

Materia:

Bases de datos

Docente:

Jesús Revilla

**Proyecto Sistema de Optimización de Corte de Materia Prima en PostgreSQL-
Manual Usuario**

Integrantes del equipo:

Castillo Silva Ángel	179752
Guerrero Guevara Yazmin	182483
López González Cruz Ángel	183213
Martínez Huerta Patricia Michelle	181730
Pulido Alvarado Ximena	179800
Sánchez Saucedo Rafael	182712
Vidales Ramírez Luis David	183060

Fecha de entrega:

24 de noviembre de 2025

Contenido

Visión General	3
Gestión de Usuarios (Seguridad).....	3
Paso 1: Configurar el Taller (Materia Prima)	3
Paso 2: Definir el Trabajo (Productos)	4
Paso 3: Ejecución del Corte (Posicionamiento).....	4
Paso 4: Revisión de Resultados	4
Anexo: Solución de Problemas (Testing).....	5

Visión General

Este manual está dirigido al Administrador del Sistema y a los Operadores. Describe cómo realizar las tareas diarias de registro de inventario y ejecución de cortes utilizando las funciones predefinidas del sistema.

Nota: Al ser un sistema basado en SQL, las "Acciones" descritas a continuación corresponden a comandos que pueden ser ejecutados desde una consola SQL (como pgAdmin) o son llamados por su aplicación de software.

Gestión de Usuarios (Seguridad)

El sistema divide a los usuarios en dos niveles:

1. **Admin:** Control total.
2. **Operador:** Solo puede ver datos, no puede modificar configuraciones críticas.

Cómo crear un usuario nuevo: No inserte datos directamente en la tabla. Use la función segura que encripta y asigna roles correctamente.

Ejemplo de comando:

```
-- Crea un usuario 'juan' con rol de operador  
SELECT fn_crear_usuario('juan','1234','operador');
```

(Referencia: sql/02_crud_usuarios_rols.sql)

Paso 1: Configurar el Taller (Materia Prima)

Antes de cortar, debe decirle al sistema qué materiales tiene disponibles.

Acción: Dar de alta una hoja de material.

- Parámetros:
 - `_numero_parte`: Código interno (ej. 'MADERA-01').
 - `_ancho`: Ancho total de la hoja.
 - `_alto`: Alto total de la hoja.

Código de ejecución:

```
-- Registra una hoja de 100x100 unidades  
CALL sp_alta_materia_prima('MP-01', 100, 100);
```

(Referencia: sql/03_materia_prima.sql)

El sistema rechazará automáticamente cualquier registro si intenta poner distancias negativas (gracias al trigger de validación).

Paso 2: Definir el Trabajo (Productos)

Registre qué es lo que va a fabricar. Un producto puede requerir múltiples copias.

Código de ejecución:

```
-- Registra el producto 'Silla Modelo A', se requieren 50 unidades  
CALL sp_alta_producto('PROD-SILLA', 'Silla de pino básica', 50);
```

(Referencia: sql/04_productos_piezas.sql)

Paso 3: Ejecución del Corte (Posicionamiento)

Esta es la operación diaria principal. Cuando el operario o el algoritmo de optimización decide dónde colocar una pieza en la hoja, se debe ejecutar el procedimiento de rotación.

Escenario: Usted necesita mover la pieza #1 a la posición X=10, Y=20 y rotarla 45 grados para que encaje mejor.

Acción en el sistema: Ejecute sp_rotar_posicionar_figuras. El sistema actualizará la geometría y guardará un registro histórico del evento.

Código de ejecución:

```
-- Mueve la pieza 1, rotación 45°, coords (10, 20), y etiqueta el evento como 'ajuste manual'  
CALL sp_rotar_posicionar_figuras(  
    1,      -- ID Pieza  
    45,     -- Ángulo  
    10,     -- X  
    20,     -- Y  
    '{"evento":"ajuste manual", "usuario":"juan"}' -- JSON de auditoría  
);
```

(Referencia: sql/07_procedimientos.sql)

Paso 4: Revisión de Resultados

Al final del turno, usted querrá saber qué tan bien se aprovechó el material.

Acción: Consultar la función de utilización. El sistema dividirá el área de todas sus piezas entre el área de la materia prima.

Código de ejecución:

```
SELECT fn_calcular_utilizacion(1); -- 1 es el ID de la materia prima
```

(Referencia: sql/08_funciones.sql)

Interpretación:

- Si el resultado es 0.90: Excelente, se aprovechó el 90% del material.
- Si el resultado es 0.40: Malo, hay mucho desperdicio (60%).

Anexo: Solución de Problemas (Testing)

Si sospecha que el sistema está fallando, el equipo técnico ha preparado un script de pruebas automatizadas en Python (tests/test_proyecto.py).

Puede verificar que la lógica de rotación funciona correctamente revisando si las pruebas pasan. El test inserta una geometría base y verifica que, tras llamar al procedimiento, los datos coincidan:

```
# Extracto de tests/test_proyecto.py
def test_rotar_posicionar(conn):
    # ... código ...
    cur.execute("""
        CALL sp_rotar_posicionar_figuras(1,45,10,20,{"evento":"rotar"});
    """)
    # ... validación ...
    assert geo["angulo"] == 45
    assert geo["x"] == 10
```