ESTANDAR CODE REVIEW

Versión 1.0.2

Arquitectura y Estándares TI

07/05/2024

Tabla de contenido

1. VERSIONES

Descripción	Versión	Autor	Fecha
Publicación de documento original	1.0.0	Reynaldo Corila Mauricio	30/04/2024
Publicación de documento original	1.0.1	Reynaldo Corila Mauricio	06/05/2024
Publicación de documento original	1.0.2	Reynaldo Corila Mauricio	07/05/2024

2. DEFINICIONES

Concepto	Definición
Code Review	Proceso en el que un revisor de Código revisa el código de un desarrollador.
Main/Master	Rama principal del repositorio, contiene la versión más reciente del código.
Revisión de pares	Proceso en el que dos o más desarrolladores revisan el código de otro desarrollador.
Pull Request	Forma de enviar cambios de código de una rama a otra.
Unit Test	Prueba que se utiliza para verificar una pequeña unidad de código.
SonarQube	Herramienta de análisis de código estático que se utiliza para identificar errores, duplicaciones y posibles problemas en el código
Frameworks	Marcos de trabajo, estructuras de software que proporcionan funcionalidad básica y patrones de diseño para ayudar a los desarrolladores a crear aplicaciones.
Vulnerabilidades	Debilidades en el software que pueden ser explotadas por los atacantes para obtener acceso no autorizado a un sistema o datos.
GitHub Enterprise Security	Conjunto de características de seguridad adicionales que se ofrecen con GitHub Enterprise.

3. OBJETIVO

Incluir un control adicional en la codificación de los equipos de desarrollo, dentro del marco de desarrollo seguro, buscando reducir o eliminar posibles vulnerabilidades y brechas de seguridad, antes de pasar a producción.

4. ALCANCE

Todos los aplicativos, proyectos, soportes y/o modificaciones que se realicen con el propósito de ser implementados en producción.

5. REQUISITOS PREVIOS A LA REVISIÓN:

DevOps:

- Colección de proyectos en el repositorio de **Azure DevOps**.
- Estándar de ramas **Git Flow** (**develop**, **release** y **main**/**master**).
- Pipeline de compilación integrado con **SonarQube**.
- Líder técnico y el grupo de dominio "**GD_ARQUITECTURA**" Configurados como revisores obligatorios del **pull request** (**develop** a **release**).

Revisión de pares:

- Revisión de pares, revisión y aprobación del líder técnico (**pull request**)

6. EVALUACIÓN (REVISIÓN DEL CÓDIGO)

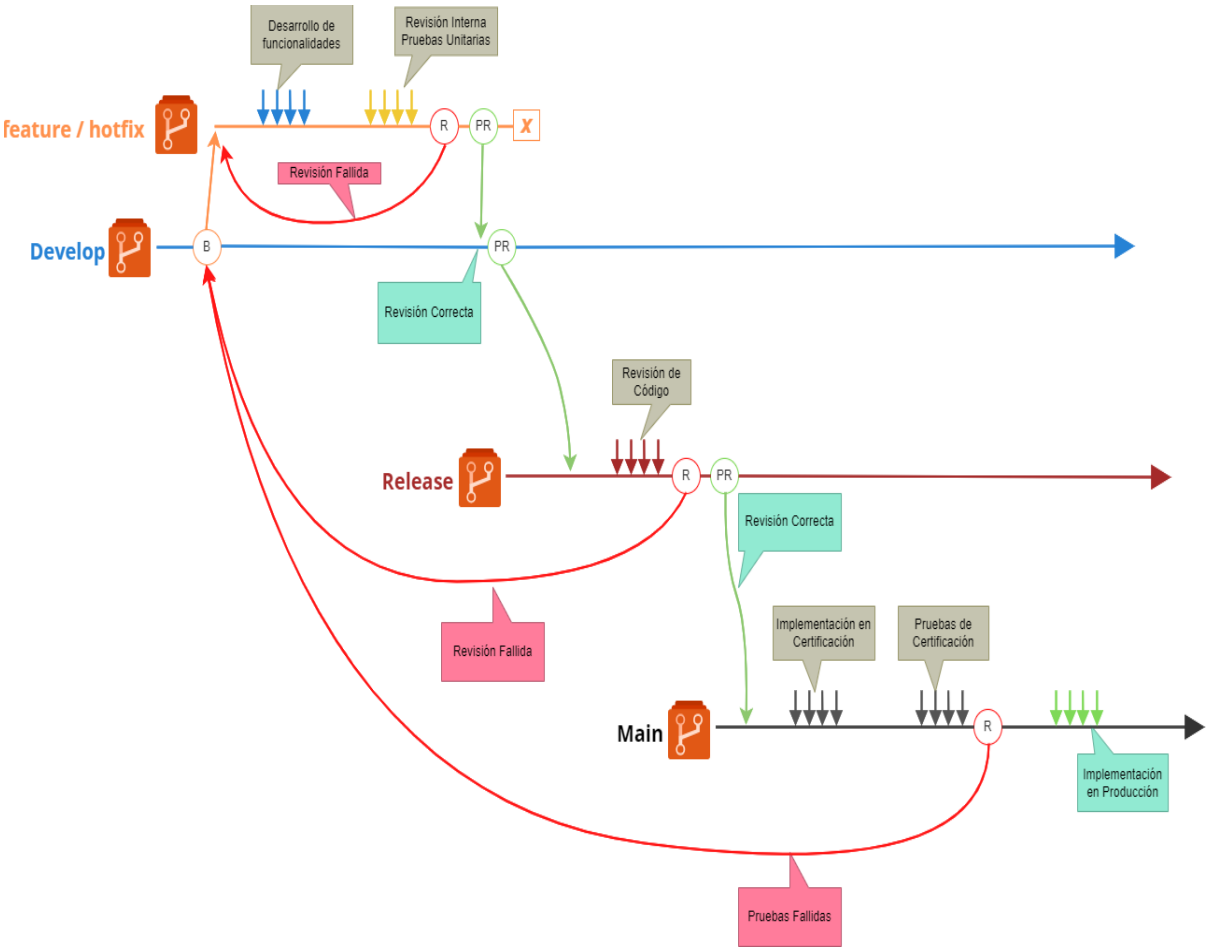
- **Vulnerabilidades comunes:** Inyección de código, fugas de información confidencial y otras amenazas a la seguridad.
- **Gestión de errores:** Manejo de errores y excepciones adecuado para garantizar la estabilidad y recuperación ante fallos del sistema.
- **Librerías y frameworks:** Uso de librerías y frameworks de fuentes confiables, con actualizaciones regulares para minimizar vulnerabilidades conocidas.
- **Validación de datos de entrada:** Validación y saneamiento de los datos de entrada para prevenir ataques de inyección, scripts maliciosos y otras amenazas.
- **Controles de acceso:** Mecanismos de autenticación y autorización de gestionar permisos y privilegios.
- **Gestión de contraseñas:** Gestión de contraseñas seguras, incluyendo cifrado en archivos de configuración, eliminación de contraseñas en archivos de tipo Log.

7. ENTREGABLE

- **Informe de revisión (Checklist):** Informe de hallazgos de la revisión:
  - Lista de las vulnerabilidades de seguridad identificadas.
  - Recomendaciones para corregir dichas vulnerabilidades.
  - Clasificación de la severidad de estas vulnerabilidades (baja, media, alta o crítica).
  - Recomendaciones para mejorar la seguridad y la arquitectura del código.
  - Sugerencias para mejorar la calidad del código en Gral.
- **Ejemplos de código:** Código con buenas prácticas para evitar vulnerabilidades.
- **Seguimiento:** Proceso de seguimiento para garantizar que las vulnerabilidades se corrijan de manera oportuna.

8. FLUJO

- **Equipos Desarrollo/Squad:**
  1. **Implementación:** Las funcionalidades y/o correcciones.
     a. **feature:** Desarrollo de nuevas funcionalidades
     b. **hotfix:** Corrección errores urgentes.
     c. **develop:** Rama principal donde se integra toda la funcionalidad nueva y/o correcciones (revisada por el **líder técnico**).
  2. **Revisión interna:** Revisión de pares y pruebas unitarias en el equipo de desarrollo.
  3. **Integración en la rama reléase**:
     d. Se crea un **pull request** para integrar los cambios de la rama **develop** a **release**.
     e. EL **líder técnico** revisa el código a profundidad, si la revisión es satisfactoria aprueba la integración. De lo contrario rechaza y vuelve al paso 1.
- **Arquitectura:**
  4. **Revisión de código:** Se realiza revisión de código al integrar en la rama **release**.
  5. **Informe/checklist:** Se genera un **checklist** con los aspectos revisados.
  6. **Integración en la rama main/máster:** Si la revisión es satisfactoria, se integra los cambios en la rama **main/máster**. De lo contrario, se rechaza y vuelve al paso 1.
- **Certificación:**
  7. **Despliegue:** Despliegue en ambiente de certificación.
  8. **Certificación:** Se ejecutan pruebas de certificación**.** Si las pruebas son satisfactorias, autoriza la implementación en **producción**. De lo contrario, rechaza y vuelve al paso 1.



9. ANÁLISIS DE CÓDIGO

Análisis automatizado:

- **SonarQube:** Análisis de código estático integrado en el pipeline de compilación para identificar posibles errores y problemas de calidad de código.

Análisis manual

- **Revisión de código:** Un revisor “experto” analiza manualmente el código.

Tiempo de revisión:

- En función a la cantidad de líneas de código en el **pull request**

10. ROLES Y RESPONSABILIDADES

- **Revisor de código:** Responsable de realizar la revisión del código y documentar los hallazgos.
- **Desarrollador:** Encargado de resolver los problemas identificados durante la revisión del código y de generar el **pull request** de la rama **develop** a la rama **release**.
- **Líder técnico:** Encargado de asegurar que se realice la revisión de pares y las pruebas unitarias en el equipo de desarrollo.

11. OBSERVACIONES

En caso de tener alguna observación o agregar alguna consideración, enviarlo a los siguientes correos:

Bolivia - Arquitectura y Estándares TI	BolArqEstaTI@bancred.com.bo
Jose Zurita Plata	JZurita@bcp.com.bo
Braian Espejo Peralta	BEspejo@bcp.com.bo
Diego Tarquino Tapia	DTarquino@bcp.com.bo
Reynaldo Corila Mauricio	RCorila@bcp.com.bo