

UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS

(Universidad del Perú, DECANA DE AMÉRICA)

FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

ESCUELA PROFESIONAL DE INGENIERÍA DE SOFTWARE



“Plan de gestión de la configuración del software V1.1 ”

Grupo: 2

Presentan los estudiantes:

Bravo Pimentel, Rafael Alexander

Huaman Marquez Christian Hilbert

Illesca Avila Aaron

Dinca, Alexandru-Andrei

Laguna Santa Cruz, Oscar Isaac

Sotelo Arce, Jocelyn Estrella

Docente:

Wong Portillo, Lenis Rossi

LIMA – PERÚ

2023



Indice

Indice	2
PLAN DE GESTIÓN DE LA CONFIGURACIÓN DEL SOFTWARE	3
1. Introducción	4
• Organigrama de la empresa	5
a) Propósito de Implementar la Gestión de la Configuración del Software	7
b) Benchmarking de herramientas (Hacer un análisis de las herramientas existentes de Control de versiones)	8
Arquitectura de Apache subversión	9
Ventajas:	10
Desventajas:	11
1. Plan de la Gestión de Configuración de Software	12
1.1. Identificación de la configuración:	12
1.2. Control de cambios:	12
1.3. Gestión de versiones:	13
1.4. Auditorías de configuración:	13
1.5. Herramientas y recursos:	13
2. Identificación	14
2.1. Clasificación del Ítem	14
2.2. Diseño de nomenclatura para documentos	15
2.3. Diseño de la estructura del repositorio	16
2.4. Definición de Línea Base	18
REFERENCIAS	19



PLAN DE GESTIÓN DE LA CONFIGURACIÓN DEL SOFTWARE



1. Introducción

1.1 Situación de la empresa

La empresa de software "CondorCode S.A.C." es una pequeña empresa que se dedica al desarrollo de software a medida para clientes de diversos sectores. Desde su fundación hace 5 años, ha completado tres proyectos exitosos y continúa realizando sus operaciones en el mercado.

El primer proyecto fue para una empresa de logística que necesitaba un sistema de seguimiento de envíos en tiempo real. CondorCode desarrolló una aplicación web personalizada que permitía al cliente realizar un seguimiento de sus envíos en tiempo real y recibir notificaciones en caso de retrasos o problemas.

Debido a que fue el primer proyecto de la empresa, surgieron algunos problemas que se tuvieron que corregir. El más grave fue que los clientes se quejaban de que el sistema no se estaba actualizando en tiempo real, lo que les impedía realizar un seguimiento preciso de sus envíos y causaba problemas de logística y de comunicación con sus propios clientes.

El segundo proyecto fue para una empresa de comercio electrónico que necesitaba un sistema de gestión de inventario automatizado. Para ello, la empresa desarrolló una aplicación de software personalizada que permitía al cliente realizar un seguimiento en tiempo real de su inventario, recibir notificaciones en caso de que los niveles de inventario alcanzaran un umbral bajo y automáticamente realizar pedidos a proveedores para reabastecerse.

El problema que se encontró en este proyecto fue que el sistema no estaba realizando pedidos de reabastecimiento de manera eficiente, lo que originaba retrasos en la entrega de los productos y afectaba la satisfacción de los clientes.

El tercer proyecto fue para una empresa de servicios financieros que necesitaba un sistema de gestión de riesgos automatizado. CondorCode desarrolló una aplicación de software personalizada que permitía al cliente realizar un seguimiento en tiempo real del riesgo en su cartera de inversiones y

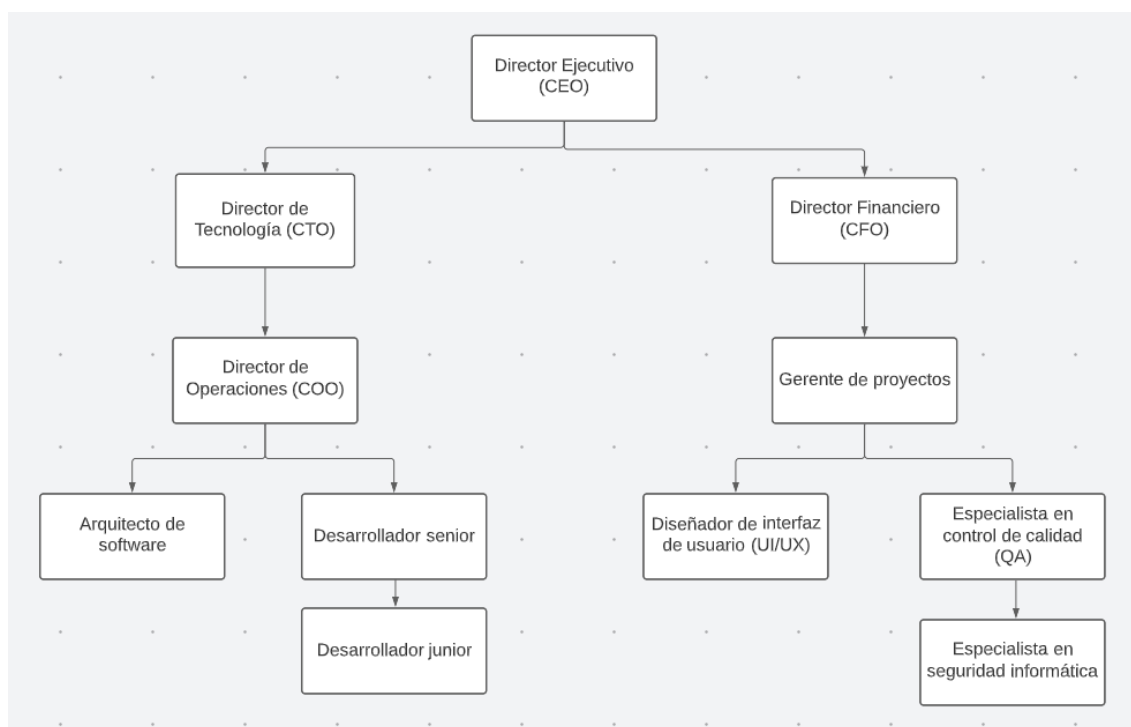


recibir notificaciones en caso de cambios importantes en los mercados financieros. Este proyecto fue desarrollado sin inconvenientes.

Actualmente, CondorCode está trabajando en su cuarto proyecto el cual es el SFPAH (Sistema de Filtrado de Precios de Alimentos e Historial). Este viene orientado a los compradores del hogar, con el fin de ofrecer facilidades en el proceso de adquisición de bienes de la canasta básica familiar.

Se espera que este proyecto se complete en los próximos meses y se espera que Tech Solutions continúe creciendo y recibiendo más proyectos en el futuro.

- **Organigrama de la empresa**



- **CEO (Director ejecutivo):** Encargado de liderar la empresa y tomar decisiones estratégicas importantes. Es responsable de la gestión general y la dirección de la empresa.
- **CTO (Director de tecnología):** Encargado de liderar el equipo de tecnología y de asegurar que la empresa utilice las tecnologías más innovadoras y eficientes. Es



responsable de la estrategia tecnológica de la empresa y de la implementación de proyectos de desarrollo de software.

- **CFO (Director financiero):** Encargado de supervisar las finanzas de la empresa y de tomar decisiones importantes en cuanto a presupuesto, inversiones, impuestos, y otros aspectos financieros. Es responsable de asegurar que la empresa sea financieramente saludable y sostenible.
- **COO (Director de operaciones):** Encargado de liderar y coordinar las operaciones de la empresa. Es responsable de la gestión de los recursos humanos, de la planificación estratégica y de la implementación de políticas y procedimientos operativos.
- **Gerente de proyectos:** Encargado de liderar y supervisar los proyectos de desarrollo de software. Es responsable de la planificación, ejecución y seguimiento de los proyectos, asegurando que se cumplan los objetivos, plazos y presupuestos establecidos.
- **Arquitecto de software:** Encargado de diseñar y planificar la arquitectura de los sistemas y aplicaciones de software de la empresa. Es responsable de la elección de tecnologías y herramientas, y de asegurar que los sistemas sean eficientes, escalables y seguros.
- **Desarrollador senior:** Encargado de liderar y supervisar el desarrollo de software. Es responsable de escribir código limpio y eficiente, de resolver problemas complejos de programación, y de guiar y formar a los desarrolladores junior.
- **Desarrollador junior:** Encargado de escribir y mantener el código de los sistemas y aplicaciones de software de la empresa. Es responsable de aprender y mejorar continuamente sus habilidades de programación.



- **Diseñador de interfaz de usuario (UI/UX):** Encargado de diseñar la interfaz de usuario de los sistemas y aplicaciones de software de la empresa. Es responsable de asegurar que la experiencia de usuario sea intuitiva, atractiva y fácil de usar.
- **Especialista en control de calidad (QA):** Encargado de garantizar la calidad de los sistemas y aplicaciones de software de la empresa. Es responsable de planificar y ejecutar pruebas de software, de identificar y reportar errores, y de asegurar que el software se comporte de manera esperada.
- **Especialista en seguridad informática:** Encargado de garantizar la seguridad de los sistemas y aplicaciones de software de la empresa. Es responsable de identificar y corregir vulnerabilidades de seguridad, de monitorear la seguridad de la red y de los sistemas, y de asegurar que se cumplan los estándares de seguridad establecidos.

a) Propósito de Implementar la Gestión de la Configuración del Software

En los 2 primero proyectos llevados a cabo por CondorCode si bien fueron los que presentaron problemas, en el primero de ellos el cual es el sistema de seguimiento de envíos en tiempo real la gran queja por parte de los usuarios fue motivo suficiente para realizar una identificación del problema que se estaba presentando el cual no mostraba la actualización en tiempo real, los miembros del equipo desarrolladores revisaron el código fuente e identificaron fallas en el apartado de sincronización en la actualización de datos de tiempo real, el cual fue corregido para solución del problema y mejorado para su futura escalabilidad, y no llegase a presentarse nuevamente.

En el segundo proyecto de el sistema de gestión de inventario automatizado el inconveniente que presentaba era de que no estaba realizando pedidos de reabastecimiento de manera eficiente, se tomó atención al caso y se llevó una reunión en equipo para evaluar las



soluciones a dicha problemática, el problema hallado fue el asunto de integración con los otros sistemas que manejaba la empresa, el cuál ocasiona problemas de ejecución del inventario, este problema fue solucionado por el equipo de desarrolladores altamente capacitados.

En el cuarto proyecto que está llevando a cabo CondorCode, se está considerando todos los problemas que se llegó a corregir, para que en este proyecto no llegase a ocurrir los mismos incidentes como otros relacionados y proporcionar al cliente un proyecto de alta calidad

b) Benchmarking de herramientas (Hacer un análisis de las herramientas existentes de Control de versiones)

Tabla 1

Benchmarking de herramientas de control de versiones

Criterios	Herramientas			
	GitKraken ^[1]	Apache Subversion ^[2]	Mercurial ^[3]	Bazaar ^[4]
Funcionalidad	Completa	Bien equipada	Completa	Amplia
Facilidad de uso	Medio	Bajo	Medio	Medio
Escalabilidad	Si	Si	Si	Si
Rendimiento	Excelente	Bueno	Excelente	Bueno
Costo	Costoso	Gratuito	Gratuito	Moderado
Arquitectura	-	Cliente - Servidor	Distribuida	Distribuida
Programado en:	C#	C	Python	Python
Lenguajes que soportan	Java, Python, C ++, C #, Ruby, etc	C, C ++, Java, Python, Ruby, etc	Python, C, C ++, Java, JavaScript, etc	Python, C, C ++, Java, etc

En la tabla 1 se muestra el benchmarking de 4 herramientas de control de versiones, específicamente GitKraken, Apache Subversion, Mercurial y Bazaar.



En términos de funcionalidad, GitKraken y Mercurial son consideradas herramientas completas, mientras que Apache Subversion y Bazaar son bien equipadas y amplias, respectivamente. La facilidad de uso varía, con GitKraken y Bazaar siendo más fáciles de usar que Apache Subversion y Mercurial.

En cuanto a escalabilidad, todas las herramientas tienen la capacidad de escalar. En términos de rendimiento, GitKraken y Mercurial son excelentes, mientras que Apache Subversion y Bazaar son buenos.

El costo también varía, con GitKraken siendo la herramienta más costosa, Apache Subversion y Mercurial siendo gratuitos y Bazaar siendo moderadamente costoso.

En términos de arquitectura, GitKraken no especifica una arquitectura, mientras que Apache Subversion es una arquitectura cliente-servidor y Mercurial y Bazaar tienen arquitecturas distribuidas.

Basándonos en la información de la tabla, se puede concluir que Apache Subversion es una buena opción para el control de versiones debido a que es una herramienta bien equipada, con una arquitectura cliente-servidor, buen rendimiento y es gratuita.

a) Herramienta elegida: Apache subversion[Describir la herramienta elegida, arquitectura, beneficios, etc]

Arquitectura de Apache subversión

La arquitectura de SVN es de tipo cliente-servidor y consta de varios componentes que trabajan juntos para proporcionar un sistema de control de versiones robusto y confiable



(*Version Control with Subversion*, s/f). A continuación se describen los componentes principales :

1. Repositorio centralizado: El corazón de la arquitectura de SVN es el repositorio centralizado, que almacena todo el código fuente y los archivos de un proyecto. El repositorio centralizado es un repositorio de base de datos que mantiene un registro completo de todos los cambios realizados en el código fuente a lo largo del tiempo.

2. Cliente SVN: El cliente SVN es una aplicación que se instala en la computadora de cada desarrollador y se utiliza para acceder y modificar el código fuente almacenado en el repositorio centralizado. El cliente SVN se comunica con el servidor SVN a través de un protocolo de red para recuperar y enviar cambios de código.

3. Servidor SVN: El servidor SVN es el componente central que gestiona el repositorio centralizado. El servidor SVN acepta conexiones de clientes SVN y se encarga de manejar todas las solicitudes de lectura y escritura en el repositorio.

4. Protocolo de red: El protocolo de red utilizado por SVN es el protocolo HTTP (HyperText Transfer Protocol) o HTTPS (HTTP Secure). Esta elección se debe a que estos protocolos son ampliamente utilizados en la web y son conocidos por la mayoría de los administradores de redes y firewalls.

5. Almacenamiento de datos: SVN almacena los datos de código fuente y de los cambios en el repositorio centralizado utilizando un sistema de archivos propio llamado FSFS (File System with Snapshot). Este sistema de archivos permite una alta eficiencia y velocidad de acceso al código fuente y proporciona una forma segura y confiable de almacenar los datos de un proyecto.



Ventajas:

1. Control de versiones: SVN es una herramienta de control de versiones que permite a los desarrolladores mantener un registro completo de todos los cambios realizados en el código fuente a lo largo del tiempo. Esto ayuda a mantener el código organizado, reducir los errores y facilitar la colaboración entre los miembros del equipo.
2. Gestión de cambios: SVN permite a los desarrolladores realizar cambios en el código fuente de forma ordenada y controlada. Los cambios se pueden realizar en diferentes ramas de código y luego fusionarlos de manera eficiente y precisa en el código principal.
3. Seguridad: SVN cuenta con un sistema de autenticación y autorización que permite a los desarrolladores autorizados acceder al código fuente y realizar cambios. Además, SVN mantiene un registro completo de todos los cambios realizados en el código, lo que facilita la identificación y corrección de errores.
4. Personalización: SVN es altamente personalizable y adaptable a las necesidades de cada equipo de desarrollo. Existen herramientas y plugins disponibles que permiten integrar SVN con diferentes editores de código, herramientas de integración continua y sistemas de seguimiento de problemas.
5. Escalabilidad: SVN es capaz de manejar grandes repositorios de código y proyectos de software empresariales complejos. Esto lo convierte en una herramienta ideal para proyectos de software de cualquier tamaño.



Desventajas:

1. Curva de aprendizaje: SVN tiene una curva de aprendizaje empinada para los nuevos usuarios debido a su complejidad y variedad de características.
2. Costo: Aunque SVN es una herramienta de código abierto, algunas empresas pueden requerir de soporte técnico profesional y esto puede generar costos adicionales.
3. Dificultad en la gestión de archivos binarios: SVN puede tener dificultades para manejar archivos binarios grandes o complejos, lo que puede ralentizar el sistema o aumentar el tamaño del repositorio.
4. Necesidad de una conexión a internet: SVN requiere una conexión a internet para acceder al repositorio centralizado, lo que puede ser una desventaja si el equipo de desarrollo está trabajando en una ubicación sin acceso a internet.

1. Plan de la Gestión de Configuración de Software

1.1. Identificación de la configuración:

- Cada versión de software desarrollado por la empresa será identificada por un número de versión único que seguirá el formato "vX.Y", donde "X" representa la versión principal, y "Y" representa la versión de revisión.
- Cada proyecto de software tendrá un nombre único y un número de proyecto asociado.
- Todos los archivos de código fuente, documentación y recursos de proyecto serán etiquetados con el nombre del proyecto, iniciales del ítem y la extensión en que se presenta.



1.2. Control de cambios:

- Todos los cambios en el código fuente, la documentación y los recursos del proyecto se registran en un sistema de seguimiento de problemas o un software de gestión de cambios.
- Cada cambio será revisado y aprobado por el gerente del proyecto, y al menos 1 miembro del equipo de desarrollo antes de que se realice.
- Se mantendrá una línea de base de cada versión de software para garantizar la integridad del código.

1.3. Gestión de versiones:

- Todas las versiones anteriores del software se almacenarán en la biblioteca de desarrollo del repositorio.
- Los registros de cambios y las líneas de base se almacenarán junto con el código fuente correspondiente.
- Los miembros del equipo de desarrollo deben etiquetar cada cambio que realicen en el repositorio de control de versiones para garantizar la trazabilidad.

1.4. Auditorías de configuración:

- Se realizarán auditorías regulares de la configuración para garantizar que el proceso de gestión de la configuración se esté siguiendo correctamente y se esté aplicando de manera uniforme en toda la empresa.
- Se llevará un registro de todas las auditorías y los resultados se utilizarán para mejorar continuamente el proceso de gestión de la configuración.



1.5. Herramientas y recursos:

- La empresa utilizará software de control de versiones y herramientas de seguimiento de problemas para ayudar en la gestión de la configuración.
- Se proporcionará capacitación a los miembros del equipo sobre el uso de estas herramientas y se mantendrán actualizados para garantizar la eficacia de la gestión de la configuración.

2. Identificación

2.1. Clasificación del Ítem

Se presenta los distintos tipos de ítems con los que se trabajarán en este proyecto, los cuales se clasifican según el tipo de documento:

- Evolución (E)

Documentos o archivos ejecutables.

- Fuente (F)

Código fuente y archivos utilizados para compilar una aplicación de software.

- Soporte (S)

Sistema operativo o software base

Tipo	Nombre del Ítem	Nomenclatura	Extensión	Proyecto
E	Plan de Proyecto			SFPAH
E	Documento de especificación de requisito 1	SFPAH-BP	PDF	SFPAH



E	Documento de especificación de requisito 2	SFPAH-SP	PDF	SFPAH
E	Documento de especificación de requisito 3	SFPAH-CP	PDF	SFPAH
E	Documento de especificación de requisito 4	SFPAH-VHP	PDF	SFPAH
E	Documento de especificación de requisito 5	SFPAH-LPF	PDF	SFPAH
E	Documento de especificación de requisito 6	SFPAH-RP	PDF	SFPAH
E	Documento para el diseño de la IU	SFPAH-DI	DOCX	SFPAH
E	Documento para la especificación de la Base de Datos	SS-DBD_01	DOCX	SFPAH

2.2. Diseño de nomenclatura para documentos

Estructuras que se utilizan para nombrar y clasificar los documentos en nuestra organización. La nomenclatura general será:

Acrónimo del proyecto + “-” + Acrónimo del elemento

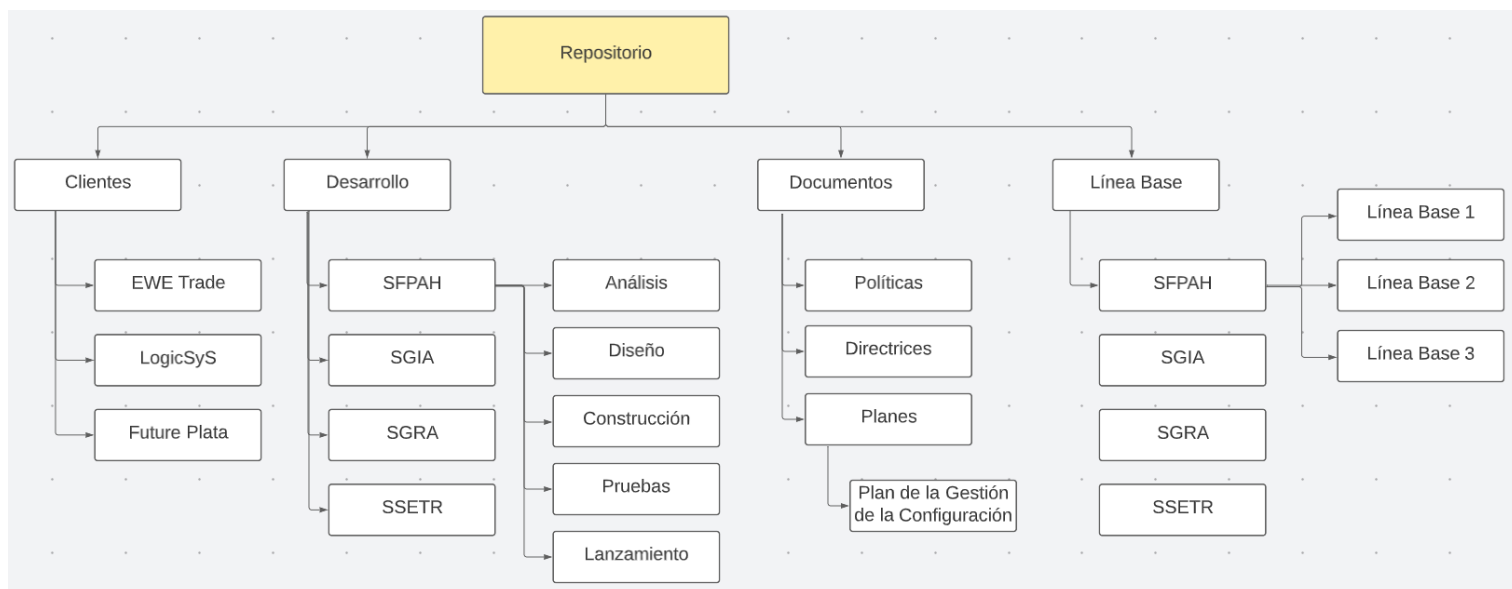
- Respecto al acrónimo de nuestro proyecto, hemos tomado como acrónimo las letras “SFPAH” dado que el nombre de nuestro proyecto es el de Sistema de filtrado de precios de alimentos e historial.
- Respecto al acrónimo del elemento, se podrá nombrar siguiendo la fórmula de la tabla a continuación.



Clasificación del elemento	Ejemplo	Nomenclatura
Ítems que no pertenecen a proyectos en específico (políticas, procesos, guías, etc.).	Política de gestión de proyectos de software	PGPS
Ítems específicos a un proyecto que no estén asociados con un componente del mismo.	Plan de gestión de la configuración del SFPAH	SFPAH-PGC
Ítems en evolución específicos de un proyecto asociados con un componente específico.	Documento de especificación de la base de datos del SFPAH	SFPAH-DBD

2.3. Diseño de la estructura del repositorio

Respecto a la estructura del repositorio en donde se almacenará nuestro proyecto, está estructurada de la siguiente manera:



Nuestro repositorio se compone de 4 bibliotecas las cuales se describen a continuación.

- **Clientes**



- Responsable: Gerente de proyecto
- Actividades: Reuniones, documentación de requerimientos, evaluación de presupuestos.
- Contenidos: Empresas para las que CondorCode S.A.C. entregó productos de software.
- Accesos: Leer, escribir, ejecutar, eliminar
- **Desarrollo**
 - Responsable: Equipo de desarrollo del proyecto
 - Actividades: Diseño de arquitectura, diseño de base de datos, diseño de interfaz de usuario, pruebas de calidad, construcción de módulos.
 - Contenidos: Proyectos desarrollados por la empresa para su lanzamiento al mercado.
 - Accesos: Leer, escribir, ejecutar
- **Documentos**
 - Responsables: Gerente de configuración, Gerente de proyecto
 - Actividades: Elaboración de políticas, lineamientos a seguir, aprobación del plan de gestión de configuración.
 - Contenidos: Documentos que regulan el método de trabajo de la compañía.
 - Accesos: Leer, escribir, ejecutar, eliminar
- **Línea Base**



- Responsable: Gerente de proyecto
- Actividades: Definición de hitos, definición de líneas base de proyectos.
- Contenidos: Lograr un programa funcional al terminar la iteración del hito.
- Accesos: Leer, ejecutar

2.4. Definición de Línea Base

Las líneas base de los proyectos están divididas por hitos para la finalización exitosa del proyecto.

- SFPAH

Línea Base	Actividades	Ítems
Línea Base 1	Fases de análisis y diseño del sistema	SFPAH-CP.XLS SFPAH-LHU.DOCX SFPAH-BP.DOCX SFPAH-SP.DOCX SFPAH-CP.DOCX SFPAH-LPF.DOCX SFPAH-VHP.DOCX SFPAH-RP.DOCX SFPAH-DI.DOCX SFPAH-EW.DOCX SFPAH-DBD.DOCX SFPAH-AYDS.DOCX SFPAH-EAS.PDF SFPAH-RPS.PDF
Línea Base 2	Fases de codificación y pruebas unitarias	-
Línea Base 3	Fases de codificación, pruebas integradas, y lanzamiento	-



REFERENCIAS

Atlassian. (s/f). *Collaboration software for software, IT and business teams*. Atlassian.

Recuperado el 20 de abril de 2023, de <https://www.atlassian.com>

Bazaar. (s/f). Recuperado el 20 de abril de 2023, de <https://bazaar.canonical.com/en/>

GitKraken Legendary Git Tools | *GitKraken*. (s/f). Recuperado el 20 de abril de 2023, de

<https://www.gitkraken.com>

GitKraken Legendary Git Tools | *GitKraken*. (s/f). Recuperado el 20 de abril de 2023, de

<https://www.gitkraken.com>

Control de versiones con Subversion. (s/f). Recuperado el 20 de abril de 2023, de

<https://svnbook.red-bean.com/>

Apache Subversion. (s/f). Recuperado el 20 de abril de 2023, de

<https://subversion.apache.org/>