistema o de algún componente de este.
- **Change Request (Petición de Cambio):** Una petición por parte de un cliente interno o externo para hacer un cambio a la línea base de la configuración.
- **Codeline (line de Código):** Es un conjunto de versiones de un componente de software y otros elementos de la configuración
- **Configuration Control (Control de Configuración):** Se trata de un elemento consistente en todas las actividades que tienen relación con la gestión de configuración trans su previo establecimiento formal de la configuración.
- **Configuration Control Board (Consejo de control de Configuración):** grupo de empleados cualificados cuya responsabilidad es evaluar, aceptar o rechazar y asegurar la implementación de los cambios propuestos a la línea base del proyecto.
- **Configuration Item (Elemento de Configuración):** Agregación de productos de trabajo que son tratados como entidades simples en los procesos asociados y son integrados en el control de la configuración.
- **Configuration Management (Gestión de Configuración):** dentro del desarrollo del software, se trata de la disciplina orientada a la identificación y documentación de las características de los elementos de la configuración, control y registro de los cambios y su respectivo estado de implementación, además de verificar el grado de satisfacción de los requisitos previamente establecidos.
- **Configuration Management Plan (Plan de Gestión de Configuración):** Documento que establece los elementos necesarios para ejercer la gestión de la configuración.

- **Configuration Management Authority (Autoridad de Gestión de Configuración):** Cualquier persona o grupo designado para ser responsable de las actividades de la gestión de configuración para que sean planificadas y ejecutadas.

- **Configuration Management Database (Base de datos de la gestión de configuración):** Repositorio para el mantenimiento de la información de la gestión de configuración, que normalmente es un almacén de datos.

- **Deltas (Incrementos):** avance generado tras una iteración del desarrollo, el cual puede ser cualquier elemento de configuración en la base de datos de gestión de configuración.

- **Minimum Viable Product (Producto Mínimo Viable):** Es una versión del producto con las características mínimas para determinar si se satisfacen los requisitos más importantes para los clientes.

- **Release (Liberación / Publicación):** Versión de software que se pone formalmente a disposición de la comunidad, puede contener todo o parte de dicha aplicación.

- **Release Alpha (Liberación Alpha):** primera fase donde comienzan las pruebas de software. Estas versiones, las cuales pueden contener problemas de estabilidad, en general son probadas por empleados relacionados con el proyecto, con un nivel de conocimiento sobre la materia elevado.

- **Release Beta (Liberación Beta):** se trata de una fase secundaria de las pruebas del software, la cual se lleva a cabo una vez terminadas todas las funcionalidades del código y corregidos los errores encontrados en la versión Alfa.

- **Release Plan (Plan de liberación):** plan encargado de describir que partes de la funcionalidad del sistema serán llevadas a cabo en cada release acompañado de sus fundamentos.

- **Software Release Management (Gestión de liberación del software):** Gestión de los procesos dirigidos a la publicación de una o más versiones de software.

- **Tag (Etiqueta de identificación de versión software):** Identificador explícito e inmutable de una versión de software insertado para cada elemento de la configuración.

- **Tools (herramientas de Gestión de Configuración):** usadas para hacer un correcto seguimiento de los cambios, almacenar versiones de los componentes del sistema, construir sistemas sobre dichos componentes y rastrear las liberaciones de versiones para los clientes.

- **Versión:** Una publicación o publicación de un elemento de configuración software, asociado a una compilación completa del elemento de configuración.

- **Versioning (Versionado):** Asignación de un identificador único a estados de los elementos de configuración del software, para un propósito específico.

ELEMENTOS DE CONFIGURACIÓN

- **Línea base:** la línea base de la arquitectura del sistema quedará definida en la iteración 0. Si fuera necesario la realización de algún cambio se debe crear una reunión con todo el equipo para informar sobre los cambios, documentarlo y que todos los integrantes estén de acuerdo.

- En cuanto a los **archivos** relacionados con los **modelos de requisitos, análisis y diseño** se encontrarán en el **archivo .vpp** en el repositorio de GitHub del proyecto con sus fases e iteraciones correspondientes.

- Con respecto a los **planes de gestión de Configuración y Pruebas** además de la planificación del proyecto, estos se encontrarán en la **Wiki** del repositorio de GitHub.

- Los **archivos .java** que contendrán el **código fuente** del proyecto también se encontrarán en el repositorio de GitHub, que se irán subiendo a la **rama development** y cuando se tenga todo el sistema implementado se realizará un **merge a la rama principal**.

- El **archivo pom.xml** que es el que contiene la **configuración de nuestro proyecto** también se encontrará en el mismo lugar que los **archivos .java** comentados anteriormente.

LIMITACIONES Y SUPOSICIONES QUE AFECTAN AL PLAN

Limitaciones laborales:

- Todos los recursos **trabajarán** un total de **14h/semanales**, doblando su horario habitual en la última interacción.
- El total de **tiempo estimado** para este proyecto es de un total de **141 horas**.
- Se establecerán los **días de trabajo** de **Lunes a Domingo**.
- El equipo **no tendrá derecho a descanso** debido algún día **festivo**, ya sea regional, nacional o mundial.
- El recurso que realice la **fase de implementación** de una iteración no podrá realizar las **pruebas** de esta.

- Durante la realización del proyecto se llevarán a cabo otros **proyectos distintos** de manera concurrente**, ralentizando** la realización de este.
- Cualquier integrante del equipo podrá **no trabajar algún día** por **motivos de salud o personales**.
- La única forma de **contactar con el cliente** será a través de **correo electrónico o reuniones esporádicas**.

RESPONSABILIDADES

- Los responsables de la actividad de **aprobar los requisitos** son Javier Santana Delgado y Julio Sánchez de las Heras Martin-Consuegra**,** ya que estos se encargarán de recoger las necesidades de los clientes realizando una serie de reuniones para posteriormente realizar un estudio de los requisitos sacados y después contrastarlos con los clientes para ver si están de acuerdo.

- El **responsable de los componentes** es Jose Antonio Oliver González-Ortega el cual será el que realizará un seguimiento de los componentes y su estructuración en el proyecto.

- El responsable de **aprobar los componentes es** Alejandro Riquelme Castaño**,** una vez Jose Antonio Oliver González-Ortega los haya identificado**,** Alejandro Riquelme Castaño dará su veredicto para decidir si estos son válidos para el proyecto que estamos tratando.

- Los responsables del **plan de versiones de los componentes** serán Tamara Redondo Soto y Jose Antonio Santacruz Gallego los cuales**,** además**,** se encargarán de **la gestión de las versiones de las construcciones del proyecto,** que garantizarán el seguimiento de las distintas versiones de los componentes del sistema y que los cambios hechos no interfieran entre sí.

ORGANIZACIÓN DEL PROYECTO

- El responsable de la **gestión de configuración del software** se encargará de fomentar el plan de gestión de configuración**,** asegurarse de la correcta planificación**,** identificación… de los elementos que se encontraran en la base de datos de la configuración**,** dirigir la evaluación de los procesos**,** ya sea para revisar los elementos de la configuración**,** estructura de la base de datos**,** etc. También realizan la acción de aprobar los cambios estructurales de la base de datos de configuración.

- El **coordinador de la configuración** deberá asegurar que todos los elementos de la configuración se registran convenientemente en la base de datos de configuración. Además, deberá gestionar todas las acciones que se realizan sobre las líneas base del proyecto, así como garantizar la consistencia e integridad de los datos de configuración.

- El **responsable de elementos de configuración** tratará de asegurar la integridad de los elementos de la configuración, también verificará los cambios que podrán realizarse sobre los elementos de configuración teniendo en cuenta que siguen el proceso de definido. Por último este trabajará conjuntamente con el gestor de configuración por si hubiera cualquier discrepancia.

- El **responsable de la gestión de cambios** se ocupará de la supervisión de la correcta actualización de los elementos debido a los cambios posibles realizados y también estimará el impacto de dichos cambios y sus riesgos.

POLÍTICAS

POLÍTICAS DE CONTROL DE CAMBIOS

- **Todos los cambios que se deban realizar** serán definidos en un **manifiesto de cambios** en el que se debe especificar entre otras cosas la fecha, el recurso que solicita el cambio, el recurso que debe realizar el cambio y los cambios específicos de los componentes.

- Una vez considerados los cambios y contemplando su impacto se **remitirá su aprobación o declinación** por parte de cada uno de los integrantes del proyecto.

- Los cambios referidos a la **implementación** quedarán especificados al final del proyecto en el **diagrama de clases** y también en el **modelo entidad-relación de la base de datos.**

POLÍTICA DE REPOSITORIO

- Para realizar un seguimiento de todos los cambios que se realizan en cada elemento del que está compuesto el proyecto, se utilizará la herramienta de **control de versiones Git,** más concretamente GitHub para tener la posibilidad de que todos los integrantes del equipo de desarrollo puedan trabajar de manera conjunta en un repositorio remoto.

- Para el uso correcto del repositorio, crearemos las ramas necesarias como nos indica **Git Flow** y en cada una de ellas trabajar de la manera correcta. Más específicamente, además de la **rama master,** se creará las **siguientes ramas:**

- **Development:** donde se encontrarán los elementos que estén en desarrollo dentro de cada iteración.

- **Hotfix:** donde se encontrarán las correcciones de posibles fallos que se encuentren durante el desarrollo.

- **Features:** donde se encontrarán las nuevas características que se vayan desarrollando de forma conjunta con la implementación normal. Una vez implementadas en su completitud, dichas características deben integrarse con la rama Development.

- En cuanto al versionado nos basaremos en **Semantic Versioning,** el cual tiene el formato **MAJOR.MINOR.PATCH.**
  El **Patch** se irá incrementando en el caso de realizar ligeras modificaciones o arreglos que sean compatibles con versiones anteriores. El **Minor** se incrementará por cada fase y también cuando se añadan funcionalidades en el código que sean compatibles con las versiones anteriores. Por último, decidimos incrementar el **Major** en cada iteración del proyecto. Las versiones Patch y Minor se resetean a *0* cuando se incrementa el Major.

- Los **nombres de los archivos** que se subirán deberán tener **letras y números,** no estará permitido el uso de tildes, ñ o cualquier carácter especial. Dentro de la subida de archivos tampoco estará permitido la subida de **archivos binarios** y **archivos comprimidos.**

ACTIVIDADES

GESTIÓN DEL CAMBIO

Se tratan de las **peticiones realizadas por los clientes o por los propios desarrolladores,** para ello se realizará un **manifiesto** con la solicitud de cambios y estos serán analizados y aprobados o denegados, evaluando su impacto en los costes y lo que supondría en el propio proyecto ese cambio.

GESTIÓN DE VERSIONES

Se trata de la forma de **establecer las versiones de los componentes de acuerdo con lo anterior comentado en "Política de repositorio"** de forma que se consigan desacoplar los distintos cambios realizados por los diferentes desarrolladores.

CONSTRUCCIÓN DEL SISTEMA

Consiste en un **proceso de integrar todos los componentes, datos y librerías del programa para luego más tarde compilarlos,** de manera que todos estos están vinculados unos con otros para crear un software ejecutable y que tenga un correcto funcionamiento.

GESTIÓN DE LIBERACIONES/ ENTREGAS DEL SOFTWARE

El proceso de preparación del software para su respectiva entrega al cliente **se organizará de la siguiente forma:**

- Una vez implementada toda la funcionalidad correspondiente a una iteración y tras haber realizado sus pruebas, se le **entregará al cliente una parte de software con funcionalidad propia.**

- Una vez terminadas todas las iteraciones del desarrollo del proyecto y, por consiguiente, teniendo todas las funcionalidades del sistema implementadas y probadas, se procederá a **realizar una "release Alpha",** la cual tiene como objetivo comenzar las pruebas del producto software completo, donde se someterá a dicho software a unas pruebas por parte de los empleados del proyecto. Más tarde, se realizará una **"release Beta"** para someter al software a unas pruebas de manera masiva con unos usuarios con menos conocimientos en la materia. Así, una vez pasado por las fases descritas anteriormente, **se lanzará el producto software al mercado.**

MANTENIMINETO

El plan de configuración será sometido a **revisión al final de cada iteración del proyecto,** así después de realizar las pruebas, se analizará si dicho plan ha satisfecho las necesidades de los integrantes del equipo y sigue siendo correcto con respecto al trabajo realizado en dicha iteración.

En caso de necesitar **realizar algún cambio,** el responsable de la gestión de los cambios tendrá que analizarlo y emitir su veredicto final, aprobando o rechazando el cambio.

HERRAMIENTAS

- **GitHub:** Sistema web de control colaborativo de revisión y desarrollo de software utilizado para guardar proyectos que son gestionados por Git.
- **Git:** Sistema de control de versiones.
- **Visual Paradigm:** Herramienta UML para realizar el ciclo de vida completo de un software, soporta la fase de requisitos, análisis y diseño (también puede generar código java para más tarde utilizarlo en un desarrollador) orientado a objetos, construcción... permite dibujar todos los tipos de diagramas y por último nos permite utilizar tanto ingeniería inversa como ingeniería directa.
- **MySQL Workbench:** Herramienta que ofrece la posibilidad de realizar un diseño, modelado, generación y administración de una base de datos. Permite crear modelos de entidad-relación, aplicar ingeniería directa e inversa o incluso ofrece características para realizar gestión de cambios y documentación que llevarían consigo un gran tiempo y esfuerzo.
- **Maven:** Herramienta de software dedicada a la gestión y construcción de proyectos. Permite una integración continua para poder realizar test unitarios, test de integración, ... hacer posible la creación de software con dependencias dentro de la estructura del JAR, etc.
- **JUnit:** Un frameword java con el cuál se pueden llevar a cabo la ejecución de clases de manera ponderada, para que se puedan comprobar que los métodos realizan correctamente su tarea. Además, se puede usar para las pruebas de regresión, cuando modificamos una parte del código y es preciso verificar que se sigue dando con todos los requisitos.