

Índice de documentación de PAE

Qué es este repo

PAE es un monorepo Android multi-módulo con dos aplicaciones coordinadas:

- **Machine**: ejecuta entregas en campo con validaciones de peso, evidencia por cámara, verificación facial y persistencia local.
- **RutaPAE**: coordina operación de ruta, visibilidad de máquinas, integración P2P y soporte de sincronización con backend.

Objetivo principal: mantener una operación confiable y auditable bajo conectividad intermitente y variabilidad de hardware.

Stack

- Android + Kotlin + Gradle (KTS)
- Jetpack Compose
- Coroutines + WorkManager
- CameraX + ML Kit + ONNX Runtime
- Persistencia local basada en SQLite (+ capacidades vectoriales en VectorialDB)
- Comunicación P2P guiada por contrato (Contract, DirectLink)
- Servicios de sincronización HTTP (MachineDomain, RutaPAEDomain)

Ambientes

- `development`
- `qa`
- `production` (actualmente mapeado a dominio preproductivo en la configuración vigente)

La selección de host/ambiente se define por configuración de runtime y variables de build.

Mapa de documentación

"arquitectura/"

- [Vision General](#)
- [Capas Y Modulos](#)
- [Ciclo De Vida De Datos](#)

"dispositivos/"

- [Bascula](#)
- [Camara](#)

"operacion-sin-conexion/"

- [Estados De Maquina](#)
- [Arquitectura P2p](#)
- [Estrategia De Sincronizacion](#)

"flujos-clave/"

- [Flujo De Entrega](#)
- [Flujo De Generacion De Embedding](#)
- [Flujo Del Gestor De Estados](#)
- [Ciclo De Vida De Jornada](#)
- [Flujo De Sincronizacion Http](#)
- [Filtrado De Jornadas En Shift Selection](#)

"api/"

- [Puntos Finales](#)
- [Contratos P2p](#)
- [Modelo De Errores](#)

"decisiones/"

- [Adr 0001 Secuencia Del Gestor De Estados](#)
- [Adr 0002 Persistencia Primero Local](#)

Fuentes de verdad

- `settings.gradle.kts` para límites y composición de módulos.
- `gradle/libs.versions.toml` para versiones de dependencias.
- `MachineDomain/src/main/java/co/ada/domain/DomainManager.kt` para armado del flujo y estados.
- `MachineDomain/src/main/java/co/ada/domain/state/StateManager.kt` para orquestación y recuperación.
- `MachineDomain/src/main/java/co/ada/domain/state/states/*.kt` para comportamiento de cada estado.
- `MachineData/src/main/java/co/ada/data/emitters/*` para señales de estado/mensaje.
- `MachineData/src/main/java/co/ada/data/local/Server.kt` para hosts por ambiente.

Ultima verificacion

- Fecha: 2026-04-23
- Método: revisión manual de código + reconciliación de documentación.
- Alcance: estructura de docs, flujo de estados, stack base y mapeo de ambientes.

Nota

Las carpetas legadas (explicaciones/, referencia/, guías/, producto/) se mantienen para compatibilidad mientras se consolida la nueva estructura.

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

Permanent link:
<http://wiki.adacsc.co/doku.php?id=ada:howto:sicoferp:factory:new-migracion-sicoferp:machine&rev=1778174502>

Last update: **2026/05/07 17:21**

