Sistema Integrado de Información

Plan de Gestión de la Configuración

Versión 1.7

Andres Fernando López Avila

Juan Esteban Moreno Rodríguez

Historial de revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 02/Septiembre/2014 | 1.0 | Se da inicio con la plantilla y su organización. | Andres Fernando López Avila. |
| 03/Septiembre/2014 | 1.1 | Se da inicio a la realización de los primeros ítems de la plantilla | Andres Fernando López Avila. |
| 04/Septiembre/2014 | 1.2 | Se da inicio a llenar el punto 2 de la plantilla | Juan Esteban Moreno Rodríguez |
| 04/Septiembre/2014 | 1.3 | Realización del punto estimación del estado de la configuración | Juan Esteban Moreno Rodríguez |
| 04/septiembre/2014 | 1.4 | Realización ítems de la configuración | Juan Esteban Moreno Rodríguez |
| 04/septiembre/2014 | 1.5 | Corrección ortográfica luego de la realización de los ítems anteriores | Andres Fernando López Avila |
| 04/septiembre/2014 | 1.6 | Continuación con la realización de la plantilla, tablas con los productos de trabajo, elemento de configuración e id. | Andres Fernando López Avila |
| 04/septiembre/2014 | 1.7 | Finalización del documento con el desarrollo de todos los campos. | Andres Fernando López Avila |

Tabla de contenidos

1. Introducción 4

1.1 Propósito 4

1.2 Alcance 4

1.3 Definiciones, acrónimos y abreviaturas 4

1.4 Referencias 4

2. Gestión de la Configuración del Software 4

2.1 Organización, responsabilidades e Interfaces 4

2.2 Desarrollo, ambiente e infraestructura 5

3. Programa de Gestión de la Configuración 6

3.1 Identificación de la Configuración 6

3.1.1 Identificación de elementos de configuración 6

3.1.2 Identificación de elementos de configuración y productos de trabajo 6

3.1.3 Línea base del proyecto 7

3.2 Configuración y control de cambios 8

3.2.1 Procedimientos y Aprobaciones para las Solicitudes de Cambios 8

3.2.2 Junta de control de cambios (CCB) 9

3.3 Estimación del Estado de la Configuración 9

3.3.1 Medios de Almacenamientos del Proyecto y Procesos de Distribución 9

3.3.2 Reportes y auditorias 9

4. Hitos 10

5. Formación y Recursos 10

6. Subcontratista y proveedor de software de control 11

Plan de Gestión de la Configuración

# Introducción

## Propósito

En este documento Plan de Gestión de la Configuración (SCM) se describirán las actividades de gestión de configuración de software que deben ser llevadas a cabo durante el proceso de desarrollo del proyecto, nos proporciona una visión general de la organización, actividades, tareas y los objetivos de gestión de la configuración, es una actividad de garantía de calidad que se aplica en todas las fases del proceso del software.

## Alcance

Este Plan de Gestión de la Configuración involucrara todas las fases del proyecto y estará basado en algunos lineamientos que se especificaran:

* El tiempo de realización del proyecto está restringido a 16 semanas, por lo que se necesitaran respuestas rápidas a los cambios.
* El desarrollo del proceso se realizara de manera incremental por lo que se debe llevar un control sobre cada una de las entregas y de los cambios sugeridos, evaluados y aprobados.
* Se elegirán los elementos de configuración en base a las entregas realizadas.

## Definiciones, acrónimos y abreviaturas

* **SCM** (Software Configuration Management) Gestión de Configuración del Software.
* **Línea Base**: Es documento o producto oficial aprobado que sirve como punto de partida para futuras versiones.
* **CI**: Ítem de configuración.
* **SCMR:** (SCM Responsable) Responsable de SCM.
* **SCR:** (System/Software Change Request) Petición de Cambio en el Sistema/Software.
* **CCB:** junta de control de Cambios.

## Referencias

[1] ANSI/IEEE Std 828-1990, IEEE Standard for Software Configuration Management Plans.

[2] CMMI para desarrollo, Versión 1.3 Noviembre de 2010 – [www.sei.cmu.edu](http://www.sei.cmu.edu).

# Gestión de la Configuración del Software

## Organización, responsabilidades e Interfaces

|  |  |
| --- | --- |
| Responsable | Actividad |
| Andres Fernando López Avila | - Revisar todas las peticiones de cambio.  - Asignar las peticiones de cambio aprobadas una fecha de implementación.  - Controlar y facilitar los cambios a las características de un CI.  - Administrar la base de datos de administración de configuración. |
| Juan Esteban Moreno Rodríguez | - Asegurarse de que la acción se toma en las solicitudes de cambio de manera oportuna.  - Mantener el Plan de Gestión de la Configuración.  - Realizar auditorías para verificar el cumplimiento del Plan. |

## Desarrollo, ambiente e infraestructura

Los sistemas de control de versiones (SVC – System Version Control) son herramientas que facilitan la administración de distintas versiones de cada producto que se está desarrollando, brindan la posibilidad de hacer cambios sobre los elementos guardados y al mismo tiempo permite devolver los cambios o guardar la nueva versión de cada uno de estos elementos para no perder el control del proyecto a desarrollar.

Para realizar el proyecto del módulo administrativo se ha elegido como SVC el sistema de control de versiones GIT, son algunas características de esta herramienta las siguientes:

* Manejo de ramas en el repositorio para dar la opción de mezclarlas entre si cuando sea necesario
* Cada una de las nuevas ramas que se van formando y los nuevos cambios van quedando guardados en la nube, por lo que no ocupa mucho espacio en el disco duro.
* Libre y de código abierto

El repositorio del proyecto del módulo administrativo se puede visitar por medio del siguiente enlace:

* <https://github.com/andresflopez13/Proyecto-Software-III>

|  |  |  |
| --- | --- | --- |
| Nombre de Herramienta | Tipo | Descripción |
| Git | CVS | Git es un sistema de control de versiones distribuido de código abierto. |
| Github | Repositorio Remoto | Es la herramienta que se utiliza para alojar los proyectos utilizando el sistema de control de versiones GIT. |
| Mantis | Bug Tracker | Es el sistema que permite realizar las peticiones de cambios al proyecto, también permite notificar los problemas hallados. |
| Netbeans | Entorno de desarrollo | Permite crear los proyectos con JAVA primordialmente pero también se puede instalar varios módulos, para el caso de este proyecto se instala el módulo de PHP que es el lenguaje de programación que se va a utilizar. |
| YII | Framework | YII es el framework elegido para desarrollar el proyecto, es orientado a objetos, lo que ayuda a mejorar la curva de aprendizaje. |
| MySql | SGBD | Es un SGBD multihilo y multiusuario, utilizado para almacenar datos y comunicar modelos. |
| Apache | Servidor Web | Es un servidor web de código abierto usado para desarrollar de forma óptima el proyecto. |

# Programa de Gestión de la Configuración

## Identificación de la Configuración

### Identificación de elementos de configuración

Los elementos de configuración son los que describen como el proyecto y sus artefactos van a ser identificados, numerados, marcados y organizados, la decisión de cuales de los entregables serán elementos de configuración será tomada por el SCMR, esto para que al momento de realizar un cambio o recuperar una versión anterior del sistema no ocurra una incompatibilidad con los otros artefactos manteniendo la integridad del proyecto.

|  |  |
| --- | --- |
| Elemento de configuración | Productos de trabajo |
| Planeación, análisis y diseño | * Plan de administración de riesgos * Plan de aseguramiento de la calidad * Cronograma de actividades * Estructura WBS * Documento de estimación * Plan de medición y análisis * Documento de estrategia SCM del proyecto * Documento para gestión de solicitudes de cambio. * Documento de selección de frameworks * Documento arquitectura de software * Diagrama de secuencia * Diagrama de clases * Diagrama Componentes * Diagrama de despliegue * Prototipos de interface |
| Implementación del proyecto | * Implementación de los casos de uso * Bases de datos * Prototipos * Manuales de usuario |
| Auditorias | * Plan de pruebas * Reporte de defectos |

### Identificación de elementos de configuración y productos de trabajo

|  |  |  |  |
| --- | --- | --- | --- |
| Id | Elemento de configuración | Id | Productos de trabajo |
| PAD | Planeación, análisis y diseño | PAD-001 | Plan de administración de riesgos |
| PAD-002 | Plan de aseguramiento de la calidad |
| PAD-003 | Cronograma de actividades |
| PAD-004 | Estructura WBS |
| PAD-005 | Documento de estimación |
| PAD-006 | Plan de medición y análisis |
| PAD-007 | Documento de estrategia SCM del proyecto |
| PAD-008 | Documento para gestión de solicitudes de cambio. |
| PAD-009 | Documento de selección de frameworks |
| PAD-010 | Documento arquitectura de software |
| PAD-011 | Diagrama de secuencia |
| PAD-012 | Diagrama de clases |
| PAD-013 | Diagrama Componentes |
| PAD-014 | Diagrama de despliegue |
| PAD-015 | Prototipos de interface |
| IDP | Implementación del proyecto | IDP-001 | Implementación de los casos de uso |
| IDP-002 | Bases de datos |
| IDP-003 | Prototipos |
| IDP-004 | Manuales de usuario |
| AU | Auditorias | AU-001 | Plan de pruebas |
| AU-002 | Reporte de defectos |

### Línea base del proyecto

En esta sección se detallaran las líneas bases para el proyecto Sistema de Información de Colciencias, estos serán los artefactos que ya cuentan con un grado de madurez sustancial, deben haber pasado por un proceso de auditoría y estar aprobado por el SQA, también será la base sobre la cual se continuara desarrollando el proyecto y sus cambios tendrán que pasar por un proceso de solicitud de cambio.

|  |  |  |  |
| --- | --- | --- | --- |
| Línea Base | Descripción | | Hito |
| LB-1 | 1.1 | Documento arquitectura de software | 1 |
| 1.2 | Realizaciones de caso de uso |
| 1.3 | Artefacto de trazabilidad |
| LB-2 | 2.1 | Estimación del proyecto final | 2 |
| 2.2 | Plan de medición y análisis |
| 2.3 | Instrumento Excel |
| LB-3 | 3.1 | Estrategia SCM aplicada al proyecto | 3 |
| 3.2 | Cronograma de implementación |
| 3.3 | Estrategia solicitudes de cambio |
| 3.4 | Auditoria interna al proyecto final |
| 3.5 | Primer build del proyecto |
| LB-4 | 4.1 | Funcionalidades | 4 |
| 4.2 | Calidad de código |
| 4.3 | Integración |
| LB-5 | 5.1 | Funcionalidades II | 5 |
| 5.2 | Calidad del código |
| 5.3 | Integración |
| LB-6 | 6.1 | Modelo de pruebas | 6 |
| 6.2 | Casos de prueba |
| 6.3 | Ejecución de pruebas |
| 6.4 | Reportes de pruebas |
| LB-7 | 7.1 | Despliegue | Final |

## Configuración y control de cambios

### Procedimientos y Aprobaciones para las Solicitudes de Cambios

En esta sección se detallara cual serán las actividades de solicitud, evaluación, aprobación e implantación de cambios a los artefactos o a los elementos de las líneas base, estos cambios será para la corrección y mejoramiento de producto.

El procedimiento que se describe a continuación será el que regirá el sistema Integrado de Información para Colciencias cada que se realice una solicitud de cambio al proyecto.

Se entenderá por solicitud de cambio las modificaciones y mejoras que afecten a las líneas bases como:

* Cambios en el diseño.
* Cambios de la arquitectura.
* Cambios en las herramientas de desarrollo.
* Cambios en la documentación.
* Cambios en las pruebas.
* Cambios en la funcionalidad.

**Solicitud:**

Cuando se realiza una solicitud de cambio en la plataforma mantis se creara una nueva instancia del formulario en el cual el interesado llenara todos los campos que se requieren y suministrara toda la información necesaria para poder registrar este cambio en el sistema.

Plataforma mantis para este proyecto: <http://software3gestioncambios.esy.es/mantis/login_page.php>

Se debe ingresar con un usuario y contraseña dispuesto por el administrador de la configuración

**Evaluación:**

En la evaluación será donde se determinara que tan necesario se hace el cambio y cuál sería su costo en cuestión de recursos y tiempo, esto se realizara en dos pasos:

1. Planificación:

* Se revisara la solicitud de cambio para determinar cuál será el alcance.
* Determinar las personas que se verán involucradas.
* Determinar un plan para evaluar dicha solicitud.

1. Valoración:

La valoración se dará por medio del impacto que causara el cambio en la aplicación y puede ser de las siguientes maneras:

* Los riesgos del proyecto.
* Tiempo.
* Especificaciones.

**Aprobación:**

El comité de control de configuración será el encargado de aprobar la solicitud de cambio pero este podrá designar a un integrante de la organización dependiendo del tipo de cambio que se solicita.

Implementación:

Una vez esté finalizada la evaluación del cambio, se decidirá quién será el encargado de implementar, se determinara un tiempo establecido, la forma de llevar un seguimiento y el momento en el que el cambio debe ser integrado junto con toda la aplicación recordando que nos basamos en un proceso de desarrollo incremental y de calidad.

### Junta de control de cambios (CCB)

La junta de control de cambios se realiza con el fin de evaluar y verificar las solicitudes de cambio que se han realizado, los cambios hechos a los artefacto o artefactos nuevos, esto con el fin de avalar cada uno de ellos, analizando cuales serían las consecuencias para el proyecto ya sean positivas o negativas. Esta junta se realizara con los encargados de la gestión de configuración del proyecto, quienes decidirán y verificaran la calidad e integridad de las líneas base del proyecto.

## Estimación del Estado de la Configuración

### Medios de Almacenamientos del Proyecto y Procesos de Distribución

Los medios de almacenamiento que se van a utilizar son varios, el primero de ellos es en los discos duros físicos, cada integrante del equipo debe de tener guardado el proyecto en su versión más reciente en su disco duro físico, el segundo medio de almacenamiento es el repositorio, todos los integrantes del equipo deben de tener acceso a este repositorio para subir cada uno de los cambios o para obtener la versión más actualizada del proyecto, para una mayor seguridad de la información, cada vez que se termine una entrega se hará una copia de seguridad en un CD o DVD.

Según mencionado anteriormente el proceso de distribución será por medio de un canal directo, es decir, se va a poder acceder a todo el proyecto de una forma directa, sin necesidad de ir a terceros, se puede acceder desde un equipo que contenga el proyecto, también desde cualquier lugar con internet al acceder al repositorio y con el CD o DVD se podrá obtener una copia del proyecto.

### Reportes y auditorias

A medida que el proyecto está en desarrollo es importante que este en constante revisión, ya que esto nos ayuda a mejorar la calidad del software, cada vez que el proyecto está dado a revisión se maneja un formato para el reporte, este formato es proporcionado por el software de solicitudes de cambio que se utilizara en este proyecto, llamado mantis el cual incluye el número de defectos que se encontró, que tan crítico es cada uno de los defectos, cuales son las consecuencias de no corregir cada defecto y cual es la estimación de resolver cada uno de los defectos ente otras características o atributos que se debe llenar para poder solicitar el cambio.

El objetivo de cada reporte es la solicitud de cambios a posibles errores o defectos que se hayan encontrado en los artefactos que componen el proyecto, con lo cual se busca una solución rápida y a tiempo impidiendo una situación crítica para el mismo, otro de los objetivos que tienen estos reportes es que sirven para verificar si al momento de hacer modificaciones o cambios que se propusieron, el proyecto no produjo algún error o si quedaron defectos en alguna parte.

Las tareas de auditoria de la configuración son comprobar que haya una secuencia entre cada una de las fases del proyecto, también es que se pueda validar el estado actual del software y que se pueda valorar el avance en una determinada línea base.

Por eso es que se hacen informes y auditorias de forma continua durante todo el proceso de desarrollo del proyecto.

# Hitos

Para un óptimo desarrollo del proyecto se han definidos hitos, a medida que cada uno de estos hitos se vallan realizando este documento va a ir siendo actualizado.

A continuación se muestran en forma de tabla los hitos del proyecto con su respectiva fecha de entrega y los productos de trabajo.

|  |  |  |
| --- | --- | --- |
| Entrega | Fecha entrega | Productos de trabajo |
| 1 | 8/septiembre/2014 | - Documento arquitectura de software  - Realizaciones de caso de uso  - Artefacto de trazabilidad |
| 2 | 17/septiembre/2014 | - Estimación del proyecto final  - plan de medición y análisis  - instrumento Excel |
| 3 | 6/octubre/2014 | - Estrategia SCM aplicada al proyecto  - cronograma de implementación  - Estrategia para gestión de incidencias y solicitudes de cambio  - Auditoria interna al proyecto final  - Primer build del proyecto |
| 4 | 29/octubre/2014 | - Funcionalidades  - calidad de código  - integración |
| 5 | 12/noviembre/2014 | - Funcionalidades II  - Calidad del código  - Integración |
| 6 | 19/noviembre/2014 | - Modelo de pruebas  - casos de prueba  - Ejecución de pruebas  - Reportes de pruebas |
| Final | 26/noviembre/2014 | - despliegue |

# Formación y Recursos

A continuación se describe el personal y la formación que se debe de tener para usar las herramientas anteriormente mencionadas y así cumplir con las actividades propuestas.

**Github:** Para manejar esta herramienta es supremamente necesario que cada una de las personas que la utilicen sepan cómo subir, bajar, registrar y combinar las diferentes versiones, también que sepan utilizar la interfaz gráfica pero al mismo tiempo que utilicen la línea de comandos.

**Mantis:** También es necesario que esta herramienta pueda ser usada por cualquier miembro del equipo de trabajo para corregir los errores e implementar las modificaciones en el menor tiempo posible y por la persona o personas que sean las más adecuadas para trabajar en ello.

**Netbeans:** Al igual que las herramientas mencionadas anteriormente, se necesita que todo el equipo de trabajo conozca muy bien esta herramienta, para instalar los diferentes componentes que hagan falta y para que se agilice la elaboración del código fuente, además de esto también es necesario conocimiento en el lenguaje de programación PHP ya que es el lenguaje usado en la elaboración de todo el proyecto.

**YII:** YII es un framework usado en PHP, por lo tanto se necesita que por lo menos una persona sepa usarlo, conozca sus características para no causar conflictos y para aprovechar al máximo sus características.

**MySql:** Es muy importante saber utilizar esta herramienta para poder realizar cada una de las consultas y para poder crear un buen modelo.

**Apache:** Es indispensable que usemos este servidor para el proyecto, ya que se necesita para poder ver el proyecto como una web, así que por este motivo debe de haber por lo menos una persona que conozca sus servicios y como funciona para que no hayan problemas.

# Subcontratista y proveedor de software de control

A continuación se especifican diferente software que se utilizan en el proyecto desarrollado fuera del entorno del proyecto:

* Github: para instalar esta herramienta solo es necesario ejecutar el instalador e ingresar una cuenta de usuario valida. Es de código abierto, por lo tanto se puede utilizar sin problemas de licencia.
* YII: antes de usar yii hay que configurar las variables de entorno y descargar la versión que se desea utilizar, también es libre.
* MySql: Para utilizarlo en el proyecto hay que descargar el instalador y ejecutarlo, seguir la instalación y dar finalizar, es código libre por lo tanto tampoco hay problemas de licencias con esta aplicación.
* Apache: la forma que se usa para instalar apache en este proyecto es por medio de una herramienta llamada xammp, es fácil de implementar y al igual que las otras aplicaciones es libre y no hay problemas con las licencias.