

# **Perseverance**

## **Solitario Apolo**

### **Plan de Gestión de Configuración**

**Autores:**

- Espósito, Javier
- Sartori, Gastón Emilio
- Tántera, Julián
- Zanotti, Emiliano

**Fecha:** 5 de Mayo de 2021

**Versión:** 1.0.1

# ÍNDICE

<b>Historial de Cambios</b>	<b>3</b>
<b>Introducción</b>	<b>5</b>
Propósito y alcance del plan	5
Propósito del plan de gestión de configuraciones	5
Herramientas para la administración de configuraciones	5
Roles y Responsabilidades	5
<b>Esquema de directorios</b>	<b>6</b>
Estructura de directorios y propósito de cada uno	6
Normas de etiquetado y nombramiento de los archivos	6
Normas de nombramiento de archivos	7
Normas de etiquetado	7
<b>Gestión de la configuración del código</b>	<b>8</b>
Esquema de ramas	8
Política de etiquetado de las ramas	8
Política de fusión de archivos	8
<b>Gestión de Cambios</b>	<b>9</b>
Change Control Board (CCB)	9
Introducción y objetivos	9
Miembros	9
Frecuencia de reunión de trabajo	10
Proceso de control de cambios	10
Herramienta de gestión de cambios	10
<b>Gestión de Entregas</b>	<b>11</b>
Formato de entrega de releases	11
Formato de entrega del instalador	11
Instrucciones mínimas de instalación	11

## Historial de Cambios

Versión	Fecha	Cambios
1.0.0	5 / 5 / 2021	Documento inicial.
1.0.1	22 / 5 / 2021	Cambios en sección 1.3, 2.1 y 2.2.1.

# 1. Introducción

## 1.1. Propósito y alcance del plan

Este documento presenta el plan de Gestión de las Configuraciones del proyecto “Solitario Apolo”. El objetivo de este documento es presentar y controlar las políticas, estrategias, herramientas y documentos utilizados en el desarrollo de este proyecto de software.

## 1.2. Propósito del plan de gestión de configuraciones

- Asegurar la integridad en las prácticas del desarrollo de software.
- Definir los roles dentro del equipo de trabajo.
- Mejorar la eficiencia de los procesos.
- Crear un historial verificable de los estados de trabajo.
- Informar del estado del proyecto a los participantes del equipo.

## 1.3. Herramientas para la administración de configuraciones

Herramienta	Uso	Links de acceso
Apolo_GitHub	Control de versiones.	<a href="#">GitHub - Proyecto Apolo</a>
Apolo_CircleCI	Control de integración continua. Realización de test automáticos.	<a href="#">CircleCI - Proyecto Apolo</a>
Apolo_GitHub-issues	Gestión de Cambios.	<a href="#">GitHub Issues - Proyecto Apolo</a>
Apolo_Monday	Gestión de Tareas	<a href="#">Monday - Proyecto Apolo</a>

## 1.4. Roles y Responsabilidades

Rol	Responsabilidad	Responsable
Gestor global de configuración	<ul style="list-style-type: none"> <li>Procurar el cumplimiento de los elementos de configuración establecidos en este documento.</li> <li>Aplicar las etiquetas de las ramas principales según lo establecido.</li> </ul>	Gaston Sartori
Gestor de configuración de proyecto	<ul style="list-style-type: none"> <li>Controlar cambios a la configuración y su documentación adjunta.</li> <li>Mantener el repositorio del proyecto actualizado y bien organizado con las versiones del proyecto.</li> </ul>	Emiliano Zanotti
Coordinador de configuración	<ul style="list-style-type: none"> <li>Asegurar que los datos se encuentran ordenados en la base de datos del proyecto y hallar fallas en la estructura del sistema.</li> <li>Reportar cualquier defecto en cuanto a la configuración del proyecto.</li> </ul>	Julián Tántera
Responsable de elementos de configuración	<ul style="list-style-type: none"> <li>Asegurar que los elementos de configuración se registren en la base de datos de configuración correctamente.</li> <li>Verificar que los cambios sobre los elementos de configuración se realizan de acuerdo al proceso de cambios definido.</li> <li>Asegurar la integridad de los elementos de configuración.</li> </ul>	Javier Espósito

## 2. Esquema de directorios

### 2.1. Estructura de directorios y propósito de cada uno

El proyecto contará con un directorio base, el cual tendrá el nombre de Apolo. Dentro del mismo se encontrará los siguientes subdirectorios:

<b>Apolo/release</b>	Donde se encontrarán los paquetes y documentos de los releases realizados.
<b>Apolo/documentos</b>	Donde se encontraran documentos y archivos propios del proyecto. Informes.
<b>Apolo/src</b>	Donde se encontrará todo el código del proyecto y el código de pruebas.
<b>Apolo/images</b>	Donde se encontrarán las imágenes utilizadas.

### 2.2. Normas de etiquetado y nombramiento de los archivos

#### 2.2.1. Normas de nombramiento de archivos

- No usar espacios.
- No usar caracteres especiales.
- Uso de guiones bajos (\_) para separar palabras.
- Uso de letras minúsculas.

Ejemplo: Para el caso del archivo del Plan de Gestión de las Configuraciones, el nombre del mismo será: `plan_gestion_config.pdf`

#### 2.2.2. Normas de etiquetado

##### 2.2.2.1. Etiqueta de versión

Para la identificación de las versiones de los archivos dentro del repositorio se utilizara etiquetas siguiendo el “Versionado Semántico”, las características del mismo son las siguientes:

- Un número normal de versión debe tomar la forma X.Y.Z, donde X,Y y Z son enteros no negativos. Cada elemento debe incrementarse numéricamente en incrementos de 1.
- Z es la versión "PATCH", debe incrementarse cuando se introducen parches y soluciones a bugs.
- Y es la versión "MINOR", debe incrementarse si se introduce una nueva funcionalidad compatible con la presente versión "MAJOR". La versión "PATCH" debe ser reseteada a 0 cuando la versión "MINOR" es incrementada.
- X es la versión "MAJOR", debe incrementarse si se produce cualquier cambio no compatible con la versión anterior. Modificaciones importantes en la API. Las versiones "PATCH" y "MINOR" deben ser reseteadas a 0 cuando se incrementa la versión "MAJOR".

Ejemplo: se tiene la versión 2.4.0, si se realiza un cambio del tipo "PATCH", la siguiente versión será la 2.4.1. Si a esta se le realiza un cambio del tipo "MAJOR", la siguiente versión será la 3.0.0.

### 2.2.2.2. Etiqueta de fecha de creación

Para la etiqueta correspondiente a la fecha de creación se utilizará el formato de: AAAA-MM-DD. Ejemplo: 2021-04-30

## 3. Gestión de la configuración del código

### 3.1. Esquema de ramas

- **Master:** Rama principal que contiene el código que está en producción.
- **Develop:** Rama que contiene las características a incluir en el release.
- **Feature:** Copia de Develop donde se desarrollarán nuevas funcionalidades del proyecto. También se solucionan bugs menores.
- **Hotfix:** Copia de Master, que se crea cuando se detecta un bug a corregir inmediatamente.

### 3.2. Política de etiquetado de las ramas

**Rama principal:** master

**Rama de producción:** develop

**Rama de implementación:** feature\_Funcion

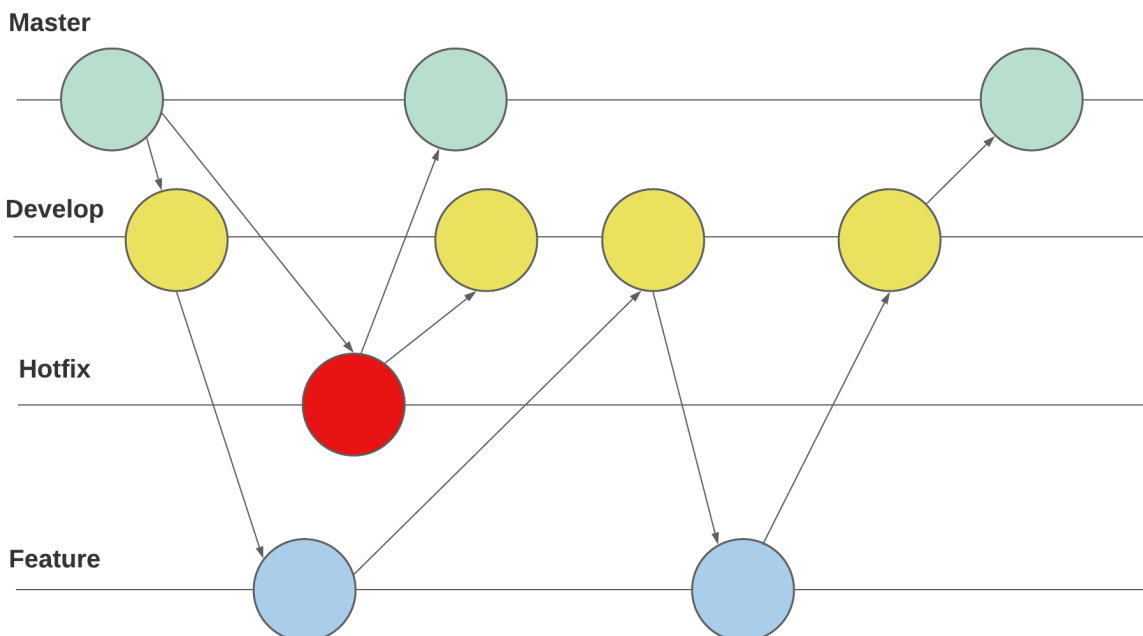
*"Funcion":* Nombre de la función que se está implementando.

**Rama de corrección:** hotfix\_ID

*"ID":* Número de ID del bug hallado.

### 3.3. Política de fusión de archivos

- **Develop:** Se fusiona con Master cuando el código está listo para salir a producción.
- **Feature:** Se fusiona con Develop cuando las funciones desarrolladas están listas para ser agregadas a una futura versión.
- **Hotfix:** Se fusiona con Master cuando se arregla el bug para corregirlo en la versión final. Se fusiona con Develop para que se reflejen los cambios al solucionar el bug en la versión que se está desarrollando.





## 4. Gestión de Cambios

### 4.1. Change Control Board (CCB)

#### 4.1.1. Introducción y objetivos

El Change Control Board se encarga de evaluar los pedidos de cambios que se quieran realizar en la documentación o en el código fuente del producto. Tiene como objetivo que dichos pedidos de cambios sean verificados por todos los miembros, y una vez aprobados por todos, se apliquen en el proyecto.

#### 4.1.2. Miembros

Rol	Responsabilidad	Responsable
Director del CCB	<ul style="list-style-type: none"> <li>Aprobar o rechazar el pedido de cambio.</li> <li>Moderación de las reuniones.</li> </ul>	Gaston Sartori
Gerente de Manejo de las Configuraciones	<ul style="list-style-type: none"> <li>Controlar que el procedimiento de corrección de errores cumpla con su objetivo.</li> <li>Diseñar políticas de control de seguimiento.</li> </ul>	Emiliano Zanotti
Gerente de Ingeniería	<ul style="list-style-type: none"> <li>Responsable de evaluar los costos de implementación de cambios.</li> </ul>	Javier Espósito
Gerente de Pruebas	<ul style="list-style-type: none"> <li>Organizar y ejecutar las pruebas.</li> <li>Evalúa qué impacto tiene el cambio en la calidad y fecha de la entrega.</li> </ul>	Julian Tantera

#### 4.1.3. Frecuencia de reunión de trabajo

Las reuniones de trabajo quedarán establecidas según el siguiente cronograma:

Lunes	De 18:00 hs a 20:00 hs
Miércoles	De 18:00 hs a 20:00 hs
De ser necesario, se organizarán reuniones extra.	

## 4.2. Proceso de control de cambios

El proceso consta de las siguientes etapas:

- 1) **Solicitud de cambio:** Se recibe una solicitud de cambio.
- 2) **Evaluación:** Se analiza si es posible y rentable su aplicación. Además se verifica qué impacto tendrá en el resto del sistema para ver qué decisiones se tendrán que tomar para implementar el cambio.
- 3) **Implementación:** Previa aprobación de la solicitud, se procede a la aplicación del cambio, con todo lo que incluye.
- 4) **Pruebas:** Se testea el software para verificar que el sistema funcione correctamente luego de aplicar el cambio.
- 5) **Aceptación del cambio:** Por último se registra el cambio realizado y se actualiza la versión del sistema.

## 4.3. Herramienta de gestión de cambios

La herramienta que utilizaremos será GitHub-Issues. Es una funcionalidad de GitHub que permite trackear bugs, establecer tareas por hacer y funciones por implementar.

# 5. Gestión de Entregas

## 5.1. Formato de entrega de releases

El formato que se emplea para la entrega de releases incluirá diferentes archivos, estos son:

- Ejecutable del programa
- README.txt, donde se informará sobre la versión del release, ayuda para la instalación y ejecución del software y bugs conocidos.

## 5.2. Formato de entrega del instalador

El formato de entrega será a través de un archivo comprimido .zip, incluyendo los ítems definidos en el Formato de entrega de releases.

## 5.3. Instrucciones mínimas de instalación

Las instrucciones de instalación se incluirán en el documento README.txt que se incluye en la entrega del release.