

# Documentación del Proyecto PAE

## Propósito

Este directorio contiene documentación técnica y arquitectónica del proyecto PAE (Plataforma de Análisis de Entregas), incluyendo guías de desarrollo, arquitectura sistema, patrones de diseño, deployment y troubleshooting.

## Tabla de contenidos rápida

Pregunta	Documento
Tengo una pregunta rápida	<a href="#">faq.md</a>
¿Por dónde empiezo?	<a href="#">newdev</a>
¿Cómo funciona la arquitectura?	<a href="#">architecture.md</a>
¿Cuál es el flujo de entrega?	<a href="#">delivery-service-flow.md</a>
¿Cómo funciona Machine online/offline?	<a href="#">machine-online-offline.md</a>
¿Cómo compilo y despliego?	<a href="#">build-and-deployment.md</a>
¿Tiene solución mi error?	<a href="#">troubleshooting.md</a>
¿Cómo hago tests?	<a href="#">testing-guide.md</a>
¿Cuáles son las APIs disponibles?	<a href="#">api-reference.md</a>
¿Qué significa este término?	<a href="#">glossary.md</a>

## Estructura de documentos (23 archivos)

### Fundamentals (4 archivos)

- [overview.md](#) - Visión general, módulos, flujos principales
- [architecture.md](#) - Estructura modular, 3 capas, dependencias
- [layers.md](#) - Responsabilidades de cada capa
- [module-interactions.md](#) - Grafo de dependencias, casos de uso

### Arquitectura y Diseño (4 archivos)

- [patterns.md](#) - Patrones de diseño, convenciones, SOLID
- [adr-architecture-decisions.md](#) - 13 decisiones arquitectónicas con justificación
- [best-practices.md](#) - Convenciones de código, patterns vs anti-patterns, quality gates
- [module-interactions.md](#) - Flujos de datos, casos de uso, error handling

### Configuración e Instalación (3 archivos)

- [environment-setup.md](#) - JDK 17, Android Studio, SDK, emulador, ADB
- [dependencies.md](#) - Gestión de versiones, matriz de librerías, conflictos

- [build-and-deployment.md](#) - Gradle, signing, ProGuard/R8, CI/CD, release APKs

## Desarrollo (4 archivos)

- [developer-guide.md](#) - APIs clave, componentes, tareas comunes
- [testing-guide.md](#) - Test pyramid, JUnit 4, Mockito, Espresso, fixtures, coverage
- [api-reference.md](#) - APIs públicas con tipo signatures
- [glossary.md](#) - Glosario técnico, siglas, acrónimos PAE

## Funcionalidad específica (5 archivos)

- [delivery-service-flow.md](#) - StateManager con 8 estados
- [p2p-architecture.md](#) - P2P contracts, models, P2PGestor
- [p2p-flow.md](#) - Workflows P2P con diagramas Mermaid
- [local-storage.md](#) - SQLite, VectorDB, persistencia
- [machine-online-offline.md](#) - Funcionamiento online/offline de Machine

## Seguridad y Operaciones (2 archivos)

- [security-and-privacy.md](#) - Permisos, encriptación, HTTPS, GDPR, compliance
- [hardware-integration.md](#) - Camera2, Bluetooth scale, GPIO LED, mocking

## Soporte y Referencia (1 archivo)

- [troubleshooting.md](#) - 20+ errores comunes y soluciones
- [faq.md](#) - Preguntas frecuentes por tema (¡Ver primero!)

# Rutas de aprendizaje por rol

## □ Nuevos Desarrolladores

Entender proyecto en 1-2 horas:

1. [overview.md](#) - El qué y por qué (15 min)
2. [architecture.md](#) - Estructura general (20 min)
3. [layers.md](#) - Cómo se comunican (15 min)
4. [environment-setup.md](#) - Instalar todo (30 min)
5. [developer-guide.md](#) - Componentes clave (20 min)
6. [build-and-deployment.md](#) - Primer build (15 min)
7. [faq.md](#) - Preguntas rápidas (según necesidad)

→ Profundizar: [best-practices.md](#) → [patterns.md](#)

## □ Arquitectos de Software

Entender decisiones y trade-offs:

1. [adr-architecture-decisions.md](#) - Las decisiones (30 min)
2. [architecture.md](#) - Cómo se conecta todo (20 min)
3. [module-interactions.md](#) - Flujos de datos (20 min)
4. [patterns.md](#) - Qué patterns usamos (15 min)
5. [layers.md](#) - Boundaries entre capas (15 min)

→ Profundizar: [p2p-architecture.md](#) → [security-and-privacy.md](#)

## □ Feature Developers

Implementar funcionalidades nuevas:

1. [developer-guide.md](#) - APIs disponibles (20 min)
2. [best-practices.md](#) - Cómo escribir código (20 min)
3. [patterns.md](#) - Patterns a usar (15 min)
4. [api-reference.md](#) - Firmas de API específicas (15 min)
5. [testing-guide.md](#) - Escribir tests (20 min)
6. [faq.md](#) - Problemas rápidos (según necesidad)

→ Si trabajas con entregas: [delivery-service-flow.md](#) → Si usas P2P: [p2p-flow.md](#)

## ⚙️ DevOps / Build Engineers

Deployment, CI/CD, versionado:

1. [build-and-deployment.md](#) - Compilation, signing, release (30 min)
2. [dependencies.md](#) - Versiones, conflictos, transitividad (20 min)
3. [environment-setup.md](#) - Build environment (20 min)
4. [troubleshooting.md](#) - Build failures (20 min)
5. [faq.md](#) - "How to compile?" (según necesidad)

## □ QA / Testing

Testing strategy y coverage:

1. [testing-guide.md](#) - Test pyramid, fixtures, mocking (40 min)
2. [best-practices.md](#) - Code review checklist (20 min)
3. [api-reference.md](#) - APIs to mock (20 min)
4. [troubleshooting.md](#) - Common test failures (15 min)
5. [faq.md](#) - Test help (según necesidad)

# Índice alfabético de archivos

## A-D

- [adr-architecture-decisions.md](#) - ADRs: decisiones arquitectónicas justificadas (13 records)
- [api-reference.md](#) - APIs públicas: StateManager, P2PManager, Services, Hardware
- [architecture.md](#) - Estructura de módulos, 3 capas, dependencias, versionado
- [best-practices.md](#) - Convenciones, patterns/anti-patterns, quality gates
- [build-and-deployment.md](#) - Gradle, ProGuard, keystore, signing, CI/CD, APKs
- [delivery-service-flow.md](#) - StateManager: 8 estados del flujo de entrega

## D-L

- [dependencies.md](#) - Librerías, versiones, matriz módulos, conflictos, exclusiones
- [developer-guide.md](#) - APIs clave, componentes, servicios, modelos, tareas comunes
- [environment-setup.md](#) - JDK 17, Android Studio, AndroidSDK, emulador, ADB, variables
- [faq.md](#) - Preguntas frecuentes por categoría (setup, build, dev, P2P, hardware, etc.)
- [glossary.md](#) - Glosario técnico, siglas, acrónimos específicos PAE
- [hardware-integration.md](#) - Camera2, Bluetooth scale, GPIO LED, power management

## L-T

- [layers.md](#) - Presentación/Dominio/Datos: responsabilidades, comunicación, ejemplos
- [local-storage.md](#) - SQLite, VectorDB ORM, persistencia, migración, limpieza
- [machine-online-offline.md](#) - Funcionamiento online/offline de Machine: conectividad, P2P, sincronización
- [module-interactions.md](#) - Grafo dependencias, casos uso, ciclo vida, error handling
- [overview.md](#) - Visión, módulos (11 core + 13 integrados), flujos principales
- [p2p-architecture.md](#) - P2P contracts, modelos, P2PGestor, P2PManager interface
- [p2p-flow.md](#) - Workflows P2P: descubierta, conexión, sincronización (con diagramas Mermaid)
- [patterns.md](#) - 5 patrones (State, Repository, Observer, Factory, Singleton), SOLID, convenciones
- [security-and-privacy.md](#) - Permisos runtime, encriptación, HTTPS/pinning, GDPR, sanitización
- [testing-guide.md](#) - Test pyramid, JUnit 4, Mockito, Espresso, fixtures, parametrized, coverage
- [troubleshooting.md](#) - 20+ errores comunes: Gradle, permisos, P2P, BD, OutOfMemory, tests

---

## Estadísticas de documentación

- **Total archivos:** 23 markdown files
- **Tamaño total:** ~120KB de contenido
- **Tiempo lectura completa:** 8-10 horas
- **Rutas onboarding rápido:** 1-2 horas por rol
- **Cobertura:** Arquitectura, desarrollo, testing, deployment, seguridad, hardware,

troubleshooting, FAQ

---

## Cómo contribuir documentación

Cada documento debe tener:

1. **Tabla de contenidos** al inicio (para docs > 500 líneas)
2. **Ejemplos de código** para conceptos complejos
3. **Links internos** a documentos relacionados
4. **Diagramas Mermaid** para flujos/grafos
5. **Time estimate** para lectura (ej: "□ 15 min")

Agregar nuevo doc:

1. Crear archivo en docs/
  2. Agregar a la matriz de "Índice de documentos"
  3. Update README.md con nueva entrada
  4. Asegurar cross-links a documentos relacionados
- 

## Cambios recientes

**Fase 5 (Actual):** Agregados 9 archivos faltantes

- build-and-deployment.md
  - testing-guide.md
  - security-and-privacy.md
  - hardware-integration.md
  - dependencies.md
  - best-practices.md
  - troubleshooting.md
  - api-reference.md
  - glossary.md
  - **faq.md** (nuevo!)
- 

## Links útiles

- [Proyecto GitHub](#)
- [Reporte de errores](#)
- [Decisiones arquitectónicas](#)
- [Guía de testing](#)

From:  
<http://wiki.adacsc.co/> - **Wiki**

Permanent link:  
<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:new-migracion-sicoferp:machine>

Last update: **2026/04/07 19:37**

