|  |
| --- |
| WhatsApp%20Image%202018-04-03%20at%203.03.05%20PM.jpeg  **Plan de Gestión de la Configuración**  **V 1.4** |

**Contenido**

[1. Introducción 3](#_Toc513502781)

[1.1. Propósito del documento 4](#_Toc513502782)

[1.2. Alcance 4](#_Toc513502783)

[1.3. Definiciones 5](#_Toc513502784)

[ Abreviaturas 5](#_Toc513502785)

[1.4. Referencias 6](#_Toc513502786)

[2. Gestión de la Configuración de Software 6](#_Toc513502787)

[2.1. Organización 6](#_Toc513502788)

[2.2. Roles y Responsabilidades 8](#_Toc513502789)

[2.3. Políticas, directrices y procedimientos 10](#_Toc513502790)

[2.4. Herramientas, entorno e infraestructura 10](#_Toc513502791)

[ Herramientas de control de versiones 12](#_Toc513502792)

[ Herramientas de entorno 13](#_Toc513502793)

[2.5. Calendario 14](#_Toc513502794)

[3. Actividades de la SCM 15](#_Toc513502795)

[3.1. Identificación de la configuración 15](#_Toc513502796)

[ Ítems de configuración 15](#_Toc513502797)

[ Definiciones de nomenclatura de los ítems 16](#_Toc513502798)

[ Lista de los ítems con nomenclatura 18](#_Toc513502799)

# Introducción

NN-Consulting es una empresa que se dedica al desarrollo de software, ofreciendo soluciones Web y Mobile, utilizando las mejores técnicas como Scrum, Kanban y Design Thinking. También, brindamos soporte 360 a nuestros clientes, teniendo así casos de éxito como:

* SOSFido
* SaluDate
* Sistema de asistencia en Pavisola S.A

Actualmente, la empresa se encuentra a cargo de 3 proyectos de gran envergadura:

* SPACIA: Sistema de Gestión de Ambientes
* COMEDOR: Sistema de tickets en el comedor de UNMSM
* Sistema de asistencia: Para la empresa Backus

En el tiempo en que NN-Consulting lleva manejando proyectos de desarrollo de software, han surgido inconvenientes en el manejo de las solicitudes de cambios (tanto previstos como imprevistos) que han llevado a la empresa a realizar múltiples versiones (no controladas) del mismo producto durante todo el ciclo de vida de los proyectos. Con esta mala práctica, la empresa pasaba por momentos de confusión e incertidumbre, debido a que no se contaba con una guía o un documento donde se encontrarán establecidas ciertas reglas para poder llevar a cabo una correcta gestión del versionamiento. Surgían inconvenientes, tales como, que los desarrolladores trabajaban sobre una versión que no era la última aprobada; o que, más de un desarrollador realizaba la misma tarea, pues en la versión que tenían no estaba completa. En consecuencia, era muy complicado realizar un seguimiento y auditoría a las versiones de un mismo producto de software y los documentos relacionados.

Con todos estos inconvenientes identificados, se consideró necesario que las versiones de los productos de software desarrollados por NN-Consulting estén sujetas a mantenimiento y gestión para minimizar los riesgos asociados al tener varias versiones de los distintos ítems, además de la cantidad de los mismos que se maneja.

Ante la necesidad de establecer los roles, responsabilidades, herramientas, políticas y procedimientos para una correcta gestión de la configuración del software; y de la necesidad de llevar a cabo las tareas necesarias para el mantenimiento del software desarrollado, se desarrolló este documento; el cual describe las actividades relacionadas con la gestión de la configuración y el mantenimiento de los productos de software y productos de trabajo de software (documentos de software) desarrollados por NN-Consulting. Este documento permitirá a la empresa gestionar adecuadamente la configuración y mantenimiento de los proyectos a cargo.

## Propósito del documento

Este documento está diseñado para servir de soporte al equipo encargado de la gestión de la configuración o mantenimiento, en el uso de métodos (y los requisitos para el cumplimiento de estos) y en la ejecución de actividades para una adecuada configuración y mantenimiento de los proyectos de software de NN-Consulting, solucionando así las malas prácticas de control del versionamiento, gestión de cambios, gestión de documentos y gestión de proyectos en los distintos proyectos de NN-Consulting, obteniendo lo siguientes beneficios:

* Documentación que comprende la descripción completa del producto final, la identificación de todos y cada uno de sus componentes, los atributos y prestaciones de los componentes.
* Control de gestión cambios del producto y/o sus componentes
* Posibilidad de recuperar las configuraciones en vigor, las fabricadas y las entregadas a lo largo de toda la vida útil del producto.

## Alcance

Este documento tendrá el siguiente alcance:

* Describir los roles, responsabilidades, políticas, procedimientos, herramientas y actividades de gestión de configuración de software que deben ser llevadas a cabo durante el proceso de desarrollo de cualquier proyecto de software de la empresa.
* Definir tanto los productos que se pondrán bajo control de configuración como los procedimientos que deben ser seguidos por los integrantes del equipo de trabajo.
* Identificar y definir la línea base de los elementos para cualquier proyecto de software de la empresa.
* Establecer de los métodos, procedimientos y herramientas para el control y registro del estado de las versiones de cualquier proyecto de software de la empresa.
* Procedimientos para la solicitud de cambios (modificación de cualquier proyecto de software de la empresa).

## Definiciones

* **Ítem de configuración:** Unidad estructural fundamental de un sistema de gestión de configuración.
* **Línea base:** Punto de referencia en el ciclo de vida del desarrollo de software marcado por la finalización y aprobación formal de un conjunto de productos de trabajo predefinidos
* **Identificación de configuración:** Esta actividad se encarga de identificar los elementos, esquemas y herramientas.
* **Control de configuración:** Gestiona los cambios que se realizarán al software durante su ciclo de vida. Determina que cambio se debe hacer, si es que es correcto y la manera de su implementación.
* **Contabilidad del estado de configuración:** Proceso por el cual en el estado actual del desarrollo puede ser juzgado y la historia del ciclo de vida del desarrollo se puede rastrear.
* **Auditoría de configuración:** Identificar los elementos del sistema que satisfacen los requerimientos del cliente.

## Abreviaturas

A continuación, se procederá a mencionar las abreviaturas utilizadas en el desarrollo de documento. Es importante resaltar, que las siglas se manejan en inglés, tomando en cuenta que este documento podría ser usado por terceros. No obstante, se describe la definición traducida.

* CCB: Tablero de Control de Configuración (Configuration Control Board)
* CI: Ítem de Configuración (Configuration Item)
* CM: Gestión de Configuración (Configuration Management)
* SCM: Gestión de la Configuración de Software (Software Configuration Management)
* CMO: Oficial de Gestión de Configuración (Configuration Management Officer)
* SQA: Aseguramiento de calidad de software (Software Quality Assurance)

## Referencias

[1] ANSI/IEEE Std 828-1983, IEEE Standard for Software Configuration Management Plans

[2] ANSI/IEEE Std 1042-1987, IEEE Guide to Software Configuration Management

# Gestión de la Configuración de Software

## Organización

La Figura 1 muestra cómo está organizado el desarrollo de los productos de software de la empresa NN-Consulting. Este modelo está alineado a la metodología de desarrollo para que pueda cubrir las expectativas de nuestros clientes, tomando en cuenta la prevención de las actividades de configuración y mantenimiento, así como, de las demás fases de desarrollo de software. Actualmente la empresa usa la metodología SCRUM. En la figura 1, se observa que, para cada Sprint según el ciclo de desarrollo de software, se tiene alineado la configuración y el mantenimiento de manera transversal. Esto se traduce, en que la SCM estará presente durante cada una de las fases de del proyecto integrada con las actividades del proyecto.

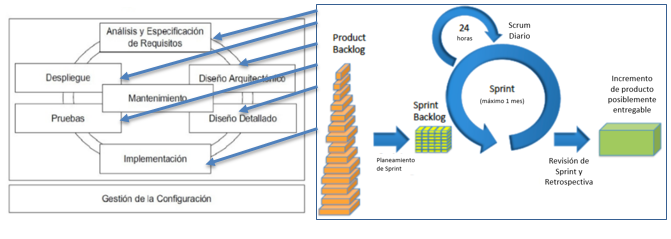


Figura 1: Relación de la metodología de desarrollo usada, respecto a las etapas del proyecto, incluyendo la configuración y mantenimiento.

La siguiente tabla muestra las actividades específicas de la SCM durante el desarrollo de un proyecto (Ver Tabla 1).

*Tabla 1: Relación entre las actividades específicas de la SCM.*

|  |
| --- |
| Actividades |
| Planificar la SCM |
| Identificar la SCM |
| Controlar la SCM |
| Obtener el estado de contabilidad |
| Realizar auditoría SCM |
| Realizar entrega de releases |

En la Figura 2, se observa la relación de las actividades de la SCM con las etapas del proyecto.

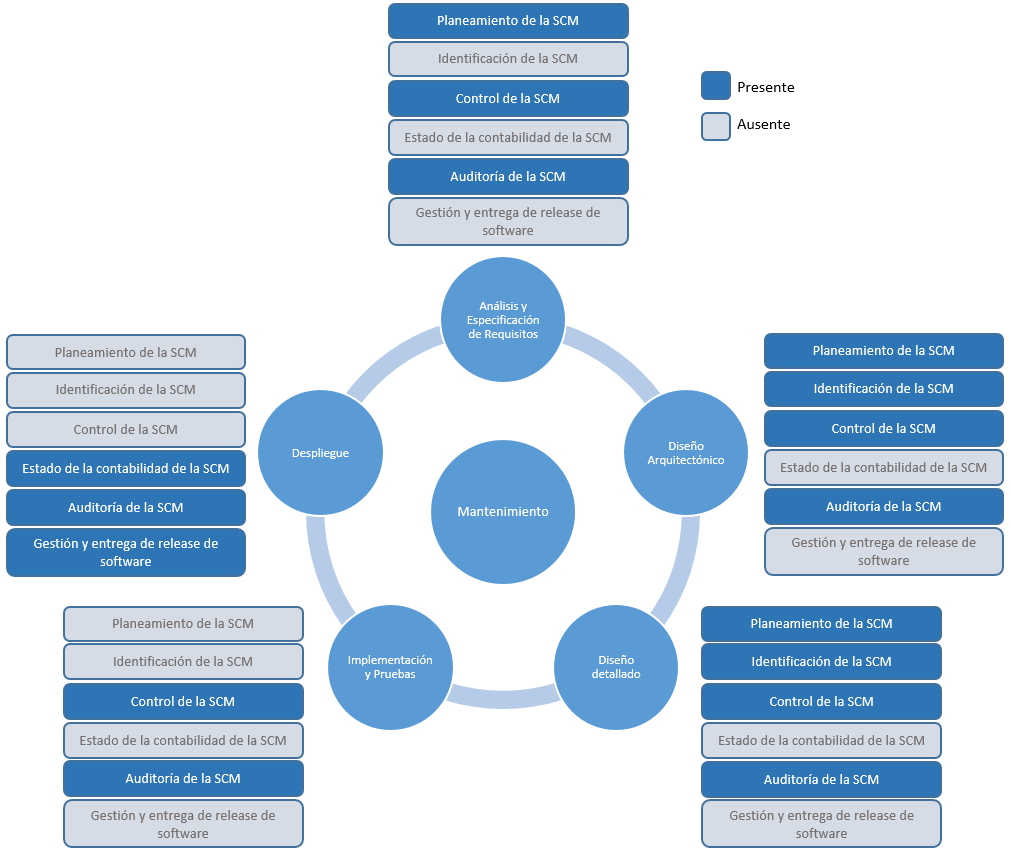


Figura 2: Relación actividades SCM con el desarrollo del proyecto de software

## Roles y Responsabilidades

La unidad de SCM de la NN-Consulting es liderada por el responsable de SCM (Configuration Management Officer, CMO), responsable por las tareas de gestión de la unidad. En general, la unidad actúa como un equipo en el cual sus miembros cooperan para llevar a cabo las tareas de SCM. A continuación, se presenta la Tabla 2 con los roles y sus responsabilidades correspondientes.

*Tabla 2: Roles y Responsabilidades*

|  |  |
| --- | --- |
| Rol | Responsabilidades |
| Oficial de Gestión de Configuración | * Gestionar la unidad de SCM. * Definir una estrategia para la implantación de las prácticas de SCM. * Proveer el personal de SCM requerido para las actividades de desarrollo. * Planificar las actividades de SCM. * Informar a los niveles superiores sobre el estado del proceso y las actividades de SCM en los proyectos. * Garantizar la capacitación del personal de SCM y los recursos necesarios para el desarrollo de sus actividades. * Promover el mejoramiento continuo del proceso de SCM. * Gestionar el desarrollo de herramientas para facilitar el proceso de SCM. |
| Tablero de Control de Configuración | * Revisar el hardware o solicitudes de cambio de clientes, en términos del cronograma del proyecto, el costo y el impacto en los clientes. * Recomienda la aprobación o desaprobación del cambio. * Priorización de la incorporación de cambios aprobados. * Reenvía la decisión de aprobación o desaprobación del cambio al CMO * Informa por escrito al cliente de la decisión de cambio. |
| Ingenieros | * Desarrollar y coordinar las actividades de SCM planificadas para el proyecto. * Establecer y participar de la identificación, control, auditoría e informe sobre el estado de la configuración. * Participar en el establecimiento de la biblioteca del software. * Monitorear el correcto funcionamiento de la biblioteca del software. * Interactuar con los desarrolladores y la unidad de SQA. |
| Bibliotecarios | * Diseñar y establecer la biblioteca del software para cada proyecto de desarrollo durante la etapa de planificación. * Establecer y mantener el software y la documentación de cada proyecto de acuerdo con un proceso documentado * Proveer a los desarrolladores las copias de las líneas bases requeridas para sus diferentes tareas. * Mantener y distribuir un índice con el contenido de cada biblioteca. * Entregar la copia original para la implementación de los cambios aprobados por el CCB. * Informar a los desarrolladores sobre los cambios a los ítems * Registrar y mantener copias de las antiguas versiones. * Apoyar la elaboración de informe sobre el estado de la configuración. |
| Personal de apoyo | * Desarrollar tareas administrativas para los representantes de SCM. * Completar los informes de SCM. * Interactuar con las diferentes herramientas utilizadas para SCM durante un proyecto. * Interactuar con el personal de SCM según sea necesario. |

## Políticas, directrices y procedimientos

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

**Políticas y Directrices**

* Las políticas y directrices se encuentran en un documento aparte. Ver repositorio: Documentos/PD.docx.
* Para ver el documento de Alcance del Sistema de Gestión de Seguridad de la Información. Ver repositorio: Documentos/DSGSI.docx.
* Para ver el Manual de Organización y Funciones. Ver repositorio: Documentos/MOF.docx.

**Procedimientos**

* Respecto a la modificación de archivos pertenecientes a la gestión de la configuración:
  + - Indicar a demás miembros encargados de los documentos qué documento está siendo manipulado, con el fin de evitar modificaciones en el mismo por parte de otros miembros, lo cual repercutiría en un sobreescribimiento del ítem.
    - Modificar el ítem apropiadamente.
    - Subir los cambios.
    - Hacer los merge necesarios para que el ítem modificado se encuentre disponible para todos.
    - Informar que el cambio fue realizado.

## Herramientas, entorno e infraestructura

En la Figura 3, se muestra el mapeo del uso de las diferentes herramientas que se usan para el manejo de las versiones de los ítems. Se puede observar la clasificación de las herramientas en: Entorno de desarrollo, Control de versiones y Repositorios. No obstante, las últimas dos fueron separadas en el gráfico para un mejor entendimiento, pues ambas funcionan como parte del Control de versiones. Las herramientas de Entorno de Desarrollo, son las que son usadas por el equipo de desarrollo para la creación del producto de software. Si bien, en muchos de estos entornos, existen interfaces para el manejo de las herramientas de control de versiones, se observa que los desarrolladores usan otras interfaces para el control de versiones, propias de las herramientas para el Control de versiones (Git). En cuanto a estas últimas, se tiene a Git Kraken como herramienta Git con interfaz gráfica y el Git Bash/CMD para líneas de comando. Actualmente, el equipo web y Back-end trabajan con la interfaz gráfica. Solo el equipo de desarrollo móvil trabaja con ambos tipos de interfaces. En cuanto a los Repositorios, se observa que todos los equipos trabajan con un repositorio local, un repositorio remoto (GIT) y un repositorio para producción (servidor Digital Ocean-utilizado solo por las áreas de web y Back-end).

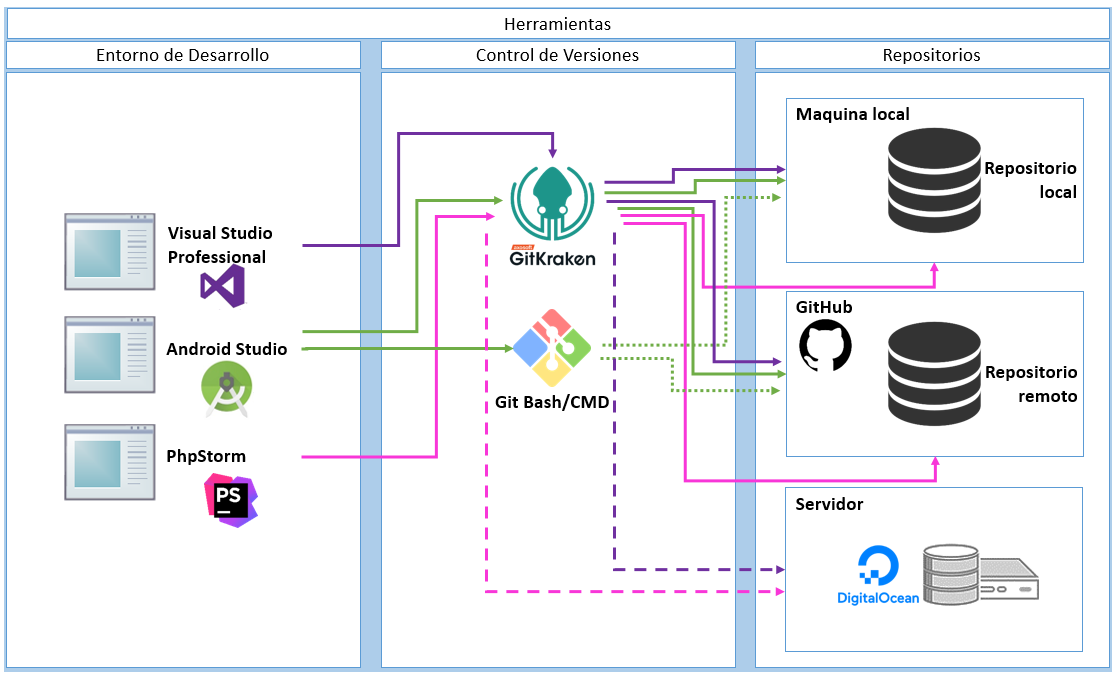


Figura 3: Mapeo del uso de las herramientas de control de versiones y entorno

## Herramientas de control de versiones

* + - Github

Es una plataforma de desarrollo colaborativo online, utiliza control de versiones git para el mantenimiento y versionado del código fuente, añadiendo una serie de servicios extras para la gestión del proyecto y el código fuente. Permite alojar nuestro código en repositorios públicos, la versión Premium permite tener repositorios privados, pero para el desarrollo de este proyecto se usará la versión gratuita. 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 hacer mejoras, sugerencias y optimizaciones.
    - Herramienta de versión de código, donde puedes añadir anotaciones en cualquier punto del proyecto.
    - Un visor de ramas o branch: donde se puede conocer el progreso que llevamos en nuestro proyecto, mostrando las actualizaciones o commits realizados a partir de nuestro master.
    - Git

Es una herramienta de tipo consola que posee múltiples comandos que ayudan a tener un buen control de versiones, permite manipular y gestionar todo el proceso a realizarse en el proyecto. Posee la misma funcionalidad tanto en Windows, Linux o Mac, permite un desarrollo no lineal lo que permite una gran rapidez en la gestión de ramas. La herramienta será usada tanto en el desarrollo Front End, Back End, Android y documentación. Git Bash es la herramienta para consola más utilizada por los desarrolladores de NN-Consulting.

* + - Git Kraken

Proporciona una interfaz gráfica multiplataforma para git. Es bastante amigable y nos permite visualizar y gestionar de una forma más sencilla y hacer un seguimiento a nuestro historial en el repositorio y sus distintas ramas al igual que git esta herramienta será usada tanto en el desarrollo FrontEnd, BackEnd, Android y documentación.

* + - Servidor-Digital Ocean

DigitalOcean es un proveedor estadounidense de servidores virtuales privados, basado en la ciudad de Nueva York. La compañía alquila instalaciones de centros de cómputo existentes, incluyendo sitios como Nueva York, Toronto, Bangalore, Ámsterdam, San Francisco, Londres y Singapur. El servidor con el que cuenta NN-Consulting tiene las siguientes características: VPS Linux (Ubuntu 16.04) 1 CPU, 1GB RAM, 25 GB SSD.

## Herramientas de entorno

* + - Visual Studio Professional (Frontend-Web)

Visual Studio Professional es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows y **MacOS** (Actualmente-2018 Sistema Operativo más utilizado por los desarrolladores en NN-Consulting). Soporta múltiples lenguajes de programación, tales como C++, **C#** (Actualmente-2018 el lenguaje más utilizado por los desarrolladores en NN-Consulting), Visual Basic .NET, F#, Java, Python, Ruby y PHP, al igual que entornos de desarrollo web, como **ASP.NET MVC** (Actualmente-2018 el más utilizado por los desarrolladores en NN-Consulting), Django, etc. Permite administrar el código en repositorios GIT que hospede cualquier proveedor, como GitHub y Visual Studio Team Services.

* + - Android Studio (Móvil)

Es el entorno de desarrollo integrado oficial para la plataforma Android. Fue anunciado el 16 de mayo de 2013 en la conferencia Google I/O, y reemplazó a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. La primera versión estable fue publicada en diciembre de 2014. Está basado en el software IntelliJ IDEA de JetBrains, y es publicado de forma gratuita a través de la Licencia Apache 2.0. Está disponible para las plataformas Microsoft Windows, Mac OS X y GNU/Linux.

* + - PhpStorm (Back-End)

PhpStorm es un IDE de programación desarrollado por JetBrains. Es uno de los entornos de programación más completos de la actualidad, permite editar código no sólo del lenguaje de programación php como lo indica su nombre. PhpStorm permite trabajar con Symfony, Drupal, WordPress, Zend Framework, **Laravel** (Actualmente-2018 framework más utilizado por el equipo de desarrollo en NN-Consulting), Magento, Joomla!, CakePHP, Yii, y otros frameworks.

## Calendario

A continuación, se presenta la tabla de desarrollo de actividades (hitos) incluidas dentro del Proceso de Gestión de la Configuración de Software para NN-Consulting (Ver Tabla 3).

Tabla 3: Calendario del Plan de Gestión de la Configuración

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Proceso | Actividad | Rol | Duración parcial | Duración total |
| Realizar planificación de la GCM | Realizar el plan de gestión de la configuración (Introducción, propósito, alcance, definiciones, abreviaturas, referencias; Organización, Roles y Responsabilidades, Herramientas, entorno e infraestructura y Calendario) | CMO, Ingenieros | 2 semanas y media | 2 semanas y media |
| Realizar la actividad de identificación de SCM | Realizar identificación de SCM (Nomenclatura, Clasificación de ítems y Lista con nomenclatura) | CMO, Ingenieros, Personal de apoyo | 1 semana y media | 1 semana y media |
| Realizar el Control de la SCM | Realizar la definición de la línea base y estructura de librerías | CMO, CCB, Ingenieros, Bibliotecarios, Personal de apoyo | 1 semana | 2 semanas |
| Realizar el plan de gestión de cambios | CMO, CCB, Ingenieros, Personal de apoyo | 1 semana |
| Verificar el Estado de Contabilidad de la SCM | Realizar los reportes de estado | CMO, CCB, Ingenieros, Bibliotecarios, Personal de apoyo | 1 semana | 1 semana |
| Realizar Auditoría de la SCM | Realizar los reportes de auditorias | CMO, CCB, Ingenieros, Personal de apoyo | 1 semana | 1 semana |
| Gestionar la entrega de los releases del software | Realizar la gestión de releases | CMO, CCB, Personal de apoyo | 1 semana | 1 semana |

# Actividades de la SCM

## Identificación de la configuración

## Ítems de configuración

En la Tabla 4, se procede a listar los ítems clasificados por su tipo, fuente y extensión.

Tabla 4: Lista de Ítems de configuración

|  |  |  |  |
| --- | --- | --- | --- |
| Tipo  E: Evolución F: Fuente  S:Soporte | Nombre de Ítem | Fuente  E: Empresa  P: Proyecto  C: Cliente  V:Proveedor | Extensión |
| E | Plan de la gestión de la configuración | E | DOCX |
| E | Políticas y Directrices | E | DOCX |
| E | Alcance SGSI | E | DOCX |
| E | Manual de Organización y Funciones | E | DOCX |
| E | Documento de especificación de requisitos | P | DOCX |
| E | Documento de Caso de uso del Sistema | P | DOCX |
| E | Documento de diseño y arquitectura de software | P | DOCX |
| E | Documento de Pruebas | P | DOCX |
| E | Manual de usuario | P | DOCX |
| E | Aplicación móvil | P | APK |
| F | Código fuente de la aplicación móvil | P | ZIP |
| F | Script de la base de datos | P | ZIP |
| F | Scripts de front-end | P | ZIP |
| F | Vistas del front-end | P | ZIP |
| F | Compilados del código fuente front-end | P | ZIP |
| E | Documentación del Api | P | DOCX |
| F | Código fuente back-end | P | ZIP |
| F | Manual de instalación | P | DOCX |

## Definiciones de nomenclatura de los ítems

A continuación, se muestran las fórmulas para la asignación de los nombres a los ítems a utilizar durante los proyectos, clasificados según los tipos de ítems.

* **Ítems en evolución y de soporte**

***Documentos***

* Documentos no específicos a un proyecto:

/\*Acrónimo del documento\*/

*Ejemplo:***PGC** *(Plan de Gestión de la Configuración)*

* Documentos específicos a un proyecto, no de un componente del mismo:

/\*Acrónimo del proyecto + “\_” + Acrónimo del documento\*/

*Ejemplo:***SPC\_PP** *(Plan de Proyecto del proyecto SPACIA)*

* Documentos específicos a un proyecto y de un componente del mismo:

/\*Acrónimo del proyecto + “\_” + Acrónimo del componente + “\_” + Acrónimo del tipo del documento\*/

*Ejemplo:***SPC\_ IS\_ CUS1** *(Caso de Uso del Sistema número 1: Ingreso al Sistema del proyecto SPACIA)*

* **Ítems fuente**

***Directorios***

* Directorios contenedores-denominación libre:

/\*minúscula de 1°palabra + (mayúscula de hasta máximo 4 palabras sin separación) \*/

*Ejemplo:***recursosEstilos** *(Carpeta de CSS para web)*

***Documentos***

* Documentos que se relacionen directamente con el código fuente:

/\*Acrónimo del proyecto + “\_” + Acrónimo del documento\*/

*Ejemplo:***SPC\_CFAM** *(Código Fuente de la Aplicación móvil del proyecto Spacia)*

***Archivos propios del desarrollo***

* Archivos usados como parte del software-libre denominación:

/\*minúscula de 1°palabra + “\_” + (minúscula de hasta máximo 4 palabras separadas por un “\_”) + “.” + extensión\*/

*Ejemplo:***activity\_Login.xml** *(activity de ingreso de una aplicación)*

***Archivos propios para el desarrollo***

* Archivos usados como parte del software-denominación condicionada por el entorno o según se considere conveniente:

/\*mayúscula de 1°palabra + (mayúscula de hasta máximo 4 palabras sin espacio+ “.” + extensión autogenerada\*/

*Ejemplo:***LoginActivity.java** *(Código Java-class para el activity del Login)*

* **Casos generales y excepciones**

***Imágenes***

/\*minúscula de 1°palabra + “\_” + (mayúscula de hasta máximo 4 palabras separadas por un “\_”) + “.” + extensión de imagen\*/

*Ejemplo:***logo\_Aplicación.png** *(Imagen del logo de la aplicación)*

***En caso de repetición en los nombres***

* Documentos:

Caso: **SPC\_PP** (Plan de Proyecto del proyecto SPACIA) y **SPC\_PP** (Procesos Planificados)

El primero en aparecer mantiene su nombre, a los siguientes en aparecer con repetición en el nombre según la fórmula asignada, se le añadirá la siguiente letra, en minúscula, a cada inicial del acrónimo. Se le irá añadiendo una letra más según vayan apareciendo los conflictos.

*Ejemplo:***SPC\_PP** *(Plan de Proyecto del proyecto SPACIA),* **SPC\_PrPl** *(Procesos Planificados del proyecto SPACIA),* **SPC\_PoPr** *(Población de Prueba para el proyecto SPACIA),* **SPC\_ProPla** *(Procedimientos Planificados para el proyecto SPACIA)*

* Archivos propios del desarrollo-en un mismo directorio:

Caso: **activity\_Login.xml** (activity de ingreso de una aplicación) y **activity\_Login.xml** (otro activity de ingreso de una aplicación)

El primero en aparecer mantiene su nombre, a los siguientes en aparecer con repetición en el nombre según la fórmula asignada, se le añadirá un número sin espacios, empezando desde el 1.

*Ejemplo:***activity\_Login.xml** (activity de ingreso de una aplicación) y **activity\_Login1.xml** (otro activity de ingreso de una aplicación)

## Lista de los ítems con nomenclatura

En esta sección, a cada uno de los ítems que ya fueron identificados

anteriormente en una tabla, se les asigna una nomenclatura en base a las reglas que fueron definidas. (Tabla 5)

Tabla 5: Nomenclatura de los ítems durante el ciclo del desarrollo del software

|  |  |  |
| --- | --- | --- |
| Nomenclatura | Descripción del Item(CI) | Fuente  E: Empresa  P: Proyecto  C: Cliente  V:Proveedor |
| PGC | Plan de la gestión de la configuración | E |
| PD | Políticas y Directrices | E |
| DSGSI | Documento del SGSI | E |
| MOF | Manual de Organización y Funciones | E |
| SPC\_DER | Documento de especificación de requisitos | P |
| SPC\_DDAS | Documento de diseño y arquitectura de software | P |
| SPC\_IS\_CUS1 | Documento de Caso de uso del Sistema | P |
| SPC\_DP | Documento de Pruebas | P |
| SPC\_MU | Manual de usuario | P |
| SPC\_AM | Aplicación móvil | P |
| SPC\_CFAM | Código fuente de la aplicación móvil | P |
| SPC\_SBD | Script de la base de datos | P |
| SPC\_SFE | Scripts de front-end | P |
| SPC\_VFE | Vistas del front-end | P |
| SPC\_CFE | Compilados del código fuente front-end | P |
| SPC\_DAPI | Documentación del Api | P |
| SPC\_CFBE | Código fuente back-end | P |
| SPC\_MI | Manual de instalación | P |

## Control de la Configuración

### Definición de Librerías

|  |  |
| --- | --- |
| **Línea Base** | |
| Responsable | Bibliotecarios |
| Actividades | * Establecer y mantener la línea base de cada proyecto de acuerdo con lo establecido en el Plan de Gestión de la Configuración. * Proveer copias de las líneas base a quienes lo soliciten y tengan los permisos necesarios. |
| Contenido | * Esta librería contiene las líneas base de todos los proyectos que se encuentran en desarrollo dentro de la empresa. |
| **Accesos** | |
| Rol | Tipo de Acceso |
| Jefe del Proyecto | * Leer |
| Oficial de Gestión de la Configuración | * Leer |
| CCB | * Leer |
| Bibliotecario | * Leer * Escribir * Modificar * Eliminar |

|  |  |
| --- | --- |
| **Producción** | |
| Responsable | Jefe de Proyecto |
| Actividades | * Establecer y mantener las versiones de producción conforme a los establecido en el Plan de Proyecto de cada proyecto de software. |
| Contenido | * En esta librería se encontrarán las versiones de software, de los distintos proyectos en desarrollo, que se encuentren en producción. |
| **Accesos** | |
| Rol | Tipo de Acceso |
| Jefe de Proyecto | * Leer * Escribir * Modificar * Eliminar |
| Ingenieros (Desarrolladores) | * Leer * Escribir * Modificar * Eliminar (con autorización del jefe de proyecto) |
| Oficial de Gestión de la Configuración | * Leer |
| CCB | * Leer |
| Bibliotecario | * Leer |

* + 1. **Reportes de Estado para el Jefe de Proyecto**
       1. **Reporte de Estado 007**

La Tabla X muestra el Reporte de estado 007.

|  |  |
| --- | --- |
| ID | RE\_007 |
| Título | Lista de los ítems del proyecto de software |
| Propósito | El propósito de este reporte es brindar información sobre los elementos de desarrollo de software de un proyecto de software. |
| Entradas | ID del Proyecto de software (SPC) |
| Salidas | * ID del ítem del proyecto * Nombre del ítem del proyecto * Autor de la última modificación. * Fecha de la última modificación. * Ubicación del ítem de configuración (Ruta). |

Tabla 30: Reporte de estado 007

Ejemplo de Reporte de Estado 007:

La tabla 31 muestra el caso de ejemplo para el reporte de estado 007.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID ítem de proyecto | Nombre del ítem del proyecto | Autor de la última modificación | Fecha de la última modificación | Ubicación del ítem de configuración |
| 001 | SPC\_PP.docx | Óscar Abad | 08/06/2018 | Desarrollo/1.SPACIA/1.Gestion/ |
| 002 | SPC\_DN.docx | Kenny Horna | 25/05/2018 | Desarrollo/1.SPACIA/2.Negocio/ |

Tabla 31: Caso de ejemplo para RE\_007

* + - 1. **Reporte de Estado 008**

En la Tabla X se muestra el reporte de estado 008.

|  |  |
| --- | --- |
| ID | RE\_008 |
| Título | Lista de participantes de un proyecto de software específico a cargo de un determinado jefe de proyecto |
| Propósito | El propósito de este reporte es brindar la lista del personal involucrado en un determinado proyecto de software. |
| Entradas | ID del Proyecto de software (SPC) |
| Salidas | * ID del desarrollador * Nombre del desarrollador * Rol * Inactividad * Tiempo de colaboración (en días) |

Tabla 30: Reporte de estado 008

Ejemplo X de Reporte de Estado 008:

La tabla 31 muestra el caso de ejemplo para el reporte de estado 008.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID del desarrollador | Nombre del desarrollador | Rol | **Inactividad (en días)** | **Tiempo de colaboración (en días)** |
| D001 | Jair Barzola | Desarrollador móvil | 2 | 40 |
| D002 | Eduardo Angulo | Desarrollador web | 3 | 25 |
| D003 | Gustavo Huaracc | Desarrollador web | 1 | 30 |

Tabla 31: Caso de ejemplo para RE\_008

* + - 1. **Reporte de Estado 009**

La Tabla X muestra el reporte de estado 009.

|  |  |
| --- | --- |
| ID | RE\_009 |
| Título | Lista de casos de uso de un proyecto de software específico. |
| Propósito | El propósito de este reporte es brindar la lista de todos los casos de uso de un proyecto de software específico. |
| Entradas | ID del Proyecto de software (SPC) |
| Salidas | * Caso de uso * Nombre del CUS * Cantidad de personas asignadas * Número de vistas relacionadas |

Tabla 30: Reporte de estado 009

Ejemplo X de Reporte de Estado 009:

La tabla 31 muestra el caso de ejemplo para el reporte de estado 009.

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de Uso | Nombre del CUS | Cantidad de Personas Asignadas | Número de vistas relacionadas |
| CUS1 | Iniciar Sesión | 4 | 6 |
| CUS2 | Recuperar contraseña | 5 | 13 |
| CUS3 | Visualizar Dashboard | 2 | 6 |

Tabla 31: Caso de ejemplo para RE\_009

* + - 1. **Reporte de Estado 010**

En la Tabla X se muestra el reporte de estado 010.

|  |  |
| --- | --- |
| ID | RE\_010 |
| Título | Lista de encargados de un proyecto de software |
| Propósito | El propósito de este reporte es brindar la lista del personal a cargo de un determinado proyecto de software. |
| Entradas | ID del Proyecto de software (SPC) |
| Salidas | * ID del desarrollador * Cantidad de proyectos en los que está participando * Caso de uso asignado * Ultimo cambio realizado |

Tabla 30: Reporte de estado 010

Ejemplo X de Reporte de Estado 010:

La tabla 31 muestra el caso de ejemplo para el reporte de estado 010.

|  |  |  |  |
| --- | --- | --- | --- |
| ID del desarrollador | Nro. de proyectos participando | Caso de uso asignado | Ultimo cambio realizado |
| D001 | 2 | CUS2 | 20/06/2018 |
| D002 | 2 | CUS1 | 17/06/2018 |
| D003 | 1 | CUS3 | 21/06/2018 |

Tabla 31: Caso de ejemplo para RE\_010

**Reportes de Auditoría**

* + 1. **Reportes de Auditoría 002**

La Tabla 46 muestra el reporte de auditoría 002.

|  |  |
| --- | --- |
| ID | RA\_002 |
| Tipo de Reporte | Auditoría a la configuración funcional |
| Nombre del Reporte | Listado de funcionalidades programadas y que fueron subidas al repositorio de un proyecto en un rango de fecha determinado. |
| Propósito | El propósito de este reporte es verificar el avance de las funcionalidades programadas en una fecha determinada y verificar si cumplen con el cronograma de actividades del proyecto de software establecido. Historial de commits en un periodo de fechas determinado. |
| Entrada | * ID de proyecto (SPC) * Rango de fechas |
| Salidas | * Nombre de funcionalidad programada * Caso de uso asociado * Detalle del commit realizado * Fecha del commit realizado * Persona que realizo el commit * Ruta del código fuente en repositorio |

Tabla 46: Reporte de auditoría 002

Caso de Ejemplo:

La tabla 47 muestra el caso de ejemplo para el reporte de auditoría 002.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nombre de funcionalidad | Caso de uso asociado | Detalle del commit | Fecha del commit | Persona que realizó el commit | Ruta del código fuente |
| Login | CUS\_01 | Se agregó validación login,forgot password y logout | 01/05/2018 | Jair Barzola | Desarrollo/1.SPACIA/5.Codificacion/Android/Spacia/app/build.gradle |
| Gestionar Reservas | CUS\_04 | Se añadieron CSS al navegation Drawer | 3/05/2018 | Eduardo Andulo | Desarrollo/1.SPACIA/5.Codificacion/Web |

Tabla 47: Caso de ejemplo para RA\_002

* + 1. **Reportes de Auditoría 003**

La Tabla 48 muestra el reporte de auditoría 003.

|  |  |
| --- | --- |
| ID | RA\_003 |
| Tipo de Reporte | Auditoría a la configuración funcional |
| Nombre del Reporte | Listado de cambios realizados respecto a un caso de uso |
| Propósito | Verificar que el flujo tenga consistencia en lo que se realiza y que los cambios realizados no varíen demasiado el propósito del caso de uso. |
| Entrada | * ID del Caso de Uso (CUS1) * ID del proyecto (SPC) |
| Salidas | * Nombre del caso de uso * Descripción del caso de uso * Versión del caso de uso * Fecha de última modificación * Detalle de último cambio realizado * Autor del cambio realizado |

Tabla 48: Reporte de auditoría 003

Caso de Ejemplo:

La tabla 49 muestra el caso de ejemplo para el reporte de auditoría 003.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nombre del CUS | Descripción del CUS | Versión del CUS | Fecha de última modificación | Detalle de último cambio realizado | Autor del cambio realizado |
| Iniciar Sesión | Login | 2.8 | 01/05/2018 | Se agregó el módulo Login, splash screen | Jair Barzola |
| Iniciar Sesión | Login | 3 | 07/04/2018 | Se agregó validación login,forgot password y logout | Jair Barzola |

Tabla 49: Caso de ejemplo para RA\_003

* + 1. **Reportes de Auditoría 004**

En la Tabla 50 se muestra el reporte de auditoría 004.

|  |  |
| --- | --- |
| ID | RA\_004 |
| Tipo de Reporte | Auditoría a la configuración física |
| Nombre del Reporte | Listado de herramientas consideradas en la gestión de la configuración |
| Propósito | Este reporte tiene el propósito de verificar las herramientas consideradas en el plan de gestión de configuración y ver si se cumplen con el uso establecido en el documento. |
| Entrada | * Código de Área u Organización (ISS) * Plan de Gestión de la Configuración (PGC) |
| Salidas | * Nombre de la herramienta * Propósito de la herramienta * Versión en uso de la herramienta * Página oficial de la herramienta * Ultima fecha de actualización de la herramienta |

Tabla 50: Reporte de auditoría 004

Caso de Ejemplo:

La tabla 51 muestra el caso de ejemplo para el reporte de auditoría 004.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nombre de la herramienta | Propósito de la herramienta | Versión en uso de la herramienta | Ultima fecha de actualización de la herramienta | Página oficial de la herramienta |
| Git | Gestionar el control de versiones de los ítems de la configuración | 2.13.0 | 01/05/2018 | *https://git-scm.com/* |

Tabla 51: Caso de ejemplo para RA\_004

* 1. **Entrega y Gestión de Release**
     1. **Entrega o Pase a Producción**

El Pase a Producción debe ser considerado como una de las fases más importantes debido a que con esta el proyecto ya puede ponerse a disposición del usuario. No obstante, se identificaron los siguientes factores que dificultaban en un inicio el pase a producción: miedo al cambio, miedo al resultado, desinterés por el producto, búsqueda de la perfección, mala ejecución de los trabajos y entornos distintos entre estados de prueba pre-paso a producción y post-paso a producción.

Ante la imperativa necesidad de establecer un procedimiento estándar para esta fase, se deben identificar los roles que intervienen en la misma (Tabla Y):

|  |  |
| --- | --- |
| **ROLES** | **FUNCIONES** |
| **Jefe de Proyecto** | Responsable del desarrollo del software, mantiene la documentación técnica necesaria para el proceso de control de calidad. |
| **Coordinador de Calidad** | Responsable de supervisar y verificar la calidad del producto desarrollado, comprobando el cumplimiento de estándares, listas de chequeo y calidad de las funcionalidades del software y todos sus servicios. Da, conformidad al producto según las especificaciones dadas y solicita el Pase a Producción, verificando que la documentación requerida para el Pase a Producción se elabore correctamente. También, se encarga de dar conformidad al software posterior al Pase a Producción. |
| **Analista de Control de Calidad** | Responsable de realizar las pruebas funcionales y técnicas a los productos de software. |
| **Responsable de Pase a Producción** | Responsable de realizar los pases de producción de los productos de software, a los servidores de aplicaciones o a las tiendas de aplicaciones móviles. Además de registrar las incidencias y emitir informes de Pase a Producción. |

**Tabla Y: Roles que intervienen en el Pase a Producción**

* + - 1. **Procedimiento de Pase a Producción**

1. **Liberación del producto**

Una vez culminada la construcción del sistema, el Jefe de Proyecto procede a designar un responsable para el Pase a Producción.

El Responsable del Pase a Producción, recibe el versionamiento del producto y/o el link de almacenamiento, asegurándose que los productos recibidos estén conformes y procede a entregar el producto liberado (Instaladores, UpDate, Scripts).

El producto es entregado paralelamente a:

* Control de calidad para las pruebas funcionales.
* Pruebas técnicas y de Seguridad.

1. **Ejecución de pruebas funcionales**

El trabajo del área de Control de Calidad, consiste en la revisión de los aspectos funcionales del sistema, para que el entregable sea liberado con un nivel de errores cero.

El coordinador de Control de Calidad asigna el equipo revisor del Sistema.

De existir observaciones, el coordinador de Control de Calidad, solicitará el levantamiento de observaciones, mediante un e-mail al Jefe de Proyecto con comunicación al Responsable del Pase a Producción.

Una vez que el Sistema está libre de errores, se compila la versión final y se procede con la entrega del producto para las Pruebas técnicas y de Seguridad.

1. **Ejecución de pruebas técnicas y de seguridad de los SI de OGTI**

El Jefe de Pruebas Técnicas y de Seguridad, recibe el ejecutable y procede a designar un responsable para la ejecución de las pruebas técnicas y de seguridad.

El responsable del Pase a Producción procede a verificar el recurso y disponer del

entorno de producción (si es web, un servidor y un navegador; y si es mobile, un dispositivo móvil con los requerimientos necesarios), conforme las especificaciones técnicas elaboradas por el equipo de Desarrollo.

Seguidamente se ejecutan las pruebas de operación, si producto de la ejecución,

existen errores o inconvenientes se procede con lo siguiente:

* Elaborar un informe de las incidencias, y Remitir al Responsable del Pase a Producción para el levantamiento de las observaciones correspondientes. Además de informar al Coordinador de Operaciones para conocimiento y fines respectivos.

1. **Verificación por control de calidad**

El Coordinador de Control de Calidad toma conocimiento de la conformidad de la

ejecución de las pruebas por el Jefe de Pruebas Técnicas y de Seguridad y procede a revisar el software para asegurar que el entregable a desplegar es la misma versión revisada por el área de control de calidad.

De estar conforme, el Coordinador de Control de Calidad, solicita al responsable del

Pase a Producción, se encargue de enviar el formato digitalizado a la cuenta del

correo electrónico [nnconsulting@solutions.com](mailto:nnconsulting@solutions.com).

1. **Liberación del aplicativo.**

Procede la liberación del producto cuando se han culminado satisfactoriamente las

pruebas de funcionales, técnicas y de seguridad y se cuenta con la validación de la versión del software por parte del área de control de calidad.

Finalmente, el operador de turno firma el formato del Pase a Producción y lo digitaliza y almacena.

* + - 1. **Estructura de Paquete de Liberación**

A continuación, se definirán las estructuras de liberación, tanto para sistemas web como para aplicaciones móviles.

**Para sistemas web:**

|  |
| --- |
| La documentación, scripts, archivos fuentes (dependiendo de lo acordado y contexto), ejecutables, API (dependiendo de lo acordado y contexto-ej. servidor del cliente) u otro documento involucrado en el pase a producción serán entregados a la oficina de Cómputo por el Coordinador de Calidad. |

**Para aplicaciones móviles:**

|  |
| --- |
| La documentación, scripts, archivos fuentes (dependiendo de lo acordado y contexto), APK, API (dependiendo de lo acordado y contexto) u otro documento involucrado en el pase a producción serán entregados a la oficina de Cómputo por el Coordinador de Calidad. |

* + - 1. **Formato de Documento de Liberación**

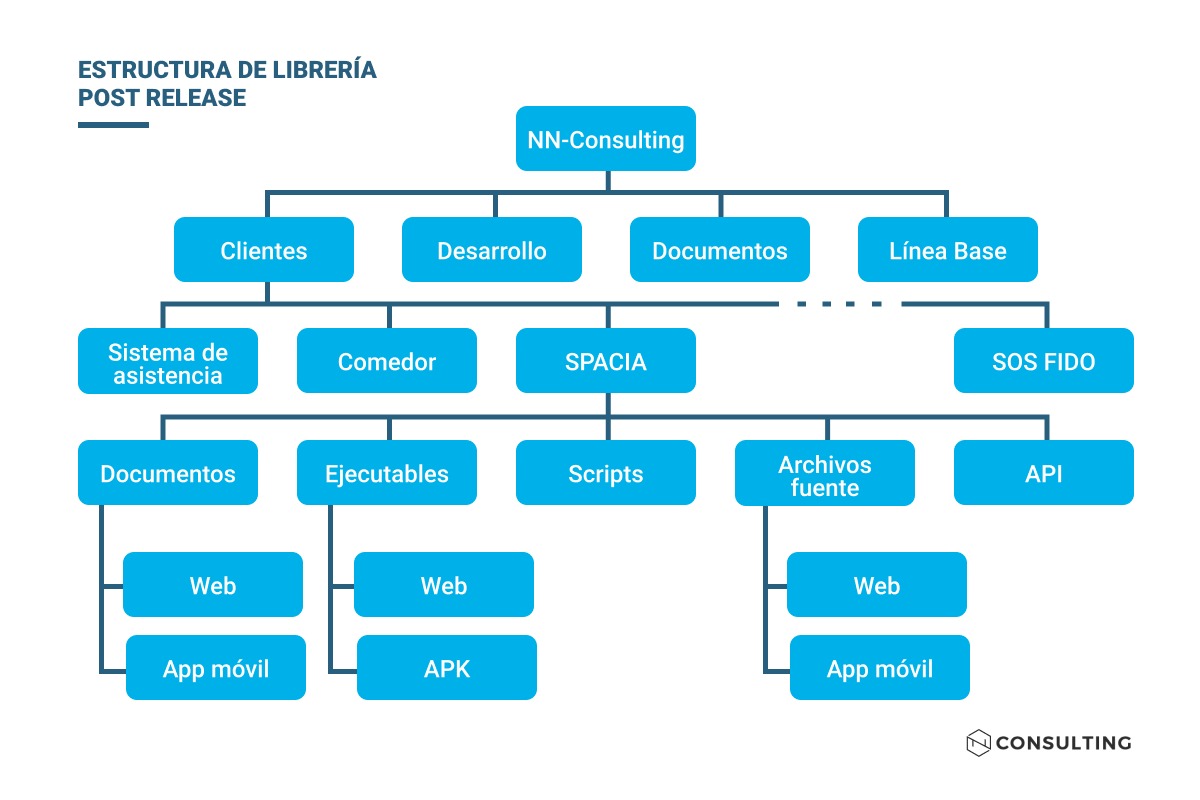
La Tabla X muestra el formato de Pase a Producción, el cual está dividido en cuatro partes: Datos generales de la solicitud, Datos generales del producto de software, Datos técnicos del producto de software y Documentos relacionados al release.

|  |  |  |  |
| --- | --- | --- | --- |
| **FORMATO DE PASE A PRODUCCIÓN** | | | |
| **Número de pase** |  | **Fecha y hora de solicitud** |  |
| **Responsable de Pase** |  | **Firma y Fecha/Hora** |  |
| **Responsable de Calidad** |  | **Firma y Fecha/Hora** |  |
| **Jefe de Proyecto** |  | **Firma y Fecha/Hora** |  |
| **Datos generales del producto de software** | | | |
| **Tipo de software** | **Web ( ) API ( ) App ( )** | | |
| **Tipo de Cliente** | **Interno ( ) Externo ( )** | | |
| **Objetivo de la versión** |  | | |
| **Datos técnicos del producto de software** | | | |
| **Sistema** |  |  |  |
| **Módulo** |  |  |  |
| **Función Principal** |  |  |  |
| **Versión** |  |  |  |
| **Detalles técnicos** | **Servidor:** | **Lenguaje de programación:** | **IDE:** |
|  | **Framework:** | **Librerías de despliegue:** |  |
|  | **Gestor de Base de Datos:** | **Otros:** |  |
| **Documentos relacionados al release** | | | |
|  | | | |

**Tabla X: Formato de Pase a Producción**

* + 1. **Gestión de Release**
       1. **Librería actualizada**

La Figura X muestra la estructura de las librerías post-release. Se observa el ejemplo para el sistema SPACIA, el cual deberá tomarse de referencia para el resto de proyectos de software. Es importante indicar que, los archivos a entregarse a los clientes varían de acuerdo a lo pactado al comienzo del proyecto junto con el cliente.

****

**Figura X: Estructura del repositorio post-release**