

Anexo 1

ARQUITECTURA DEL SISTEMA

Arquitectura del sistema para el desarrollo de la nueva aplicación de gestión de casos de FUNDAPA

Este documento es propiedad de FUNDAPA, y su contenido es confidencial. Este documento no puede ser reproducido, en su totalidad o parcialmente, ni mostrado a otros, ni utilizado para otros propósitos que los que han originado su entrega, sin el previo permiso escrito de FUNDAPA. FUNDAPA no podrá ser considerada responsable de eventuales errores u omisiones en la edición del documento.

Índice

Índice	2
1. Modelo de arquitectura de software recomendado	3
1.1. Arquitectura general:	3
1.2. Modelo objetivo	3
2. Arquitectura lógica	4
2.1. Capas	4
2.2. Tecnologías propuestas	4
2.2.1. Backend	4
2.2.2. Frontend	4
2.2.3. Base de datos.....	4
2.2.4. Gestión documental	5
2.2.5. Generación documental	5
2.2.6. Seguridad	5
2.2.7. Autenticación y autorización	5
2.2.10. Cifrado y seguridad	6
2.2.11. Interoperabilidad e integraciones.....	6
2.2.12. APIs	6
2.2.13. Integración Microsoft	6
2.2.14. Automatizaciones y workflows	6
2.2.15. Requisito de calidad del software	6
2.2.16. Testing.....	7
2.2.17. Accesibilidad	7
2.2.18. Propiedad y sostenibilidad.....	7
2.2.19. Sistema Operativo.....	7
2.2.20. Infraestructura física servidora	7
2.2.21. Infraestructura de Publicación	7

1. Modelo de arquitectura de software recomendado

Nuestra propuesta tecnológica contempla principalmente el ecosistema Microsoft por su robustez, capacidad de integración, soporte empresarial y alineación con entornos corporativos modernos y actualmente en uso en FUNDAPA. Tecnologías como .NET, Azure y SQL Server ofrecen una base sólida para el desarrollo, despliegue y mantenimiento de soluciones escalables y seguras.

No obstante, la arquitectura planteada mantiene un enfoque flexible y agnóstico, permitiendo incorporar otras tecnologías líderes del mercado cuando el proyecto lo requiera. En este sentido, frameworks y entornos como Java, Spring Boot o React pueden integrarse de forma natural para cubrir necesidades específicas de frontend, microservicios, interoperabilidad o experiencia de usuario avanzada.

Este enfoque híbrido garantiza una solución adaptable, sostenible y preparada para evolucionar según los requerimientos funcionales y estratégicos del proyecto.

1.1. Arquitectura general:

Se considera necesario garantizar:

- interoperabilidad futura,
- reutilización de servicios,
- integración con aplicaciones corporativas,
- y posible evolución hacia aplicaciones móviles u otros canales.

1.2. Modelo objetivo

Se establece como modelo objetivo una arquitectura web multicapa basada en:

- Frontend web SPA (Single Page Application),
- Backend desacoplado mediante API REST,
- Base de datos relacional corporativa,
- Sistema de almacenamiento documental,
- Servicios de autenticación y autorización centralizados

2. Arquitectura lógica

2.1. Capas

La arquitectura propuesta estará compuesta, al menos, por las siguientes capas.

- Capa de presentación: Interfaz web responsive accesible desde navegador.
- Capa de servicios/API: Servicios REST para lógica de negocio e integración.
- Capa de dominio: Gestión de reglas de negocio, workflows y automatizaciones.
- Capa de persistencia: Acceso a datos y almacenamiento documental.
- Capa de integración: Integración con sistemas corporativos y servicios externos.

2.2. Tecnologías propuestas

2.2.1. Backend

La solución podría desarrollarse sobre tecnología Microsoft .NET, utilizando:

- .NET 8 (LTS)
- ASP.NET Core
- Entity Framework Core

2.2.2. Frontend

La interfaz de usuario podría desarrollarse mediante:

- Angular (versión LTS vigente)
- TypeScript

2.2.3. Base de datos

Motor de base de datos:

- Microsoft SQL Server
- PostgreSQL
- MySQL

Requisitos de persistencia: La base de datos deberá:

- garantizar integridad transaccional,
- soportar auditoría completa,
- permitir trazabilidad histórica,
- optimizar búsquedas complejas.

Se deberán implementar:

- índices optimizados
- versionado histórico
- control de concurrencia
- backups automatizados
- políticas de recuperación.

2.2.4. Gestión documental

La solución deberá incorporar capacidades de:

- almacenamiento documental
- anexo de expedientes
- generación documental automática
- exportación a formatos estándares

El sistema deberá permitir:

- almacenamiento seguro de documentos
- versionado documental
- metadatos asociados
- búsquedas
- control de acceso por permisos

2.2.5. Generación documental

La generación automática y semiautomática de documentos deberá soportar, conforme a los requisitos funcionales descritos en el análisis del sistema.

- plantillas parametrizadas
- combinación de datos
- campos dinámicos

2.2.6. Seguridad

La solución deberá cumplir obligatoriamente la siguiente normativa, todo ello conforme a las exigencias funcionales definidas.

- RGPD,
- LOPDGDD,
- ENS,
- normativa de administración electrónica aplicable,
- requisitos de trazabilidad y auditoría.

2.2.7. Autenticación y autorización

La solución deberá integrarse con entornos Microsoft corporativos mediante:

- LDAP (Active Directory de Gobierno de Navarra)

2.2.8. Control de acceso

Se deberá implementar un modelo RBAC avanzado:

- permisos por rol,
- permisos por módulo,
- permisos por ámbito funcional,
- y permisos específicos por expediente o entidad.

2.2.9. Auditoría

La aplicación deberá registrar:

- accesos,
- modificaciones,
- eliminaciones,
- exportaciones,

- consultas,
- y acciones administrativas.

2.2.10. Cifrado y seguridad

La solución deberá garantizar:

- cifrado TLS 1.2 o superior
- cifrado de datos sensibles
- almacenamiento seguro de credenciales
- y protección frente a vulnerabilidades OWASP Top 10

2.2.11. Interoperabilidad e integraciones

La solución deberá diseñarse para permitir integración futura con:

- plataformas corporativas
- sistemas administrativos
- sistemas documentales
- herramientas Microsoft 365
- plataformas de administración electrónica

2.2.12. APIs

Se valorará positivamente:

- APIs REST documentadas,
- especificación OpenAPI/Swagger
- autenticación segura

2.2.13. Integración Microsoft

Se valorará positivamente:

- integración con Microsoft 365
- SharePoint
- Exchange
- Teams,
- Power BI.

2.2.14. Automatizaciones y workflows

La solución deberá soportar:

- tareas automáticas,
- avisos,
- cambios de estado,
- workflows,
- validaciones,
- y automatizaciones de negocio.

2.2.15. Requisito de calidad del software

- código mantenible,
- modularidad,
- cobertura de pruebas,
- documentación técnica,
- estándares de calidad.

2.2.16. Testing

Se deberán contemplar:

- pruebas unitarias,
- pruebas de integración,
- pruebas funcionales,
- pruebas de seguridad,
- pruebas de carga.

2.2.17. Accesibilidad

La interfaz deberá cumplir:

- WCAG 2.1 nivel AA,
- y normativa de accesibilidad vigente en administración pública.

2.2.18. Propiedad y sostenibilidad

La solución deberá garantizar:

- entrega completa del código fuente,
- ausencia de bloqueo tecnológico,
- documentación técnica y funcional,
- posibilidad de mantenimiento por terceros.

2.2.19. Sistema Operativo

Los sistemas operativos sobre los que correrá la aplicación deben ser los estándares para SSPP proporcionados por Nasertic:

- Windows 2022 server datacenter
- RedHat 9.6 Enterprise

2.2.20. Infraestructura física servidora

La aplicación correrá en la infraestructura virtual para SSPP proporcionada por Nasertic

- Vmware 8.0.3
- Almacenamiento L1 red SAN

2.2.21. Infraestructura de Publicación

La aplicación deberá ser publicada en la infraestructura de publicación para SSPP proporcionada por Nasertic

- Publicador frontal
- Proxy inverso
- Permitir segmentación de redes para securización por capas