



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

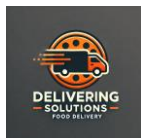
PÁGINA

0 - 14

Plan de configuración

Producto: Sistema de gestión para repartos de
comida a domicilio

Versión: 1.0.0



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

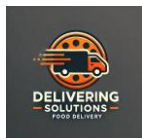
0

PÁGINA

1 - 14

Contenido

1.	Introducción.....	2
1.1	Objetivo y alcance	2
1.2	Terminología	2
1.3	Referencias	3
2.	Gestión de SCM.....	3
2.3.	Control de cambios	3
2.4.	Control de versiones	4
2.5.	Registro de Estado de Configuración	4
2.6.	Auditoría de Configuración.....	4
3.	Roles y Responsabilidades.....	4
3.1.	Gestor de configuración.....	4
3.2.	Desarrollador	5
3.3	Comité de Control de Cambios (CCB)	5
3.4.	Equipo de Calidad (QA)	5
3.5.	Verificador de Configuración	5
3.6.	Usuario Final	5
4.	Herramientas y Entorno de SCM.....	6
4.1.	Soporte a las 3C	6
4.2.	Java y Spring	6
4.3.	Metodología de Desarrollo Iterativa e Incremental.....	6
4.3.1.	Planificación de Iteraciones	6
4.3.2.	Versionado y Gestión de Configuración	6
4.3.3.	Monitorización del Avance	7
4.3.4.	Herramientas SCM y Construcción de Software	7
5.	Procedimientos de Cambio y Flujo de Trabajo	7
5.1.	Procedimientos de Cambio	7
5.2.	Flujo de Trabajo	8
6.	Baseline.....	9
7.	Calendario	9
8.	Puntos de Control.....	9
9.	Gestión de Riesgos.....	10
9.1.	Identificación de riesgos	10



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

2 - 14

9.2. Evaluación de Riesgos	11
10. Procedimientos de Auditoría y Reportes	11
10.1. Objetivos de la Auditoría:	11
10.2. Alcance de la Auditoría:	11
10.3. Procedimiento de la Auditoría:	12
10.4. Documentación de la Auditoría	12
10.5. Seguimiento de Auditoría	12

1. Introducción

El documento presente (basado en el estándar **IEEE 828-2012**) describe las actividades de gestión de configuración de software que se deben llevar a cabo durante el desarrollo del proyecto.

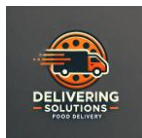
1.1 Objetivo y alcance

La gestión de configuración establecida en el proyecto **Reparto a domicilio** busca asegurar que todas las versiones de los componentes del sistema estén debidamente documentadas, controladas y accesibles para los miembros del equipo, manteniendo así la coherencia, rastreabilidad y estabilidad del sistema a lo largo de su ciclo de vida.

Esta política se aplica a todos los elementos de configuración, incluidos el código fuente, documentación, configuraciones de entorno, bases de datos, y cualquier otro artefacto necesario para la construcción, prueba, y despliegue del sistema.

1.2 Terminología

- Configuration Item (CI): Componente individual del sistema de configuración. Puede incluir código fuente, documentación, scripts de configuración, etc.
- Software Configuration Management (SCM): Control y documentación de los cambios en los elementos de configuración a lo largo del ciclo de vida del software.
- Configuration Control Board (CCB): Comité de control de configuración
- Software Change Authorization (SCA): Proceso de aprobación para cualquier cambio propuesto en un elemento de configuración.
- System Change Request (SCR): Solicitud para realizar un cambio en el sistema (código fuente, arquitectura, CIs...)
- Verification Responsible (VR): Responsable de verificar los elementos de configuración y cambios realizados.
- Baseline: Producto considerado punto de referencia aprobado. Sirve de base para futuros desarrollos.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

3 - 14

- Branching: Proceso de creación de líneas de desarrollo paralelas en el sistema de control de versiones.
- Build: Conjunto compilado de elementos de configuración que forman una versión ejecutable del sistema o subsistema. Incluye código fuente y configuraciones específicas.

1.3 Referencias

- IEEE Standard for Configuration Management in Systems and Software Engineering, IEEE Standard 828-2012, IEEE, 2012.
- Git. (2024). *Documentation*. Retrieved from <https://git-scm.com/doc>.
- Pérez-Castillo, R. (2024). *Ingeniería del Software en Acción. De la velocidad, a la excelencia*

2. Gestión de SCM

En este apartado se describen las responsabilidades y responsables para la realización de las actividades de gestión de configuración dentro del proyecto

2.2. Identificación de la configuración

Se debe establecer una identificación única para cada **CI**, teniendo un nombre único y una estructura jerárquica clara en el repositorio.

- Código fuente en la carpeta **/src**, dividiendo las rutas en **/entidades**, **/dominio**, **/persistencia** y **/service**.
- Documentación en la carpeta **/docs**
- Archivos de configuración en **/config**
- BBDD local en **/bbdd**

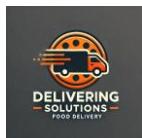
2.3. Control de cambios

Todo cambio debe solicitarse mediante un **SCR** que debe incluir:

- Descripción del cambio propuesto.
- Justificación y prioridad.
- Evaluación de impacto y plan de pruebas

Las **SCR** son revisadas por el **CCB**, para evaluar el impacto y la necesidad del cambio

Solo los cambios aprobados pueden ser implementados, siendo documentados en el repositorio de **SCM**.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

4 - 14

2.4. Control de versiones

Con la intención de documentar y controlar las versiones de cada elemento de configuración, usaremos **git** como sistema de control de versiones.

La estructura es la siguiente:

- **master**: Para la versión estable en producción
- **development**: Para la integración de nuevas funcionalidades antes de su despliegue
- **feature**: ramas temporales para desarrollo de nuevas funcionalidades
- **hotfix**: rama para corrección de errores

Se empleará el versionado semántico para etiquetar versiones importantes (ej: v1.0.0)

2.5. Registro de Estado de Configuración

Con el fin de mantener un registro del estado actual de cada elemento, se actualizarán los estados de cada CI a medida que avanza el proyecto (*en desarrollo, en pruebas, en producción...*)

Se deben generar informes semanales del estado de configuración.

2.6. Auditoría de Configuración

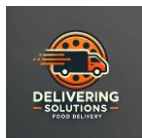
Para asegurar que se cumple con las políticas de SCM, se realizarán auditorías de configuración antes de cada despliegue o cuando el CCB designe un punto crítico del ciclo de vida.

3. Roles y Responsabilidades

Dado que nuestra empresa dispone únicamente de 4 socios, la segregación de responsabilidades es prácticamente inviable. No obstante, mostramos los siguientes cargos/roles de cada persona, pudiendo compartir y solapar roles entre socios.

3.1. Gestor de configuración

- Debe supervisar la implementación de la SCM asegurando que se cumplan los procedimientos y políticas.
- Debe mantener y gestionar el repositorio de configuración, incluyendo la organización de ramas, etiquetado de versiones y la documentación de configuraciones.
- Aprobar cambios críticos y coordinar el proceso de revisión de cambios en conjunto con el CCB.
- Generar y distribuir informes de estado de configuración y de auditorías.
- El encargado será Pablo Verdúñez Gervaso.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

5 - 14

3.2. Desarrollador

- Debe asegurar que los cambios en el código fuente sigan las políticas de SCM establecidas.
- Debe crear y gestionar las ramas de trabajo correspondientes.
- Debe documentar y describir cada cambio realizado en el repositorio.
- Debe completar las SCR cuando se proponga un cambio significativo y colaborar en el proceso de revisión de cambios.
- Los encargados serán: Ismael López, Pablo Verdúñez, Jorge López y Marco Muñoz.

3.3 Comité de Control de Cambios (CCB)

- Debe revisar y evaluar todas las SCR, determinando su viabilidad, impacto y prioridad.
- Debe aprobar o rechazar cambios propuestos.
- Debe asegurar que cada cambio contenga un plan de pruebas y evaluación de riesgos.
- Debe coordinarse con el gestor de configuración para integrar correctamente los cambios
- Los encargados serán: Ismael López, Pablo Verdúñez, Jorge López y Marco Muñoz.

3.4. Equipo de Calidad (QA)

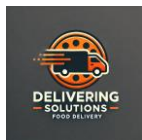
- Debe verificar que los cambios realizados en los elementos de configuración cumplan con los requisitos de calidad establecidos
- Debe participar en las auditorías de configuración para verificar que todos los elementos cumplen con las especificaciones y están correctamente versionados y documentados.
- Los responsables serán Jorge López Gómez y Marco Muñoz García.

3.5. Verificador de Configuración

- Realizará verificaciones de configuración durante las auditorías y en los puntos de control establecidos en el plan SCM.
- Debe asegurar que todos los CI estén en las versiones correctas y que el sistema no tenga discrepancias.
- Debe colaborar en la creación de informes de auditoría y en la detección de inconsistencias.
- El responsable será Ismael López Marín

3.6. Usuario Final

- Debe proporcionar retroalimentación al equipo sobre el rendimiento y funcionalidad del sistema.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

6 - 14

- Debe colaborar con el CCB para identificar necesidades o problemas que requieran ajustes en la configuración.
- Este rol lo desempeñará D. Ricardo Pérez del Castillo.

4. Herramientas y Entorno de SCM

4.1. Soporte a las 3C

- **Comunicación:** Se empleará Office 365, los anuncios de GitHub y GitHub Project para facilitar la comunicación constante entre los miembros del equipo, realizando reuniones virtuales, compartiendo archivos y coordinando tareas. Se programarán reuniones semanales (tras finalizar cada sprint).
- **Control:** Emplearemos GitHub para el control de versiones. Nos permitirá estructurar las ramas, hacer Pull Requests y establecer etiquetas y versiones.
- **Colaboración:** Emplearemos Office 365 para la documentación colaborativa y GitHub para la colaboración en código y gestión de tareas.

4.2. Java y Spring

Emplearemos SpringBoot para facilitar la creación de la aplicación java y un entorno de desarrollo rápido.

Spring nos permite la inyección de dependencias, gestionando así las instancias de clases y mejorando el desacoplamiento.

4.3. Metodología de Desarrollo Iterativa e Incremental

4.3.1. Planificación de Iteraciones

El desarrollo se dividirá en iteraciones cortas (sprints de 1 semana), en las que se planificarán los objetivos y se entregarán incrementos de software funcionales al final de cada iteración.

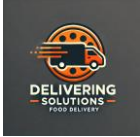
Al final de cada iteración, se documentará el sprint y se ajustará el plan de la siguiente iteración.

4.3.2. Versionado y Gestión de Configuración

Emplearemos el versionado semántico **<major.minor.patch>** para etiquetar las versiones.

Los cambios se deben registrar y aprobar por el CCB antes de ser integrados a la rama master.

Se empleará GitHub para asegurar el control de versiones y facilitar la colaboración continua.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

7 - 14

4.3.3. Monitorización del Avance

Para la gestión de tareas, se utilizará GitHub Projects, estableciendo las tareas bajo las siguientes fases:

- Sprint Backlog
- Sprint Ready
- Sprint In Progress
- Sprint In Review
- Spring Done

4.3.4. Herramientas SCM y Construcción de Software

Se empleará GitHub para el SCM, permitiendo crear ramas para cada tarea y revisar los cambios mediante pull requests.

Para la construcción de software, se empleará Maven, ya que nos permite automatizar la compilación, gestión de dependencias y testing.

Para la gestión de dependencias se utilizará el archivo pom.xml de Maven.

5. Procedimientos de Cambio y Flujo de Trabajo

5.1. Procedimientos de Cambio

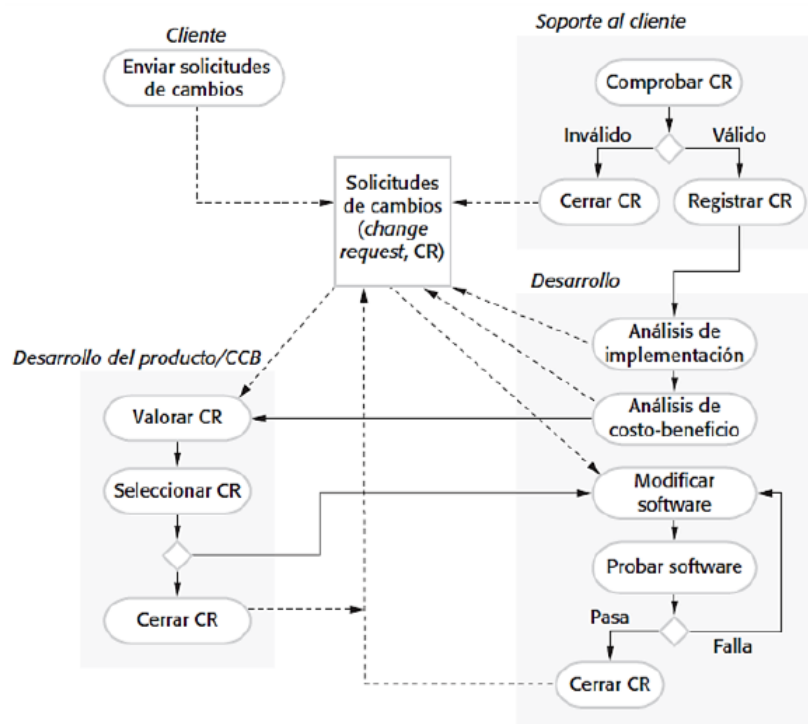
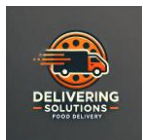


Figura 31. Proceso de gestión de cambios.

- Pérez-Castillo, R. (2024). Ingeniería del Software en Acción. De la velocidad, a la excelencia



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

8 - 14

1. Solicitud de cambio:

- Identificación de la necesidad de cambio.
- Se documenta el cambio solicitado, describiendo la razón y el impacto esperado.

2. Evaluación del cambio:

- Se realiza un análisis de impacto para entender cómo afectará el cambio a los recursos, cronogramas, costos y calidad.
- Se deben evaluar los riesgos en esta fase.

3. Aprobación del cambio:

- Se revisa la solicitud de cambio y, según el impacto y la naturaleza del cambio, se aprueba o rechaza por el CCB.

4. Implementación del cambio:

- Una vez aprobado, se procede a implementar el cambio siguiendo el plan y las directrices establecidas.
- Los responsables del cambio coordinan su implementación para asegurar que sea efectiva y cause la mínima interrupción.

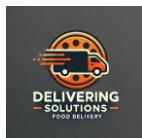
6. Revisión y cierre:

- Después de implementar el cambio, se realiza una revisión para verificar que el cambio haya producido los resultados esperados.
- La solicitud de cambio se cierra y se documenta el proceso y los resultados.

5.2. Flujo de Trabajo

Aquí se definen los pasos a seguir para completar una tarea.

1. Entrada de tareas: Define los recursos e información necesaria para comenzar el proceso.
2. Asignación de tareas: Determina quién es el responsable de cada tarea.
3. Ejecución de tareas: Cada tarea se lleva a cabo de acuerdo con el flujo de trabajo
4. Control de calidad: Se revisa la tarea para asegurar que cumple con los estándares de calidad.
5. Salida de tareas: Cuando todas las tareas han sido completadas y se cumple con el objetivo final, se genera el producto.
6. Documentación: Se recaba información sobre el proceso para el siguiente desarrollo y futuras mejoras.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

9 - 14

6. Baseline

Estableceremos la línea base en la versión 1.0.0.

Esta versión contendrá todas las funcionalidades solicitadas por el cliente de manera básica, y servirá de punto de referencia para los siguientes cambios.

Esta línea base nos permite medir el progreso y evaluar la desviación del plan inicial.

7. Calendario

Actividad	Depende de:	Comienzo
Planificar el Plan de Configuración	Nada	Semana 1
Definir Baseline	Plan de Configuración	Semana 2
Sprint 1: Esqueleto	Plan de Configuración Baseline	Semana 3
Sprint 2: Funcionalidades de Entidades	Esqueleto	Semana 5
Sprint 3: Funcionalidades operativas	Esqueleto Funcionalidades	Semana 6
Sprint 4: Finalizar 1.0.0	Esqueleto Funcionalidades	Semana 7
Implementación control de cambios	Versión 1.0.0	Semana 8
Realizar Informe Final	Versión Final	Semana 13

8. Puntos de Control

Definiremos puntos de control, estableciendo donde se realiza una revisión/verificación antes de avanzar a la siguiente etapa.

De esta manera aseguramos que los cambios se implementen correctamente, manteniendo la calidad y alineación con los objetivos del proyecto.

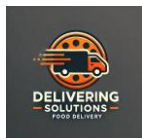
Estos serán:

1. Revisión de Objetivos y Requisitos de la Iteración

Se revisan y priorizan los requisitos y objetivos a abordar, así como identificar posibles riesgos.

2. Revisión de Diseño para la Iteración

Se proponen soluciones que se adapten correctamente al resto del proyecto, y se opta por una.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

10 - 14

3. Revisiones de Código (Code Reviews)

Durante el desarrollo de cada incremento, se realizarán revisiones y correcciones continuas.

4. Revisión de Avance en la Iteración

Al final de cada iteración, se lleva a cabo una revisión formal en la que el equipo demuestra el incremento desarrollado.

5. Pruebas Unitarias, de Integración y Funcionales

Se realizarán pruebas unitarias para asegurar que cada componente individual funciona de manera correcta

6. Demostración y Retroalimentación de la Iteración (Sprint Review)

Al final de cada iteración, se lleva a cabo una revisión formal en la que el equipo demuestra el incremento desarrollado.

7. Revisión de Procesos y Mejora Continua (Sprint Retrospective)

El equipo analiza la iteración y discute las posibles mejoras.

8. Validación de Preparación para el Despliegue

Antes de lanzar el incremento a producción, se valida que esté listo para su despliegue, aprobando todas las pruebas y documentando los cambios.

9. Monitoreo Post-Despliegue y Retroalimentación del Incremento

Después de desplegar el incremento en producción, se monitorea su rendimiento y se recopila retroalimentación del cliente.

9. Gestión de Riesgos

9.1. Identificación de riesgos

La identificación de riesgos se debe hacer de manera continua, ya que en cada iteración, los riesgos pueden cambiar.

Debemos verificar los siguientes riesgos:

- **Riesgos técnicos:** Tecnologías inestables, integraciones difíciles o falta de experiencia con herramientas específicas.
- **Riesgos de requisitos:** Cambios en los requisitos, requisitos incompletos o ambiguos.
- **Riesgos de cronograma:** Retrasos en el desarrollo, dependencias entre tareas o subestimación del tiempo necesario.
- **Riesgos de calidad:** Baja cobertura de pruebas, problemas de rendimiento o errores críticos en producción.
- **Riesgos de recursos:** Disponibilidad limitada de personal, rotación de equipo o falta de habilidades específicas.
- **Riesgos de cumplimiento:** Cumplimiento de regulaciones o estándares específicos, como protección de datos.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

11 - 14

9.2. Evaluación de Riesgos

Evaluaremos los riesgos en función de la probabilidad de que se den, y el impacto que tengan sobre el proyecto.

Probabilidad\ Impacto	Bajo	Medio	Alto
Baja	Riesgo Menor	Riesgo Menor	Riesgo Moderado
Media	Riesgo Menor	Riesgo Moderado	Riesgo Crítico
Alta	Riesgo moderado	Riesgo Crítico	Riesgo grave

Una vez evaluado, se establecen prioridades y se definen las estrategias para mitigar el error (Evitar, Mitigar, Transferir, Aceptar)

10. Procedimientos de Auditoría y Reportes

La auditoría es una revisión sistemática y documentada de los procesos y productos del proyecto, destinada a evaluar la calidad, el cumplimiento de los requisitos, y la gestión de riesgos y cambios. En el contexto de este proyecto, se realizarán auditorías después de cada sprint, en presencia el **Change Control Board (CCB)**.

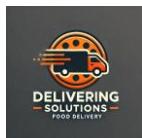
10.1. Objetivos de la Auditoría:

- Asegurar el cumplimiento de los requisitos establecidos.
- Identificar problemas en el flujo de trabajo y áreas de mejora.
- Revisar la efectividad de la gestión de riesgos y cambios.
- Confirmar la calidad del producto y el cumplimiento de estándares.

10.2. Alcance de la Auditoría:

La auditoría abarcará los siguientes aspectos:

- **Calidad del Producto:** Evaluación de la calidad del software en términos de funcionalidad, rendimiento y usabilidad.
- **Cumplimiento de Requisitos:** Verificación de que todos los requisitos se están cumpliendo de acuerdo con las especificaciones acordadas.
- **Gestión de Cambios:** Revisión de los procesos de control de cambios para asegurar que solo los cambios aprobados se integran al proyecto.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

12 - 14

10.3. Procedimiento de la Auditoría:

1. Recolección y Revisión de Documentación:

- Recopilar y revisar toda la documentación relevante del sprint, incluyendo especificaciones de requisitos, registros de cambios, informes de pruebas y documentación de diseño.

2. Revisión de Procesos y Procedimientos:

- Verificar que los procesos y procedimientos documentados se están siguiendo y evaluar su efectividad. Esto incluye el cumplimiento de las metodologías de desarrollo y la correcta aplicación de los puntos de control.

3. Evaluación de la Calidad del Producto:

- Realizar una revisión de calidad de los productos intermedios y finales del software, como el código fuente, la arquitectura y los casos de prueba, para asegurar que cumplen con los estándares de calidad y los requisitos del proyecto.

4. Detección de No Conformidades:

- Identificar cualquier área donde el proyecto no cumple con los estándares o requisitos establecidos. Cada no conformidad debe documentarse y clasificarse según su impacto potencial en el proyecto.

5. Recomendaciones de Mejora:

- Sugerir mejoras para resolver las no conformidades y optimizar los procesos, teniendo en cuenta las lecciones aprendidas de iteraciones anteriores.

10.4. Documentación de la Auditoría

Cada auditoría generará un informe que incluirá:

- **Resumen de Hallazgos:** Descripción de los resultados, incluyendo áreas de conformidad y no conformidad.
- **Recomendaciones:** Lista de acciones sugeridas para mejorar el proceso y asegurar el cumplimiento en futuras iteraciones.
- **Plan de Acción Correctiva:** Propuesta de un plan para abordar las no conformidades detectadas, especificando responsables y plazos.

10.5. Seguimiento de Auditoría

- Las acciones correctivas se revisarán en las auditorías posteriores para verificar su implementación y efectividad.



DOCUMENTO PLAN CONFIGURACION

CODIGO

DS-GDC001

VERSION

0

PÁGINA

13 - 14

- El CCB podrá ajustar las estrategias de mitigación y control según los resultados de cada auditoría, manteniendo un ciclo de mejora continua.