***Universidad de las Fuerzas Armadas***

*Un dibujo de salsa de tomate

Descripción generada automáticamente con confianza baja*

***Tecnologías de la Información y Comunicación***

***Ingeniería de Software II, NRC 16417***

***Actividad de Aprendizaje No. 2***

***Integrantes***

* *Almeida Garófalo Alan Joel*
* *Flores Ortiz, Jose Miguel*

***10 de enero del 2024***

**Gestión de la Configuración**

**Introducción**

La ingeniería de software juega un papel vital en la creación de aplicaciones eficientes y funcionales que satisfagan las necesidades del mercado. En el contexto del estudio de caso de una empresa minorista de libros electrónicos, la gestión de la configuración se presenta como un elemento crítico para garantizar la coherencia e integridad del software en desarrollo. Dirigido por el ingeniero de Edison, Andrés De la Torre D., MS, InfoSecurity asumió el desafío de modelar una aplicación que no solo mantuviera su posición en el mercado, sino que también se esforzara por superar las expectativas de los usuarios y los objetivos comerciales.

En este contexto, la gestión de la configuración es el proceso fundamental para controlar sistemáticamente los artefactos de software desde los requisitos iniciales hasta la implementación final. Dado que la aplicación se simplificará como una aplicación cliente-servidor distribuida con una arquitectura de tres niveles, la complejidad inherente a la interacción entre la presentación, el proceso y los datos resalta la necesidad de una gestión de configuración eficaz. Este documento cubrirá aspectos clave de la gestión de la configuración al crear aplicaciones, teniendo en cuenta los requisitos técnicos y comerciales establecidos, así como los diferentes tipos de usuarios que interactuarán con la plataforma.

Para garantizar que el software sea rastreable, consistente y adaptable a medida que evoluciona, la gestión de la configuración se convierte en una piedra angular estratégica. A medida que profundizamos en los detalles del estudio de caso de Ingeniería de software II, veremos cómo las aplicaciones cliente/servidor distribuidas diseñadas para clientes, administradores de TI y agentes de ventas se beneficiarán enormemente de prácticas sólidas de gestión de configuración. Este enfoque no sólo garantiza un desarrollo fluido y colaborativo, sino que también permite a la empresa perseguir sus objetivos comerciales en el competitivo mercado de ventas de libros electrónicos.

**Alcance**

El alcance de la gestión de configuración en este caso práctico abarcará los siguientes aspectos clave:

**Control de Versiones:**

* Implementación y mantenimiento de un sistema de control de versiones para rastrear y gestionar las diferentes iteraciones del software a lo largo del ciclo de vida del desarrollo.
* Etiquetado y documentación de versiones específicas para facilitar la identificación de hitos y la reversión a estados anteriores si es necesario.

**Gestión de Cambios:**

* Establecimiento de un proceso formal para la solicitud, evaluación y aprobación de cambios en los artefactos de software.
* Registro detallado de las modificaciones realizadas, incluyendo la justificación y el impacto previsto en el sistema.

**Configuración de Ambientes:**

* Definición y mantenimiento de configuraciones específicas para ambientes de desarrollo, prueba y producción.
* Gestión de dependencias y configuraciones de terceros para garantizar la consistencia entre entornos.

**Trabajo Colaborativo:**

* Facilitación de la colaboración entre los diferentes equipos involucrados en el desarrollo, como los encargados de la interfaz de usuario, lógica de negocio y administración de bases de datos.
* Integración continua para garantizar la compatibilidad y funcionalidad conjunta de las distintas partes del sistema.

**Auditoría y Seguimiento:**

* Establecimiento de mecanismos de auditoría para registrar cambios y acceder al historial de configuración.
* Implementación de un proceso de seguimiento para evaluar el rendimiento de la gestión de configuración y realizar ajustes según sea necesario.

**Objetivos**

* Crear un sistema que garantice la integridad del código fuente y facilite la colaboración entre desarrolladores.
* Proporcionar versiones estables y etiquetadas para gestionar puntos de control y reversiones.
* Establecer un proceso estructurado de gestión de cambios, desde la solicitud hasta la aprobación e implementación.
* Minimizar los riesgos asociados con el cambio mediante una evaluación cuidadosa y una comunicación clara de las consecuencias esperadas.
* Mantener configuraciones específicas para entornos de desarrollo, prueba y producción para garantizar la coherencia y minimizar los problemas de incompatibilidad.
* Administrar dependencias externas y configuraciones de terceros para garantizar una implementación sin problemas.
* Mejorar la colaboración entre equipos mediante la integración continua y la sincronización efectiva de las contribuciones individuales.
* Garantizar la interoperabilidad y la funcionalidad compartida entre diferentes capas de una aplicación cliente-servidor distribuida.

**Roles y Responsabilidades**

|  |  |
| --- | --- |
| **Roles** | **Responsabilidades** |
| **Gerente de Configuración:** | - Supervisar la planificación, identificación, control y auditoría de todos los elementos de configuración en la aplicación de venta electrónica de libros.  - Desarrollar y ejecutar el plan de gestión de configuración.  - Fomentar la utilización efectiva de la base de datos de configuración entre los equipos de desarrollo.  - Monitorizar y reportar cualquier cambio no autorizado en los elementos de configuración.  - Garantizar la consistencia e integridad de los datos a través de procedimientos de verificación y auditoría.  - Liderar evaluaciones del proceso y aprobar cambios estructurales en la base de datos de configuración. |
| **Coordinador de Configuración:** | - Registrar adecuadamente todos los elementos de configuración en la base de datos, asegurando su integridad.  - Garantizar consistencia e integridad mediante procedimientos de verificación y auditoría.  - Reportar discrepancias o no conformidades al Gerente de Configuración.  - Participar activamente en la mejora continua del proceso de gestión de configuración. |
| **Responsable de Elementos de Configuración:** | - Registrar en la base de datos los elementos de configuración bajo su responsabilidad con datos apropiados.  - Verificar que los cambios en los elementos sigan el proceso definido.  - Asegurar la idoneidad e integridad de los elementos a su cargo.  - Colaborar con el Gerente de Configuración para abordar discrepancias identificadas en auditorías e implementar acciones correctivas. |
| **Gestor de Cambio:** | - Evaluar el impacto y riesgo de los cambios propuestos en la aplicación.  - Asegurar que los responsables de la configuración actualicen los históricos de los elementos afectados con los cambios implementados. |

**Actividades de Gestión de Configuración**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Actividad** | **Rol Responsable** | **Descripción** | **Entradas** | **Salidas** |
| Planificación de la Gestión de Configuración. | Gerente de Configuración | Elaborar el plan de gestión de configuración para la aplicación de venta electrónica de libros. | Requisitos del proyecto. Planificación de proyecto. | Plan de gestión de configuración aprobado. |
| Identificación de Elementos de Configuración. | Gerente de Configuración | Identificar y registrar los elementos de configuración relevantes para la aplicación. Establecer la estructura del directorio de gestión de configuración. | Documentación de requisitos. Productos del proyecto. | Elementos de configuración identificados. Línea base. Estructura del directorio de gestión de configuración. |
| Control de Cambios y Líneas Base. | Responsable del Elemento de Configuración | Evaluar y controlar los cambios en los elementos de configuración. Obtener la aprobación para solicitudes de cambio en productos de trabajo de línea base. | Solicitudes de cambio. | Registro de solicitud de cambio. Solicitud de cambio aprobada. Línea base actualizada. |
| Informe de Estado de Configuración. | Gerente de Configuración | Mantener y publicar de forma regular el estado actualizado de los elementos de configuración. | Elementos de configuración. | Informe de estado de elementos de configuración. |
| Verificación y Auditoría. | Gerente de Configuración | Realizar auditorías para evaluar la eficacia de la gestión de configuración. | Registros de gestión de configuración. Línea base. Registros de cambios. | Informe de auditoría de gestión de configuración. |

**Identificación de Elementos de Configuración**

Las tareas clave a realizar dentro de esta actividad incluyen:

**Identificación de Productos:**

Identificar los productos que estarán bajo gestión de configuración para el proyecto.

**Asignación de Identificadores Únicos:**

Asignar identificadores únicos para cada elemento de configuración, incluyendo propiedades como autor, tipo de documento, persona responsable, entre otros.

**Definición de Estructura de Almacenamiento:**

Definir la estructura de almacenamiento para organizar y gestionar los elementos de configuración.

**Control de Acceso:**

Definir un nivel de control de acceso para los miembros del equipo sobre la infraestructura de almacenamiento.

**Selección de Herramientas:**

Seleccionar herramientas específicas para la gestión de configuración, considerando las necesidades del proyecto.

**Criterios para Inclusión bajo Gestión de Configuración:**

Especificar cuándo se incluirá cada elemento bajo gestión de configuración en el ciclo de vida del proyecto.

**Autorización y Línea Base:**

Obtener la autorización para incluir documentos bajo gestión de configuración (línea base) y aplicar los procedimientos definidos.

**Procedimientos de Cambios:**

Desarrollar procedimientos para solicitar e implantar cambios, especificando roles y responsabilidades en el proceso.

**Criterios para la Selección de Elementos de Configuración (Ejemplos):**

* Productos de trabajo compartidos entre dos o más grupos.
* Productos susceptibles a cambios debido a modificaciones en requisitos o errores.
* Productos interdependientes, donde un cambio en uno afecta a otros.
* Productos críticos para el proyecto.

**Posibles Elementos de Configuración para la Aplicación:**

Planes

* Plan de proyecto.
* Plan de calidad.
* Plan de gestión de configuración.
* Plan de gestión de riesgos.

**Registros del Proyecto:**

Material de apoyo al cliente.

**Especificación de Requisitos:**

* Requisitos de negocio.
* Requisitos de usuario.
* Requisitos de sistema.
* Matriz de trazabilidad de requisitos.

**Documentos de Diseño:**

Resultados de la resolución y análisis de decisiones.

* Código Fuente.
* Plan de Integración de Software.

**Informes de Revisiones y Pruebas:**

Planes de pruebas (unitarias, integración, sistemas, aceptación de usuario, regresión).

Datos de pruebas y casos de pruebas.

Plan de Instalación/Mantenimiento.

1. Documentos de Manual de Usuario.
2. Plan de Entrega de Servicios.
3. Informes de Investigación.
4. Informes de Estimación.
5. Informes de Cierre del Proyecto.
6. Prototipos.
7. Informes de Métricas.
8. Entregables al Cliente.

**Métricas de Software**

Métricas de Código

1. Cobertura de Código:
   * Definición: Porcentaje del código cubierto por pruebas automatizadas.
   * Objetivo Educativo: Comprender la importancia de las pruebas y cómo afectan la calidad del código.
2. Complejidad Ciclomática:
   * Definición: Medida de la complejidad del código basada en la cantidad de caminos de ejecución posibles.
   * Objetivo Educativo: Concienciar sobre la simplicidad y mantenibilidad del código.
3. Número de Defectos:
   * Definición: Cantidad total de problemas identificados durante revisiones de código y pruebas.
   * Objetivo Educativo: Enseñar la importancia de la revisión y detección temprana de errores.
4. Densidad de Defectos:
   * Definición: Número de defectos por líneas de código.
   * Objetivo Educativo: Destacar la relación entre calidad del código y densidad de defectos.

Métricas de Proceso

1. Tiempo Promedio de Resolución de Problemas:
   * Definición: Promedio de tiempo necesario para abordar y corregir problemas identificados.
   * Objetivo Educativo: Comprender la eficiencia del proceso de resolución de problemas.
2. Frecuencia de Despliegues:
   * Definición: Número de despliegues realizados por semana/mes.
   * Objetivo Educativo: Introducir conceptos de desarrollo continuo y entrega incremental.
3. Eficiencia del Proceso de Desarrollo:
   * Definición: Comparación de estimaciones de tiempo versus tiempo real empleado.
   * Objetivo Educativo: Enseñar la importancia de la planificación y gestión del tiempo.

Métricas de Diseño

1. Acoplamiento:
   * Definición: Medida de la dependencia entre módulos o componentes.
   * Objetivo Educativo: Entender la modularidad y la independencia en el diseño del sistema.
2. Cohesión:
   * Definición: Evaluar la relación entre los elementos internos de un módulo.
   * Objetivo Educativo: Destacar la importancia de la organización interna de los componentes.
3. Complejidad del Diseño:
   * Definición: Evaluar la complejidad de la arquitectura y el diseño del sistema.
   * Objetivo Educativo: Concienciar sobre la simplicidad y claridad en el diseño.

**Proceso de control de cambios**

El proceso de control de cambios está diseñado para gestionar eficazmente las modificaciones que afectan a los productos controlados por configuración en aplicaciones de venta de libros electrónicos. Estas son las etapas del proceso:

**1. Recepción de Solicitud de Cambio:**

* Cuando surge la necesidad de un cambio, ya sea debido a un error identificado, una mejora propuesta o un nuevo requisito, el proceso se inicia enviando una solicitud de cambio.

**2. Registro y Documentación:**

* Las solicitudes de cambio se registran en el sistema de gestión de configuración y registran detalles como la descripción del cambio, el motivo, la prioridad y cualquier información relacionada.

**3. Evaluación Inicial:**

* El Gerente de Configuración y el Gerente de Elementos de Configuración revisarán inicialmente la solicitud para determinar su viabilidad y evaluar el impacto potencial en los productos y el proyecto en general.

**4. Análisis de Impacto y Evaluación Técnica:**

* Se realizará un análisis detallado del impacto del cambio propuesto, tomando en cuenta factores como la complejidad técnica, los recursos requeridos y el cumplimiento de los objetivos del proyecto.

**5. Revisión Formal:**

* Para considerar la solicitud de cambio, se convoca a una reunión formal para evaluar los resultados del análisis de impacto y determinar si se aprueba o rechaza el cambio.

**6. Aprobación o Rechazo:**

* Basándose en la revisión formal, se toma una decisión sobre la aprobación o el rechazo del cambio. Si se aprueba, se pasa a la siguiente etapa.

**7. Modificación de Productos Afectados:**

* Con la aprobación, se procede a implementar el cambio en los productos afectados, siguiendo los procedimientos definidos en el plan de gestión de configuración.

**8. Comunicación de Cambios:**

* Se notifican todas las partes interesadas y personas afectadas sobre la implementación exitosa del cambio, proporcionando detalles sobre las modificaciones realizadas.

**9. Actualización de la Línea Base:**

* Después de la implementación exitosa, se actualiza la línea base para reflejar la nueva configuración de los productos, asegurando la coherencia en el seguimiento de versiones.

**10. Registro y Seguimiento:**

* Cada cambio desde la solicitud hasta la implementación se registra en detalle en el sistema de gestión de configuración. Se realiza un seguimiento continuo para evaluar el impacto a lo largo del tiempo.

**11. Informe de Estado de Configuración:**

* La información sobre los cambios realizados se incluye en informes de estado de configuración, que se mantienen y publican periódicamente para mantener informadas a todas las partes.

**Informe del estado de los elementos de configuración**

Los informes del estado de la configuración son una herramienta esencial para gestionar eficazmente la configuración del software en un proyecto de libro electrónico. Este informe proporcionará información importante para evaluar el progreso del desarrollo y mantener la integridad del sistema. La estructura y funcionamiento de este sistema se muestran a continuación:

**1. Identificación y Recopilación de Información:**

Se identificarán y recopilarán los siguientes tipos de información para respaldar el proceso de gestión de la configuración del proyecto:

* Registro de documentación de configuración aprobada.
* Designación de responsables de elementos de configuración.
* Estado de cambios propuestos y desviaciones de configuración.
* Implementación del estado de los cambios aprobados.
* Configuración de todas las unidades de los elementos de configuración en el inventario.
* Resultados de auditorías de configuración.

**2. Herramienta Automatizada de Soporte:**

Se implementará una herramienta de gestión de configuración automatizada para respaldar la recopilación, el seguimiento y la presentación de informes. Esta herramienta proporcionará características tales como:

* Registro y aprobación de cambios.
* Asignación de responsables de configuración.
* Seguimiento del estado de implementación de cambios.
* Mantenimiento de inventario de configuración.
* Auditorías automatizadas para garantizar la coherencia.

**3. Métricas e Información a Incluir en el Informe:**

El informe de estado de configuración incluirá, entre otras, las siguientes métricas e información:

* Número total de cambios propuestos.
* Estado actual de implementación de cambios aprobados.
* Designación y carga de trabajo de responsables de configuración.
* Inventario actualizado de todas las unidades de configuración.
* Resultados de auditorías, destacando cualquier no conformidad.

**4. Periodicidad y Distribución del Informe:**

Los informes se generarán periódicamente según un cronograma predeterminado (por ejemplo, semanal o mensual) para garantizar que la información se actualice continuamente. El informe se compartirá con todas las partes interesadas, incluido el equipo de desarrollo, los clientes y los responsables de la configuración.

**5. Evaluación Continua y Mejora:**

Se establecerá un proceso de evaluación continua para verificar la eficacia de los informes y herramientas automatizadas. Se realizarán los ajustes necesarios para mejorar la calidad y la idoneidad de los informes para la gestión de la configuración.

**Herramientas**

**Github:** Sistema web de control colaborativo de revisión y desarrollo de software utilizado para alojar proyectos que sean gestionados con Git.

**Definición de Plan de Gestión de la Configuración**

Un plan de gestión de la configuración (CMP) es un documento estratégico que establece políticas, procesos y procedimientos para identificar, controlar, gestionar y validar elementos de configuración durante todo el ciclo de vida del proyecto. Su objetivo principal es garantizar la integridad y coherencia del producto o sistema y facilitar el seguimiento de los cambios durante el desarrollo. Estos son los principales elementos incluidos en PGC:

**1. Objetivo del Plan:**

El objetivo principal de PGC es proporcionar la estructura organizacional y los procesos para definir, controlar y probar los componentes de configuración del proyecto, asegurando la consistencia y calidad del software producido.

**2. Alcance:**

Definir claramente los límites y el alcance del control de la configuración, especificando qué elementos serán controlados y qué actividades serán gestionadas por el plan. Para proyectos de libros electrónicos, el alcance puede incluir código fuente, documentación y productos relacionados.

**3. Identificación de Elementos Configuración:**

Configurar un esquema para identificar de forma única cada elemento de configuración. En el contexto de un proyecto, esto puede incluir convenciones de nomenclatura para versiones, lanzamientos y otras entidades no estándar.

**4. Proceso de control de cambios**

Describir los procedimientos para realizar comprobaciones de configuración. Esto incluye revisar sistemáticamente los elementos de configuración para garantizar que cumplan con regulaciones y estándares específicos. Para mantener la integridad del sistema, se pueden realizar pruebas con regularidad.

**5. Roles y Responsabilidades:**

Definir los roles y responsabilidades de los miembros del equipo relacionados con la gestión de la configuración. Esto puede incluir roles como administrador de configuración, coordinador de configuración y propietario de elementos de configuración junto con sus roles y responsabilidades asociados.

**6. Herramientas**

Proporcionar una lista de herramientas y recursos necesarios para una gestión de configuración eficaz. Esto puede incluir herramientas de control de versiones, sistemas de seguimiento de problemas y cualquier otra herramienta relevante para el proyecto.

**7. Métricas y Evaluación Continua:**

Establecer métricas para evaluar la efectividad del Plan de Gestión de la Configuración y recomendar un proceso de evaluación continua para realizar los ajustes y mejoras necesarios.

**Especificación del Entorno Tecnológico para la Gestión de Configuración**

En esta tarea, se delineará el entorno tecnológico específico que respaldará la gestión de configuración del sistema de información, considerando la automatización de procesos y controles definidos en el plan de gestión de la configuración. El enfoque será garantizar la eficacia y eficiencia en la mecanización de los procesos, proporcionando la base tecnológica necesaria para el control de versiones, gestión de cambios y seguimiento de la configuración.

**1. Revisión del Plan de Gestión de la Configuración:**

Analizar detalladamente el Plan de Gestión de la Configuración proporcionado como entrada para comprender los requisitos específicos y controles establecidos.

**2. Identificación de Componentes Tecnológicos Necesarios:**

En colaboración con el equipo de desarrollo, identificar los componentes hardware y software necesarios para respaldar los procesos de control de versiones, gestión de cambios y auditoría de la configuración.

**3. Selección de Herramientas de Control de Versiones:**

Evaluar y seleccionar una herramienta de control de versiones que se alinee con los requisitos del proyecto. Puede incluir sistemas como Git o SVN, considerando la preferencia del equipo y la compatibilidad con el entorno de desarrollo.

**4. Herramientas de Gestión de Proyectos y Comunicación:**

Seleccionar herramientas de gestión de proyectos y comunicación, como Jira o Trello, para facilitar la colaboración y la asignación de tareas relacionadas con la gestión de configuración.

**5. Automatización de Integración Continua:**

Evaluar y configurar herramientas de integración continua, como Jenkins o GitLab CI, para automatizar la verificación del código y la detección temprana de problemas de integración.

**6. Herramientas de Auditoría y Métricas:**

Seleccionar herramientas de auditoría automatizadas, como SonarQube, y herramientas de presentación de métricas, como Grafana o Kibana, para evaluar la calidad del código y realizar un seguimiento del rendimiento de la gestión de configuración.

**7. Entornos de Desarrollo, Prueba y Producción:**

Establecer entornos utilizando tecnologías como Docker para asegurar ambientes consistentes y reproducibles para desarrollo, prueba y producción.

**8. Seguridad y Control de Acceso:**

Implementar prácticas de seguridad, incluida la autenticación de dos factores, y establecer niveles de acceso controlado en las herramientas utilizadas para garantizar la seguridad de los artefactos y la información de configuración.

**9. Documentación del Entorno Tecnológico:**

Documentar detalladamente el entorno tecnológico especificado, proporcionando información clara sobre las herramientas seleccionadas, configuraciones y flujos de trabajo.

**10. Validación del Entorno Tecnológico:**

Validar el entorno tecnológico con el equipo de desarrollo para asegurar su compatibilidad y eficacia en la gestión de configuración.

**Participantes:**

* Responsable de Gestión de Configuración
* Equipo de Desarrollo

|  |  |
| --- | --- |
| **Aspecto** | **Herramienta/Recurso** |
| **Control de Versiones** | Git |
| **Plataforma de Colaboración** | GitHub |
| **Gestión de Ramas** | Gitflow |
| **Herramienta de Seguimiento de Problemas** | Jira, Redmine |
| **Automatización de Auditorías** | SonarQube, Jenkins |
| **Base de Datos de Configuración** | MySQL, PostgreSQL |
| **Herramienta de Entregas Continuas** | Jenkins, Travis CI |
| **Documentación** | Confluence |
| **Comunicación y Notificación** | Slack, Email |
| **Herramienta de Pruebas** | Selenium, JUnit, Jest |
| **Ambiente de Desarrollo Integrado (IDE)** | IntelliJ IDEA, Visual Studio Code |

**Consideraciones:**

1. **Control de Versiones (Git):** Para el seguimiento de versiones y control de cambios.
2. **Plataforma de Colaboración (GitHub):** Facilita la colaboración y revisión de código.
3. **Herramienta de Seguimiento de Problemas (Jira, Redmine):** Permite gestionar y dar seguimiento a problemas y cambios.
4. **Automatización de Auditorías (SonarQube, Jenkins):** Para garantizar la calidad del código y realizar auditorías automáticas.
5. **Base de Datos de Configuración (MySQL, PostgreSQL):** Almacena información sobre la configuración del proyecto.
6. **Herramienta de Entregas Continuas (Jenkins, Travis CI):** Automatiza el proceso de integración y entrega.
7. **Documentación (Confluence):** Para la creación y almacenamiento de documentación.
8. **Comunicación y Notificación (Slack, Email):** Canales de comunicación eficientes y notificaciones para cambios importantes.
9. **Herramienta de Pruebas (Selenium, JUnit, Jest):** Para realizar pruebas automáticas y garantizar la calidad del software.
10. **Ambiente de Desarrollo Integrado (IDE):** Herramientas para el desarrollo y programación eficiente.