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 | Project Manager |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 ]

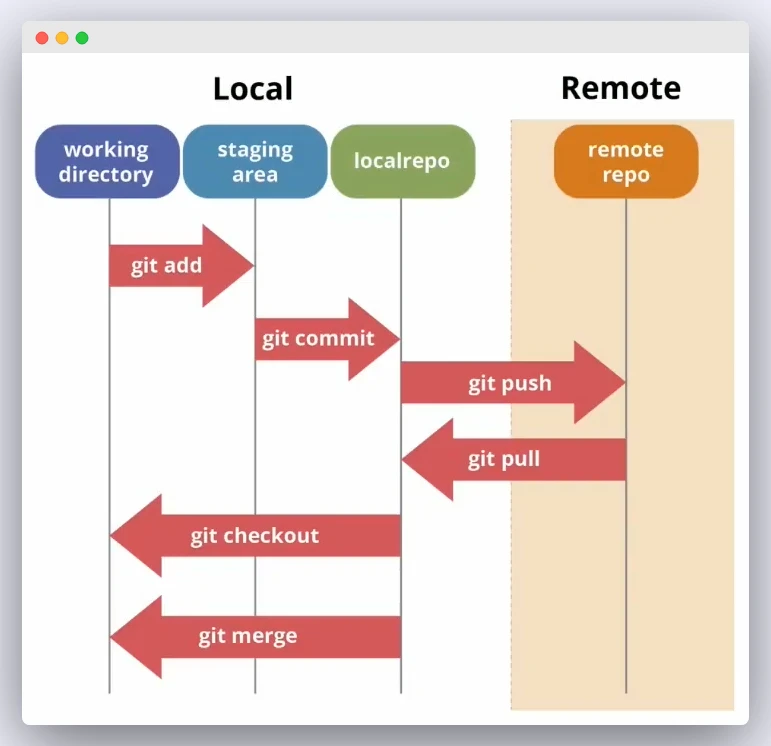
|  |  |  |  |
| --- | --- | --- | --- |
| **CRITERIOS** | **GIT** | **Mercurial** | **Subversion** |
| Gratuito | Sí | Sí | Sí |
| Curva de aprendizaje | Media | Baja | Media |
| Centralizado | No | No | Si |
| Distribuido | Si | Sí | No |
| Enfocado a instantaneas | Si | Si | No |
| Enfocado a deltas | 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

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Explicación:

* Clientes: Contiene proyectos relacionados con los diferentes clientes.
  + Clínica Ricardo Palma
    - HT: 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.
  + Digital212
    - TM: 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.
  + UNMSM
    - PVCU: 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.
* Desarrollo: Agrupa a las diferentes etapas del desarrollo de los proyectos.
  + PVCU
    - Análisis: Archivos relacionados con la etapa de análisis de negocio, historias de usuario y el reporte de QA.
    - Diseño: Contiene los documentos y artefactos relacionados con el diseño del software, como el diseño de la base de datos, el diseño de la UI, el diseño de la arquitectura de software, etc.
    - Gestión: Archivos relacionados con la gestión, la planificación y seguimiento del proyecto, como el cronograma del proyecto y el Project Charter.
    - Negocio: Contiene el documento de negocio.
* Documentos: Están los diferentes documentos de referencia o soporte para el desarrollo y la gestión del proyecto.
  + Manuales: Documentos de instrucciones para la implementación del proyecto.
    - PVCU: Manual específico del proyecto PVCU.
  + Planes: Documentos que tienen al plan de gestión.
* Línea Base: Agrupa a los diferentes hitos del proyecto en base al cliente.
  + PVCU
    - Línea Base 1: Contiene el primer hito del proyecto.
    - Línea Base 2: Agrupa el estado de los módulos del segundo hito.
    - Línea Base 3: Contiene el tercer hito del proyecto.
    1. Definición de Líneas Base

|  |  |  |
| --- | --- | --- |
| **Hito** | **Nombre de línea base** | **Items (nomenclatura)** |
| Hito 1: 20/09/24 | Línea Base 1 | Plan del proyecto (PVCU-PC.docx)  Cronograma del proyecto (PVCU-CP.xlsx)  Documento de Negocio (PVCU-DN.docx)  Lista de Épicas (PVCU-LE.xlsx)  Historia de Usuario 1 de la Épica 1 (PVCU-HU1.1.docx)  Historia de Usuario 2 de la Épica 1 (PVCU-HU1.2.docx)  Historia de Usuario 1 de la Épica 2 (PVCU-HU2.1.docx)  Historia de Usuario 2 de la Épica 2 (PVCU-HU2.2.docx)  Historia de Usuario 1 de la Épica 3 (PVCU-HU3.1.docx)  Historia de Usuario 2 de la Épica 3 (PVCU-HU3.2.docx)  Historia de Usuario 3 de la Épica 3 (PVCU-HU3.3.docx)  Historia de Usuario 4 de la Épica 3 (PVCU-HU3.4.docx)  Historia de Usuario 1 de la Épica 4 (PVCU-HU4.1.docx)  Historia de Usuario 2 de la Épica 4 (PVCU-HU4.2.docx)  Historia de Usuario 3 de la Épica 4 (PVCU-HU4.3.docx)  Historia de Usuario 1 de la Épica 5 (PVCU-HU5.1.docx)  Historia de Usuario 2 de la Épica 5 (PVCU-HU5.2.docx)  Historia de Usuario 3 de la Épica 5 (PVCU-HU5.3.docx)  Historia de Usuario 1 de la Épica 6 (PVCU-HU6.1.docx)  Historia de Usuario 2 de la Épica 6 (PVCU-HU6.2.docx)  Historia de Usuario 3 de la Épica 6 (PVCU-HU6.3.docx)  Historia de Usuario 1 de la Épica 7 (PVCU-HU7.1.docx)  Historia de Usuario 2 de la Épica 7 (PVCU-HU7.2.docx)  Historia de Usuario 1 de la Épica 8 (PVCU-HU8.1.docx)  Historia de Usuario 2 de la Épica 8 (PVCU-HU8.2.docx)  Historia de Usuario 3 de la Épica 8 (PVCU-HU8.3.docx)  Reporte de QA 1 (PVCU-RQA1.docx)  Reporte del Primer Sprint (PVCU-RPS.docx) |
| Hito 2: 12/10/24 | Línea Base 2 | Documento de Lógica de Negocio(PVCU-DLN.docx)  Documento de la Arquitectura de Software(PVCU-DAS.docx)  Documento del Modelado de Clases(PVCU-DMC.docx)  Documento del Modelado de Base de Datos(PVCU-DMBD.docx)  Documento de la User Interface(PVCU-DUI.docx)  Guía de Estilos(PVCU-GE.docx)  Reporte de QA 2(PVCU-RQA2.docx)  Reporte del Segundo Sprint(PVCU-RSS.docx) |
| Hito 3: 14/11/24 | Línea Base 3 | Épica 1(//E1)  Épica 2(//E2)  Épica 3(//E3)  Épica 4(//E4)  Épica 5(//E5)  Épica 6(//E6)  Épica 7(//E7)  Épica 8(//E8)  Documento de Pruebas de Software(PVCU-DPS.docx)  Manual de Usuario(PVCU-MU.docx)  Reporte del Tercer Sprint(PVCU-RTS.docx)  Acta de Cierre del Proyecto(PVCU-ACP.docx) |

* + 1. Lista de clasificación de Items

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tipo**  (E= Evolución  F= Fuente  S= Soporte) | **Nombre del Item** | **Nomenclatura** | **Extensión** | **Proyecto** |
| E | Cronograma del Proyecto | PVCU-CP.xlsx | xlsx | PVCU |
| E | Lista de Épicas | PVCU-LE.xlsx | xlsx | PVCU |
| E | Historia de Usuario 1 de Épica 1 | PVCU-HU1.1.docx | docx | PVCU |
| E | Historia de Usuario 2 de Épica 1 | PVCU-HU1.2.docx | docx | PVCU |
| E | Historia de Usuario 1 de Épica 2 | PVCU-HU2.1.docx | docx | PVCU |
| E | Historia de Usuario 2 de Épica 2 | PVCU-HU2.2.docx | docx | PVCU |
| E | Historia de Usuario 1 de Épica 3 | PVCU-HU3.1.docx | docx | PVCU |
| E | Reporte de QA 1 | PVCU-RPQA1.docx | docx | PVCU |
| E | Documento de Lógica de Negocio | PVCU-DLN.docx | docx | PVCU |
| E | Documento de la Arquitectura de Software | PVCU-DAS.docx | docx | PVCU |
| S | Python intérprete para Windows | PVCU-PIW.exe | exe | PVCU |
| S | Python intérprete para Linux | PVCU-PIL.tar.xz | tar.xz | PVCU |
| E | Documento del Modelado de Base de Datos | PVCU-DMBD.docx | docx | PVCU |
| E | Documento de la User Interface | PVCU-DUI.docx | docx | PVCU |
| E | Documento del Modelado de Clases | PVCU-DMC.doc | docx | PVCU |
| E | Reporte de Primer Sprint | PVCU-RPS.docx | docx | PVCU |
| F | Épica 1 | //E1 | js/py | PVCU |

* + 1. Definición de la Nomenclatura de ítems

Acrónimo del proyecto de software: Plataforma virtual de comercio universitario (PVCU)

Nomenclatura

Caso 1: Para ítems que pertenecen a un proyecto

|  |
| --- |
| Acrónimo del proyecto + “-”+Acrónimo del ítem +”.”+ extensión |

Caso 2: Si son historias de usuario

|  |
| --- |
| Acrónimo del proyecto + “-”+Acrónimo de historia de usuario + Número de Épica + “.”+Número de historia de usuario +”.”+ extensión |

Caso 3: Si son reportes de QA

|  |
| --- |
| Acrónimo del proyecto + “-”+Acrónimo del ítem + Acrónimo de Quality Assurance+Número de reporte+”.”+ extensión |

Caso 4: Si los nombres son iguales

|  |
| --- |
| Acrónimo del proyecto + “-”+Acrónimo del ítem + dos siguientes letras de la última palabra del ítem+”.”+ extensión |

Caso 5: Si se es un directorio

|  |
| --- |
| “//”+Acrónimo del ítem |

* 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

Diagrama

Descripción generada automáticamente

**Break**