



ARQUITECTURA DE SOFTWARE
DOSSIER DE INGENIERÍA

Código: DOC-ARCH-001
Estado: VIGENTE
Clasificación: INTERNO

Ambato – Ecuador
28/01/2026

Contents

1. PROPÓSITO DEL DOCUMENTO.....	3
2. ALCANCE DE LA ARQUITECTURA	3
3. PRINCIPIOS ARQUITECTÓNICOS.....	4
4. ESTILO ARQUITECTÓNICO	4
5. DESCRIPCIÓN DE COMPONENTES PRINCIPALES.....	4
6. ARQUITECTURA DE CAPAS.....	5
7. CONSIDERACIONES DE SEGURIDAD.....	5
8. CONSIDERACIONES DE PERFORMANCE	5
9. INTEGRACIONES EXTERNAS	5
10. RESTRICCIONES Y DECISIONES TÉCNICAS	6

1. PROPÓSITO DEL DOCUMENTO

El presente documento define la arquitectura de software del sistema, estableciendo la estructura técnica de alto nivel que guiará el diseño detallado, la construcción, las pruebas y el despliegue del producto.

Este documento tiene como objetivo:

- Traducir los requisitos funcionales y no funcionales (DOC-REQ-002) en una solución técnica coherente.
- Servir como referencia única para desarrolladores, QA, DevOps y líderes técnicos.
- Garantizar decisiones técnicas alineadas con los objetivos de calidad definidos por ISO/IEC 25010.
- Reducir riesgos técnicos y ambigüedades antes de iniciar la construcción.

Normas de referencia:

ISO/IEC 12207 – Procesos de ciclo de vida

ISO/IEC 42010 – Descripción de arquitecturas

ISO/IEC 25010 – Modelo de calidad del producto

2. ALCANCE DE LA ARQUITECTURA

Este documento describe la arquitectura **a nivel conceptual y lógico**, no el diseño detallado de clases ni implementación específica de código.

Incluye:

- Estilo arquitectónico seleccionado
- Componentes principales del sistema
- Relaciones entre capas
- Consideraciones de seguridad y rendimiento
- Integraciones externas previstas
- Restricciones técnicas relevantes

No incluye:

- Diagramas UML detallados (ver DOC-MOD-001)
- Diseño físico de base de datos (ver DOC-DB-001)
- Manuales de despliegue operativo (ver DOC-TEC-001)

3. PRINCIPIOS ARQUITECTÓNICOS

Toda arquitectura definida en CSJ Software Develop debe cumplir los siguientes principios:

1. **Separación de responsabilidades**
Cada componente debe cumplir una función clara y única.
2. **Escalabilidad**
La arquitectura debe permitir crecimiento en usuarios, datos y funcionalidades sin rediseños mayores.
3. **Seguridad por diseño**
La seguridad no es un agregado posterior; se considera desde el diseño inicial.
4. **Trazabilidad completa**
Todo componente debe poder vincularse a uno o más requisitos en la Matriz de Trazabilidad (DOC-TRZ-001).
5. **Mantenibilidad**
El sistema debe facilitar correcciones, mejoras y evolución controlada.

4. ESTILO ARQUITECTÓNICO

Se adopta un estilo de Arquitectura en Capas, seleccionado para garantizar la separación de responsabilidades y la mantenibilidad del sistema. Esta elección se justifica por la necesidad de aislar la lógica de negocio de la infraestructura y facilitar la ejecución de pruebas unitarias aisladas.

Justificación técnica:

- **Mantenibilidad:** Facilita la evolución de componentes sin afectar a todo el sistema.
- **Escalabilidad:** Permite escalar capas específicas según la demanda de carga.

5. DESCRIPCIÓN DE COMPONENTES PRINCIPALES

Esta sección describe los **bloques funcionales de alto nivel** del sistema.

Para cada componente se debe documentar:

- Nombre del componente
- Responsabilidad principal
- Interfaces expuestas
- Dependencias con otros componentes
- Consideraciones de seguridad asociadas

El nivel de detalle debe permitir a cualquier desarrollador comprender el rol del componente sin necesidad de revisar código.

6. ARQUITECTURA DE CAPAS

El sistema se organiza en las siguientes capas lógicas, manteniendo una jerarquía estricta:

1. **Presentación:** Interfaces de usuario y manejo de sesiones.
2. **Aplicación:** Orquestación de servicios y casos de uso.
3. **Dominio / Negocio:** Entidades y reglas de negocio puras.
4. **Persistencia / Datos:** Acceso a repositorios y base de datos.
5. **Infraestructura:** Servicios externos, mensajería y utilitarios.

Regla Corporativa Obligatoria: No se permite el acceso directo entre capas que rompa la jerarquía definida (ej. la capa de Presentación no puede acceder directamente a la Persistencia sin pasar por Aplicación).

7. CONSIDERACIONES DE SEGURIDAD

Siguiendo el principio de Seguridad por diseño, se implementan las siguientes tácticas:

- **Autenticación:** Gestión mediante tokens seguros vinculados al servicio de identidad.
- **Autorización:** Control de acceso basado en roles (**RBAC**) conforme al requisito RNF-SEC-XX.
- **Mínimo Privilegio:** Cada componente opera con los permisos estrictamente necesarios para su función.

8. CONSIDERACIONES DE PERFORMANCE

Para cumplir con los umbrales de latencia establecidos en el SRS ($< 2s$), se definen:

- **Caché:** Implementación de capas de caché para consultas de alta lectura.
- **Concurrencia:** Estrategias de manejo de hilos para optimizar transacciones concurrentes.

9. INTEGRACIONES EXTERNAS

En esta sección se identifican los sistemas externos con los que interactúa la solución:

- APIs de terceros
- Servicios de autenticación
- Pasarelas de pago
- Servicios en la nube

Para cada integración se debe indicar:

- Propósito

- Tipo de comunicación
- Riesgos asociados
- Dependencia crítica o no crítica

10. RESTRICCIONES Y DECISIONES TÉCNICAS

Se deben documentar todas las restricciones relevantes, tales como:

- Tecnologías obligatorias
- Lenguajes permitidos
- Entornos de ejecución
- Limitaciones legales o contractuales

Asimismo, se deben registrar decisiones técnicas clave que afecten la arquitectura futura.

11. TRAZABILIDAD Y CONTROL

Toda arquitectura definida en este documento debe:

- Estar trazada a requisitos en DOC-TRZ-001
- Ser validada antes del inicio de desarrollo
- Ser actualizada ante cualquier cambio aprobado mediante DOC-CHG-001

La arquitectura es un documento **vivo**, pero cualquier modificación debe seguir el proceso formal de control de cambios.

12. APROBACIÓN INTERNA

Sin la aprobación formal de este documento, **no** se autoriza el inicio de la fase de construcción.

Recordatorios de Gobernanza para este Documento:

- **Trazabilidad:** Cada componente aquí descrito debe estar vinculado a un requisito en la Matriz de Trazabilidad (DOC-TRZ-001).
- **Documento Vivo:** Cualquier cambio en la arquitectura tras esta aprobación requiere una Solicitud de Cambio (DOC-CHG-001).