

## 1. GUÍA DE DESPLIEGUE

### Prerrequisitos del Sistema

#### Requisitos Mínimos de Hardware:

- Procesador: Dual-core 2.0 GHz o superior
- RAM: 4 GB mínimo (8 GB recomendado)
- Almacenamiento: 500 MB de espacio libre
- Sistema operativo: Windows 10+, Linux, macOS

#### Requisitos de Software:

- Java JDK 8 o superior
- Permisos de escritura en directorio de instalación
- Resolución pantalla: 1024x768 mínimo

### Paso a Paso de Instalación

#### Paso 1: Verificar Java

bash

# En terminal/console

java -version

javac -version

#### Salida esperada:

text

java version "1.8.0\_291"

Java(TM) SE Runtime Environment (build 1.8.0\_291-b10)

#### Paso 2: Descargar e Instalar

##### Opción A - Instalación Automática (Windows)

1. Descargar SistemaContable-Instalador.exe
2. Ejecutar como administrador
3. Seguir asistente de instalación
4. La aplicación se instalará en C:\Program Files\SistemaContable\

##### Opción B - Instalación Manual

bash

# 1. Crear directorio

mkdir SistemaContable

```
cd SistemaContable
```

# 2. Copiar archivos JAR y librerías

```
cp sistema-contable.jar ./
```

```
cp lib/gson-2.8.9.jar ./lib/
```

# 3. Estructura final:

SistemaContable/

├─ sistema-contable.jar

├─ start.bat      # Windows

├─ start.sh      # Linux/Mac

├─ lib/

|    └─ gson-2.8.9.jar

└─ config/

    └─ application.properties

### Paso 3: Scripts de Ejecución

**Windows (start.bat):**

```
batch
```

```
@echo off
```

```
title Sistema Contable Empresarial
```

```
echo Iniciando Sistema Contable...
```

```
java -cp "sistema-contable.jar;lib\*" SistemaContable
```

```
pause
```

**Linux/Mac (start.sh):**

```
bash
```

```
#!/bin/bash
```

```
echo "Iniciando Sistema Contable Empresarial..."
```

```
java -cp "sistema-contable.jar:lib/*" SistemaContable
```

### Paso 4: Verificar Instalación

1. Ejecutar start.bat (Windows) o ./start.sh (Linux/Mac)
2. Debe aparecer ventana de login

3. Verificar que se creen archivos JSON iniciales

## 2. GUÍA DE OPERACIÓN/USO

### Flujo Básico del Sistema

#### 2.1 Inicio de Sesión

1. **Ejecutar** la aplicación
2. **Ingresar credenciales:**
  - *Jefatura Financiera:* admin / admin123
  - *Asistente Contable:* asistente / asistente123
3. **Verificar** que el rol se muestre correctamente

### Operaciones Esenciales

#### Registrar Factura

1. Seleccionar "Registrar Factura"
2. Completar formulario:
  - **Fecha:** DD/MM/AAAA (no puede ser futura)
  - **Cliente:** Nombre o razón social
  - **Subtotal:** Monto antes de IVA (ej: 100.00)
  - **Cuenta Contable:** Código contable (ej: 2101.01)
  - **Número Factura:** Identificador único (ej: FAC-2024-001)
3. El sistema calcula automáticamente:
  - IVA (15%): \$15.00
  - Total: \$115.00
4. Confirmar registro

#### Aprobar/Rechazar Transacción (Solo Jefatura)

1. Seleccionar "Revisar Transacciones Pendientes"
2. Ver lista con estado "Registrado"
3. Seleccionar transacción para revisar detalles
4. Elegir "Aprobar" o "Rechazar"
5. Ingresar motivo (opcional para aprobación, obligatorio para rechazo)

#### Generar Reporte de Retención IVA

1. Ir a "Generar Reportes"
2. Seleccionar año fiscal

3. El sistema calcula:
  - Total IVA en compras
  - Retención (30% del IVA compras)
  - Transacciones consideradas
4. Opción exportar a CSV

### **Mensajes del Sistema**

#### **Éxito:**

- "Factura FAC-2024-001 registrada exitosamente"
- "Transacción #5 aprobada"
- "Respaldo generado: transacciones\_respaldo\_2024.json"

#### **Errores:**

- "Error: Campos incompletos"
- "No tiene permisos para esta operación"
- "Fecha no puede ser futura"
- "El subtotal debe ser mayor a 0"

### **Mantenimiento Rutinario**

#### **RespalDOS Diarios**

bash

# El sistema genera respaldos automáticos

transacciones\_respaldo\_2024\_01\_15.json

bitacora\_respaldo\_2024\_01\_15.json

#### **Limpieza de Archivos Temporales**

- Los archivos JSON se mantienen automáticamente
- Verificar espacio en disco periódicamente
- No eliminar manualmente archivos JSON

## **4. POST-MORTEM: LECCIONES APRENDIDAS Y RIESGOS RESIDUALES**

### **Lecciones Aprendidas**

#### **Lo que Funcionó Bien:**

1. **Arquitectura en Capas**

- Separación clara entre controladores, modelos y persistencia
- Facilita mantenimiento y pruebas

## 2. Patrón Singleton en Controladores

- Garantiza una única instancia de controladores críticos
- Evita inconsistencia de datos

## 3. Persistencia JSON

- Fácil debuggeo (archivos legibles)
- No requiere servidor de base de datos
- Backup y restauración simples

## 4. Bitácora de Auditoría

- Trazabilidad completa de operaciones
- Cumple con requerimientos de compliance

### Lecciones Técnicas:

#### 1. Validación Temprana

- Implementar validación en UI y modelos
- Mensajes de error específicos y útiles

#### 2. Manejo de Excepciones

- Logging consistente en todos los niveles
- Excepciones personalizadas para errores de negocio

#### 3. Pruebas Automatizadas

- Invertir en pruebas unitarias desde el inicio ahorra tiempo
- Pruebas de integración críticas para flujos complejos

### Riesgos Residuales

#### Riesgos Técnicos (Medio Impacto):

##### 1. Escalabilidad de Archivos JSON

- **Riesgo:** Performance con >10,000 registros
- **Mitigación:** Plan de migración a base de datos
- **Monitorizar:** Tiempo de respuesta >2 segundos

##### 2. Corrupción de Archivos

- **Riesgo:** Pérdida de datos por fallo de escritura
- **Mitigación:** Respaldos automáticos cada 4 horas

- **Procedimiento:** Restauración desde respaldo más reciente

#### **Riesgos Operacionales (Alto Impacto):**

##### **3. Error en Cálculos Fiscales**

- **Riesgo:** Incumplimiento con normativa tributaria
- **Mitigación:** Doble verificación manual mensual
- **Contingencia:** Recalculo manual y ajuste contable

##### **4. Acceso No Autorizado**

- **Riesgo:** Exfiltración de información financiera
- **Mitigación:** Auditoría trimestral de bitácoras
- **Procedimiento:** Rotación de contraseñas cada 90 días

#### **Riesgos Críticos (Requieren Atención Inmediata):**

##### **5. Falta de Capacitación**

- **Riesgo:** Uso incorrecto del sistema
- **Mitigación:** Sesiones de entrenamiento mensuales
- **Métrica:**  $\leq 2$  errores de usuario por semana

#### **Recomendaciones para Próxima Versión**

##### **Prioritario (Sprint 1):**

1. Implementar recuperación de contraseñas
2. Mejorar reportes con gráficos
3. Exportación a Excel/PDF

##### **Medio Plazo (Sprint 2-3):**

4. Integración con sistema bancario
5. Módulo de conciliación automática
6. Notificaciones por email

##### **Largo Plazo (Roadmap):**

7. Versión web/móvil
8. Machine learning para detección de anomalías
9. API para integración con otros sistemas