

Plan de Gestión de Configuración

2017

ÍNDICE

[1. Introducción 3](#_Toc494450685)

[2. Gestión de Configuración 3](#_Toc494450686)

[2.1. Organización 3](#_Toc494450687)

[2.2. Roles o responsabilidades 4](#_Toc494450688)

[2.3. Políticas, Directrices y procedimientos 5](#_Toc494450689)

[2.4. Herramienta, entorno e infraestructura 5](#_Toc494450690)

[Herramientas de control de versiones 6](#_Toc494450691)

[Herramientas de entorno 7](#_Toc494450692)

[2.5. Calendario 7](#_Toc494450693)

[3. Actividades de la SCM: 9](#_Toc494450694)

[3.1. Identificación de la configuración 9](#_Toc494450695)

[3.1.1. Ítems de configuración: 9](#_Toc494450696)

[3.1.2. Ítems de configuración: 9](#_Toc494450697)

[3.1.3. Lista de los ítems con Nomenclatura 11](#_Toc494450698)

[3.2. Control de los ítems de la configuración 11](#_Toc494450699)

Plan de Gestión de la Configuración

# Introducción

El propósito del presente documento es la brindar una alternativa para el control de versiones de los proyectos de la empresa, actualmente en la empresa se tiene diversos productos sin versionar y esto genera desorden tanto en el personal actual como el nuevo, sin contar en el tiempo que pierde el equipo intentando descifrar como trabajan o restaurando versiones funcionales. Por lo cual estamos haciendo la siguiente propuesta.

El presente plan debe ser aplicado a todos los proyectos grandes o pequeños, de esta forma el orden y el versionado se aplicarán a todos los productos de software de empresa. El presente plan de configuración se estará realizando teniendo en cuenta lo siguiente:

* En la empresa actualmente hay 10 proyectos; 9 de ellos en producción y 1 en desarrollo que es SLM. El presente plan se enfocará en 3 proyectos de software; 2 de ellos en producción (SMO y STC) y el único en desarrollo que es SLM, dando mayor prioridad a SLM y en segundo plano a los otros 2.
* Se usará la herramienta GitHub en su versión gratuita durante este proyecto, cabe resaltar que el repositorio estará expuesto a copia de terceros. Se plantea adquirir la versión de pago durante o luego de la implementación del proyecto donde los usuarios ya estarán correctamente entrenados y proteger el repositorio de la empresa de manera que solo los empleados tengan acceso a ella. O migrar hacia alternativas con repositorios privados sin costo como GitLab.

# Gestión de Configuración

## Organización

A continuación, en la Figura 1 se ilustra como las actividades de la Gestión de la Configuración se relacionan con las diferentes etapas de un software. Como se ve en la imagen, abarca todas las etapas donde se deben ir identificando elementos de configuración que deberán ser controlados y mantenidos a la par con el software.

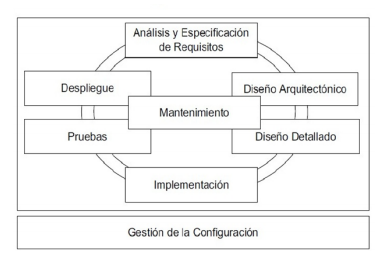


Figura 1: Relación del modelo de proceso usando, respecto a la gestión de configuración y mantenimiento

## Roles o responsabilidades

* **Responsable del Plan de Gestión de la Configuración (*RPGC*)**
  + Debe proveer la infraestructura y el entorno de configuración para el proyecto. Debe preocuparse porque todos los integrantes del grupo entiendan y puedan ejecutar las actividades de SCM que el Plan les asigna, así como asegurar que éstas sean llevadas a cabo. Seguir la línea base, controlando las versiones y cambios de ella, son tareas correspondientes a él. Debe definir y construir el Ambiente Controlado e informar al resto del equipo sobre la manera de usarlo.
  + Personal necesario: 1
* **Bibliotecario (*BIB*)**
  + Encargado de respaldar con las tareas de almacenamiento de los *baselines*. Un *baseline* o línea base es uno o más documentos formalmente diseñados y corregidos en un tiempo específico del ciclo de vida de los ítems de configuración, estos a su vez, son el conjunto de versiones de un item aceptado por el cliente.
  + Personal necesario: 1
* **Comité de Control de Configuración (*CCB*)**,
  + Máxima autoridad en la autorización de cambios.
  + Tiene la autoridad para aceptar o rechazar las propuestas de cambio a componentes de configuración. Cómo estos cambios tienen sentido controlarlos una vez que se crean las líneas base, el comité de control de cambios tiene la autoridad para gestionar las líneas base del producto y asegurar que los cambios son adecuadamente considerados y coordinados.
  + Personal necesario: 2

## Políticas, Directrices y procedimientos

El detalle de las directivas, directrices y procedimientos que aplican al PGC están en los siguientes documentos:

**Políticas de Seguridad. – Documento Políticas de Seguridad de la Gestión de la Configuración**

**Directrices de Autorización. – Documento de Directrices de Autorización de la Gestión de la configuración**

**Estándares de programación. – Manuales de estándares de programación para cada lenguaje**

**Manuales de Revisión de código. – Manual para revisión de código en cada lenguaje**

**Procedimiento de Despliegue. – Manual para despliegues en diferentes tecnologías**

## Herramienta, entorno e infraestructura

Durante el proceso de gestión de configuración se utilizará la herramienta GitHub para el control de versiones del producto. Cuando algún miembro haga una modificación en el proyecto, deberá acceder al repositorio donde está alojada esta aplicación para almacenar la parte modificada en él, teniendo el resto del equipo de desarrollo la última versión actualizada en dicho servidor.

Esta gestión de acceso al servidor para la actualización se hará mediante la herramienta git para los documentos y el código fuente y se tendrá un control de “Ramas” para evitar incidentes dentro de la Rama principal o “Master”.

El control de cambios, así como la gestión de defectos, se llevará a cabo mediante la plantilla “Solicitud de Cambio”, permitiendo al equipo de desarrollo ir detallando los defectos encontrados para que la persona responsable pueda corregirlos

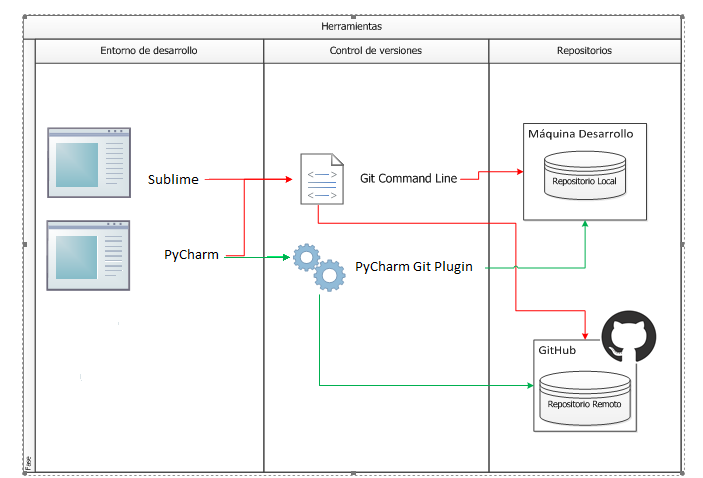
Todos los miembros del equipo de trabajo tendrán acceso a estas herramientas.

Figura 2 : Estructura del trabajo de github con las herramientas Sublime y PyCharm

### Herramientas de control de versiones

● Github

Un hosting online para nuestros repositorios que utiliza git para el mantenimiento y versionado del código fuente, añadiendo una serie de servicios extras para la gestión del proyecto y el código fuente. La parte gratuita de este hosting permite alojar nuestro código en repositorios públicos, si queremos repositorios privados entramos a la parte “premium”. En la actualidad ofrecen varias herramientas útiles para el trabajo en equipo, entre ellos cabe destacar

- Una wiki: para el mantenimiento de distintas versiones de las páginas.

- Un sistema de seguimiento de problemas, que permite a un miembro de tu equipo detallar el problema con tu software.

- Herramienta de versión de código, donde puedes añadir anotaciones en cualquier punto de un fichero.

- Un visor de ramas: donde se puede comparar los progresos realizados en las distintas ramas de nuestro repositorio.

● Git CLI

Git es un sistema rápido, escalable, distribuido de control de versiones con un número muy grande de comandos que proporciona las operacione de alto nivel y acceso completo al nivel inferior. La herramienta a usar para el caso de proyectos Front-end y Back-end será Git CLI(Command Line).

● Pycharm Git

PyCharm te permite consultar (en términos de Git, **clonar**) un repositorio existente y crear un nuevo proyecto basado en los datos que ha descargado. Aparte de clonando un repositorio remoto, puede crear un repositorio local basado en los orígenes de un proyecto existente.

### Herramientas de entorno

● Sublime (Backend and Frontend)

Sublime Text es un editor de código multiplataforma, ligero y con pocas concesiones a las florituras. Es una herramienta concebida para programar sin distracciones. Su interfaz de color oscuro y la riqueza de coloreado de la sintaxis, centra nuestra atención completamente.

El sistema de resaltado de sintaxis de Sublime Text soporta un gran número de lenguajes (C, C++, C#, CSS, D, Erlang, HTML, Groovy, Haskell, HTML, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, Matlab, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile and XML).

## Calendario

Las diferentes actividades que se tomara en cuenta durante todo el proceso de gestión de la configuración y mantenimiento del software (Ver Tabla 2.0).

Tabla 2: Calendario del Plan de Gestión de la Configuración

|  |  |
| --- | --- |
| **Actividad** | **Dependencia** |
| Realizar el plan de gestión de la configuración | Nada |
| Organización  Roles y responsabilidades  Políticas, Directrices y procedimientos  Herramientas | Plan de gestión de la configuración |
| entorno e Infraestructura  Calendario | Plan de gestión de la configuración |
| Realizar la actividad de identificación de SCM | Plan de gestión de la configuración |
| * Lista de la clasificación de CI | Plan de gestión de la configuración |
| * Definición de la Nomenclatura de Item | Plan de gestión de la configuración |
| * Lista de Item con la nomenclatura | Plan de gestión de la configuración |
| Realizar la definición de la línea base y estructura de librerías | Plan de gestión de la configuración |
| Realizar el plan de gestión de cambios | Plan de gestión de la configuración |
| Realizar los reportes de estado | Plan de gestión de la configuración y plan de gestión de cambios |
| Realizar los reportes de auditorias | Plan de gestión de la configuración y plan de gestión de cambios |
| Realizar la gestión de release | Plan de gestión de la configuración |

# Actividades de la SCM:

## Identificación de la configuración

## Ítems de configuración:

En la siguiente tabla 3.0 se señalan los diferentes ítems a realizar en este proyecto con su respectiva clasificación tomada:

Tabla 3: *Ítems a realizar en el proyecto según su clasificación*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tipo de item | Nombre del ítem (CI) | Fuente del documento | Extensión del documento | Proyecto Origen |
| E | Plan de la Gestión de la configuración | E | DOC |  |
| E | Documento de Acta del Proyecto | P | DOC | SLM |
| E | Modelo de Datos | P | MDJ | SLM |
| E | Manuales de Usuario | P | DOC | SLM |
| E | Documento de Requerimientos | C | DOC | SLM |
| F | Reporte de Locales | P | PY | SLM |
| F | Pantalla administrativa | P | PY | SLM |
| F | Reporte de usuarios | P | PY | SLM |
| F | Historial de alquiler | P | PY | SLM |
| Soporte | Laravel | P | PHP | SLM |
| Soporte | Framework de apoyo | P | JS | SLM |
| Soporte | Sublime text | P | EXE | SLM |
| (E = Evolución  F = Fuerte  S = Soporte) |  | (E= Empresa  P= Proyecto  C= Cliente  V=Proveedor) |  |  |

## Ítems de configuración:

Teniendo en cuenta la clasificación tomada a los ítems de configuración se tomará la siguiente nomenclatura:

* **Ítems en evolución**:
  + Sea el caso de ítems de empresa

Ejm: **P**lan de **G**estión de **C**onfiguración → **PGC.docx o PGC.pdf**

* + Sea el caso de ítems de proyecto

Ejm: El documento **P**lan de **P**royecto en el proyecto **S**istema de **L**ocales **M**usicales → **SLM\_PP.docx**

* Excepciones para Casos de Uso

Ejm: Caso de Uso para **Reg**istro de **Usu**ario en el proyecto de **SLM** → **SLM\_CUS\_REG\_USU.docx**

* **Ítems fuente:**
  + Para los items de desarrollo de proyectos FrontEnd:
    - Todo item que cuente con más de 2 palabras, deberá ser nombrado en Kebab\_case.
    - Los componentes: <NOMBRE DEL COMPONENT O FEATURE>.component.js
    - Las vistas o templates: <NOMBRE DEL COMPONENT O FEATURE>.template.html
    - Los controladores <NOMBRE DE COMPONENT O FEATURE>.controller.js
    - Los services(servicios) <NOMBRE DEL SERVICIO>.service.js
  + Para los items de proyectos BackEnd:
    - Lenguaje de desarroll Python y sus ítems serán nombrados: [nombre del ítem en kebab-case].py
    - Todo item con más de dos palabras deberá ser nombrado: [nombre del ítem en kebab-case]. {js|json}
* **Ítems de soporte:**
  + Para los proyectos de Front-end, Back-end y Android,los archivos de configuración serán nombrados: [nombre del archivo en kebab-case].[extesion]
* **Ítem de Archivo:**
  + Serán nombrados: [Iniciales del proyecto]\_ [Iniciales del documento].docx

## Lista de los ítems con Nomenclatura

En la siguiente tabla 4.0 se dará a conocer la nomenclatura de algunos ítems teniendo en cuenta la definición de nomenclatura tomada (se tomará de ejemplo a “Sistema de Locales Musicales”):

Tabla 4*: Nomenclatura de ítems durante el ciclo de vida del software*

|  |  |  |
| --- | --- | --- |
| NOMENCLATURA | ENTREGABLE | ETAPA |
| SLM\_PP | Plan de Proyecto | Requerimientos |
| SLM\_DN | Elaborar el Documento Negocio - BPMN | Requerimientos |
| SLM\_CUN | Documento de CUN | Diseño |
| SCAE\_DA | Diseño de la Arquitectura | Diseño |
| PGC | Plan de gestión de configuración | SCM |
| PGC\_CMB | Plan de Control de Cambios | SCM |

## Control de los ítems de la configuración