**Plan de Gestión de la Configuración**

**2025-2**

**Grupo**

**1**

**Integrantes**

**Rojas Palomino Gonzalo Sebastian**

**Moreno Zevallos Eva Lucia**

**Ramirez Patiño Fabrizio Gabriel**

**San MarCODING S.A.C.**

1. **Introducción**

1.1 Situación de la empresa

SanMarCODING S.A.C es una consultora de software que cuenta con 5 años de experiencia en el desarrollo y mantenimiento de sistemas empresariales. La empresa se desempeña en el rubro tecnológico, ofreciendo soluciones para distintos sectores como banca, retail, salud, educación y servicios de delivery.

La empresa ha logrado consolidar una cartera de clientes reconocidos, como Interbank S.A.A., para el que desarrollamos un aplicativo móvil de transferencias bancarias (PocketBank), Saga Falabella S.A. para el que desarrollamos un Sistema de inventario para almacenes (StockNums), Auna S.A.A. para el que desarrollamos “Medical” que es un sistema de mantenimiento de equipos biomédicos y control sanitario. También colaboramos con la Universidad Nacional Mayor de San Marcos (UNMSM), para los cuales desarrollamos “UniReg” una plataforma de matrícula para universidades, PedidosYa Perú S.A.C. para el cual desarrollamos un sistema de gestión de incidencias y tiempos de entrega para deliverys(FleetGuide), también realizamos para el Banco de la Nación un sistema de monitoreo de servidores, redes y ciberseguridad (SecureOps), con la Universidad de Lima (UL) desarrollamos “SmartCampus” que es un sistema de mantenimiento de infraestructura digital en campus, con Norky’s S.A.C. realizamos un amplio trabajo realizando “Inventarium” (Sistema de control de insumos para restaurantes), con N.G. Restaurants S.A. se logró la meta de FoodnFast (Aplicativo de delivery para restaurantes) y nuestro trabajo más reciente es PharmaLink (Sistema de gestión para farmacias) que lo desarrollamos para InRetail Pharma S.A., sin embargo, un problema identificado es la gestión del control de versiones en sus proyectos. Debido a la cantidad de sistemas en desarrollo y en mantenimiento, surgen dificultades para llevar un seguimiento claro de cambios, coordinar equipos en paralelo y asegurar la consistencia entre entornos de prueba y producción.

1.2. Objetivo

La empresa propone implementar el plan de la gestión de configuración de software para facilitar el control de versiones de los ítems que componen cada uno de nuestros proyectos, trazar los cambios hechos a dichos ítems (o a sus versiones) y reducir el riesgo de conflicto entre los miembros del equipo.

1. **Gestion**
   1. Roles y Responsabilidades

| **Roles** | **Cantidad** | **Responsabilidad** |
| --- | --- | --- |
| Gestor de la Configuración del Software | 1 | Responsable de toda el área de la configuración del Software.  Diseñar el plan de gestión de configuración, supervisar el uso de herramientas de control de versiones, coordinar auditorías y liberar versiones oficiales. |
| Bibliotecario de la Configuración | 1 | Organizar los elementos de la configuración, controlar accesos, distribuir versiones y asegurar la trazabilidad de líneas bases, todo relacionado a una herramienta de control de versiones. |
| Administrador de la Línea Base(Baseline) | 1 | Definir y gestionar las líneas base aprobadas, asegurar de que se integren los cambios de manera correcta en cada versión y mantiene la integridad y consistencia en los release. |
| Auditor | 1 | Revisar que los cambios cumplan con normas de calidad y estándares, realizar las auditorías de la línea base y verificar que los cambios autorizados fueron implementados. Realizar las auditorías físicas y funcionales de los elementos de la configuración del software. |
| Comité de Control de Cambios | 1 | Grupo encargado de evaluar, aprobar o rechazar solicitud de cambios, y analizar impactos en costo, tiempo y calidad antes de autorizar un cambio. |

* 1. Benchmarking de Herramientas de control de versiones, Gráfica de la infraestructura (de la herramienta elegida) mínimo 4

| **Criterios** | **GitHub** | **Mercurial** | **Bazaar** | **IBM Rational ClearCase** |
| --- | --- | --- | --- | --- |
| Disponibilidad (gratuita/de pago) | Gratuita y de pago | Gratis | Gratis | De pago |
| Curva de aprendizaje | Alta | Media | Baja | Muy alta |
| Ecosistema | Amplio y activo | Poco activo | Obsoleto | Amplio y activo |
| Escalabilidad | Alta | Alta | Baja | Alta |
| Soporte multiplataforma | Óptimo | Inadecuado | Óptimo | Óptimo |
| Flexibilidad | Muy alta | Alta | Baja | Media |

La herramienta elegida: GitHub

Gráfica de arquitectura

1. Actividades de la GCS
   1. Identificación de la GCS
      1. Definir estructura de la librerías del repositorio



* + 1. **Definición de Nomenclatura**

Proyecto: BookStore+

Sistema de Venta de Libros (SVL)

**Fórmula 1:** Para ítems que pertenecen a un proyecto:

**Acrónimo del proyecto + ”\_”+ acrónimo del item +extension**

**Fórmula 2:** Para ítems que pertenecen a un proyecto, pero que si se utilizara la fórmula 1, causarían nomenclaturas repetidas de archivos distintos.

**Acrónimo del proyecto + “\_”+ tres primeras letras del item + extensión**

En el caso que se siguen repitiendo nombres, agregar la siguiente letra del ítem.

**Fórmula 3:** Para los ítems de caso de uso, se utilizará

**Acrónimo del proyecto + “\_” + CU + número consecutivo de 3 cifras extensión**

Item:

Lista de requisitos (LR), Nomenclatura: SVL\_LR.xlsx

Project Charter(PC), Nomenclatura: SVL\_PC.docx

Cronograma(C), Nomenclatura: SVL\_C.xlsx

Especificación de Caso de Uso 01(CU001), Nomenclatura: SVL\_CU001.docx

Especificación de Caso de Uso 02(CU002), Nomenclatura: SVL\_CU002.docx

Especificación de Caso de Uso 03(CU003), Nomenclatura: SVL\_CU003.docx

Especificación de Caso de Uso 04(CU004), Nomenclatura: SVL\_CU004.docx

* + 1. Lista de Items (Clasificación de Items)

| **Tipo**  (Evolucion  Fuente  Soporte**)** | **Item** | **Nomenclatura** | **Proyecto**  **(Acronimo)** |
| --- | --- | --- | --- |
| Evolución | Lista de Requisitos | SVL\_LR.xlsx | SVL |
| Evolución | Project Charter | SVL\_PC.docx | SVL |
| Evolución | Cronograma | SVL\_C.xlsx | SVL |
| Evolución | Especificación de Caso de Uso 01 | SVL\_CU001.docx | SVL |
| Evolución | Especificación de Caso de Uso 02 | SVL\_CU002.docx | SVL |
| Evolución | Especificación de Caso de Uso 03 | SVL\_CU003.docx | SVL |
|  | Plan de Gestión de la Configuración | PGC |  |

* + 1. Definicion de Linea Base
  1. **Control de la GCS**
     1. **Ejemplos de Solicitudes de cambio (1 por alumno)**
     2. Proceso de Control de Cambios (De la Empresa ETB)
        + 1. Grafica del proceso
          2. Tipos de estados de la Solicitud de cambio
          3. Ejemplos de reportes (3)
  2. Estados de la GCS

Ejemplos de reporte de estado de Github (Commit, Contribuciones, Ramas)

* 1. Auditoria de la GCS
     1. Auditoria Fisica

Lista de Items que se encuentran en la linea base 3 de un proyecto

* + 1. Auditoria Funcional

| Historia de Usuario | Fecha de auditoria | Auditor | Estado  Cumple (√)  No cumple (X) |
| --- | --- | --- | --- |
| HU01-Toma de decisiones | 25/11/24 | Diego Gamarra | √ |
| HU02-Aprobacion del alumnado | 25/11/24 | Diego Gamarra | X |
|  |  |  |  |
|  |  |  |  |

* 1. Gestion de Release y **Entrega del Software**

*6.1 Gestion de Release (Como voy a hacer la entrega de los release de los clientes)*

6.2 Entrega del Software (Pase a producción)

Proponer como hacen el pase a produccion, actividades, herramientas, etc