

**TALLER GESTION DE LA CONFIGURACION**

DEFINICION DE POLITICAS

**INTEGRANTES**

Andrés Rodríguez Roa

Luis Sabroso Cortes

Tito Maturana De La Cruz

Edward Ramos Pérez

Diego Milquez

Pontificia Universidad Javeriana

Facultad de Ingeniería – Departamento de Ingeniería de Sistemas

Especialización Arquitectura Empresarial de Software

Ingeniería de Software

Bogotá, Agosto de 2022

Contenido

[1 Identificación de los Ítem de Configuración. 3](#_Toc112337462)

[2 Control de Configuración 4](#_Toc112337463)

[3 Seguimiento Estado 6](#_Toc112337464)

[4 Auditoria Configuración 7](#_Toc112337465)

Políticas Gestión de la Configuración

La gestión de la configuración establece 4 grandes etapas que se deben cumplir para un adecuado manejo para lograr mantener un producto de Software.

Estas incluyen

# Identificación de los Ítem de Configuración.

|  |  |  |
| --- | --- | --- |
| **ETAPA** | **CONCEPTO** | **POLITICA** |
| Identificación Configuración | Planeación IC | Se establecen los siguientes como los Ítems de Configuración (CI) a ser controlados a lo largo del proceso de Desarrollo del producto de Software:  -Casos de Pruebas -Versión del Fuente en el repositorio -Generación de Línea Base -Reporte de defectos -Modelos de datos  -Scripts -Plan de Instalación |
| Estructura de configuración | Se maneja 2 ambientes desarrollo y producción  MAIN: esta rama contendrá todas las versiones estables del producto, con lo cual cualquier commit que se realice sobre ella, implica que el producto está listo para ser usado por los usuarios. Este paso implica que el producto haya pasado los casos de pruebas y que esté debidamente documentado.  DESARROLLO: En esta rama se aloja el código que conformará la siguiente la versión. Sobre esta rama se trabajará con versiones inestables, que buscan agregar los cambios necesarios para cumplir con los requerimientos. Además de estas dos ramas principales se establece el uso de ramas de soporte. Dichas ramas son:  Feature: Nuevas características del producto. Hotfix: Cambios en “caliente”, corrección de bugs reportados mediante el uso de tickets en versiones del servicio que se encuentran en la rama main.  La ruta de los directorios dependerá de la configuración interna que tenga el repositorio seleccionado que para nuestro caso en GitHub,.  Distribución igual y recursos cercanos en los ambientes de Desarrollo, QA, Producción |
| Convención de Nombre | - Utilizar solo caracteres alfanuméricos, a excepción de guiones(-) o barras bajas(\_) no deben utilizarse puntos o dejar espacios.  -Solo utilizar minúsculas. Esto ayuda a evitar conflicto cuando se migran objetos a un sistema operativo que sea sensible al uso de mayúsculas. |
| Localización Nemónica | -Identificar una única ubicación para el almacenamiento de los archivos que constituyen hitos y archivos maestros. - Sincronizar los archivos ubicados en diferentes localizaciones de forma regular. |
| Numeración de Versiones | En los proyectos desarrollados se deben seguir unas normas de versionado. Esta política incluye:  -Uso de tres dígitos para la numeración: major, minor y revisión. Esta numeración estará definida por la política de Control de Números descrita dentro de este documento. |
| Planeación Línea Base | Para el proyecto de pre-simulación Se toma como línea base la versión hasta la historia PDCA-1 como versión v.1.0.0.  A partir de ahí se tomarán las funcionalidades o releases que se definan. |
| Control Números | El esquema para la identificación de las distintas versiones del producto es un esquema numérico conformado de la siguiente manera:  v<major\_version>.<minor\_version>.<patch\_number>  Major Version: indica un cambio de funcionalidad sustancial a otras versiones del sistema Minor Version: indica que el sistema es funcionalmente idéntico, pero es distinto desde el punto de vista no-funcional a otras versiones Patch Versión: indica el número de un parche aplicado al sistema, que significa la corrección de un defecto encontrado. |

# Control de Configuración

|  |  |  |
| --- | --- | --- |
| **ETAPA** | **CONCEPTO** | **POLITICA** |
| Control de Configuración | Librería Controlada | No Aplica. |
| Sumisión CI Dependencia | - Registrar las relaciones entre componentes cuando sea necesario.  Dentro lo definido para el proyecto se tendrá dentro del repositorio un diagrama indicando las dependencias que se tienen por cada uno de los componentes. |
| Control versión estado | Se debe generar inventario de los Cambios solicitados y cambios aprobados. Estado de implementación de cada cambio aprobado. |
| Control Distribución Retiro | Se debe mantener Historial de los procesos aplicados a cada elemento de configuración |
| Reporte Defectos | Se debe mantener inventario de los defectos generados en la elaboración y pruebas del producto este inventario se debe llevar en la respectiva Bitacora |
| Investigación | Para cada CI en el reporte, revisar el estado actual del CI contra los datos contenidos en la CMS. Investigar cómo solucionar las inconsistencias y cómo prevenir que esas inconsistencias vuelvan a ocurrir. |
| Limpieza | -Identificar las ultimas 5 versiones del GitHub que deberán ser guardadas y respaldadas que no sean usadas en la versión actual. |
| Control Cambio | Los cambios al producto se manejarán con los dueños del producto. Tito Maturana y Luis Sabroso |
| Análisis de Impacto | Crear un mapa mental que nos permita identificar los puntos de mejora para así alcanzar las metas propuestas para el proyecto de manera más ágil proponiendo funcionalidades o acciones.  Sacar provecho de la experticia de los integrantes del grupo para identificar riesgos y soluciones a problemas presentados en iteraciones anteriores del proyecto.  El control de camio para este proyecto estará a cargo de los integrantes del grupo con una validación de pares con respecto a los cambios a realizar en cada despliegue |
| Revisión/Pruebas | No aplica |

# Seguimiento Estado

|  |  |  |
| --- | --- | --- |
| **ETAPA** | **CONCEPTO** | **POLITICA** |
| Seguimiento Estado | Colección de Datos | Los actores involucrados deben registrar la configuración y mantenida de acuerdo con los criterios definidos en la presente norma, manteniendo la fecha, la versión y el elemento de configuración creado, actualizado o eliminado, esto se realizar en el documento definido para este fin. |
| Generacion reporte | Se generará reporte del inventario del ítem de configuración, su estado, los cambios registrados y el responsable de hacerlo. Se debe llevar un registro de la configuración aprobada el estado de los cambios propuestos para la implementación del estado de los cambios aprobados. |
| Análisis de Datos | Se debe producir un reporte periódico para verificar los CI´s a ser revisados en donde se podrá considerar las siguientes fuentes de información: - CMDB (Auditoria periódica) - Buscar todos los incidentes donde se haya detectado una inexactitud en la CMS - Revisar los reportes de las herramientas de descubrimiento automático. Para cada uno se debe registrar cada uno de los Ci que reporten inconsistencias en el último análisis realizado. |

# Auditoria Configuración

|  |  |  |
| --- | --- | --- |
| **ETAPA** | **CONCEPTO** | **POLITICA** |
| Auditoria Configuración | Conformidad Procedimental | Se debe verificar la conformidad con estándares y procedimientos aplicables a la gestión de configuración, por ejemplo, comprobando si se usa la versión correcta del documento de diseño para realizar la codificación. Como resultado de la auditoría se deberá generar un informe donde se registren todas las no conformidades detectadas y así iniciar un plan de mejora para solucionarlas. Después de una auditoría de configuración exitosa se puede establecer una línea base del producto.  Esta verificación se debe realizar cada semana para poder comprobar que esta de acuerdo al procedimiento. |
| Verificación Ítem de Configuración | Se debe verificar que, en un momento dada, la línea base se compone de una colección consistente y bien definida de productos. Los elementos de configuración se encuentran en el directorio apropiado. El estado actual de los elementos de configuración es consistente. La información de línea base se mantiene de forma correcta.  Se debe generar informe de las verificaciones realizadas |