

Aplicación Móvil para la Gestión de Reservas del Hotel Miraflores de la ciudad de Ibarra

Plan de Configuración

Versión [1.0]

Historia de revisiones

Fecha	Versión	Descripción	Autor
[23/02/2025]	[1.0]	[Primer avance]	[Francisco Quinteros, Xavier Torres y Matías Mosquera]

Contenido

1. INTRODUCCIÓN	3
1.1. PROPÓSITO	3
1.2. ALCANCE	3
1.3. TERMINOLOGÍA	3
1.4. REFERENCIAS	3
2. GESTIÓN DE SCM	3
2.1. ORGANIZACIÓN	3
2.2. RESPONSABILIDADES	4
2.3. POLÍTICAS, DIRECTIVAS Y PROCEDIMIENTOS APLICABLES	5
3. ACTIVIDADES DE SCM.....	5
3.1. IDENTIFICACIÓN DE LA CONFIGURACIÓN	5
3.1.1. <i>Elementos de configuración</i>	5
3.1.2. <i>Nomenclatura de Elementos</i>	6
3.1.3. <i>Elementos de la Línea Base del Proyecto</i>	8
3.1.4. <i>Recuperación de los Elementos de configuración</i>	9
3.2. CONTROL DE CONFIGURACIÓN	10
3.2.1. <i>Solicitud de cambios</i>	10
3.2.2. <i>Evaluación de cambios o Análisis de Impacto</i>	10
3.2.3. <i>Aprobación o desaprobación de cambios</i>	11
3.2.4. <i>Implementación de cambios</i>	11
3.3. ESTADO DE LA CONFIGURACIÓN	11
3.4. AUDITORIAS Y REVISIONES DE CONFIGURACIÓN	12
3.5. CONTROL DE INTERFASES	12
3.6. CONTROL DE SUBCONTRATOS Y VENDEDORES	13
4. CALENDARIO.....	13
5. RECURSOS	13
6. MANTENIMIENTO DEL PLAN DE SCM.....	14

1. Introducción

Este Plan de Configuración establece las actividades y procedimientos de Gestión de Configuración de Software (SCM) que serán aplicados en el desarrollo de la aplicación móvil para la gestión de reservas del Hotel Miraflores de la ciudad de Ibarra. Su objetivo es garantizar la integridad, trazabilidad y control de los elementos de configuración del sistema, facilitando el mantenimiento y la evolución del software.

1.1. Propósito

El propósito de este documento es definir los productos sujetos a control de configuración, así como los procedimientos y responsabilidades del equipo en la gestión de cambios y control de versiones.

1.2. Alcance

El Plan de Configuración está basado en algunos supuestos clave:

- El proyecto tiene una duración limitada a 13 semanas, por lo que se busca una rápida respuesta a los cambios, minimizando la burocracia en los procedimientos.
- El modelo de proceso se basa en un desarrollo incremental, con múltiples iteraciones que requieren un control riguroso sobre los productos generados y los cambios aprobados.
- Se incluirán la mayor cantidad de productos posibles dentro del control de configuración, considerando las restricciones de tiempo y capacidad organizativa del equipo.
- La elección de los elementos de configuración se realizará en función de los entregables, siendo esta responsabilidad del responsable de SCM (SCMR) con apoyo del equipo de desarrollo.

1.3. Terminología

- CCB (Configuration Control Board) Comité de Control de Configuración.
- CI (Configuration Item) elemento bajo gestión de Configuración.
- SCA (Software Change Authorization) Autorización de Cambio en el Software.
- SCM (Software Configuration Management) Gestión de Configuración del Software.
- SCMR (SCM Responsable) Responsable de SCM.
- SCR (System/Software Change Request) Petición de Cambio en el Sistema/Software.
- SQA (Software Quality Assurance) Aseguramiento de la Calidad del Software.
- SQAR (SQA Responsable) Responsable de SQA.

1.4. Referencias

- ANSI/IEEE Std 828-1990, IEEE Standard for Software Configuration Management Plans.
- Modelo de Proceso basado en Scrum y desarrollo incremental.

2. Gestión de SCM

2.1. Organización

La gestión de configuración estará organizada en torno a un equipo liderado por el **responsable de SCM (SCMR)**, quien garantizará el cumplimiento de los procesos de control de versiones, gestión de cambios y mantenimiento de la línea base del software. El equipo de trabajo se estructura en diferentes roles, cada uno con responsabilidades específicas dentro del **ciclo de vida del software**.

A nivel organizacional, se identifican las siguientes líneas de trabajo dentro del proyecto:

Línea de Trabajo	Responsabilidades en SCM
SCMR (responsable de Gestión de Configuración)	Supervisión y control de versiones, aseguramiento de la línea base del proyecto, auditoría de SCM, definición de políticas de nomenclatura y control de versiones.
Desarrolladores	Implementación de cambios, mantenimiento del historial de versiones, validación de estabilidad del código.
Equipo de Pruebas y Calidad (SQA)	Evaluación de la integridad de los cambios antes de la liberación, pruebas de regresión, generación de reportes de estado.
Gerente del Proyecto	Aprobación de cambios significativos, coordinación con los stakeholders.
Comité de Control de Configuración (CCB)	Evaluación, aprobación o rechazo de cambios críticos en la línea base del software.

El **Comité de Control de Configuración (CCB)** estará compuesto por el **SCMR**, el **Gerente del Proyecto** y representantes de los equipos de **Desarrollo y Pruebas**, quienes analizarán el impacto de los cambios antes de su implementación.

2.2. Responsabilidades

Se definen las siguientes responsabilidades dentro del proceso de gestión de configuración:

Responsable	Actividad
Francisco Quinteros (SCMR)	Definir políticas de gestión de configuración, establecer lineamientos de nomenclatura y control de versiones, coordinar con los desarrolladores y equipo de pruebas. Realizar auditorías de configuración.
Matías Mosquera (Desarrollador Principal)	Implementar cambios conforme a los procedimientos establecidos, mantener el historial de versiones, garantizar la estabilidad del código y documentar los cambios.
Xavier Torres (Equipo de Pruebas y Calidad - SQA)	Validar la integridad de los elementos de configuración antes de su liberación, realizar pruebas automatizadas y manuales, generar reportes de errores y sugerencias de optimización.
Gerente del Proyecto	Aprobar cambios estratégicos, coordinar con stakeholders y asegurar la alineación del desarrollo con los objetivos del negocio.
CCB (Comité de Control de Configuración)	Evaluar, aprobar o rechazar cambios críticos que afecten la línea base del software. Definir criterios de aceptación de cambios y su impacto en el proyecto.

El SCMR debe proveer la infraestructura y el entorno de configuración para el proyecto, asegurando que todos los integrantes del equipo comprendan y puedan ejecutar las actividades de SCM establecidas en este plan.

Además, el SCMR deberá:

- Identificar los **elementos de configuración** para establecer la **línea base del proyecto**.
- Definir y aplicar **políticas de nomenclatura** para facilitar la identificación y rastreo de los elementos de configuración.
- Gestionar el **control de configuración**, estableciendo estándares y procedimientos para el manejo de cambios.
- Generar **reportes de estado de la configuración**, documentando el historial de revisiones y liberaciones.

- **Auditar la línea base del software** para verificar que el sistema en desarrollo es consistente y que todos los cambios han sido aprobados formalmente.

2.3. Políticas, directivas y procedimientos aplicables

Las **políticas y procedimientos** aplicables incluyen:

- ✓ **Gestión de cambios:** Todo cambio en los elementos de configuración debe ser **documentado, evaluado y aprobado** antes de su implementación.
- ✓ **Control de versiones:** Se utilizará un **sistema de control de versiones centralizado** (GitHub) para garantizar la trazabilidad y recuperación de versiones anteriores.
- ✓ **Auditorías y verificación:** Se realizarán **auditorías periódicas** de la línea base del software para garantizar la conformidad del sistema con los estándares de calidad y seguridad.

Estos procedimientos aseguran una **gestión de configuración estructurada**, minimizando errores y facilitando la recuperación de versiones previas cuando sea necesario.

3. Actividades de SCM

Las actividades clave de SCM incluyen:

- ✓ Identificación y control de los elementos de configuración.
- ✓ Control de versiones y cambios.
- ✓ Gestión de la línea base del proyecto.
- ✓ Verificación y auditoría de la configuración.

3.1. Identificación de la configuración

La identificación de la configuración es fundamental para garantizar la trazabilidad y el control del sistema. Se definen los siguientes aspectos clave:

3.1.1. Elementos de configuración

Para este proyecto, los elementos de configuración estarán alineados con los entregables del Modelo de Proceso, asegurando su trazabilidad en cada iteración. No todos los entregables serán elementos de configuración, solo aquellos esenciales para garantizar la integridad del sistema.

El responsable de SCM (SCMR) determinará qué productos serán incluidos como elementos de configuración, tomando en cuenta la recuperación de versiones anteriores y su impacto en el desarrollo del software.

Se establece una línea base por iteración en cada fase del desarrollo, definiendo:

- ✓ Eventos que generan una nueva línea base.
- ✓ Elementos bajo control en la línea base.
- ✓ Procedimientos para establecer y modificar la línea base.
- ✓ Autorizaciones requeridas para cambios en documentos de línea base.

Los elementos bajo control de configuración incluyen:

- ✓ Código fuente: Archivos en Ionic, Angular, MySQL.
- ✓ Diseño de UI/UX: Prototipos creados en Figma.
- ✓ Documentación técnica: Manuales de usuario, especificaciones técnicas.

- ✓ Modelos de datos: Esquemas de base de datos.
- ✓ Reportes de pruebas: Documentación de pruebas de validación y verificación.

3.1.2. Nomenclatura de Elementos

Para garantizar la correcta identificación de los elementos de configuración, se utilizará un sistema de nomenclatura estructurado. Este permitirá diferenciar versiones y asegurar la trazabilidad en la gestión del software.

El formato para la nomenclatura será: **[Nomenclatura][GX][vY].[extensión]**

- ✓ Nomenclatura: Nombre asignado al entregable.
- ✓ X: Número de grupo.
- ✓ Y: Número de versión.
- ✓ Extensión: Formato del archivo.

Ejemplo: **RQALS_G1_v2.doc** → Documento de "Alcance del Sistema", Grupo 1, versión 2.

Los elementos se almacenarán según la fase e iteración, facilitando la recuperación de versiones y el seguimiento de cambios.

Para asegurar la coherencia en la gestión de la configuración, cada entregable en el modelo de proceso sigue una nomenclatura específica según su disciplina:

Requerimientos:

Nomenclatura	Entregable
RQACT	Acta de Reunión de Requerimientos
RQDRQ	Especificación de Requerimientos
RQMOD	Modelo de Casos de Uso
RQRSU	Requerimientos Suplementarios
RQDVC	Documento de Validación con el Cliente
RQPIU	Pautas para Interfase de Usuario
RQRCA	Requerimientos Candidatos
RQALS	Alcance del Sistema
RQGLO	Glosario
RQOOMDO	Modelo de Dominio
RQOODRP	Documento de Requerimientos para el Prototipo
RQGXXNOM	Nomenclatura

Diseño:

Nomenclatura	Entregable
DSMDI	Modelo de Diseño
DSARQ	Descripción de la Arquitectura
DSOOMDA	Modelo de Datos
DSOODDP	Documento de Diseño del Prototipo

Implementación:

Nomenclatura	Entregable
IMEDT	Estándar de Documentación Técnica
IMEI	Estándar de Implementación
IMPR	Prototipo
IMIIN	Informe de Integración
IMDT	Documentación técnica
IMIVU	Informe de Verificación Unitaria

IMOOPII	Plan de Integración de la Iteración
IMOOMIM	Modelo de Implementación
IMOOEJI	Ejecutable de la Iteración
IMOORRP	Reporte de Revisión por Pares
IMOOCVU	Clases de la Verificación Unitaria de Módulo
IMGXICO	Informe de Consolidación
IMGXEST	BC Con Estilos
IMGXCON	BC Consolidado
IMGXNUC	BC Núcleo
IMGXMOD	BC Módulo

Verificación:

Nomenclatura	Entregable
VRPVV	Plan de Verificación y Validación
VRDAP	Documento de Evaluación y Ajuste del Plan de V & V
VRPVI	Plan de Verificación de la Iteración
VRMCP	Modelo de Casos de Prueba
VRIVD	Informe de Verificación de Documento
VRIVI	Informe de Verificación de Integración
VRIVS	Informe de Verificación del Sistema
VRRPR	Reportes de Pruebas
VREV	Evaluación de la Verificación
VRIFV	Informe Final de Verificación

Implantación (IP):

Nomenclatura	Entregable
IPMSU	Materiales para Soporte al Usuario (Se pueden usar sufijos para identificar cada ítem dentro del material Ej. IPMSUMU para Manual de Usuario)
IPMCA	Materiales para Capacitación
IPPS	Presentación del Sistema
IPPLA	Plan de Implantación
IPVPR	Versión del Producto
IPOOEDU	Estándar de Documentación de Usuario
IPOORFPA	Reporte Final de Pruebas de Aceptación

Gestión de Configuración y Control de Cambios (SCM):

Nomenclatura	Entregable
SCMPLA	Plan de Configuración
SCMMAC	Manejo del Ambiente Controlado
SCMGC	Gestión de Cambios
SCMRV	Registro de Versiones
SCMILB	Informe de la Línea Base del Proyecto
SCMIF	Informe Final de SCM

Gestión de Calidad (SQA):

Nomenclatura	Entregable
SQAPLA	Plan de Calidad
SQADAP	Documento de Evaluación y Ajuste del Plan de Calidad
SQARTF	Informe de RTF
SQAES	Entrega Semanal de SQA
SQAIR	Informe de Revisión de SQA
SQADV	Descripción de la Versión
SQANV	Notas de la Versión
SQAIF	Informe Final de SQA

Gestión de Proyecto (GP):

Nomenclatura	Entregable
GPPLA	Plan de Proyecto
GPISP	Informe de Situación del Proyecto
GPEM	Estimaciones y Mediciones
GPDRI	Documento de Riesgos
GPRAC	Registro de Actividades
GPIFP	Informe Final de Proyecto
GPARE	Acta de la Reunión de Equipo
GPPIT	Plan de la Iteración
GPPDE	Plan de Desarrollo
GPICF	Informe de Conclusiones de la Fase
GPPDIP	Presentación en Diapositivas del Proyecto
GPPDP	Presentación al Director del Proyecto
GPARD	Acta de la Reunión con el Director del Proyecto
GPOODAP	Documento de Evaluación y Ajuste al Plan de Proyecto
GPIARI	Acta de la Reunión de Integración

Comunicación (COM):

Nomenclatura	Entregable
COMDI	Documento Informativo
COMENS	Encuesta de Satisfacción del Cliente
COMEVS	Evaluación de Satisfacción del Cliente

3.1.3. Elementos de la Línea Base del Proyecto

La **Línea Base del Proyecto** se define como el conjunto de elementos que serán controlados y gestionados en cada fase e iteración del desarrollo de la aplicación móvil del **Hotel Miraflores**. Cada iteración genera productos que deben ser rastreables y verificables en el sistema de control de versiones.

A continuación, se presentan los elementos de la línea base organizados por **fase del proyecto** e **iteraciones dentro de cada fase**:

FASE	ITERACIÓN	Elemento	Descripción	Disciplina
Requerimientos	Iteración 1	RQDRQ - Especificación de Requerimientos	Documento detallado con los requisitos del sistema	Gestión de Requerimientos
Diseño	Iteración 2	DSARQ - Descripción de la Arquitectura	Definición de la arquitectura de software y tecnologías utilizadas	Arquitectura de Software
Implementación	Iteración 3	IMPR - Prototipo funcional	Primer prototipo funcional de la aplicación	Desarrollo de Software
Verificación	Iteración 4	VRPVV - Plan de Verificación y Validación	Estrategia y metodología de pruebas para el software	Pruebas y QA
Implantación	Iteración 5	IPMSU - Manual de Usuario	Documentación dirigida al usuario final	Soporte y Capacitación
SCM y Control de Cambios	Iteración 6	SCMPLA - Plan de Configuración	Documento con estrategias y control de cambios	Gestión de Configuración

Estos elementos serán revisados en cada iteración antes de ser aprobados y almacenados en la línea base del sistema.

3.1.4. Recuperación de los Elementos de configuración}

Para garantizar la correcta administración y recuperación de los elementos de configuración, se implementará un repositorio de control de versiones (GitHub) que permitirá gestionar el código, la documentación y los datos de manera eficiente.

Cada elemento de configuración será almacenado en un entorno controlado con los siguientes criterios:

✓ **Código fuente:**

- Formato: Archivos en .ts, .html, .scss (Ionic/Angular) y .sql (Base de Datos).
- Ubicación: Repositorio de código en GitHub, organizado por ramas (main, develop, feature).
- Requerimientos de aceptación: Validación de código por revisión de pares antes del merge.
- Control de acceso: Acceso restringido mediante credenciales y permisos de read/write.

✓ **Documentación técnica y entregables:**

- Formato: Archivos en .docx, .pdf, .md.
- Ubicación: Almacenamiento en Drive/SharePoint, vinculado a GitHub Wiki.
- Requerimientos de aceptación: Validación de versión final por el SCMR antes de su publicación.
- Control de acceso: Permisos de lectura para todos los miembros, edición para responsables de documentación.

✓ **Modelos de datos y base de datos:**

- Formato: Esquemas en .sql, diagramas en .png/.pdf.
- Ubicación: Base de datos de prueba alojada en servidor local o nube.
- Requerimientos de aceptación: Pruebas de consistencia en cada iteración antes de su actualización.
- Control de acceso: Solo el equipo de backend tendrá privilegios de escritura sobre la base de datos.

✓ **Reportes de pruebas y validación:**

- Formato: Archivos en .xlsx, .pdf, .json.
- Ubicación: Repositorio de resultados en herramientas de QA (Jira/TestRail).
- Requerimientos de aceptación: Análisis de métricas y firma de aprobación por equipo de QA.
- Control de acceso: Acceso solo para equipo de pruebas y SCMR.

Esta estrategia asegura la integridad y trazabilidad de los elementos de configuración, facilitando la recuperación de versiones anteriores y garantizando la estabilidad del sistema.

3.2. Control de configuración

Esta sección detalla las actividades de **solicitud, evaluación, aprobación e implementación de cambios** a los elementos de la línea base del sistema. Los cambios pueden ser correcciones o mejoras y se aplicarán a los siguientes aspectos clave:

- **Requerimientos**
- **Diseño del sistema**
- **Arquitectura**
- **Herramientas de desarrollo**
- **Documentación del proyecto** (nuevos documentos o modificaciones de los existentes).

El control de cambios se realizará mediante un sistema de control de versiones (Git), asegurando trazabilidad y recuperación de versiones anteriores en caso de ser necesario.

3.2.1. Solicitud de cambios

Cada solicitud de cambio debe incluir los siguientes elementos:

- ✓ Descripción detallada del cambio: Explicación del ajuste que se quiere realizar.
- ✓ Justificación y análisis de impacto: Evaluación de cómo afectará el cambio al sistema.
- ✓ Evaluación de costos y plazos: Determinar el tiempo y esfuerzo necesario para implementar el cambio.
- ✓ Aprobación del Comité de Control de Configuración (CCB): Para garantizar que los cambios sean validados antes de su implementación.

Cuando se realiza una solicitud de cambio, se debe actualizar el documento de “Solicitud de Cambio” para su registro y aprobación.

3.2.2. Evaluación de cambios o Análisis de Impacto

Para evaluar un cambio, se deben analizar sus implicaciones dentro del proyecto. El proceso de evaluación consta de dos pasos:

1. Planificación de la evaluación del cambio:

- ✓ Revisión de la solicitud de cambio y su alcance.
- ✓ Determinación de los responsables del análisis.
- ✓ Desarrollo de un Plan de Evaluación del Cambio.
- ✓ Si el cambio involucra al cliente, se debe obtener su aprobación.

2. Evaluación del cambio:

- ✓ Determinar el impacto en los productos técnicos.
- ✓ Analizar cómo afecta a los planes del proyecto.
- ✓ Revisar el impacto en los acuerdos con el cliente.
- ✓ Evaluar riesgos adicionales para el proyecto.

3.2.3. Aprobación o desaprobación de cambios

Para cada cambio, se establecerá un Comité de Control de Configuración (CCB) encargado de analizar, aprobar o rechazar modificaciones críticas en el sistema.

El CCB estará compuesto por:

- ✓ Administrador del proyecto (Obligatorio).
- ✓ Arquitecto del sistema (Opcional).
- ✓ Analista de negocio (Opcional).
- ✓ Desarrolladores (Opcional, según el impacto del cambio).
- ✓ Responsable de SCM (Obligatorio).
- ✓ Cliente o stakeholders (Opcional, según la magnitud del cambio).

Para cambios estratégicos o críticos, se elevará la solicitud a un comité de nivel superior integrado por el Gerente de Proyecto, quien tomará la decisión final en caso de disputas.

3.2.4. Implementación de cambios

Una vez aprobado un cambio, se inicia el proceso de implementación en función del ciclo de vida del software.

- ✓ **Integración de cambios en el sistema** dentro del repositorio de código.
- ✓ **Monitoreo del progreso** en cada etapa del desarrollo.
- ✓ **Ejecución de pruebas unitarias y de integración** para validar el cambio.
- ✓ **Liberación en entornos de prueba y producción**, asegurando la estabilidad del sistema.

Dado que se sigue un enfoque **incremental e iterativo**, cada cambio debe ser introducido en la fase correspondiente (**Requerimientos, Análisis, Diseño, Implementación, Verificación**), garantizando que todas las áreas impactadas sean actualizadas.

3.3. Estado de la configuración

El estado de la configuración del sistema debe ser constantemente **monitoreado y documentado**.

Las actividades clave incluyen:

- ✓ **Seguimiento de versiones y cambios realizados** mediante herramientas de control de versiones.
- ✓ **Generación de reportes de estado**, detallando el estado actual de la línea base del proyecto.
- ✓ **Control de acceso a los datos de configuración**, limitando la edición y modificación de archivos críticos solo a los usuarios autorizados.

Reportes de Estado

Los reportes de configuración deben incluir la siguiente información mínima:

- ✓ Versión inicial aprobada de cada elemento de configuración.
- ✓ Estado de las solicitudes de cambio registradas.
- ✓ Estatus de la implementación de cambios aprobados.

Si se utiliza una herramienta automática para gestionar la configuración (**Git, Jenkins, SonarQube**), se especificará su funcionalidad y modo de uso dentro del equipo.

3.4. Auditorías y revisiones de configuración

Las auditorías serán realizadas antes de cada liberación o actualización de una versión prioritaria del sistema.

- ✓ **Objetivo:** Verificar que la línea base del software sea consistente y bien definida.
- ✓ **Elementos auditados:** Se seleccionarán componentes clave de la línea base.
- ✓ **Agenda de auditorías:** Antes de liberar una nueva versión del software.
- ✓ **Conducción:** Dirigida por el SCMR con el equipo de desarrollo.
- ✓ **Participantes:** SCMR, desarrolladores y revisores asignados.
- ✓ **Documentos requeridos:** Solicitudes de cambio aprobadas, reportes de estado.
- ✓ **Reporte de deficiencias y acciones correctivas:** Se documentarán ajustes necesarios antes de la liberación.
- ✓ **Criterios de aprobación:** El SCMR decidirá si la versión está lista para producción.

Estas auditorías garantizarán que la línea base del software esté alineada con los objetivos del proyecto y que los cambios se hayan implementado de manera efectiva.

3.5. Control de Interfases

El **control de interfases** se encarga de la gestión de cambios en los elementos de configuración que modifican la interacción con sistemas externos o módulos fuera del alcance del Plan de Configuración.

El **responsable de SCM (SCMR)** será el encargado de:

- ✓ **Supervisar y documentar los cambios en las interfaces de integración** con otros sistemas.
- ✓ **Evaluar el impacto de los cambios en la interoperabilidad**, garantizando que no afecten la funcionalidad del sistema.
- ✓ **Realizar pruebas de compatibilidad** cuando se actualicen APIs externas o módulos conectados.
- ✓ **Hay que asegurar que las interfaces cumplan con los estándares de comunicación** definidos en la arquitectura del sistema.

Herramientas utilizadas para el control de interfases:

- ✓ Postman para pruebas de APIs.
- ✓ Swagger para documentación de servicios REST.
- ✓ Docker para simulación de entornos de integración.

3.6. Control de subcontratos y vendedores

N/A. En este proyecto no se han considerado subcontratos de desarrollo externo ni adquisición de software de terceros. Sin embargo, en caso de incluir proveedores de software o servicios en el futuro, se establecerán los siguientes lineamientos:

- ✓ **Evaluación de proveedores** en función de la compatibilidad con los requisitos del sistema.
- ✓ **Garantía de cumplimiento con estándares de seguridad y calidad**, verificando certificaciones como **ISO 27001** o **GDPR** si aplican.
- ✓ **Monitoreo del rendimiento de los proveedores**, asegurando entregables alineados con el cronograma del proyecto.

4. Calendario

El **cronograma de actividades de SCM** seguirá el desarrollo iterativo basado en **Scrum**, asegurando que cada sprint concluya con la **revisión y validación de los elementos de configuración**.

Hitos principales de SCM

Actividad	Fecha de Inicio	Fecha de Finalización
Definición de la línea base	Semana 1	Semana 2
Implementación del control de cambios	Semana 3	Semana 4
Auditoría inicial de configuración	Semana 5	Semana 6
Primera entrega de código controlado	Semana 6	Semana 7
Verificación de la consistencia de la línea base	Semana 8	Semana 9
Auditoría final antes de la implementación	Semana 12	Semana 13
Revisión post-implementación	Semana 13	Semana 14

Dependencias entre actividades:

- ✓ La **definición de la línea base** debe completarse antes de la implementación del control de cambios.
- ✓ No se podrán hacer **entregas de código** sin pasar por auditoría previa de configuración.
- ✓ La **auditoría final** se llevará a cabo antes de la implementación del software en producción.

5. Recursos

Para la correcta ejecución de la gestión de configuración, se identifican los siguientes **recursos técnicos y humanos** necesarios:

✓ **Herramientas de software:**

- ✦ **GitHub:** Para control de versiones y administración del código.
- ✦ **Figma:** Para diseño y documentación de interfaces de usuario.
- ✦ **MySQL y XAMPP:** Para la gestión de bases de datos y simulación de entornos locales.
- ✦ **Jenkins/GitHub Actions:** Para integración continua y despliegue automatizado.
- ✦ **Jira:** Para la gestión de tareas y seguimiento de cambios.

✓ **Técnicas utilizadas:**

- ✦ **Revisión de código** por pares antes de fusionar cambios en la rama principal.
- ✦ **Automatización de pruebas unitarias** y de regresión para garantizar calidad del software.
- ✦ **Gestión de configuraciones basada en Infraestructura como Código (IaC)** cuando aplique.

✓ **Capacitación del equipo:**

- ✦ **Entrenamiento en gestión de configuración**, incluyendo el uso de Git y control de versiones.
- ✦ **Formación en ciberseguridad**, enfocada en el manejo de accesos y prevención de ataques.
- ✦ **Capacitación en herramientas de control de cambios**, auditoría y pruebas de software.

6. **Mantenimiento del Plan de SCM**

Para garantizar que el **Plan de Configuración (SCM)** continúe siendo efectivo a lo largo del proyecto, se establecen las siguientes directrices de mantenimiento y revisión:

✓ **Responsable de monitoreo:**

- ✦ El **SCMR (Francisco Quinteros, Matías Mosquera y Xavier Torres)** serán los encargados de supervisar el cumplimiento del plan y gestionar actualizaciones cuando sea necesario.

✓ **Frecuencia de revisión:**

- ✦ Se realizará una **revisión al inicio de cada fase del proyecto** y ajustes según sea necesario.
- ✦ Se evaluará la **efectividad de los procedimientos** de gestión de configuración al final de cada iteración.

✓ **Evaluación y aprobación de cambios:**

✚ Cualquier modificación al **Plan de SCM** debe ser evaluada por el **Comité de Control de Configuración (CCB)** y aprobada por el **Gerente del Proyecto**.

✓ **Comunicación de cambios:**

✚ Los cambios aprobados serán **documentados y distribuidos** al equipo a través del repositorio del proyecto en **GitHub** y mediante reuniones de seguimiento en **Jira**.

✓ **Registro de mejoras y lecciones aprendidas:**

✚ Se documentarán **buenas prácticas** identificadas durante la ejecución del proyecto para su aplicación en futuras fases.

✚ Se realizará una **evaluación post-implementación** para identificar áreas de mejora en la gestión de configuración.