Política de Desarrollo Seguro

Esquema Nacional de Seguridad [RD 311/2022]
ISO 27001:2022 SGSI Seguridad de la Información

Departamento IT & PMO

Control de Versiones

Versión	Responsable	Fecha	Cambio
1.0	Responsable PMO	02/12/2020	Creación del documento
2.0	Responsable SGSI	27/08/2021	Actualización del documento
3.0	Responsable SGSI	30/09/2024	Actualización del documento
4.0	Responsable SGSI	23/04/2026	Actualización del documento

Revisado	Aprobado
Responsable SGI	**Responsable Seguridad**

ÍNDICE

1. Objetivo y compromiso de la Dirección

La Dirección de EXCELTIC, S.L., en base a su Política de Calidad y Seguridad fundamentada en la coherencia con sus Valores, se compromete a un crecimiento y mejora continua en sus Servicios y Procesos de Desarrollo Tecnológico. El presente documento materializa ese compromiso en el ámbito del desarrollo seguro de software

Durante el ciclo de desarrollo de aplicaciones (SDLC) es obligatorio considerar los requerimientos de seguridad y control para garantizar la seguridad de la información tratada por nuestras herramientas. A tal fin, se establecen las directrices a seguir en cada fase del ciclo de desarrollo; dichas directrices deberán reflejarse en los documentos y artefactos correspondientes a cada fase. Es obligado el cumplimiento de esta política para todos los sistemas y aplicaciones desarrollados por EXCELTIC.

Como marco de referencia para la mejora y evaluación del proceso de software, EXCELTIC adopta las normas ISO/IEC 12207:2017 (procesos del ciclo de vida del software) e ISO/IEC 33000 (evaluación de procesos), complementadas con los requisitos de seguridad establecidos por el Esquema Nacional de Seguridad (RD 311/2022) nivel MEDIO y por la norma ISO/IEC 27001:2022.

Principios generales que rigen el desarrollo tecnológico en EXCELTIC:

- Generar productos seguros y de calidad de forma predecible
- Conseguir la satisfacción de los clientes cumpliendo los plazos, la seguridad y la calidad exigida, colaborando estrechamente con él tanto en la definición como en el desarrollo del proyecto, a lo largo de todo el ciclo de vida.
- Todo el ciclo de desarrollo debe ajustarse a la Política de Seguridad de la Información de EXCELTIC.

2. Alcance

Esta política aplica, sin excepciones, a:

- Todo el software desarrollado, mantenido o customizado por EXCELTIC, tanto para clientes externos como para uso interno.
- Todo el personal interno y personal externo (autónomos, subcontratas y proveedores) que participe en alguna fase del SDLC
- Todos los entornos implicados en el ciclo de vida del software: desarrollo, pruebas, preproducción y producción, ya estén alojados en infraestructura de EXCELTIC o de cliente.

- Todas las herramientas de soporte al desarrollo: repositorios de código (GitHub, ScriptCase), gestión de tareas (JIRA), calidad de código (SonarQube, IonQube) y pruebas de seguridad (OWASP ZAP)

3. Marco normativo de referencia

La presente política da cumplimiento a los siguientes marcos normativos:

- **Real Decreto 311/2022**, de 3 de mayo, por el que se regula el Esquema Nacional de Seguridad (nivel MEDIO), en particular las medidas mp.sw.1 (Desarrollo de aplicaciones) y mp.sw.2 (Aceptación y puesta en servicio).
- **ISO/IEC 27001:2022** – Sistema de Gestión de la Seguridad de la Información, especialmente los controles del Anexo A: A.5.8, A.8.25, A.8.26, A.8.27, A.8.28, A.8.29, A.8.30, A.8.31, A.8.32 y A.8.33.
- **Reglamento (UE) 2016/679 (RGPD)** y Ley Orgánica 3/2018 de Protección de Datos Personales y garantía de los derechos digitales (LOPDGDD)
- **ISO/IEC 12207:2017** – Procesos del ciclo de vida del software.
- **ISO/IEC 33000** – Evaluación de procesos de software.
- **OWASP ASVS y OWASP Top-10** como guía de referencia técnica para requisitos de seguridad y codificación segura.

4. Responsabilidades

- **Responsable Seguridad:** responsable de asegurar que el cumplimiento y aplicación de las mejores prácticas, así como la mejora continua de las mismas, se gestiona adecuadamente en toda la organización. Aprueba formalmente esta política y dota de los recursos necesarios para su aplicación
- **Responsable Técnico:** responsable de garantizar que las personas que trabajan bajo su control aplican la política y los estándares establecidos dentro de EXCELTIC, y de que se impliquen y comprometan con su evolución y mejora. Supervisa el cumplimiento de las directrices del SDLC en cada proyecto.
- **Responsable SGI:** asesoran al equipo directivo y técnico, proporcionan apoyo metodológico especializado y garantizan que los informes sobre la situación de los proyectos tecnológicos están disponibles
- **Equipo Técnico:** cada miembro tiene la responsabilidad de conocer y mantener el cumplimiento de las políticas y estándares establecidos en el desarrollo tecnológico, garantizando la calidad y seguridad de sus productos de trabajo. Debe además conocer e involucrarse en la consecución de los objetivos del proyecto.
- **Proveedores y subcontratas:** deben aceptar contractualmente el cumplimiento de esta política cuando participen en cualquier fase del SDLC de EXCELTIC.

5. Análisis Funcional

En la fase de análisis funcional los requisitos de seguridad deben tratarse como requisitos funcionales de primer nivel, con idéntica prioridad y trazabilidad que los requisitos de negocio. Estos puntos deberán reflejarse en el artefacto de análisis funcional generado en la fase.
Actividades mínimas obligatorias:

- Identificación y clasificación de los activos de información tratados por la aplicación, conforme a los niveles de seguridad del ENS (bajo, medio, alto) y al esquema de clasificación corporativo.
- Identificación de los actores, roles y permisos necesarios, así como de los flujos de información críticos.
- Definición de los requisitos de confidencialidad, integridad, disponibilidad, autenticidad y trazabilidad (dimensiones ENS) aplicables a la solución
- Identificación de los requisitos legales y regulatorios aplicables (RGPD, LOPDGDD, sectoriales del cliente).
- Validación formal del análisis funcional por los Responsables de Tecnología antes del inicio del diseño técnico.

6. Gestión de la autenticación

Se deben establecer los mecanismos de autenticación de los usuarios en cada herramienta, identificando además si existen autenticaciones delegadas a sistemas de terceros (federación, SSO).
Requisitos específicos EXCELTIC:

- Cada desarrollador dispone de una máquina virtual instalada en su equipo portátil, a la que accede para realizar el desarrollo A estas máquinas se les realizan copias de seguridad diarias que se almacenan en el servidor corporativo de EXCELTIC, S L
- Se utiliza autenticación multifactor (MFA) para el acceso a los entornos de desarrollo, repositorios de código y herramientas de gestión de proyectos.
- Las contraseñas cumplen la Política de Contraseñas Corporativa (longitud mínima, complejidad, caducidad y no reutilización).
- Las credenciales de servicio y secretos (API keys, cadenas de conexión, certificados) nunca se almacenan en código fuente; se gestionan en un almacén de secretos controlado.
- En aplicaciones desarrolladas para clientes, se integrarán los mecanismos de autenticación solicitados (SSO, OAuth2, SAML, OpenID Connect) conforme a las mejores prácticas.

7. Tratamiento de la información (RGPD / LOPDGDD)

En el análisis de requerimientos de las soluciones software se deben tener en cuenta los requerimientos de tratamiento de información que requiera la solución. Se debe garantizar el correcto tratamiento de la información, así como la normativa vigente. Se trasladará y asesorará al cliente sobre dicha normativa y cómo afecta al proceso de negocio analizado

Requisitos específicos:

- Aplicación de los principios de privacidad desde el diseño y por defecto (privacy by design & by default) en todas las soluciones que traten datos personales.
- Minimización de datos: solo se recogerán y tratarán los datos estrictamente necesarios para la finalidad declarada.
- Toda solución que trate datos personales deberá estar inventariada en el Registro de Actividades de Tratamiento corporativo o del cliente
- El Responsable de Protección de Datos será consultado en fases tempranas del proyecto cuando aplique.

8. Gestión de riesgos de seguridad

Se deberán identificar los riesgos asociados con la seguridad de la información, tanto aquellos derivados del proceso de negocio definido como aquellos que puedan ser concretos del contexto en el que se utilizará la solución. Por este motivo, todas las herramientas de desarrollo se incluyen en el análisis de riesgos corporativo.

Requisitos específicos:

- El análisis de riesgos sigue la metodología adoptada por el SGSI de EXCELTIC.
- Se revisa con periodicidad mínima anual y siempre que se produzcan cambios significativos en una aplicación, en su arquitectura o en el contexto de amenazas
- Para cada riesgo identificado se definen tratamientos (mitigar, transferir, aceptar, evitar) y controles, que se integran en el backlog del proyecto como requisitos de seguridad.
- Los riesgos residuales con nivel superior al umbral aceptable por la Dirección deben ser aprobados formalmente antes del paso a producción.

9. Diseño Técnico

A continuación, se detallan las directrices a tener en cuenta en esta fase del desarrollo, que deberán reflejarse en el artefacto de diseño técnico generado

Principios de seguridad por diseño obligatorios:

- **Modelado de amenazas (threat modeling):** identificación sistemática de amenazas, vectores de ataque y superficies expuestas (STRIDE o equivalente) antes de iniciar la codificación.
- **Mínimo privilegio:** cada componente, proceso o usuario dispondrá únicamente de los permisos estrictamente necesarios para su función
- **Defensa en profundidad:** se implementarán múltiples capas de control, de forma que el fallo de una no comprometa al conjunto.
- **Fail secure:** ante cualquier error o condición anómala, la aplicación deberá degradar hacia un estado seguro, nunca más permisivo.
- **Validación en servidor:** toda validación con implicaciones de seguridad se realiza en el lado servidor; las validaciones de cliente son complementarias
- **Seguridad por defecto:** las configuraciones por defecto son las más seguras; activar funciones arriesgadas requiere acción explícita
- El artefacto de diseño técnico incluirá obligatoriamente un apartado de seguridad que cubra autenticación, autorización, gestión de sesiones, cifrado, logs y gestión de errores.

10.Sistema de logs y trazabilidad

Las aplicaciones deben disponer de un sistema de logs apropiado que permita la identificación de errores y la trazabilidad de las operaciones. En aplicaciones que requieran logs de auditoría, éstos deberán almacenarse en un medio que no permita su modificación, garantizando la integridad y la completitud de la información para disponer de trazabilidad plena.

Requisitos mínimos:

- Se registrarán, como mínimo: inicio y fin de sesión, intentos fallidos de autenticación, cambios de privilegios, acceso y modificación de datos sensibles o de configuración, errores del sistema y operaciones de administración.
- Cada evento de log incluirá, al menos: marca temporal identificador de usuario, identificador de sesión, acción realizada, resultado y origen (IP, host).
- Los logs se protegerán frente a modificación y borrado no autorizado, y se almacenarán en una ubicación segregada de la aplicación
- El periodo de retención mínimo de los logs relevantes para seguridad será de 24 meses, salvo que normativa sectorial o contractual exija un periodo superior.
- Los logs de seguridad se revisarán periódicamente y se integrarán, cuando aplique, con la herramienta de monitorización corporativa para detección temprana de incidentes.

11.Gestión de sesiones

La gestión de sesiones es importante para que los usuarios autenticados tengan una asociación robusta y criptográficamente segura con sus datos. Esto permite aplicar los controles de autorización y prevenir ataques web típicos como la reutilización, falsificación e intercepción de sesiones Para ello, en nuestras aplicaciones utilizaremos frameworks que gestionen la sesión de forma idónea En aquellas aplicaciones que implementen un sistema de gestión propio, se pondrá especial foco en las vulnerabilidades derivadas de dicha implementación.

Requisitos mínimos:

- Los identificadores de sesión se generarán con fuentes criptográficamente seguras y tendrán longitud suficiente para resistir ataques de fuerza bruta.
- La sesión se regenerará tras el inicio de sesión y tras cualquier cambio de nivel de privilegio
- Se definirá un tiempo máximo de inactividad y una duración máxima absoluta de la sesión.
- El cierre de sesión invalidará la sesión tanto en cliente como en servidor.

12.Separación de funciones

El diseño de aplicaciones debe seguir las buenas prácticas. Será especialmente importante el "Principio de responsabilidad única", por el cual una clase debe tener responsabilidad sobre una funcionalidad concreta en la aplicación.

En este sentido, EXCELTIC, S L desarrolla en base a los 5 principios SOLID de diseño de aplicaciones de software:

- **S Single Responsibility Principle (SRP):** Principio de Responsabilidad Única
- **O – Open/Closed Principle (OCP):** Principio Abierto/Cerrado.
- **L – Liskov Substitution Principle (LSP):** Principio de Sustitución de Liskov.
- **I – Interface Segregation Principle (ISP):** Principio de Segregación de Interfaces.
- **D – Dependency Inversion Principle (DIP):** Principio de Inversión de Dependencias.

Entre los objetivos de aplicar estos 5 principios a la hora de escribir código se encuentran:

- Crear un software eficaz: que cumpla con su cometido y que sea robusto y estable.
- Escribir un código limpio y flexible ante los cambios: que se pueda modificar fácilmente según necesidad, que sea reutilizable y mantenible.
- Permitir escalabilidad: que acepte ser ampliado con nuevas funcionalidades de manera ágil.

13.Manejo de errores

El manejo de errores se define en dos niveles: manejo estructurado de excepciones y control de errores funcional. El manejo estructurado de excepciones es siempre preferido, ya que es más fácil cubrir el 100 % del código. En general, las aplicaciones no deben devolver errores no controlados, dado que pueden utilizarse como mecanismo de penetración
Requisitos mínimos:

- Los mensajes de error devueltos al usuario serán genéricos y no expondrán trazas, información técnica, estructuras internas ni datos sensibles.
- La información técnica detallada del error se registrará exclusivamente en los logs del servidor.
- Se definirá un comportamiento por defecto (fail secure) para cualquier excepción no prevista

14.Implementación / Codificación

A continuación, se detallan las directrices a tener en cuenta en esta fase del desarrollo, que deberán reflejarse en todo el código desarrollado durante la implementación de la solución. Todo desarrollador debe tener un conocimiento mínimo sobre seguridad en el desarrollo de soluciones
Requisitos específicos:

- El Equipo Técnico recibirá formación en desarrollo seguro.
- Se utilizarán guías de referencia reconocidas por la industria.
- Queda prohibido introducir en el código credenciales, claves, tokens o cualquier dato sensible en texto claro
- Todo desarrollo debe reutilizar, en la medida de lo posible, componentes y librerías validadas corporativamente frente a la reimplementación de funciones críticas de seguridad.

15.Buenas prácticas de codificación

Se definen a continuación las buenas prácticas de desarrollo seguro a considerar, siguiendo la referencia estándar marcada por OWASP Se aplican las siguientes actividades:

- **Validación de entrada:** toda entrada de usuario, servicio externo o fichero se considera no fiable y se valida en servidor mediante listas blancas (tipo, longitud, rango, formato).
- **Codificación de salida:** la información devuelta al usuario se codifica en el contexto adecuado (HTML, JavaScript, URL, SQL) para prevenir XSS e inyecciones.

- **Control de accesos:** autorización basada en roles validada en servidor en cada operación Prohibido basar la seguridad únicamente en la ocultación de opciones en el interfaz
- **Protección de datos:** cifrado en tránsito y en reposo de los datos sensibles, conforme a la clasificación del activo.
- **Seguridad en comunicaciones:** TLS obligatorio para cualquier tráfico que transporte información sensible o autenticada.
- **Configuración del sistema:** deshabilitar servicios, puertos, cuentas y funciones no necesarias; aplicar el principio de mínima exposición
- **Seguridad de bases de datos:** uso obligatorio de consultas parametrizadas o ORM, cuentas de aplicación con permisos mínimos, cifrado de campos sensibles.
- **Gestión de memoria:** en lenguajes no gestionados, controlar tamaños de buffer, liberar recursos y evitar desbordamientos.
- **Gestión de dependencias de terceros:** inventario de librerías y sus versiones; análisis de vulnerabilidades (SCA) automatizado en el pipeline

16. Prácticas criptográficas

- Todas las funciones de criptografía de la aplicación deben ser implementadas en sistemas confiables (por ejemplo, el servidor).
- Proteger los secretos maestros (master secrets) del acceso no autorizado
- Los módulos de criptografía deberán, en caso de fallo, fallar de forma segura
- Todos los números aleatorios, GUIDs y frases aleatorias deberán generarse utilizando módulos aprobados para su generación (RNG criptográficamente seguros).
- Queda prohibida la implementación propia de algoritmos criptográficos; se utilizarán exclusivamente librerías estándar ampliamente auditadas.

16.1. Seguridad en las comunicaciones

- Implementar cifrado para todas las transmisiones de información sensible. Debe incluir TLS para proteger la conexión, combinable con cifrado discreto de ficheros sensibles en conexiones no basadas en HTTP.
- Los certificados TLS deben ser válidos y contener el nombre de dominio correcto, no estar expirados y estar instalados con los certificados intermedios cuando se requieran.
- Las conexiones TLS que fallen no deben transformarse en una conexión insegura
- Utilizar conexiones TLS para todo contenido que requiera acceso autenticado y para cualquier otro tipo de información sensible.
- Utilizar una única implementación estándar de TLS, configurada correctamente.
- Especificar los caracteres de codificación para todas las conexiones.

- Filtrar los parámetros que contengan información sensible de los referer HTTP cuando existan vínculos a sitios externos

16.2. Gestión del ciclo de vida de las claves

Alineado con ISO/IEC 27001 y ENS, se establecen medidas para la generación, distribución, almacenamiento, uso, rotación, archivado y destrucción segura de las claves criptográficas utilizadas. Las claves de producción residen exclusivamente en almacenes de secretos autorizados y nunca en repositorios de código

17.Calidad de código

Los desarrolladores deberán utilizar herramientas de calidad de código local como SonarQube, IonQube o la herramienta de SONAR configurada en GIT para .NET, con las reglas establecidas por la empresa para seguir la normativa de calidad de código Una vez subido el código al repositorio se lanzará la herramienta Sonar, que identificará el incumplimiento de las reglas de calidad donde proceda. Los desarrolladores deberán solucionar estas incidencias detectadas.
La herramienta IonQube sirve además para anonimizar código en la versión de ScriptCase, aportando una capa adicional de seguridad.
Las reglas SonarQube incluyen tanto reglas de calidad como reglas de seguridad (security hotspots y vulnerabilities)

18.Repositorios seguros

Se utilizará el repositorio de código corporativo para garantizar la seguridad e integridad de todo el código desarrollado (GitHub para .NET, ScriptCase para PHP). La gestión de usuarios en los repositorios tendrá caducidad para garantizar la revisión periódica de dichos permisos

Requisitos mínimos:

- Revisión periódica de los permisos sobre los repositorios.
- Protección de las ramas principales (main/master) mediante reglas que exijan revisión por pares antes del merge.
- Prohibición explícita de almacenar secretos en el repositorio; análisis automático de secret scanning activado

19.Versionado de aplicaciones

- El versionado del desarrollo .NET se gestiona en la herramienta GitHub.
- El versionado del desarrollo PHP se gestiona en la herramienta ScriptCase

- Cada entrega a preproducción y producción queda asociada a una etiqueta (tag) o release identificable que permita su trazabilidad y su reversión si fuera necesario

20. Validación y Pruebas

A continuación se detallan las directrices a tener en cuenta en esta fase del desarrollo, que deberán reflejarse en todo el código desarrollado durante la implementación de la solución Además, se generarán artefactos de definición y ejecución de pruebas que garanticen la trazabilidad del proceso de pruebas.

21. Revisión de código

Los desarrolladores no tendrán permisos para subir código directamente a producción El desarrollo se realiza en los entornos locales de cada uno de los desarrolladores; los responsables técnicos de PHP y de NET serán los encargados de validar los desarrollos realizados. En caso de encontrar algún hallazgo se procederá a un segundo control local. Posteriormente, los responsables técnicos serán los encargados de pasar de preproducción a producción tras validar la demo del desarrollo con el cliente. El entorno de producción puede ser del cliente o de EXCELTIC, S.L.

Requisitos mínimos adicionales:

- Toda modificación de código requiere revisión por pares (principio de “cuatro ojos”) antes de fusionarse en la rama principal.
- Se ejecutará análisis estático de código con SonarQube como paso bloqueante del pipeline de integración continua.
- Se realizará análisis de composición de software sobre las dependencias de terceros para detectar vulnerabilidades conocidas
- Criterios de rechazo: presencia de vulnerabilidades de severidad alta/crítica, incumplimiento del Quality Gate de Sonar, ausencia de pruebas asociadas a la funcionalidad.

22. Pruebas

Las pruebas o revisiones a realizar se marcan en la herramienta respectiva (Jira, Redmine, etc) como tareas a realizar por cada desarrollador. Dentro de estas pruebas, todos los desarrollos deben pasar por un análisis de vulnerabilidades. Para todas las pruebas se utilizan datos anonimizados.
Tipología de pruebas obligatorias:

- **Pruebas unitarias:** cobertura objetivo definida por el proyecto; bloqueantes si no alcanzan el umbral mínimo corporativo
- **Pruebas de integración:** validan el comportamiento entre componentes, incluidos los de seguridad (autenticación, autorización, cifrado, logs).
- **Pruebas de aceptación (UAT):** validadas con el cliente, cubriendo los criterios funcionales y no funcionales acordados.
- **Pruebas de seguridad DAST:** análisis dinámico automatizado sobre la aplicación desplegada en preproducción
- **Pruebas de seguridad SAST:** análisis estático con SonarQube sobre cada commit al repositorio.
- **Pruebas de intrusión (pentesting):** con periodicidad al menos anual para aplicaciones críticas y tras cambios mayores en la arquitectura o superficie expuesta.

22.1. Aceptación y puesta en servicio

Antes de autorizar el paso a producción de una aplicación o de una nueva versión, el Responsable Técnico verificará el cumplimiento de los siguientes criterios de aceptación, que se documentarán en el artefacto de aceptación del proyecto:

- Análisis de seguridad superados: Quality Gate de Sonar en umbrales aprobados, informe DAST y análisis SCA sin hallazgos de severidad alta o crítica pendientes de tratar
- Pruebas funcionales y de aceptación firmadas por el cliente o responsable funcional
- Documentación de operación entregada (instalación, configuración, operación, contingencia).
- Aprobación formal del paso a producción por parte del Responsable Técnico.

23.Separación de entornos

En cumplimiento de ISO/IEC 27001 y las buenas prácticas del ENS, EXCELTIC mantiene los entornos de desarrollo, pruebas/preproducción y producción. Este mantenimiento puede ser lógico o físico, pero en todo caso se garantizarán las siguientes condiciones:

- Los credenciales, claves y configuraciones son específicos de cada entorno y no se comparten entre ellos.
- El personal de desarrollo no dispone por defecto de permisos sobre el entorno de producción
- El paso de código entre entornos se realiza exclusivamente mediante el procedimiento de gestión de cambios.
- No se utilizan datos reales de producción en entornos de desarrollo o pruebas salvo que estén debidamente anonimizados.

- Cada entorno dispone de su propia configuración de registro de actividad y monitorización

24.Gestión de cambios

En cumplimiento de ISO/IEC 27001, todo cambio sobre una aplicación en producción se gestiona de forma controlada mediante el siguiente procedimiento:

- Registro del cambio en la herramienta corporativa con descripción, impacto esperado, componentes afectados y responsable.
- Evaluación del impacto en seguridad por parte del Responsable Técnico para cambios significativos.
- Validación técnica y funcional en preproducción antes del paso a producción.
- Aprobación formal del cambio por los responsables técnicos y para cambios relevantes en seguridad, Responsable de Seguridad
- Plan de reversión (rollback) definido y probado antes de la implantación
- Comunicación del cambio a las partes afectadas y registro posterior de su ejecución y resultado.
- Para cambios de emergencia se aplicará un procedimiento abreviado, con aprobación posterior obligatoria y revisión específica por el SGSI.

25.Datos para pruebas

En cumplimiento de ISO/IEC 27001, el tratamiento de los datos utilizados para pruebas sigue las siguientes reglas:

- Queda prohibido el uso de datos reales de producción en entornos de desarrollo y pruebas
- Cuando por necesidades técnicas se requiera utilizar datos con estructura de producción, éstos serán previamente anonimizados o pseudonimizados mediante técnicas aprobadas corporativamente (eliminación, enmascaramiento, sustitución, generalización).
- Los conjuntos de datos de prueba se protegen con el mismo nivel de control de acceso que los datos de producción del mismo nivel de clasificación.
- Los datos de prueba generados para un proyecto se eliminan de forma segura al finalizar su utilidad
- Cualquier excepción al uso de datos reales requiere aprobación formal del Responsable de Seguridad y, cuando incluya datos personales, consulta al Responsable de Protección de Datos.

26.Desarrollo externalizado

En cumplimiento de ISO/IEC 27001 cuando EXCELTIC subcontrate total o parcialmente el desarrollo de software a terceros, se aplicarán los siguientes controles:

- El proveedor aceptará contractualmente el cumplimiento de la presente Política de Desarrollo Seguro y del resto de políticas de seguridad aplicables de EXCELTIC
- Se incluirán cláusulas específicas de seguridad: confidencialidad, titularidad del código, entregables de seguridad (análisis estático, pruebas, documentación), derecho de auditoría y régimen de notificación de incidentes.
- Se supervisarán los entregables con los mismos criterios de aceptación que el desarrollo interno.
- Los accesos del proveedor a los sistemas, repositorios y datos de EXCELTIC se otorgarán bajo mínimo privilegio, con caducidad y revisión periódica
- Al finalizar la relación se revocarán todos los accesos y se verificará la devolución o destrucción segura de la información transferida.

27.Implantación y Mantenimiento

A continuación, se detallan las directrices a tener en cuenta en esta fase del desarrollo, que deberán reflejarse en los artefactos de esta fase

- La implantación se ejecuta conforme al procedimiento de gestión de cambios.
- Se mantiene documentación operativa actualizada: manual de instalación, configuración, operación, copia de seguridad y contingencia.
- Se aplica una política de parcheado y actualización del sistema operativo, plataformas y dependencias con periodicidad acorde al nivel de criticidad y a las alertas de seguridad
- Las actuaciones de mantenimiento correctivo y evolutivo quedan registradas y trazadas en herramientas corporativas y en los repositorios

28.Revisión del estado del sistema

Se dispondrá de un sistema de monitorización continuo que garantice el servicio y los recursos necesarios para la solución Además, se prestará especial atención a las alarmas producidas por intentos de ataque de seguridad sobre el propio sistema

Requisitos mínimos:

- Monitorización de disponibilidad, rendimiento y uso de recursos.
- Monitorización de eventos de seguridad: intentos de autenticación fallidos, accesos anómalos, errores repetidos, intentos de explotación de vulnerabilidades conocidas.
- Integración, cuando aplique, con la plataforma SIEM corporativa o del cliente, y con el proceso de gestión de incidentes del SGSI

- Procedimiento de escalado de alertas con tiempos de respuesta definidos según criticidad

29.Revisión y actualización de la política

La presente política se revisará con periodicidad mínima anual, y siempre que se produzca alguno de los siguientes eventos:

- Cambios significativos en la normativa de referencia (ENS, ISO 27001, RGPD, LOPDGDD u otra aplicable).
- Cambios relevantes en el marco tecnológico, herramientas o arquitectura de desarrollo de EXCELTIC.
- Incidentes de seguridad cuya causa raíz evidencie carencias en la política.
- Conclusiones de auditorías internas o externas que identifiquen oportunidades de mejora

La revisión es responsabilidad del Responsable de Seguridad y la aprobación de cualquier modificación corresponde al Responsable SGI. La política permanecerá disponible en la web corporativa y en la intranet de la organización, actualizándose regularmente.