

Gastar Group

Reproductor de Música

Configuration Managment Plan (CM_Plan)

Autores:
<Gasparini, Roman;
Tarres, Martin>

Document Version: 1.0.0

Historial de Revisiones

Version	Fecha	Resumen de Cambios	
1.0.0	27/04/2017	Primera versión	

Página de aprobación

En la siguiente tabla se listan las personas encargas de cada area, y a las que se deberá acudir en caso de que se deba realizar algun cambio mayor.

La aprobacion no es necesaria si se trata de un cambio menor.

Sólo se realizan cuando el cambio es solicitado por una persona ajena al CM.

Configuration Management	Tarres, Martín
Software Developer	Gasparini, Roman

Tabla de contenidos

1 Introducción

- 1.1 Propósito y Alcance
- 1.2 Propósito de Prácticas del SMP
- 1.3 Acrónimos
- 1.4 Herramientas del Plan de CM

2 Roles en Gestión de Configuración

- 2.1. Configuration Manager del proyecto
- 2.2 Responsabilidades en el Plan de Gestión de Configuraciones

3 Gestión de Cambios

- 3.1 Alcance
- 3.2 Módulos de clientes y releases
 - 3.2.1.1 Miembros
 - 3.2.1.2 Frecuencia
 - 3.2.1.3 Herramientas de gestión de cambios

4 Identificación de Configuraciones

5 Equipos de Desarrollo de Proyecto

6 Gestión de la Configuración del Código Fuente

7 Build Management

8 Release Management

1 Introducción

1.1 Propósito y Alcance

Dentro del siguiente documento podemos encontrar descripciones detalladas sobre el Plan de Gestión de Configuraciones para el proyecto de un reproductor de música. El objetivo de este plan es controlar la configuración de los requerimientos, documentos, software y herramientas usadas en este proyecto.

Las herramientas se utilizarán para el control de versiones, para la integración continua, gestión de defectos, entre otros.

La gestión de las configuraciones de software (SCM) consiste en las herramientas y métodos utilizados para controlar el software a lo largo de su desarrollo y uso.

1.2. Propósito de Prácticas del SMP

- Garantizar la consistencia al poner en práctica las actividades de CM.
- Definir las autoridades que soporten las prácticas del CM.
- Mantener la integridad del producto a lo largo de su ciclo de vida.
- Informar a grupos e individuos interesados sobre el estado del proyecto.
- Crear un historial de los estados actual y anteriores de los productos desarrollados.
- Mejoras en el proceso de desarrollo.

1.3. Acrónimos

Acrónimo	Descripción
CM	Control de configuraciones
CBB	ChangeControl Board
SCM	Gestión de la configuración de Software
SMP	Plan de gestión de software

1.4. Herramientas del Plan de CM

Herramienta/Proceso	Propósito
Intellij IDEA	Entorno de desarrollo del proyecto
GitHub	Servicio de repositorio y hosting de ítems puestos en control de configuración.
Travis CI	Servicio de Integración Continua

2 Roles Gestión de Configuraciones

2.1. Configuration Manager del proyecto

Las actividades de gestión de las configuraciones dentro del proyecto “Reproductor de música” son coordinadas por el Manager Global del Proyecto de Configuraciones (GPCM), rol asignado a una persona.

El GPCM será responsable por actividades como controlar las ramas principales y las versiones de lanzamiento, determinando cuándo deben ser creadas las ramas y que actividades y características corresponden a cada rama.

Las actividades de manejo de configuraciones, procesos y sus respectivos procedimientos deben ser seguidas y respetadas por todos los miembros. Es la responsabilidad de cada persona seguir y aplicar el procedimiento de CM apropiado, de acuerdo con sus rol.

2.2 Responsabilidades en el Plan de Gestión de Configuraciones

Rol	Responsabilidades
-----	-------------------

Configuration Manager	<p>Posee la responsabilidad global de todos los ítems en configuración.</p> <p>Responsable de aplicar etiquetas en los releases.</p> <p>Coordinar actividades de CM dentro del proyecto.</p> <p>Responsable de la creación de ramas y administrar las mismas.</p> <p>Asistir en actividades de merge y de construcción.</p> <p>Asegurarse de la integridad y rastreabilidad de los ítems en configuración del proyecto.</p> <p>Generar nuevo código con nuevas características periódicamente.</p> <p>Ayudar a resolver conflictos en las actividades de construcción y merge.</p> <p>Cumplir los requisitos de calidad en los distintos branches.</p>
Software Developer	<p>Seguir las practicas recomendadas en cada etapa del proyecto.</p> <p>Corregir bugs encontrados en el código.</p> <p>Asegurarse de la integridad del producto y su rastreabilidad a lo largo del tiempo.</p>

3 Gestión de Cambios

3.1 Alcance

La gestión de cambios es un proceso que ocurre después de que la documentación y el código fuente están identificados y aprobados.

Los cambios incluyen modificaciones internas al enfoque documentado originalmente, debido a resultado de tests o solicitudes de integración de nuevas funciones.

3.2 Módulos de clientes y releases

3.2.1 CCB (Change Control Board)

La CCB es un comité que se asegura de que cada modificación es considerada

apropiadamente, y autorizada antes de su implementación. La junta es responsable de aprobar, monitorear y controlar las solicitudes de cambios para establecer ramas de desarrollo de ítems de configuración. Las decisiones sobre las solicitudes de cambios son tomadas según el resultado de tests de calidad del producto.

Los integrantes de la junta se contactaran via email o por teléfono para concertar una reunión extraordinaria, o discutir cambios referidos al proyecto fuera de ellas.

3.2.1.1 Miembros

La siguiente tabla muestra los integrantes del equipo que asiste a las reuniones de la CBB.

Rol en la CBB	Nombre
Software development	Gasparini, Roman
Configuration Manager	Tarres, Martin

3.2.1.2 Frecuencia

Reunión de la CBB	Frecuencia
Reproductor de Música	2 veces por semana

3.2.1.3 Herramientas de gestión de cambios

Se usará un repositorio de Github para gestionar cambios y nuevas características del software a desarrollar.

URL: [Repositorio GitHub](#)

4 Identificación de Configuraciones

El repositorio de Github será usado para guardar y compartir recursos de diferentes releases. Por cada Release, se generará un tag que identifique y permita separar una versión de otra.

5 Equipos de Desarrollo de Proyecto

El código fuente sera gestionado por los siguientes equipos

Release Management Team: está a cargo del testing necesario para que el producto pueda llegar a ser lanzado al consumidor, asegurando su calidad. Está a cargo de la mantención del repositorio y la gestión de los tests automáticos que se realizan luego de cada commit, así como también de crear y mantener la documentación que

será entregada a los consumidores.

Development Team: a cargo de desarrollar nuevas funcionalidades, subir el código al branch correspondiente y hacer el merge a la rama adecuada según el calendario del proyecto. También pueden trabajar arreglando bugs o testeando características.

6 Gestión de la Configuración del Código Fuente

En esta sección se describen diferentes ítems de gestión de código fuente. Abarca aspectos referidos a esquema de ramificado, etiquetado, estrategia de fusión de código y niveles esperados de calidad para la totalidad del producto.

7 Build Management

El tipo de Build a utilizar y que será implementado en todas las ramas es:

Continuous Integration Builds: Son útiles para identificar tan pronto como sea posible errores en el código y así poder arreglarlos antes de hacer algún build formal. Están configurados para no aplicar ningún tag y para ser corridos automáticamente cada vez que alguien realice un commit a cualquiera de las ramas.

En la pestaña correspondiente de Travis CI se puede encontrar información útil, tal como el resumen de los tests empleados, el autor, la fecha del commit, etc.

La herramienta a utilizar para manejar el proceso de builds definido será Travis CI, que será debidamente configurada para enviar correos electrónicos a los grupos de trabajo que realicen el commit avisando de alguna falla en el build.

8 Release Management

Las Release builds serán realizadas en las ramas de desarrollo correspondiente. Estos serán enviados a los clientes por medio del repositorio de GitHub dándoles acceso al mismo para que descarguen los archivos necesarios, y puedan probar el proyecto en su IDE de Java de preferencia.

