

Plan de Gestión de Configuración

2017

ÍNDICE

[1. Introducción 3](#_Toc493177922)

[2. Gestión de Configuración 3](#_Toc493177923)

[2.1. Organización 3](#_Toc493177924)

[2.2. Roles o responsabilidades 5](#_Toc493177925)

[2.3. Políticas, Directrices y procedimientos 6](#_Toc493177926)

[2.4. Herramienta, entorno e infraestructura 7](#_Toc493177927)

[Herramientas de control de versiones 7](#_Toc493177928)

[Herramientas de entorno 8](#_Toc493177929)

[2.5. Calendario 9](#_Toc493177930)

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:

* De acuerdo al tipo de proyectos estos durarán entre 12 a 15 semanas, sin embargo para el proyecto “Sistemas de locales musicales” este durara 12 semanas.
* El presente plan abarcara proyectos del tipo Móvil, Front-end, Back-end. Como lo es para el proyecto de “Sistemas de locales musicales” a excepción del desarrollo móvil.
* El presente plan se enfocará en aquellos proyectos de software que se encuentren en la fase de desarrollo o nuevos.

El gestor de la configuración es el encargado de aplicar el siguiente plan a todos los proyectos de software en coordinación con cada gerente de proyecto. Donde se realizaran reuniones, entre el gestor de configuración y los jefes de proyectos, previas de darse cambios en los proyectos.

# Gestión de Configuración

## Organización

Para cumplir con los requerimientos exigidos por los clientes, nuestro grupo de trabajo, cuenta con un esquema organizacional, donde cada integrante de nuestro equipo conoce sus obligaciones y responsabilidades en nuestra organización, donde cumplen su función de manera muy profesional. En la siguiente tabla mostraremos la relación de actividades con los roles de proyecto.

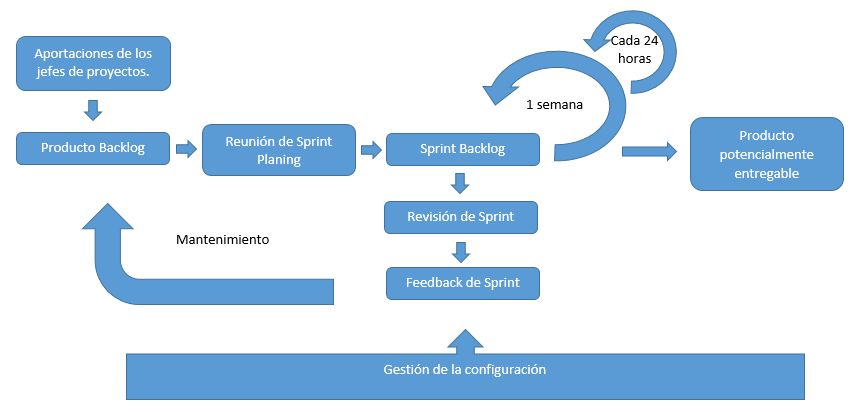
 Figura 1: Relación del modelo de proceso usando, respecto a la gestión de configuración y mantenimiento

Tabla 1 : Relación de Actividades con los roles de proyecto

|  |  |
| --- | --- |
| **Actividad** | **Responsable** |
| Planificar la Configuración del SCM | RPGC |
| Definición de la línea base del proyecto | RPGC, CCC, BIB |
| Seguimiento del proyecto de la línea base | RPGC |
| Definición del ambiente | RPGC |
| Control de cambios | RPGC, CCC |
| Descripción de la versión | RPGC |
| Realizar informe final del SCM | RPGC |

## 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

Respecto a los documentos en el repositorio de GitHub, todos tienen derechos de administrador, en otras palabras, permiso de lectura, escritura y creación de nuevos documentos. Se tiene como política la total confianza en todos los integrantes del equipo.

**Políticas. -**

* Seguir los estándares de nomenclatura plasmados en el presente documento.
* Cada rol debe cumplir las actividades designadas en el documento.
* Cada miembro debe trabajar únicamente y exclusivamente sobre su rama.
* La filtración de información a agentes externos se encuentra estrictamente prohibida.
* Correcta comunicación entre los miembros encargados de la gestión de la configuración.
* Toda dependencia en los proyectos deberá ser estrictamente documentada.

**Directrices. -**

* Se encuentra estrictamente prohibido el versionamiento en el nombre del item. Ejm: No se permite el nombre “SLM\_SCMP\_v1.1.docx”
* Para los items de gestión de la configuración:
  + Los items deberán llevar el siguiente estándar de nombre:

<INICIALES DE LA EMPRESA>\_<INICIALES DEL DOCUMENTO>.\*

Ejm: SLM\_ SCMP.docx (Plan de gestión de la configuración)

* En caso dos items de gestión de la configuración llegasen a tener la misma nomenclatura, el ítem más reciente tendrá la nomenclatura:
  + <INICIALES DEL DOCUMENTO>\_<TAG>.\*

Donde <TAG> es un identificador para dar soporte a la identificación del documento más reciente.

* Para los items de gestión de proyectos:
  + Los items deberán llevar el siguiente estándar de nombre:

<INICIALES DEL PROYECTO>\_<INICIALES DEL DOCUMENTO>.\*

Ejm: SLM\_PP.docx

## 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

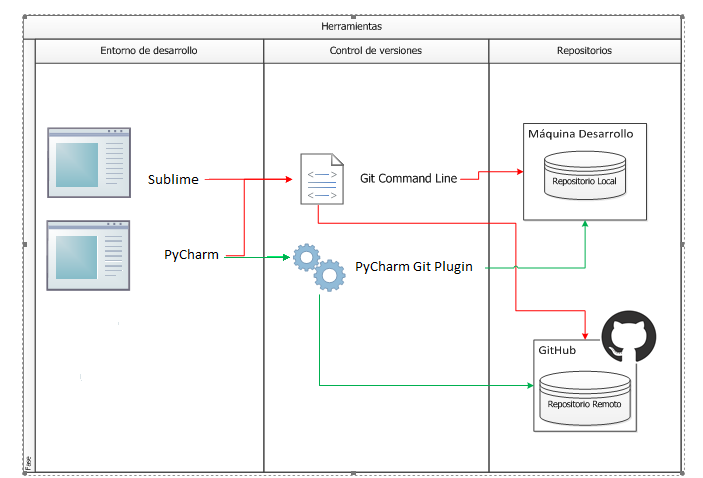
Todos los miembros del equipo de trabajo tendrán acceso a estas herramientas.

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** | **Depende de** | **Comienzo** |
| Realizar el plan de gestión de la configuración | Nada | Semana 1: 01/09/2017 |
| Realizar el documento de negocio | Plan de gestión de la configuración | Semana 2: 08/09/2017 |
| - | Plan de gestión de la configuración | Semana 3: 15/09/2017 |