DEVSLA

**Plan de Gestión de la Configuración del Software**

Integrantes

Dávila Raffo, Alwin Edu

Espinoza Peralta, Carlos Miguel

Jara Espinoza, Ángela Lucía

Madrid Ruiz, Giacomo Salvador

Mantilla Flores, Shamir

Patricio Julca, Vilberto Alberto

Versión 1.0

**Control de versión**

| **Fecha** | **Versión** | **Descripción** | **Autor** |
| --- | --- | --- | --- |
| 19/09/2024 | 1.0 | Versión inicial | Gestor de la configuración |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. **Introducción**
   1. Situación de la empresa y problemática

Devsla, fundada en 2023 en Lima, Perú, es una pequeña empresa emergente enfocada en el desarrollo de software. Desde su creación, ha enfrentado el desafío de construir una base sólida tanto a nivel de estructura interna como en su presencia en el mercado. A pesar de ser nueva en la industria tecnológica, presenta tres proyectos en su repertorio.

* TaskMaster: Herramienta de gestión de tareas diseñada para equipos pequeños destinada a Digital 212, que permite crear, asignar y priorizar tareas de manera sencilla, facilitando la colaboración y la transparencia en el trabajo.
* HealthTrack: Aplicación para el seguimiento de hábitos de salud y bienestar destinada a la Clínica Ricardo Palma, que ayuda a los usuarios a establecer y monitorear objetivos relacionados con la actividad física, la nutrición y el bienestar mental.
* EZCommerce (en desarrollo): Plataforma virtual de comercio destinada a la UNMSM, que permitirá a estudiantes vender productos y servicios de forma segura y ágil, adaptándose a las necesidades específicas del entorno académico.

En el desarrollo de EZCommerce, Devsla enfrentó una problemática crítica que impactó directamente su flujo de trabajo: la ausencia de un sistema de control de versiones adecuado. Al iniciar las primeras actividades del proyecto, el equipo se dio cuenta de que, a medida que varios miembros trabajaban en distintos aspectos simultáneamente, surgieron inconsistencias debido a la falta de un repositorio centralizado. Esto generó confusiones sobre las versiones actuales y resultó en múltiples archivos con cambios no documentados. Como consecuencia, el equipo experimentó un aumento significativo en los errores y retrabajos, lo que no solo retrasó el cronograma del proyecto, sino que también creó frustración entre los integrantes. Esta situación puso de manifiesto la necesidad urgente de establecer un control de versiones eficaz, con el objetivo de recomponer el flujo de trabajo y facilitar una gestión más ordenada y colaborativa del código en el futuro.

* 1. El objetivo del plan

El objetivo del presente plan de gestión de la configuración de software es implementar un sistema de control de versiones distribuido que permita gestionar de manera eficiente el código fuente y la documentación de los proyectos de Devsla. Esto permitirá a la empresa mejorar la colaboración entre los desarrolladores, evitando la sobrescritura de código y los errores comunes derivados de la falta de un sistema centralizado. A través de la adopción de un repositorio central y un flujo de trabajo basado en ramas, se podrá asegurar un desarrollo ágil, organizado y con trazabilidad completa de cada cambio realizado. Además, la capacidad de revertir modificaciones erróneas o integrar mejoras de manera controlada proporcionará mayor flexibilidad y seguridad en la gestión del código.

1. **Gestión de Configuración de Software**
   1. Roles y responsabilidades

| Roles | Descripción |
| --- | --- |
| Gestor de la configuración | Define, aplica y supervisa el proceso de gestión de versiones y cambios en el software. |
| Bibliotecario | Administra y asegura el almacenamiento, etiquetado y acceso adecuado a las versiones del software y documentación. |
| Comité de control de cambios | Evalúa y aprueba o rechaza las solicitudes de cambio, asegurando su impacto positivo y estabilidad. |
| Equipo de desarrollo | Implementa los cambios aprobados, documenta y sigue los procesos de configuración establecidos. |

* 1. Herramientas, entorno e Infraestructura

[**Bemchmarking** (Min 3 herramientas), Ponderar y selección ]

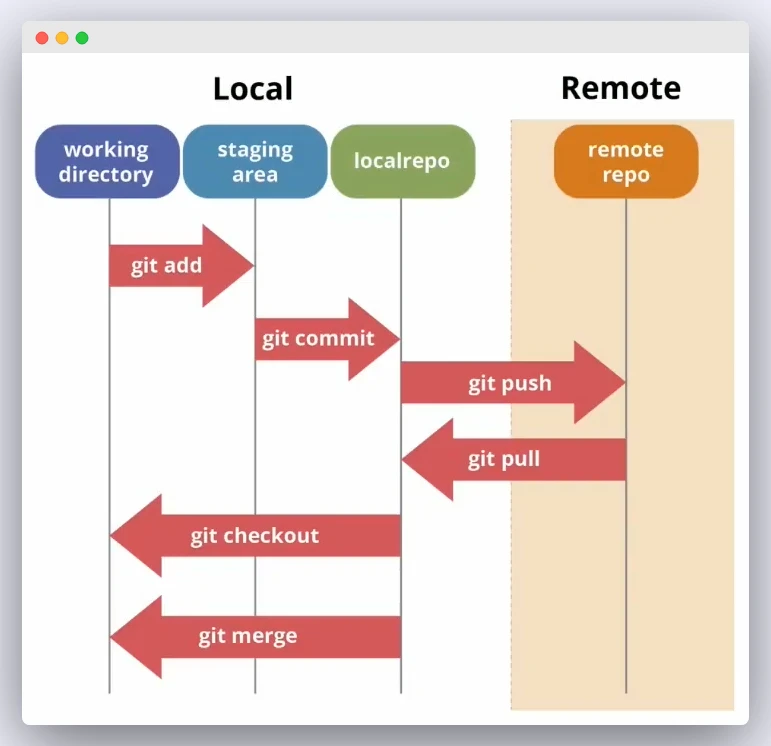
| **CRITERIOS** | **GIT** | **Mercurial** | **Subversion** |
| --- | --- | --- | --- |
| Gratuito | Sí | Sí | Sí |
| Curva de aprendizaje | Media | Baja | Media |
| Centralizado | No | No | Si |
| Distribuido | Si | Sí | No |
| Enfocado a instantaneas | Si | Si | No |
| Enfocado a deltas | No | No | Si |
| Plataformas | GitHub, GilLab,...(+20) | Bitbucket, Heptapod, …(+10) | Apache Subversion, VisualSVN Server ,.. (+5) |

En base a la disponibilidad y las capacidades que se requieren para este proyecto se ha decidido optar por GIT puesto que nos permite trabajar bajo un enfoque distribuido de manera que se tenga mayor seguimiento del versionamiento.

**Herramienta elegida**: GIT

**Descripción**: GIT es una herramienta de gestión de versiones distribuida que nos permite registrar los cambios a través de instantáneas. Su arquitectura distribuida permite que cada desarrollador tenga una copia completa del repositorio, lo que facilita el trabajo sin conexión y la colaboración eficiente.

**Arquitectura**:



1. **Actividades de la GCS**
   1. Identificación
      1. Definición de la estructura de las librerías

[Diagrama similar a este:]

Diagrama

Descripción generada automáticamente

* + 1. Definición de Líneas Base

(Por **ejemplo** el proyecto AGSCPI (Revisar el cronograma, si tiene 3 hitos, entonces tendrá 3 linea bases)

| **Nombre de línea base** | **Items** | **Nomenclatura** |
| --- | --- | --- |
| Linea Base 1 | Project Charter | AGSCPI-PC.docx |
| Cronograma del proyecto | AGSCPI-CP.docx |
| Lista de requisitos | AGSCPI-LR.XLS |
| Linea Base 2 | Lista de requisitos | AGSCPI-LR.XLS |
|  |  |  |
|  |  |  |
| Linea Base 3 |  |  |

* + 1. Lista de clasificación de Items

| **TIPO DE ITEM**  (E= Evolución F=Fuente S=Soporte) | **DESCRIPCION DE ITEM** | **PROYECTO** |
| --- | --- | --- |
| E | Documento de Arquitectura | SIC |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  | SERT |
|  |  | SIFDE |

* + 1. Definición de la Nomenclatura de ítems

[Mínimo 4 formulas para distintos Escenarios ]

Caso 1:

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

Caso 2:

Caso 3:

* + 1. Lista de ítem con la nomenclatura

| **DESCRIPCION DEL ITEM** | **NOMENCLATURA** | **EXTENSION** | **PROYECTO** |
| --- | --- | --- | --- |
| Documento de Arquitectura | SIC-DA | .DOCX | SIC |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* 1. Control de la GCS

3.2.1 Formatos de solicitudes de Cambio (1c/u)

3.2.4. Proceso de Gestión de Cambios

* 1. Estado

3.3.1 Colocar ejemplo de Reportes para el Estado de la GCS

* 1. Auditoria
     1. Colocar ejemplos de Reportes de Auditorias
  2. **Entrega del software** y Gestión de Release

3.5.1**. Proponer un Proceso de pase a producción** (Pasos a seguir)

3.5.2 Gestión de reléase

dajdkljldjkHDJKwd

dajkDJal

Diagrama

Descripción generada automáticamente

**Break**