**Plan de Gestión de la Configuración**

**Consultora ElmerX**

**Versión 1.2**

**Historial de Revisiones**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 14/09/2018 | 1.0 | Creación del documento | Steve Sader Matos Manguinuri |
| 14/09/2018 | 1.1 | Introduccion Agregada | Todo el equipo |
| 12/10/10 | 1.2 | Correcciones y Nomenclaturas Hechas | Todo el equipo. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**ÍNDICE**

[1. Planificación 3](#_Toc527109532)

[1.1. Introducción 3](#_Toc527109533)

[1.2. Roles, responsabilidades y cantidad 3](#_Toc527109534)

[1.3. Políticas, Directrices y procedimientos 4](#_Toc527109535)

[1.4. Herramientas, entorno e infraestructura 6](#_Toc527109536)

[1.5. Calendario 6](#_Toc527109537)

[2. Identificación 8](#_Toc527109538)

[2.1. Lista de clasificación de CI 8](#_Toc527109539)

[2.2. Definición de la Nomenclatura 8](#_Toc527109540)

# 

# Planificación

## Introducción

Una de las problemáticas más comunes que tenemos en nuestra consultora se generan cuando desarrollamos un proyecto de software, pues lo miembros del equipo de desarrollo hacen uso de diferentes repositorios, esto genera que el jefe de proyecto no haga un control eficiente de las versiones de los elementos, además de generar retrasos en las entregas, errores que no se tenían planificados.

Estos problemas se generan por una falta de comunicación entre los miembros del equipo y el jefe de proyecto, así como también la falta de capacitación en el uso de repositorios como github, los miembros del equipo no saben lo que deben hacer cada vez que quieren cambiar algo es por eso que cada uno trabaja por su propia cuenta.

Y en los casos que se trabaja en un mismo repositorio, se nos presenta el problema de combinar los documentos en Word. En ocasiones ocurren conflictos que dificultan el avance del proyecto.

Se desarrolla este plan con el propósito de establecer pautas para un mejor desempeño de los equipos de trabajo.

El objetivo de este plan es poder llevar un control de los proyectos, recursos y los tiempos para el desarrollo.

Y se logrará a través de este documento, donde se hará mantenimiento de todos los proyectos.

## Roles, responsabilidades y cantidad

|  |  |  |
| --- | --- | --- |
| Roles | Responsabilidades | Cantidad |
| Gerente General | Establecer un control de cambio de los proyectos de la empresa, conjuntamente con el documento de Gestión de la configuración. | 1 |
| Gestor de la configuración | Se encargara de gestionar la infraestructura global de la gestión de la configuración y el entorno del equipo de desarrollo de los productos | 2 |
| Gerente funcional | Se encargara de aprobar y vigilar la utilización de los recursos para los proyecto sin afectar notoriamente el capital o recursos internos de la empresa. | 1 |
| Bibliotecario | Mantiene los documentos y código controlado.  Además trabaja en conjunto con el gestor de la configuración. | 4 |
| Miembros del equipo | Personas que formaran parte del equipo operativo de los proyectos. | 20 |
| Auditor | Auditar la gestión de la configuración. Velar por la calidad de la gestión de configuración. | 2 |

## Políticas, Directrices y procedimientos

Las políticas, directrices y procedimientos a utilizarse son los siguientes:

|  |  |
| --- | --- |
| Documento | Origen |
| Documento de privilegios | Documentos |
| Manual de procesos | Documentos |
| Políticas de la empresa | Documentos |
| Estándares de la empresa | Documentos |
| Manual de buenas practicas | Documentos |

**Lineamientos Generales**

1. Los proyectos deberán ser ejecutado en iteraciones incrementales con una demostración del producto al finalizar cada iteración: con esta política, se conocerá el estado del proyecto, evaluando si los requisitos cumplen con las expectativas del cliente, si la calidad es la esperada, o si hay retrasos; agilizando la toma de decisiones correctivas.
2. Los requisitos se desarrollarán priorizados por el valor aportado al cliente: Esta política permitirá que los objetivos más importantes del proyecto sean atendidos.
3. El control y seguimiento de los proyectos se basará en los requisitos completados en cada iteración. Se entiende como un requisito, los entregables asociados a: análisis, desarrollo, pruebas, documentación, etc. e integrados con los entregables de las iteraciones anteriores. (ElmerX/NombreProyecto/Gestión)
4. Cada requisito debe ser independiente del resto de los requisitos, en la medida de lo posible. (ElmerX/NombreProyecto/Análisis)
5. Cada requisito debe ser demostrable, permitiendo cómo comprobar con el cliente que el requisito está completado y que se cumplen sus expectativas. (ElmerX/NombreProyecto/Análisis)
6. El requisito debe ser de un grado de esfuerzo para ser completado semejante al del resto de requisitos: de manera que la organización y el cliente, puedan realizar una extrapolación del progreso del proyecto. (ElmerX/NombreProyecto/Análisis)

**Desarrollo** (ElmerX/NombreProyecto/Desarrollo)

1. Los componentes de software deberán ser desarrollados y liberados por partes, y no entregados al final del proyecto.
2. El desarrollo del componente de software que conformarán la solución, deberá ser liberados en varias iteraciones.
3. Cada iteración deberá producir software con calidad de producción, probado, integrado, y documentado (funcional, técnica).
4. Cada iteración deberá cumplir con un subconjunto de requerimientos.
5. Cada iteración deberá contemplar (análisis, diseño, implementación, documentación, etc.).

**Pruebas**

1. Cada proyecto debe incorporar las prácticas de TDD (Test Driven Development).
2. Cada uno de los entregables, deberá contener scripts de pruebas unitarias, integrales, funcionales, etc; mediante la utilización de frameworks como junit, dbunit, mockObjtects, etc.

**Documentación**

1. La documentación de los proyectos, específicamente:  manual de usuario, manual de operaciones, arquitectura de la solución, especificaciones, etc; deberán ser entregables parciales para cada una de las iteraciones, es decir, la documentación no se liberará al final del proyecto, sino en entregables parciales. (ElmerX/NombreProyecto/Análisis)

**Control de Calidad**

1. Cada uno de los entregables, serán sometidos a un script de calidad, que ejecutara la organización, y no serán admitidos como productos del proyecto hasta alcanzar un nivel aceptable. (ElmerX/NombreProyecto/Gestión)

**Control de Riesgos**

1. Los riesgos serán identificados en la primera iteración, llevándose a cabo también una valoración inicial de la exposición al riesgo y planes de contingencia. En cada iteración se revisará y actualizará el documento “Lista de Riesgos”, añadiendo además la lista de riesgos más importantes actualizada por cada iteración. (ElmerX/NombreProyecto/Gestión)

**Control de Artefactos**

1. Cada uno de los artefactos del proyecto, deberán ser mantenidos bajo un sistema de control de versiones.
2. La organización disponibilizara un sistema de control de versiones, que deberá ser actualizado por el cliente de forma remota.

## Herramientas, entorno e infraestructura

|  |  |
| --- | --- |
| HERRAMIENTA | DESCRIPCION |
| Git | Software de control de versiones. Su propósito es llevar registro de los cambios en archivos de computadora y coordinar el trabajo que varias personas realizan sobre archivos compartidos. |
| Spring Boot | Spring Boot facilita la creación de aplicaciones basadas en Spring independientes y de grado de producción que puede "simplemente ejecutar".  Tomamos una opinión de la plataforma Spring y las bibliotecas de terceros para que pueda comenzar con el mínimo esfuerzo. La mayoría de las aplicaciones Spring Boot necesitan muy poca configuración Spring. |
| Glassfish | Servidor de aplicaciones para la plataforma Java EE. Se usará para agilizar y proporcionar una estructura adecuada al proyecto. |
| Android Studio | IDE oficial de Android. Está diseñado específicamente para que Android acelere su desarrollo y lo ayude a crear aplicaciones de la más alta calidad para cada dispositivo Android. |
| Postgresql | Base de datos relacional, se usará como contenedor de datos para la aplicación. |
| Heroku | Plataforma de computación en la nube, proporciona un conjunto de herramientas de desarrollo y servicios específicos de la aplicación bajo una sola plataforma para desarrollar, probar, implementar y administrar el proceso de desarrollo de aplicaciones. Servirá como plataforma de despliegue para la aplicación. |
| Amazon Web Services (AWS) | Plataforma de computación en la nube, que se usara como entorno de despliegue de la base de datos y del API. |

## Calendario

|  |  |  |  |
| --- | --- | --- | --- |
| ACTIVIDAD | | TIEMPO (días) | ROL |
|  | |  |  |
| 1 | Planificación de la SCM |  |  |
|  | Identificar la problemática de la empresa | 1 | Gestor de la Configuración |
|  | Definir el alcance | 1 | Gestor de la Configuración |
|  | Definir el propósito y finalidad del plan | 2 | Gestor de la Configuración |
|  | Identificar políticas y procedimientos | 2 | Gestor de la Configuración |
|  | Analizar herramientas | 3 | Gestor de la Configuración |
|  | Definir tareas | 3 | Gestor de la Configuración |
|  |  |  |  |
| 2 | Identificación de la SCM |  |  |
|  | Definir Ítems | 2 | Bibliotecario |
|  | Definir nomenclatura de elementos | 1 | Bibliotecario |
|  | Definir la estructura del repositorio | 1 | Bibliotecario |
|  | Plantear privilegios y seguridad | 3 | Bibliotecario |
|  | Inventariar los Items | 4 | Bibliotecario |
|  |  |  |  |
| 3 | Control de la SCM |  |  |
|  | Controlar los Ítems | 4 | Comité de Control de Cambios |
|  |  |  |  |
| 4 | Estado de la SCM |  |  |
|  | Revisar el estado de la SCM | 2 | Gestor de la Configuración |
|  | Analizar el estado de la SCM | 2 | Gestor de la Configuración |
|  |  |  |  |
| 5 | Estado de la contabilidad de la SCM |  |  |
|  | Revisar el estado de la contabilidad de la SCM | 2 | Gestor de la Configuración |
|  | Analizar el estado de la contabilidad de la SCM | 2 | Gestor de la Configuración |
|  |  |  |  |
| 6 | Auditoria de la SCM |  |  |
|  | Auditar la configuración | 4 | Auditor |
|  |  |  |  |
| 7 | Gestión y monitorea de la SCM |  |  |
|  | Monitorear la configuración | 2 | Gestor de la Configuración |
|  | Controlar la configuración | 2 | Gestor de la Configuración |

# Identificación

## Lista de clasificación de CI

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tipo  (E = Evolución  F = Fuente  S = Soporte) | Nombre del item | Fuente  (E = Empresa  P = Proyecto  C = Cliente  P = Proveedor) | Extensión | Proyecto |
| E | Documento de Negocio | P | DN | STSM |
| E | Plan de proyecto | P | PP | STSM |
| E | Plan Gestión de Configuración | E | PGC | - |

## Definición de la Nomenclatura

CASO 1: Para los elementos que no pertenecen a un proyecto, la nomenclatura es la siguiente:

|  |
| --- |
| Acrónimo del ítem |

Ejemplo:

* Plan de gestión de la configuración:

Acrónimo del ítem: PGC

* Política de gestión de proyectos de software:

Acrónimo del ítem: PGPS

CASO 2: Para los elementos que pertenecen a un proyecto, la nomenclatura es la siguiente:

|  |
| --- |
| Acronimo del proyecto + “\_” + Acronimo del item |

Ejemplo(s):

* Documento de Análisis:

Acrónimo del ítem: STSM\_DA

CASO 3: Para los casos de uso de un proyecto, la nomenclatura es la siguiente:

|  |
| --- |
| Acrónimo del proyecto + ”\_” + CU + ”\_” +             Tres primeras                                                                         letras del nombre del caso de uso |

Ejemplo(s):

* Caso de uso 001: Registro de viaje

STSM\_CU001\_REGVIA

* Caso de uso 002: Administración de pasajeros

STSM\_CU002\_ADMPAS

CASO 4: Para elementos que son código fuente de un proyecto, la nomenclatura es la siguiente:

|  |
| --- |
| Acrónimo del proyecto + “\_” + CF + “\_” + Tipo de Software |

Ejemplo(s):

* STSM\_CF\_BACK
* STSM\_CF\_FRONT