Ingeniería de Software

Trabajo Práctico 1

Plan de Gestión de las Configuraciones

Proyecto “.”

|  |  |
| --- | --- |
| Profesor: | Julián Nonino |
| Grupo: | Fuplace |
| Integrantes: | Vignolo, Gabriel Enrique |
|  | Losano Quintana, Juan Cruz |
|  | Velazquez, Cristian |

**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versión** | **Fecha** | **Cambios** | **Autores** |  |
|  |  |  |  |  |
| 1.0.0 | 27-Abr-2018 | Inicio de Configuration Management | Vignolo, Gabriel Enrique  Losano Quintana, Juan Cruz |  |
|  |  |
|  |  | Velazquez, Cristian David |  |
| 2.0.0 | 11-May-2018 | Versión 2 | Vignolo, Gabriel Enrique  Losano Quintana, Juan Cruz |  |
|  |  |
|  |  | Velazquez, Cristian |  |
| 2.1.0 | 11-May-2018 | Inicio secciones 2, 3 y 4 | Vignolo, Gabriel Enrique  Losano Quintana, Juan Cruz  Velazquez, Cristian David |  |
|  |  |  |  |  |
| 2.2.0 | 11-May-2018 | Realización sección 2.1 y 2.2 | Velazquez, Cristian David |  |
|  |  |  |  |  |
| 2.3.0 | 11-May-2018 | Realización sección 3.1, 3.2, 3.2.1, 3.2.1.1, 3.2.1.2, 3.2.1.3 | Vignolo, Gabriel Enrique |  |
|  |  |  |  |  |
| 2.4.0 | 11-May-2018 | Realización sección 4.3 | Losano Quintana, Juan Cruz |  |
|  |  |  |  |  |

Tabla de Contenidos

[1 Introducción 4](#_Toc481947170)

[1.1 Propuesta y alcance 4](#_Toc481947171)

[1.2 Propósito de la administración de la configuración del software (SCM) 4](#_Toc481947172)

[1.3 Referencias, abreviaturas y glosario 4](#_Toc481947172)

[1.4 Herramientas de Gestión de las Configuraciones 4](#_Toc481947173)

[2 Roles y Responsabilidades 5](#_Toc481947177)

[2.1 Administradores de la configuración del proyecto 5](#_Toc481947178)

[2.2 Responsabilidades de la administración de la configuración 5](#_Toc481947178)

[3 Gestión de Cambios 6](#_Toc481947180)

[3.1 Alcance 6](#_Toc481947181)

[3.2 Versiones principales y módulos para el cliente 6](#_Toc481947181)

[3.2.1 Consejo de control del cambio (CCB) 7](#_Toc481947182)

[3.2.1.1 Miembros 7](#_Toc481947183)

[3.2.1.2 Frecuencia de reuniones 8](#_Toc481947184)

[3.2.1.3 Herramienta de gestión de cambios 8](#_Toc481947185)

[4 Gestión de la Configuración del Código Fuente 8](#_Toc481947186)

[4.1 Esquema de directorios 8](#_Toc481947187)

[4.2 Normas de etiquetado y nombramiento de archivos. 8](#_Toc481947188)

[4.3 Plan del esquema de ramas 9](#_Toc481947189)

[4.3.1 Rama Principal 9](#_Toc481947190)

[4.3.2 Ramas de desarrollo 9](#_Toc481947191)

[4.3.3 Ramas de Lanzamiento (Release) 9](#_Toc481947192)

[4.4 Políticas de fusión 9](#_Toc481947193)

[4.5 Forma de entrega de los “releases” 10](#_Toc481947194)

**1 INTRODUCCIÓN**

**1.1** **Propuesta y alcance**

Este documento cubre el Plan de Gestión de las Configuraciones para el proyecto “.”. El propósito del Plan CM es controlar la configuración de los requisitos, documentos, hardware, software y herramientas utilizados para este proyecto.

SCM es el proceso mediante el cual se identifican métodos y herramientas para controlar el software a lo largo de su desarrollo y uso. Se definirán las distintas tareas que se deben realizar y como estas tareas afectaran al desarrollo del proyecto.

**1.2** **Propósito de la administración de la configuración del software (SCM)**

● Asegurar la consistencia de la información poniendo en práctica la SCM.

● Definir las personas que le dan soporte a las prácticas de SCM.

● Mantener la integridad a los largo de todo el ciclo de vida del producto.

● Informar a grupos y las personas el estado del proyecto.

● Crear un historial del estado anterior y actual del proyecto.

● Mejora de Procesos.

**1.3** **Referencias, abreviaturas y glosario**

|  |  |
| --- | --- |
| **Abreviatura** | **Descripción** |
|  |  |
| SCM | Administración de la configuración del software (Software Configuration Management) |

**1.4** **Herramientas de Gestión de las Configuraciones**

|  |  |  |
| --- | --- | --- |
| Referencia | Propósito | Link |
|  | Entorno de desarrollo. |  |
| Java | Lenguaje de programación. | https[://www.java.com](http://www.java.com/) |
| Git | Software de manejo de versiones. | https[://www.gitscm.com](http://www.gitscm.com/) |
| GitHub | Sistema de control de versiones. | https[://www.github.com](http://www.github.com/) |
| Git Issues | Sistema de seguimiento de errores. | https[://www.github.com/issues](http://www.github.com/issues) |
| Travis | Herramienta de integración. | https[://www.travis.org](http://www.travis.org/) |
| Gradle | Herramienta de automatización. | [http://www.gradle.org](http://www.gradle.org/) |
| Visual Paradigm | Aplicación orientada a la creación de diagramas UML. |  |

**2 ROLES Y RESPONSABILIDADES**

**2.1 Administradores de la configuración del proyecto**

La siguiente tabla muestra personas que tomarán el rol de Gerentes de Configuración Global:

|  |  |  |
| --- | --- | --- |
| Rol | Titular | Suplente |
| Global PCM | Velazquez, Cristian David | Losano Quintana, Juan Cruz |

**2.2 Responsabilidades de la administración de la configuración**

La Administración de la configuración del proyecto está a cargo del Global PCM, él es el responsable de actividades como seguimiento de las herramientas, creación de los nuevos branches, creación del release, etc.

|  |  |
| --- | --- |
| Rol | Responsabilidades |
| GPCM | Posee toda responsabilidad sobre todos los CI. Responsabilidad en la creación de branches y administración de sus políticas.  Responsabilidad y asistencia sobre el etiquetado y lanzamiento de branches.  Coordinar actividades del CM en el proyecto. Asegurar la correcta ejecución del esquema del CM. Participación en auditorias.  Analizar todas las novedades relacionadas al CM. |
| TPCM | Asistencia en la creación de tags y branches. Creación de actividades para el equipo específico en branches.  Garantizar la integridad del producto y el seguimiento de los elementos de configuración propios del equipo.  Participación en auditorias.  Analizar todas las novedades relacionadas al CM. |
| Team | Ayudar a resolver conflictos durante la actividad de merge.  Asegurarse que los criterios de calidad de los entregables a la rama principal se cumplan. Seguir todos los procesos asociados, políticas y prácticas definidas por sus roles asignados. |

**3 ADMINISTRACIÓN DE CAMBIOS**

**3.1 Alcance**

La administración de cambios es un proceso que ocurre después de que se identifica y aprueba la línea de base de la documentación, el código fuente o el hardware del producto. Los cambios a gestionar también serán internos, ya sea debido a resultados de simulación o pruebas e incluso por solicitudes externas en busca de cambios de características o funcionalidades.

El alcance del trabajo será aprobar o rechazar cambios necesarios en planes, documentos y código. Se tomarán decisiones sobre las acciones a realizar basadas en el resultado de las actividades de aseguramiento de la calidad del producto y la salud del producto después de cada ciclo de prueba.

**3.2 Versiones principales y módulos para el cliente**

**3.2.1 Consejo de control del cambio (CCB)**

Es un comité que garantiza que cada cambio sea considerado adecuadamente por todas las partes y que esté autorizado antes de su implementación. Es responsable de aprobar, supervisar y controlar las solicitudes de cambio para establecer líneas base de elementos de configuración.

Los elementos a ser revisados por el CCB son aquellos cambios presentados por el Coordinador de Emisión, el presidente de CCB u otros miembros de CCB.

El alcance del trabajo será aprobar / rechazar los cambios necesarios en los planes, documentos y códigos. Las decisiones deberán tomarse sobre las medidas que se tomarán en función del resultado de las actividades de garantía de calidad del producto y salud del producto después de cada ciclo de prueba.

**3.2.1.1 Miembros**

|  |  |
| --- | --- |
| **Integrantes** | **Rol en el CCB** |
| Velazquez, Cristian David | GPCM  Gerente de ingeniería |
| Vignolo, Gabriel Enrique | Director de ingeniería  Administrador de versiones – coordinador de problemas |
| Losano Quintana, Juan Cruz | GPCM suplente  Uber Scrum Team |

**3.2.1.2 Frecuencia de reuniones**

Tendremos reuniones semanales o en base a la demanda que tengamos. Estas serán generalmente para organizarnos e informarnos, ya que la mayor parte del trabajo se va a realizar por módulos y según la disposición del tiempo de los integrantes.

**3.2.1.3 Herramienta de gestión de cambios**

Se utilizará la herramienta GitHub para realizar el control de versiones. GitHub maneja las versiones del proyecto, registrando los cambios entre ellas y facilitando la comparación e incorporación de modificaciones ya sea la producción de código nuevo o la corrección de errores desde varios colaboradores.

**4 GESTIÓN DE LA CONFIGURACIÓN DEL CÓDIGO FUENTE**

**4.1 Esquema de directorios**

**4.2 Normas de etiquetado y de nombramiento de los archivos**

**4.3 Plan de esquemas de ramas a usar**

*Rama de integración:* se define como la rama de integración principal donde se llevarán a cabo todas las funciones liberadas. Esta rama se crea de forma predeterminada por CVS cuando se agrega un archivo al control de origen.

*Ramas de desarrollo:* son aquellas ramas donde se codifica el desarrollo de nuevas características / funcionalidades del programa. La rama de desarrollo que se va a utilizar es:

- Ramales de Sprint: este tipo de ramales serán creados por el CM y son utilizados por los equipos de scrum. Aquí se comprometerán el desarrollo de historias hasta que se fusionen con la rama de integración. Se recomienda que se creen uno por sprint y también se pueden crear por equipo y / o muchos equipos de scrum que pueden trabajar juntos en la misma rama de sprint según las circunstancias.

*Ramas de lanzamiento:* este tipo de ramas debe ser creado por el CM y se crean para indicar aquí las versiones de código de una versión específica y se bifurcarán desde la rama de integración. El nombre de la rama Release será branch- <versión> -rel

**4.4 Políticas de fusión de archivos y de etiquetado**

**4.5 Forma de entrega de los “releases”**