

Plan de Desarrollo y Arquitectura

Sistema de Gestión para Metalería

Fecha: 24 de noviembre de 2025

Objetivo: Tener el MVP listo para el 26 de diciembre de 2025

Índice (Resumen de Secciones)

1. Contexto general del sistema
2. Objetivos del MVP
3. Roles, perfiles y permisos
4. Arquitectura técnica y módulos
5. Modelo de datos (visión de alto nivel)
6. Flujos de negocio principales
7. Seguridad, sesiones y auditoría
8. Inventario y contabilidad
9. Reportes y exportaciones
10. Plan de implementación hacia el 26 de diciembre de 2025

1. Contexto general del sistema

El sistema es una aplicación web 100% enfocada en la operación y control de una metalería con múltiples sucursales.

La solución cubre dos grandes necesidades:

- Operación diaria de pesajes por parte de los trabajadores (registro de compras y ventas de material).
- Revisión, consolidación, contabilidad e inventario por parte de administradores y super administradores.

Características clave del contexto:

- Número inicial de sucursales: 3, con posibilidad de crecimiento moderado.
- Cada sucursal tiene uno o más trabajadores y uno o más administradores.
- Existe un super administrador original con control total del sistema y capacidad de crear/eliminar otros super administradores.
- Volumen esperado: máximo ~10 notas de pesaje por día por sucursal, con horizonte de diseño de 3 años sin necesidad de una re-arquitectura profunda.
- El sistema debe ser fácil de usar, especialmente en dispositivos móviles para trabajadores, y profesional/escalable del lado técnico.

2. Objetivos del MVP

El MVP (versión mínima viable) debe ser completamente funcional en producción y cubrir la operación real de la metalería.

Objetivos principales:

- Permitir a los trabajadores registrar pesajes de material con sub-pesajes, descuentos en kilogramos y evidencias fotográficas.
- Generar notas preliminares que sean revisadas, corregidas y aprobadas por los administradores.
- Actualizar automáticamente el inventario por sucursal y material al aprobar compras y ventas.
- Generar movimientos contables por cada nota aprobada (compras, ventas, cancelaciones con reverso).
- Permitir al super administrador consultar y descargar registros contables filtrados por sucursal y periodo.
- Mantener una auditoría de cambios clave para trazabilidad interna.
- Diseñar una arquitectura modular, clara y extensible para fases futuras (pagos parciales, transferencias entre sucursales, gráficas avanzadas, etc.).

3. Roles, perfiles y permisos

3.1 Super admin original

- Control total del sistema.
- Puede crear/editar/eliminar otros super admins.
- Puede crear/editar/eliminar administradores y trabajadores.
- Puede crear y configurar sucursales.
- Puede administrar el catálogo de materiales y las tablas de precios de compra/venta.
- Puede crear y administrar cuentas financieras (caja, bancos, cuentas internas).
- Puede ver y modificar cualquier nota, inventario o movimiento contable.
- Tiene acceso exclusivo a la auditoría detallada de cambios.
- Puede realizar ajustes manuales de inventario.

3.2 Super admins adicionales

- Tienen las mismas capacidades funcionales que el super admin original.
- Solo el super admin original puede crearlos o eliminarlos.

3.3 Administrador de sucursal

- Asociado a una o varias sucursales (asignadas por super admin).
- Puede ver todas las notas de sus sucursales (pendientes, aprobadas, canceladas).
- Puede editar trabajadores de sus sucursales (crear, actualizar, cambiar estado).
- Puede administrar proveedores y clientes vinculados a sus sucursales.
- Revisa y corrige notas preliminares enviadas por los trabajadores.
- Completa método de pago y asigna cuenta financiera correspondiente.
- Vincula proveedor al catálogo si fue creado de forma básica por el trabajador.
- Aprueba o cancela notas:
- Al aprobar, se actualizan inventario y contabilidad.
- Al cancelar una nota aprobada, se generan movimientos inversos según las reglas definidas.
- Puede ver reportes básicos para sus sucursales, pero no descargar reportes contables globales.

3.4 Trabajador

- Tiene acceso únicamente a:
- Iniciar sesión.
- Registrar un nuevo pesaje (nota preliminar).
- No consulta inventario ni contabilidad.
- No ve reportes globales.
- Su estado debe ser 'activo' para poder generar registros.

4. Arquitectura técnica y módulos

4.1 Stack tecnológico

- Backend: FastAPI (Python).
- Base de datos relacional: PostgreSQL.
- Frontend: HTML + CSS + JavaScript + Bootstrap, renderizado con Jinja templates.
- Almacenamiento de imágenes: Amazon S3 (con límite de tamaño razonable y soporte

para versiones futuras).

- Despliegue: Heroku (aplicación web y base de datos Heroku Postgres).
- Entornos: 'dev' y 'prod'.

4.2 Arquitectura lógica por módulos

Se propone una arquitectura modular por dominio de negocio:

- Módulo 'users':
 - Gestión de usuarios, roles y estados.
 - Autenticación y sesiones.
- Módulo 'branches':
 - Definición de sucursales (nombre, configuración básica).
- Módulo 'materials':
 - Catálogo de materiales.
 - Tablas de precios de compra (Regular, Mayorista, Menudeo) y de venta.
- Módulo 'partners':
 - Proveedores y clientes (sin acceso al sistema).
- Módulo 'weighings':
 - Notas de pesaje (compras y ventas).
 - Materiales de la nota.
 - Sub-pesajes con descuentos y fotos.
- Flujo de estados: BORRADOR, EN_REVISION, APROBADA, CANCELADA.
- Módulo 'inventory':
 - Inventario actual por sucursal y material.
 - Movimientos de inventario (entradas, salidas, ajustes, transferencias).
- Módulo 'accounting':
 - Movimientos contables por nota (compras, ventas, reversos, ajustes).
 - Pagos parciales y saldos (diseñado para MVP, aunque UI de pagos puede evolucionar).
- Módulo 'finance':
 - Cuentas financieras (caja, bancos, cuentas internas por sucursal).
- Módulo 'reports':
 - Reportes y exportaciones (inicialmente registros contables).
- Módulo 'audit':
 - Registro de auditoría de cambios relevantes.
- Módulo 'files':
 - Integración con S3 para subida y gestión de fotos de báscula.

Esta separación facilita mantener el sistema legible y escalable, sin caer en una complejidad innecesaria para el MVP.

5. Modelo de datos (visión de alto nivel)

A nivel conceptual, el modelo se organiza en los siguientes grupos:

5.1 Usuarios y sucursales

- Tabla 'users':
 - id, username, password_hash, nombre_completo, rol (super_admin, admin, trabajador), estado (activo/inactivo), sucursal_id (obligatorio para trabajadores), super_admin_original (bool).
- Tabla 'sucursales':
 - id, nombre, dirección, estado.
- El nombre de la sucursal se usa como prefijo del folio de nota (ej. 'SucursalCentro-0001').

5.2 Materiales y precios

- Tabla 'materiales':
 - id, nombre, descripción, activo.
- Tabla 'tablas_precios':
 - id, material_id, tipo (COMPRA/VENTA), precios específicos según tipo:
 - COMPRA: precio_regular, precio_mayorista, precio_menudeo.
 - VENTA: precio_venta.
 - fecha_creacion, es_ultima (indica la versión vigente para cada material y tipo).

5.3 Proveedores y clientes

- Tabla 'proveedores':
 - id, nombre_completo, telefono, correo_electronico, placas, activo.
- Tabla 'clientes':
 - id, nombre_completo, telefono, correo_electronico, placas, activo.
- Las notas pueden enlazarse a proveedores/clientes existentes o nuevos que luego formaliza el administrador.

5.4 Notas de pesaje

- Tabla 'notas':
 - id, folio, tipo (COMPRA/VENTA), sucursal_id, trabajador_id (quien creó), admin_id (quien aprobó), proveedor_id/cliente_id (según el caso), totales de kilos (bruto, descuentos, neto) e importe total, método de pago principal, cuenta_financiera_id, estado_nota (BORRADOR, EN_REVISION, APROBADA, CANCELADA), timestamps y snapshot 'nota_original_trabajador'.
- Tabla 'nota_materiales':
 - id, nota_id, material_id, tipo_precio_compra (Regular/Mayorista/Menudeo) cuando aplica, precio_compra_kg y/o precio_venta_kg (congelados), totales de kilos e importes por material.
- Tabla 'subpesajes':
 - id, nota_material_id, peso_bruto_kgs, descuentos_en_kg: tierra, humedad, basura, jumbos, peso_neto_kgs, foto_id,

fecha_registro, registrado_por_user_id.

- Tabla 'fotos':
 - id, url_s3, fecha_subida, subido_por_user_id.

5.5 Inventario

- Tabla 'inventario_actual':
 - sucursal_id, material_id, kilos_actuales.
- Tabla 'movimientos_inventario':
 - sucursal_id, material_id, tipo (ENTRADA, SALIDA, AJUSTE, TRANSFERENCIA_ENTRADA, TRANSFERENCIA_SALIDA), kilos, fecha, nota_id, creado_por_user_id, descripción.

5.6 Contabilidad y pagos

- Tabla 'cuentas_financieras':
 - id, sucursal_id (o global), tipo (CAJA, BANCO, INTERNA_SUCURSAL), nombre_mostrado, datos bancarios, activo.
- Tabla 'movimientos_contables':
 - id, tipo (COMPRA_EXTERNA, VENTA_EXTERNA, TRANSFERENCIA..., AJUSTE_INVENTARIO, REVERSO), nota_id, sucursal_id, contraparte_tipo (PROVEEDOR, CLIENTE, SUCURSAL), contraparte_id, montos (bruto, descuentos, neto, pagado, pendiente), método de pago principal, cuenta_financiera_id, fecha_movimiento, creado_por_user_id, estado, reversa_de_id.
- Tabla 'pagos_movimiento':
 - id, movimiento_contable_id, tipo_pago (PAGO_A_PROVEEDOR/COBRO_DE_CLIENTE), cuenta_financiera_id, metodo_pago, monto, fecha_pago, referencia, registrado_por_user_id.

5.7 Auditoría

- Tabla 'audit_log':
 - id, entidad, entidad_id, accion, usuario_id, fecha, datos_anteriores (JSON), datos_despues (JSON).

6. Flujos de negocio principales

6.1 Flujo de compra (nota de pesaje)

- 1) Trabajador inicia sesión.
- 2) Selecciona proveedor:
 - Puede buscar proveedores existentes.
 - Si no existe, puede capturar datos básicos (nombre, placas, etc.).
- 3) Inicia una nueva nota de tipo COMPRA para su sucursal.

4) Para cada material que trae el proveedor:

- Selecciona el material.
- Selecciona el tipo de precio de compra (Regular/Mayorista/Menudeo).
- Registra uno o varios sub-pesajes:
- Peso bruto en kg.
- Descuentos en kg: tierra, humedad, basura, jumbos.
- El sistema calcula peso neto = peso_bruto - sum(descuentos).
- Toma y sube foto de báscula (evidencia).

5) El sistema calcula totales por material y totales de la nota (kilos e importe).

6) El trabajador revisa la información y envía la nota a revisión:

- estado_nota pasa a EN_REVISION.
- Se guarda un snapshot de la nota original del trabajador para auditoría.

7) El administrador de la sucursal:

- Ve las notas en estado EN_REVISION.
- Puede corregir campos (pesos, descuentos, precios aplicados, proveedor final, etc.), pero no las fotos.
- Completa método de pago y asigna cuenta financiera correspondiente.
- Vincula proveedor al catálogo si fue creado de forma básica por el trabajador.

8) El administrador aprueba la nota:

- estado_nota = APROBADA.
- Se generan movimientos de inventario de tipo ENTRADA por material.
- Se genera un movimiento contable de tipo COMPRA_EXTERNA con los datos finales de la nota.

6.2 Flujo de venta

El flujo de venta a clientes es análogo al de compra, pero:

- La nota se registra como tipo VENTA.
- Se descuenta material del inventario de la sucursal (movimientos SALIDA).
- El movimiento contable se registra como VENTA_EXTERNA.
- El cliente puede seleccionarse de catálogo o registrarse de manera básica para que luego el admin lo formalice.

6.3 Cancelación de notas aprobadas

- Solo admins y super admins pueden cancelar notas aprobadas.
- Al cancelar una nota aprobada:
 - Se generan movimientos de inventario inversos (revirtiendo entradas o salidas).
 - Se crea un movimiento contable de tipo REVERSO, enlazado al movimiento original.
- No se permite cancelar notas que no hayan generado un movimiento contable previo (es decir, no se reversa algo que nunca se contabilizó).

6.4 Manejo de evidencias fotográficas

- Para el MVP, si falla la subida de una foto:

- Se permite avanzar con el registro, marcando la nota como “con evidencia incompleta”.
- Los administradores pueden revisar estos casos, subir la evidencia faltante o aprobar conscientemente la nota.
- Todas las fotos se almacenan en S3, asociadas a sub-pesajes individuales.

7. Seguridad, sesiones y auditoría

7.1 Seguridad y sesiones

- Contraseñas de trabajadores: gestionadas exclusivamente por administradores y super admins.
- Contraseñas de administradores: gestionadas exclusivamente por super admins.
- No se imponen reglas de complejidad estrictas en contraseñas (sólo se evita que sean vacías).
- Sesiones:
 - Trabajadores: sesiones de larga duración (pensadas para no interrumpir constantemente su flujo en patio).
 - Administradores: sesiones de aproximadamente 24 horas.
 - Se permite que un mismo usuario inicie sesión en múltiples dispositivos simultáneamente.
- Recuperación de acceso:
 - Trabajadores: el administrador puede cambiar la contraseña y comunicársela.
 - Administradores: sólo el super admin puede resetearla.
 - Autenticación basada en sesiones/ cookies y/o tokens, con protección básica frente a accesos no autorizados.

7.2 Auditoría

- Se registran eventos clave en la tabla de auditoría:
- Creación y modificación de notas.
- Cambios de estado (BORRADOR, EN_REVISION, APROBADA, CANCELADA).
- Cambios en inventario (incluyendo ajustes manuales).
- Cambios en tablas de precios.
- Creación y modificación de usuarios, proveedores, clientes y sucursales.
- Para cada evento se almacenan:
 - Usuario responsable.
 - Fecha/hora.
 - Valores antes y después (en formato JSON).
 - La auditoría detallada es visible únicamente para el super admin (principalmente el original), evitando exponer demasiada información sensible a otros roles.

8. Inventario y contabilidad

8.1 Inventario

Enfoque: inventario simple por sucursal y material, con soporte de historial a través de movimientos.

- El inventario actual se guarda en 'inventario_actual' (kilos_actuales por sucursal y material).
- Cada nota aprobada de COMPRA genera movimientos de inventario de tipo ENTRADA.
- Cada nota aprobada de VENTA genera movimientos de inventario de tipo SALIDA.
- Ajustes manuales de inventario:
- Sólo el super admin puede realizarlos.
- Se registran como movimientos de tipo AJUSTE con descripción.
- A partir de 'movimientos_inventario' se puede reconstruir la historia del inventario en el tiempo, y generar reportes simples de evolución si se requiere.

8.2 Contabilidad

Se propone un modelo contable sencillo, pero alineado con buenas prácticas:

- Cada nota aprobada genera un 'movimiento_contable':
 - COMPRA_EXTERNA para compras a proveedores.
 - VENTA_EXTERNA para ventas a clientes.
- Campos clave de cada movimiento contable:
 - vínculo a sucursal, contraparte (proveedor/cliente/sucursal), montos (bruto, descuentos, neto).
 - montos pagados y pendientes (para futuros pagos parciales).
 - método de pago principal y cuenta financiera usada.
- Pagos parciales:
 - Se modelan en 'pagos_movimiento', enlazados al movimiento contable.
 - Permiten reflejar: cuánto se ha pagado a un proveedor y cuánto ha pagado un cliente.
 - Para el MVP, los flujos de UI de pagos pueden ser simples, priorizando primero el registro contable por nota.
- Cancelaciones con reverso:
 - Si una nota aprobada se cancela, se genera un movimiento de tipo REVERSO enlazado al original.
 - Se evita restar dinero de registros que nunca se contabilizaron, siguiendo la regla: sólo se genera reverso si existe un movimiento original.

9. Reportes y exportaciones

Para el MVP, el foco principal está en el reporte contable descargable para el super admin.

9.1 Reporte contable (descargable)

- Acceso: sólo super admin.
- Filtros principales:
- Sucursal.
- Rango de fechas.
- Tipo de movimiento (COMPRA_EXTERNA, VENTA_EXTERNA, etc.).
- Contenido mínimo:
- Fecha del movimiento.
- Tipo de movimiento.
- Sucursal.
- Proveedor/cliente/otra sucursal (según el caso).
- Folio de la nota.
- Monto bruto, descuentos y monto neto.
- Monto pagado y monto pendiente.
- Método de pago y cuenta financiera.
- Estado del movimiento (ABIERTO/CERRADO/CANCELADO).

9.2 Futuros reportes (fuera del alcance mínimo, pero soportados por la arquitectura)

- Reporte de inventario actual consolidado.
- Reporte de compras/ventas por material.
- Gráficas de evolución de inventario por material y sucursal.
- KPIs de sucursales (compras, ventas, utilidad aproximada).

10. Plan de implementación hacia el 26 de diciembre de 2025

Considerando que hoy es 24 de noviembre de 2025 y la fecha objetivo para tener el MVP listo es el 26 de diciembre de 2025, se propone el siguiente plan de trabajo en bloques semanales.

Semana 1 (24 nov – 30 nov)

- Configurar el proyecto FastAPI + PostgreSQL + Jinja en Heroku (entorno dev).
- Definir modelos de datos y migraciones para:
 - users, sucursales.
 - materiales, tablas_precios.
 - proveedores, clientes.
 - notas, nota_materiales, subpesajes, fotos.
- Implementar autenticación básica (roles y estados).
- Crear pantallas iniciales:
 - Login.
- Pantalla de trabajador: creación de nueva nota de pesaje.

Semana 2 (1 dic – 7 dic)

- Implementar el flujo completo de nota de compra:
- Trabajador: creación de nota preliminar con sub-pesajes y fotos (S3).
- Envío a revisión.
- Implementar panel de administrador:
- Listado de notas EN_REVISION.
- Vista de detalle de nota, con capacidad de edición de campos (no fotos).
- Completar método de pago y cuenta financiera.
- Acción de aprobar nota:
- Generar movimientos de inventario (entradas).
- Actualizar inventario_actual.
- Iniciar registro de auditoría para notas.

Semana 3 (8 dic – 14 dic)

- Implementar módulo de contabilidad:
- cuentas_financieras.
- movimientos_contables para COMPRA_EXTERNA y VENTA_EXTERNA.
- Integración aprobación de nota → movimiento contable.
- Implementar cancelación de notas aprobadas con movimientos REVERSO e inversión de inventario.
- Implementar reporte contable descargable para super admin (CSV/Excel).

Semana 4 (15 dic – 21 dic)

- Implementar vistas de inventario por sucursal/material.
- Implementar vistas de movimientos de inventario (solo lectura).
- Construir panel de super admin:
- Gestión de sucursales.
- Gestión de admins y trabajadores.
- Gestión de materiales y precios.
- Gestión de cuentas financieras.
- Afinar permisos y restricciones de acceso.

Semana final y buffer (22 dic – 26 dic)

- Pruebas end-to-end de flujos completos (compra, venta, cancelación, descarga de reporte).
- Ajustes de usabilidad, especialmente en la interfaz móvil para trabajadores.
- Configuración de copias de seguridad automáticas de la base de datos:
- Habilitar backups diarios en Heroku Postgres.
- Configurar retención adecuada (7–30 días).
- Configurar bucket S3 con versioning y políticas de ciclo de vida.

Este plan está diseñado para entregar un MVP funcional, serio y presentable, cubriendo todas las necesidades operativas básicas de la metalería, y sentando una base sólida para funcionalidades más avanzadas en siguientes fases.