GUÍA de SCMP

**(Software Configuration Management Plan)**

**INDICE**

[**Introducción a la Guía de SCMP**](#h.gjdgxs)

[**1.** **Introducción**](#h.30j0zll)

[1.1. Propósito](#h.1fob9te)

[1.2. Alcance](#h.3znysh7)

[1.3. Definiciones](#h.2et92p0)

[1.4. Referencias](#h.tyjcwt)

[**2.** **Gestión de Configuración del Software (SCM)**](#h.3dy6vkm)

[2.1. Organización de SCM](#h.1t3h5sf)

[2.2. Responsabilidades de SCM](#h.4d34og8)

[2.3. Políticas, Directrices y procedimientos](#h.2s8eyo1)

[2.4. Herramientas, entorno e Infraestructura](#h.17dp8vu)

[2.5. Calendario o cronograma](#h.3rdcrjn)

[**3.** **Actividades de la Gestión de Configuración del Software (SCM)**](#h.26in1rg)

[3.1. Identificación de la configuración](#h.lnxbz9)

[3.1.1. Identificación de los ítems de configuración](#h.35nkun2)

[3.1.2. Nomenclatura de los ítems de configuración](#h.1ksv4uv)

[3.2. Control de configuración](#h.44sinio)

[3.2.1. Solicitud de cambios](#h.2jxsxqh)

[3.2.2. Evaluación de cambios](#h.z337ya)

[3.2.3. Aprobación o desaprobación de cambios](#h.3j2qqm3)

[3.2.4. Implementación de los cambios](#h.1y810tw)

[3.3. Estado de la configuración](#h.4i7ojhp)

[3.4. Auditorías de configuración](#h.2xcytpi)

[3.5. Gestión y entrega de Release de Software](#h.1ci93xb)

**Introducción a la Guía de SCMP**

1. **Introducción**

* 1. **Propósito**

FISUM es una consultora dedicada al desarrollo y mantenimiento de aplicaciones web y móvil para empresas PYMES.

Actualmente en FISUM no existe un proceso claro y establecido para el control de versión y entregables para los clientes. Al establecer una nueva versión de un artefacto, se crea un artefacto completamente nuevo. Esto conlleva a la existencia múltiples artefactos con el mismo nombre y mismo propósito pero con diferente grado de avance durante el ciclo de desarrollo del proyecto. Esta forma no estandarizada de control de versiones ha llegado a ocasionar que versiones erróneas de artefactos sean agregadas a una línea base de un proyecto siendo desarrollado por FISUM, lo cual repercute negativamente en la reputación de la consultora.

El implementar un plan de gestión de configuración que se aplique a todos los proyectos siendo desarrollados por FISUM impactará positivamente a la consultora. Se obtendrá un registro histórico del cambio de los artefactos a lo largo del proceso de desarrollo de los proyectos, lo cual disminuiría considerablemente el tiempo de respuesta ante una petición de cambio; incrementará considerablemente la calidad de los procesos de desarrollo de los proyectos, permitiendo la identificación rápida de versiones estables y aprobadas de los artefactos de este.

A su vez , en caso de elaborar un producto que pueda ser reutilizado para diferentes clientes , se obtendrá un registro de las versiones que poseen dichos clientes , lo cual ayudará al correcto mantenimiento de dichas versiones sin tener que comprometer a otros clientes.

Se estima que una correcta implementación del Plan de Gestión de la Configuración dentro de FISUM genere reducción de costos de producción ya que se reducirán considerablemente los errores de versiones en entregas a clientes, lo cual implica menor tiempo de desarrollo, una mayor calidad en el producto final y la posibilidad de poder reusar eficientemente un proyecto (aplicación web o móvil) para un nuevo cliente si sus necesidades lo ameritan.

* 1. **Aplicabilidad**

El plan de gestión de la configuración será aplicado en todos los proyectos de software que la consultora FISUM  a lo largo de todas las fases de desarrollo de dichos proyectos (también incluyendo el mantenimiento y la gestión de cambios en los requerimientos del proyecto).

* 1. **Alcance y Gobierno**

El gerente de configuración de la consultora es el encargado de elaborar, junto a los demás gerentes y el equipo de desarrollo, el plan de gestión de la configuración.

El gerente de cada proyecto es el responsable de asegurar que dicho proyecto se rija bajo las guías y mandatos establecidos en este plan.

* 1. **Definiciones**

FISUM: nombre de la consultora.

SCRUM: metodología ágil que ayudará a desarrollar de manera iterativa y eficaz cada proyecto que se realice.

* 1. **Referencias**

[1] Sommerville, I. Software engineering, 11end. (2011)

[2] Hass, G. (2003). Configuration management principles and practice. Addison-Wesley Longman Publishing Co., Inc..

1. **Gestión de Configuración del Software (SCM)**

* 1. **Organización de SCM**

De acuerdo a los procesos que se realizan en la SCM, se genera un marco distribuido en el cual durante cada fase por la que el sistema pasa cada uno de esos procesos también se desarrolla ya sea en gran o pequeña parte.

* Planeamiento de la Gestión de la SCM
* Identificación de la SCM
* Control de la SCM
* Estado de Contabilidad de la SCM
* Auditoria de la SCM
* Gestión y Entrega de las Releases del Software

Se mostrara una imagen resaltando cada estado del proceso, las dividiremos según las Phases de la metodología RUP:

* Iniciación

*[Grafico 1]- Porcentaje del proceso de Iniciación*

* Elaboración

*[Grafico 2]- Porcentaje del proceso de Elaboración*

* Construcción

*[Grafico 3]- Porcentaje del proceso de Construcción*

* Transición

*[Grafico 4]- Porcentaje del proceso de Transición*

A continuación se mostrará los procesos explicados gráficamente según el diagrama R.U.P.



* 1. **Responsabilidades de SCM**

Las responsabilidades y roles que se desarrollarán en la empresa serán las siguientes:

|  |  |  |
| --- | --- | --- |
| **Nombre de Rol** | **Responsabilidades** | **Nivel de Autoridad** |
| **Gerente de Proyecto** | Supervisar el funcionamiento de la Gestión de la Configuración | Máxima autoridad |
| **Gerente de Configuración** | Ejecutar todas las tareas de Gestión de la Configuración | Autoridad para operar las funciones de la Gestión de la Configuración |
| **Bibliotecario** | Brindar acceso a los miembros del equipo a los ítems de configuración | Autoridad para limitar los accesos a la información de calificada del sistema |
| **Comité de Control de Cambios** | Evaluar el impacto que tendría la implementación de un RFC en el correcto funcionamiento y estructura del proyecto | Autoridad sobre los desarrolladores y los planes de mantenimiento y desarrollo en los sistemas a desarrollar |
| **Desarrollador** | Implementar los requerimientos establecidos por los clientes para cada proyecto | Autoridad para realizar cambios previamente revisados por el Comité de Control de Cambios |

*[Tabla 1]- Listado de roles de gestión de configuración*

* 1. **Políticas, Directrices y procedimientos**

**Políticas:**

* Es obligación de cada integrante reportar dificultades si se le presentan en el camino.
* Es indispensable en el equipo la elección de una persona que se encargue de realizar las revisiones de los avances.
* Es obligación del equipo realizar las coordinaciones necesarias para llevar adelante el proyecto.

**Directrices:**

* Se establecen los roles de administradores a todos los integrantes del proyecto.
* Los integrantes del proyecto tienen todos los permisos de lectura y escritura sobre el repositorio.

**Procedimientos:**

* Habrá un encargado de establecer la versión de producción mediante la recopilación y revisión previa de los avances subidos.
* Para establecer una versión estable el producto debe pasar todos los casos de prueba establecidos en el entorno del mismo nombre.
  1. **Herramientas, entorno e Infraestructura**

**Herramientas:**

Para realizar la gestion de la configuracion se utilizara las siguientes herramientas, en las cuales se capacitara al personal de la organización:

* **GIT**: Es un software de control de versiones diseñado por Linus Torvalds, pensando en la eficiencia y la confiabilidad del mantenimiento de versiones de aplicaciones cuando éstas tienen un gran número de archivos de código fuente. Al principio, Git se pensó como un motor de bajo nivel sobre el cual otros pudieran escribir la interfaz de usuario o front end como Cogito o StGIT. Sin embargo, Git se ha convertido desde entonces en un sistema de control de versiones con funcionalidad plena.
* **Github**: Es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git. El código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago.
* **Jenkins**: Es un software de Integración continua open source escrito en Java. Está basado en el proyecto Hudson y es, dependiendo de la visión, un fork del proyecto o simplemente un cambio de nombre.

Jenkins proporciona integración continua para el desarrollo de software. Es un sistema corriendo en un servidor que es un contenedor de servlets, como Apache Tomcat. Soporta herramientas de control de versiones como CVS, Subversion, Git, Mercurial, Perforce y Clearcase y puede ejecutar proyectos basados en Apache Ant y Apache Maven, así como scripts de shell y programas batch de Windows

* **AWS**:Amazon Web Services (AWS abreviado) es una colección de servicios de computación en la nube (también llamados servicios web) que en conjunto forman una plataforma de computación en la nube, ofrecidas a través de Internet por Amazon.com. Es usado en aplicaciones populares como Dropbox, Foursquare, HootSuite. Es una de las ofertas internacionales más importantes de la computación en la nube y compite directamente contra servicios como Microsoft Azure y Google Cloud Platform. Es considerado como un pionero en este campo.

**Entornos**:

Se trabajara bajo los siguientes entornos, en cada uno se detalla la forma de trabajo.

**Desarrollo**: Entorno orientado exclusivamente al desarrollo y diseño de

nuevas clases de proceso. Al estar ubicado en instalaciones independientes de las de Producción, se garantiza su independencia hasta que sean comprobados en el Entorno de Pruebas antes de sincronizarlos con el de Producción.

**Pruebas**. Entorno donde se comprueban y certifican los nuevos desarrollos antes de pasarlos al Entorno de Producción.

**Producción**. Es en este entorno donde los usuarios trabajan diariamente en los procesos BPM introduciendo y consultando los datos reales de la organización.

**Arquitectura**



* 1. **Calendario o cronograma**

|  |  |  |  |
| --- | --- | --- | --- |
| HITO O EVENTO SIGNIFICATIVO | Duración | FECHA PROGRAMADA | Responsable |
|  | **35 días** | **28/03/2016 al 17/06/2016** |  |
| Introducción | **4 días** | **28/03/2016 al 31/03/2016** |  |
| Definir el estado actual | 1 día | 28/03/2016 | Gestor de la Configuración |
| Definir el propósito | 1 día | 29/03/2016 | Gestor de la Configuración |
| Definir el alcance | 1 día | 30/03/2016 | Gestor de la Configuración |
| Definir los términos utilizados | 1 día | 31/03/2016 | Gestor de la Configuración |
| Gestión del Proceso de Gestión de la Configuración | **6 días** | **02/05/2016 al 07/05/2016** |  |
| Definir el organigrama | 1 día | 02/05/2016 | Gestor de la Configuración |
| Definir los roles y responsabilidades | 1 día | 03/05/2016 | Gestor de la Configuración |
| Definir las políticas y procedimientos aplicables | 1 día | 04/05/2016 | Gestor de la Configuración |
| Definir la herramienta, entorno e infraestructura | 1 día | 05/05/2016 | Gestor de la Configuración |
| Elaboración del cronograma | 2 día | 06/05/2016 al 07/05/2016 | Gestor de la Configuración |
| Actividades de Gestión de la Configuración | **25 días** |  |  |
| Identificación de Elementos de Configuración | **6 días** | **09/05/2016 al 14/05/2016** |  |
| Identificar elementos de configuración | 3 día | 09/05/2016 al 11/05/2016 | Bibliotecario |
| Definir la nomenclatura de la identificación | 1 día | 12/05/2016 | Bibliotecario |
| Elaborar lista de ítems con nomenclatura | 2 día | 13/05/2016 al 14/05/2016 | Bibliotecario |
| Mantenimiento y Control de la Gestión de la Configuración | **5 días** | **23/05/2016 al 27/05/2016** |  |
| Elaborar definición de Líneas Base | 2 día | 23/05/2016 al 24/05/2016 | Gestor de la Configuración |
| Elaborar la definición del formato de solicitud de cambio | 1 día | 25/05/2016 | Comité de Control de Cambios |
| Elaborar Plan de Gestión de Cambios | 2 día | 26/05/2016 al 27/06/2016 | Comité de Control de Cambios |
| Informe de Estado de la Configuración | **4 días** | **31/05/2016 al 03/06/2016** |  |
| Elaborar la Definición de reportes para el Estado | 2 días | 31/05/2016 al 01/06/2016 | Gestor de la Configuración |
| Mantener y actualizar el estado de los elementos de la configuración | 2 días | 02/06/2016 al 03/06/2016 | Gestor de la Configuración |
| Verificación y Auditoría | **5 días** | **06/06/2016 al 10/06/2016** |  |
| Elaborar informe de auditoría de Gestión de la Configuración | 5 días | 06/06/2016 al 10/06/2016 | Auditor |
| Gestión de Versiones | **5 días** | **13/06/2016 al 17/06/2016** |  |
| Elaborar la Configuración del Paquete de Liberación | 1 día | 13/06/2016 | Bibliotecario |
| Mantener la librería actualizada | 2 día | 14/06/2016 al 15/06/2016 | Bibliotecario |
| Elaborar el Documento de Liberación | 2 días | 16/06/2016 al 17/06/2016 | Bibliotecario |

1. **Actividades de la Gestión de Configuración del Software (SCM)**
   1. **Identificación de la configuración**

La primera fase de la gestión de la configuración de software es la identificación de los ítems (elementos) de configuración (los artefactos que son sujetos a todo el proceso de SCM) esta es una fase crucial en el proceso de SCM ya que una mala identificación de los ítems de configuración puede traer mayores costos (bien por gestionar la configuración de artefactos de software “*poca relevancia*” al ciclo de vida del software o por no gestionar aquellos artefactos de relevancia). Los ítems de configuración identificados serán gestionados en todas las demás fases del proceso de SCM, para facilitar la gestión en esta fase se define la clasificación de cada ítem como un identificador (el cual es definido por una nomenclatura según la clasificación dada a dicho ítem).

La fase de identificación de la configuración está compuesta de la identificación de los ítems de configuración, la clasificación de ellos, la definición de la nomenclatura de identificación según la clasificación definida y la aplicación de la nomenclatura en cada ítem de configuración.

* 1. **Clasificación de los ítems de configuración**

En esta actividad se identifica los ítems (elementos) de configuración, esto es aquellos artefactos involucrados en el ciclo de vida del software, que por su importancia, deben ser gestionados los cambios y versiones que este posee. No todos los artefactos de software son parte de los elementos de la configuración esto se debe a que el equipo de gestión de configuración solo considera aquellos artefactos que necesitan ser gestionados (aquellos que mayormente tiene gran importancia en el desarrollo y mantenimiento del software, aquellos que están sujetos a un alto grado de frecuencia, entre otros motivos), ya que considerar todo artefacto como un ítem involucra mayores costos.

El equipo de gestión de configuración teniendo claro estos aspectos por un mutuo acuerdo, con el mismo y con el equipo de desarrollo de cada proyecto, ha definido los ítems de configuración. Estos ítems se muestran en la Tabla 2.

|  |  |  |  |
| --- | --- | --- | --- |
| Tipo de ítem (E=Evolución  F=Fuente  S=Soporte) | Nombre del ítem | Fuente  (E= Empresa  P= Proyecto  C= Cliente  V=Proveedor) | Proyecto |
| E | Plan del proyecto | P | SISLAB |
| E | Cronograma del proyecto | P | SISLAB |
| E | Documento de negocio | P | SISLAB |
| E | Lista de Requisitos | P | SISLAB |
| E | Especificación del caso de uso Mantenimiento de laboratorio | P | SISLAB |
| E | Especificación del caso de uso  Reparación de laboratorio | P | SISLAB |
| E | Especificación del caso de uso  Abastecimiento de componentes | P | SISLAB |
| E | Especificación del caso de uso  Reservar laboratorio para evento | P | SISLAB |
| E | Especificación del caso de uso  Apertura de laboratorio | P | SISLAB |
| E | Especificación del caso de uso  Cierre de Laboratorio | P | SISLAB |
| E | Especificación del caso de uso  Admisión a Laboratorio Libre | P | SISLAB |
| E | Especificación del caso de uso Salida de Laboratorio Libre | P | SISLAB |
| E | Documento de análisis | P | SISLAB |
| E | Documento de diseño | P | SISLAB |
| E | Especificación de casos de prueba | P | SISLAB |
| E | Documento de arquitectura de software | P | SISLAB |
| E | Documento de despliegue | P | SISLAB |
| F | Script de generación de la base de datos | P | SISLAB |
| F | Código fuente | P | SISLAB |
| S | Spring Framework | V | SISLAB |
| S | Gestor de base de datos relacional MySQL | V | SISLAB |
| E | Plan del proyecto | P | DSO |
| E | Cronograma del proyecto | P | DSO |
| E | Documento de negocio | P | DSO |
| E | Lista de Requisitos | P | DSO |
| E | Especificación del caso de uso Mantenimiento de datos de empleado | P | DSO |
| E | Especificación del caso de uso Mantenimiento de datos de organización | P | DSO |
| E | Especificación del caso de uso Gestión de responsabilidad entre proveedor y cliente | P | DSO |
| E | Documento de análisis | P | DSO |
| E | Documento de diseño | P | DSO |
| E | Especificación de casos de prueba | P | DSO |

*[Tabla 2]- Listado de ítems de configuración*

1. **Nomenclatura de los ítems de configuración**

En esta sección se especifican la identificación y descripción única de cada elemento de configuración.

Además se especifica como se distinguirán las diferentes versiones de cada elemento.

Para todos los elementos de configuración se les deberá agregar, después del nombre del mismo, información acerca del grupo al que corresponde el elemento y la versión del mismo.

La nomenclatura a seguir ha sido acordada por los miembros del equipo de configuración, el cual es la siguiente:

|  |  |  |  |
| --- | --- | --- | --- |
| Tipo de ítem de configuración | | Nomenclatura | Ejemplo |
| Ítem en evolución | Documentos generales (no específicos a un proyecto) | {Abreviatura del documento} (esto es la primera letra de cada palabra del nombre del documento) | Para el plan de gestión de la configuración, la nomenclatura seria GCS |
| Documentos asociados a un proyecto pero no a un componente en especifico | {Abreviatura del proyecto} \_{Abreviatura documento} | Para el documento del negocio del proyecto sistema de laboratorio (SISLAB), la nomenclatura seria SISLAB\_DN |
| Documentos asociados a un proyecto y aun componente en específico | {Abreviatura del proyecto}\_{Abreviatura del componente}\_{Abreviatura del documento} | Para el documento especificación del caso de uso Generar reporte de gastos mensuales del componente Tesorería del proyecto FISIERP, la nomenclatura seria: FISIERP\_T\_GRGM |
| Archivos ejecutables | {Nombre del archivo}v{Versión}.{Revisión}{Carácter de actualización}\*  \*El carácter de actualización seguirá el orden alfabético de acuerdo al número de veces que ha sido actualizado. | Para el componente Bootstrap con versión 3 con 6 revisiones y 2 actualizaciones de bugs, la nomenclatura seria: Bootstrapv3.6b |
| Ítem fuente | | {Nombre del ítem} | Para el controlador UsuarioController, la nomenclatura será: UsuarioController |
| Ítem de soporte | | {Nombre del ítem}v{Versión}.{Revisión}{Carácter de actualización}\* | Para el componente Bootstrap con versión 3 con 6 revisiones y 2 actualizaciones de bugs, la nomenclatura seria: Bootstrapv3.6b |

*[Tabla 3]- Nomenclatura para cada ítem de configuracion*

Se aplica la nomenclatura, indicada arriba, para cada entregable en el modelo de proceso:

|  |  |  |
| --- | --- | --- |
| Nomenclatura | Entregable | Fase del proceso de desarrollo de software |
| SISLAB\_PP | Plan del proyecto | P |
| SISLAB\_CP | Cronograma del proyecto | P |
| SISLAB\_DN | Documento de negocio | P |
| SISLAB\_LR | Lista de Requisitos | P |
| SISLAB\_ECUML | Especificación del caso de uso Mantenimiento de laboratorio | P |
| SISLAB\_ECURL | Especificación del caso de uso  Reparación de laboratorio | P |
| SISLAB\_ECUAC | Especificación del caso de uso  Abastecimiento de componentes | P |
| SISLAB\_ECURLE | Especificación del caso de uso  Reservar laboratorio para evento | P |
| SISLAB\_ECUAL | Especificación del caso de uso  Apertura de laboratorio | P |
| SISLAB\_ECUCL | Especificación del caso de uso  Cierre de Laboratorio | P |
| SISLAB\_ECUALL | Especificación del caso de uso  Admisión a Laboratorio Libre | P |
| SISLAB\_ECUSLL | Especificación del caso de uso Salida de Laboratorio Libre | P |
| SISLAB\_DA | Documento de análisis | P |
| SISLAB\_DD | Documento de diseño | P |
| SISLAB\_ECP | Especificación de casos de prueba | P |
| SISLAB\_DAS | Documento de arquitectura de software | P |
| SISLAB\_DDS | Documento de despliegue del sistema | P |
| SCRIPT | Script de generación de la base de datos (script) | P |
| SOURCE | Código fuente (source) | P |
| SpringFramework v4.2.6 | Spring Framework | V |
| MysqlCommunityServer v5.7.12 | Gestor de base de datos relacional MySQL Community Server | V |
| DSO\_PP | Plan del proyecto | P |
| DSO\_CP | Cronograma del proyecto | P |
| DSO\_DN | Documento de negocio | P |
| DSO\_LR | Lista de Requisitos | P |
| DSO\_ECUMDE | Especificación del caso de uso Mantenimiento de datos de empleado | P |
| DSO\_ECUMDO | Especificación del caso de uso Mantenimiento de datos de organización | P |
| DSO\_ECUGRPC | Especificación del caso de uso Gestión de responsabilidad entre proveedor y cliente | P |
| DSO\_A | Documento de análisis | P |
| DSO\_DD | Documento de diseño | P |
| DSO\_ECP | Especificación de casos de prueba | P |

*[Tabla 4]- Nomenclatura de cada ítem a entregar*

* 1. **Control de configuración**

En esta etapa describiremos como será gestionado los cambios al sistema, dado que el cambio es un hecho vital en el desarrollo del software. Los cambios pueden darse por los siguientes motivos:

* Los clientes desean modificar los requerimientos.
* El equipo de desarrollo, el cliente o un usuario encuentra una falla (bug) que desea que sea solucionada.
* El usuario buscar actualizar el sistema debido a las nuevas versiones de los componentes (la plataforma, librerías, entre otros) que forman el sistema [1].

La causa de estas modificaciones se debe a que, a medida que transcurre el tiempo, se adquieren más conocimientos, este conocimiento adicional es la fuerza motriz de la mayoría de cambios.

El objetivo de este proceso de Control de configuración es estar apto ante cualquier cambio que podría producirse en cualquier momento del ciclo de vida del software. Siendo esta una actividad que garantiza la calidad de software en todas las fases del proceso de ingeniería de software.

**Línea Base y Elementos de Configuración del Software**

Dado que los cambios suceden en cualquier momento en el ciclo de vida de software es necesario definir puntos de control, si estos no son definidos no se podría conocer cómo se encuentra el software actual respecto a un periodo pasado (e.g. hace un mes, hace 5 meses), como se estuvo hace un periodo pasado, entre otros beneficios.

Esta parte del proceso se encarga de definir esos puntos de control, que son conocidos como líneas base. Una línea base es un artefacto del Control de configuración que nos ayuda a definir y estipular los hitos que sean necesarios en el ciclo de vida del software para que de esta manera se pueda llevar un control.

Cada línea base contiene un conjunto de ítems de configuración controlados por la misma. Las líneas base para cada proyecto de nuestra consultora se muestran en la Figura 1.

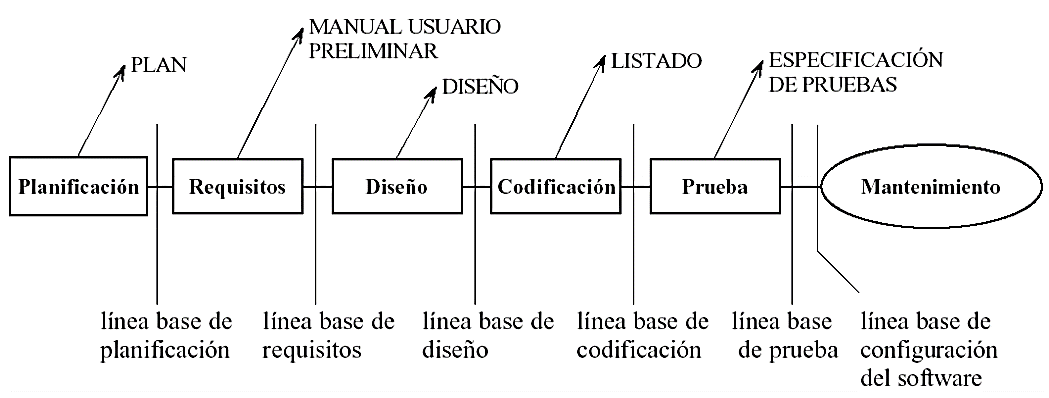


Figura Líneas base de cada proyecto de desarrollo de software

Las actividades de desarrollo posteriores a la creación de una línea base muchas veces generan cambios en la misma, cuando esto suceda se requerirá una revisión formal y una justificación de todas las modificaciones del documento (control de cambios).

Un elemento de configuración del software (ECS) es la información creada como parte del proceso de ingeniería del software. Los siguientes ECS son el objeto de las técnicas de gestión de configuraciones y forman un conjunto de líneas base

1) Especificación del sistema

2) Plan del proyecto software

3) a) Especificación de requerimientos del software

b) Prototipo ejecutable o en papel

4) Manual de usuario preliminar

5) Especificación de diseño:

a) Diseño preliminar

b) Diseño detallado

6) Listados del código fuente

7) a) Planificación y procedimiento de prueba

b) Casos de prueba y resultados registrados

8) Manuales de operación y de instalación

9) Programas ejecutables

10) Manual de usuario

11) Documentos de mantenimiento

a) Informes de problemas del software

b) Peticiones de mantenimiento

c) Órdenes de cambios de ingeniería

12) Estándares y procedimientos de ingeniería del software

Los ítems de configuración para cada línea base con sus respectivos hitos son mostrados en la Tabla 1.

|  |  |  |
| --- | --- | --- |
| Línea Base | Evento/Hito | Ítems de Configuración |
| Línea Base Funcional | | |
| Línea base de la Planificación | Fin fe fase de Planificación. | -Casos del Negocio.  -Documentos del Negocio.  -Análisis de impacto.  -Proyect Charter.  -Análisis preliminar de requisitos. |
| Línea Base de Instanciación | | |
| Línea base de la especificación de requisitos | Fin de la segunda iteración de la lista de requisitos. | -Documento de Lista de requisitos – Primera Iteración.  -Documento de Lista de requisitos – Segunda Iteración. |
| Configuración durante el desarrollo | | |
| Línea base de diseño | Fin de análisis del diseño. | -Diseño preliminar.  -Diseño detallado.  -Especificación de casos de uso.  -Documento de análisis.  -Matriz de Trazabilidad. |
| Línea base de codificación | Fin del desarrollo del sistema. | -Código Fuente.  -Documentación del código fuente.  -Casos de Prueba por casos de uso.  -Resultado de pruebas unitarias. |
| Línea base de pruebas | Fin de casos de prueba. | -Documento de la ejecución de los casos de prueba.  -Documento de conformidad. |
| Línea Base del Producto | | |
| Línea base de mantenimiento | Fin de la entrega y mantenimiento. | -Documento de arquitectura.  -Documento de despliegue.  -Presentación del sistema.  -Manual de instalación.  -Manual de usuario. |

Tabla Líneas base con sus respectivos ítems de configuración y hitos

**Estructura de Librerías**

Una de los principales objetivos de la gestión de la configuración es asegurar que el almacenamiento de un ítem no desaparecerá o será dañado, además que este pueda ser encontrado en cualquier momento y entregado en la condición en la que se espera que este, y ese registro debe ser mantenido para indicar a quien se le ha entregado un ítem o una copia de ella [2].

El almacenamiento es físico. Los ítems que son almacenados están físicamente presentes en un lugar específico. A este lugar se le conoce como librerías.

Las librerías para cada proyecto de FISUM están divididas como se muestra en la Figura 2.



Figura Estructura de las librerías para cada proyecto

* Librería Clientes

Librería donde se almacena la información en la que el cliente / usuario puede colaborar con cierta información, por ejemplo: los requisitos necesarios del cambio a realizar, detalles de interfaz, conocimientos básicos del sistema, etc.

**Responsable**

Bibliotecario

**Actividades**

Brindar acceso a los clientes a los ítems de configuración.

• Leer

• Escribir

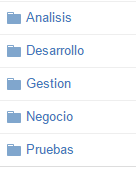
• Ejecutar

• Eliminar

**Contenido**

* Plan de proyecto.
* Cronograma de proyecto.
* Lista de requisitos.
* Documento de negocio.
* Documento de análisis.
* Documento de diseño.
* Librería De Trabajo

Librería donde se almacenan los documentos que están relacionados al desarrollo del control de cambio. Se distribuye en las etapas del proyecto.



**Responsable**

Arquitecto de Software

**Actividades**

Check in y Check out de ítems pertenecientes a la librería.

Arquitecto:

• Leer

• Escribir

• Ejecutar

• Eliminar

Desarrollador:

• Leer

• Escribir

• Ejecutar

**Contenido**

* Especificación de casos de uso Mantenimiento de Laboratorio.
* Especificación de casos de uso Abastecimiento de componentes.
* Especificación de casos de uso Apertura de Laboratorio.
* Especificación de casos de prueba.
* Documento de análisis.
* Documento de diseño.
* Documento de arquitectura de Software
* Código Fuente.
* Spring Framework.
* Script de generación de base de datos.
* Documento de despliegue.
* Librería QA

En esta librería de colocaran los documentos relacionados a la Calidad que presente el software, y sea en una etapa productiva o en un ambiente de calidad.

Al probar el software en un ambiente de calidad se ejecutarán casos de prueba más específicos y rigurosos, los defectos que se muestren quedaran registrados, al finalizar los casos de prueba el personal de Calidad enviará los defectos totales y se corregirá el componente defectuoso para nuevamente subir a Calidad. En caso no se encontraran defectos el componente de software será subido a producción con la firma y autorización del Gerente de Proyecto, Gerente de Configuración y del Comité de Control de Cambios.

**Responsable**

Comité de Control de Cambios y Personal de QA.

**Actividades**

Crear nuevos casos de prueba y documentarlos en la librería, los defectos también deben documentarse respectivamente.

Personal de QA:

• Leer

• Escribir

• Ejecutar

• Eliminar

Desarrollador:

• Leer

• Escribir

• Ejecutar

**Contenido**

* Especificación de Casos de Prueba - QA
* Documento de Caso 1.
* Documento de Caso 2.
* Documento de Caso 3.
* Documento de Caso 4.
* Defecto de Caso 1.
* Defecto de Caso 2.
* Hoja de Pase a producción.

**Solicitud de Cambio – RFC**

Al momento de solicitar un cambio, se debe se llenar la solicitud de cambio y enviársela al comité de Control de Cambios para su evaluación (Ejemplo: Plantilla – Solicitud de Cambio). Ellos determinaran si el cambio se realizará o no, dependiendo de la importancia e impacto.

En caso de la que solicitud sea aceptada se completará el Acta de Implantación (Ejemplo: Plantilla – Acta de Implantación), seguido se eso se implementará el cambio solicitado.

* + 1. **Solicitud de cambios**

<se indican los procedimientos que serán seguidos para realizar cambios en las líneas base, desde la solicitud del cambio hasta su aprobación, describiendo los documentos que serán generados en las distintas instancias del procedimiento de cambios y adjuntando el formato que tendrán dichos documentos. >

* + 1. **Evaluación de cambios**

< se indican los procedimientos para hacer la evaluación de un cambio solicitado, una vez recibida una solicitud de cambio se debe considerar el impacto que este producirá en el proyecto >

* + 1. **Aprobación o desaprobación de cambios**

< se indican las responsabilidades asignadas en el proceso de control de cambios, quien o quienes estudiarán y aprobarán las solicitudes de cambio, en general las responsabilidades están asociadas a los productos afectados. >

* + 1. **Implementación de los cambios**

< se describe como se implementará un cambio aprobado, incluyendo la información de la solicitud del cambio, los nombres y versiones de los items de configuración afectados, fecha de instalación e identificador de la nueva versión generada. >

* 1. **Estado de la configuración**

< se describen los reportes de configuración que serán realizados, el tipo, frecuencia, información que contendrán y control de acceso. >

Para este curso se utilizarán las facilidades que provee CVS para reportar el estado de los items de configuración, los que brindan información sobre la historia de las revisiones, fechas, personas responsables y numeros de revisión asignados.

* 1. **Auditorías de configuración**

< se describen las auditorías que serán realizadas sobre los items de configuración para determinar que los mismos son consistentes. Se debe incluir la información indicada en el >

* 1. **Gestión y entrega de Release de Software**