# Proyecto 4: Sistema de Facturación

## Portada

**Proyecto 4: Sistema de Facturación**
**Curso: Programación Orientada a Objetos**
**Framework: Ruby on Rails**
**Integrantes:**


- Daniel Alemán Ruiz
- Luis Meza Chavarría


---

## Índice



---

## Enlace de GitHub

Repositorio del Proyecto

---

## Descripción del Proyecto

Este proyecto es un sistema completo de facturación desarrollado con Ruby on Rails que permite a una empresa administrar inventario, emitir facturas con cálculo automático de impuestos y generar reportes de ventas y stock mediante una aplicación web moderna y funcional.

El sistema está diseñado siguiendo los principios SOLID y las buenas prácticas de la Programación Orientada a Objetos, proporcionando una solución robusta para la gestión empresarial con las siguientes características principales:

- Dashboard interactivo con estadísticas en tiempo real
- Gestión completa de inventario con control de stock y alertas
- Sistema de facturación multi-cliente con numeración automática
- Configuración flexible de tipos de impuestos
- Generación automática de facturas en formato PDF
- Control de integridad de datos con restricciones de eliminación
- Interfaz web responsiva y moderna

---

## Pasos de Instalación

## Requisitos del Sistema

- Ruby 3.x o superior
- Rails 8.0.2
- PostgreSQL 12 o superior
- Node.js (para el asset pipeline)
- Git

## Proceso de Instalación

1. **Clonar el repositorio**

```
git clone git@github.com:DanielAR27/Proyecto4-Lenguajes.git
sistema_facturacion
cd sistema_facturacion
```

2. **Instalar las dependencias de Ruby**

```
bundle install
```

3. **Configurar la base de datos**

   Crear un archivo `.env` en la raíz del proyecto con las credenciales de PostgreSQL:

```
POSTGRES_USER=usuario_postgresql
POSTGRES_PASSWORD=password_postgresql
```

4. **Crear y configurar la base de datos**

```
dotenv rails db:create
rails db:migrate
```

5. **Cargar datos iniciales**

```
rails db:seed
```

6. **Iniciar el servidor de desarrollo**

```
rails server
```

7. **Acceder a la aplicación**

   Abrir el navegador web e ir a: `http://localhost:3000`

---

# Manual de Usuario

## Dashboard Principal

Al acceder al sistema, el usuario visualiza un dashboard con estadísticas importantes:

- **Resumen de Productos**: Total de productos, productos activos y alertas de stock bajo
- **Estado de Facturas**: Cantidad total de facturas distribuidas por estado (emitidas, pagadas, anuladas)
- **Métricas de Ventas**: Ventas totales acumuladas y ventas del mes actual
- **Valor del Inventario**: Valor monetario total del stock disponible
- **Top 5 Productos Más Vendidos**: Ranking de productos por cantidad vendida
- **Facturas Recientes**: Listado de las últimas 5 facturas creadas

## Gestión de Productos

### Crear Producto

1. Acceder a la sección "Productos" desde el menú de navegación
2. Hacer clic en "Nuevo Producto"
3. Completar la información requerida:
   - Nombre del producto (obligatorio)
   - Código único (se genera automáticamente si no se especifica)
   - Precio unitario (obligatorio)
   - Stock actual y stock mínimo
   - Descripción y categoría (opcionales)
4. Confirmar la creación

### Consultar y Filtrar Productos

- Utilizar la barra de búsqueda para filtrar por nombre
- Filtrar por categoría usando el selector desplegable
- Navegar entre páginas usando los controles de paginación
- Identificar productos con stock bajo mediante el indicador visual

### Gestión de Stock

- El sistema registra automáticamente el historial de cambios de stock
- Acceder al historial desde la vista detalle de cada producto
- Las alertas de stock bajo se muestran cuando el stock actual es menor o igual al stock mínimo configurado

## Sistema de Facturación

### Crear Nueva Factura

1. Navegar a "Facturas" y seleccionar "Nueva Factura"
2. Completar los datos del cliente:
    - Nombre del cliente (obligatorio)
    - Email del cliente (opcional)
3. Agregar productos a la factura:
    - Seleccionar el producto del listado desplegable
    - Especificar la cantidad deseada
    - Elegir el tipo de impuesto aplicable
    - Visualizar el cálculo automático del total por línea
4. Revisar el resumen con subtotales, impuestos y total general
5. Confirmar la creación de la factura

#### Estados de Factura

Las facturas pueden tener tres estados diferentes:

- **Emitida**: Factura creada y lista para cobro (stock ya descontado)
- **Pagada**: Factura cobrada exitosamente
- **Anulada**: Factura cancelada con stock restaurado automáticamente

#### Acciones sobre Facturas

- **Marcar como Pagada**: Cambiar estado de "Emitida" a "Pagada"
- **Anular Factura**: Cancelar factura y restaurar stock automáticamente
- **Descargar PDF**: Generar documento imprimible de la factura
- **Eliminar**: Solo disponible para facturas en estado "Anulada"

### Configuración de Impuestos

#### Crear Tipo de Impuesto

1. Acceder a la sección "Impuestos"
2. Seleccionar "Nuevo Tipo de Impuesto"
3. Definir:
    - Nombre descriptivo (ejemplo: "IVA", "Exento")
    - Porcentaje en formato decimal (ejemplo: 0.13 para 13%)
    - Estado activo/inactivo
4. Guardar la configuración

#### Gestión de Impuestos

- Los impuestos se pueden activar o desactivar según necesidad
- Solo los impuestos activos aparecen disponibles al crear facturas
- El sistema calcula automáticamente los montos por línea y totales

---

## Arquitectura Lógica Utilizada

### Patrón Modelo-Vista-Controlador (MVC)

El sistema implementa el patrón MVC de Ruby on Rails con la siguiente estructura:

```
sistema_facturacion/
├── app/
│   ├── controllers/          # Controladores de la aplicación
│   │   ├── application_controller.rb
│   │   ├── home_controller.rb      # Dashboard principal
│   │   ├── products_controller.rb  # Gestión de productos
│   │   ├── invoices_controller.rb  # Sistema de facturación
│   │   └── tax_types_controller.rb # Configuración de impuestos
│   ├── models/               # Modelos de datos y lógica de negocio
│   │   ├── product.rb        # Modelo de productos
│   │   ├── invoice.rb        # Modelo de facturas
│   │   ├── invoice_item.rb   # Modelo de líneas de factura
│   │   ├── tax_type.rb       # Modelo de tipos de impuesto
│   │   └── stock_history.rb # Modelo de historial de stock
│   └── views/                # Vistas y plantillas HTML
│       ├── layouts/          # Layout principal de la aplicación
│       ├── home/            # Vista del dashboard
│       ├── products/        # Vistas de gestión de productos
│       ├── invoices/        # Vistas del sistema de facturación
│       └── tax_types/       # Vistas de configuración de impuestos
├── db/
│   ├── migrate/             # Migraciones de base de datos
│   └── seeds.rb            # Datos iniciales del sistema
└── config/
    ├── routes.rb           # Configuración de rutas
    └── database.yml        # Configuración de PostgreSQL
```

### Principios SOLID Implementados

- **Single Responsibility Principle**: Cada modelo tiene una responsabilidad específica y bien definida
- **Open/Closed Principle**: Los modelos son extensibles sin modificar código existente
- **Liskov Substitution Principle**: Las subclases mantienen el comportamiento esperado
- **Interface Segregation Principle**: Interfaces específicas para diferentes tipos de operaciones
- **Dependency Inversion Principle**: Dependencias abstraídas mediante el framework de Rails

### Tecnologías y Herramientas

- **Framework**: Ruby on Rails 8.0.2
- **Base de Datos**: PostgreSQL con Active Record ORM
- **Frontend**: Bootstrap 5 para diseño responsivo
- **Iconografía**: Font Awesome para iconos
- **Generación PDF**: Prawn y Prawn-Table
- **Paginación**: Implementación manual personalizada

---

## Funcionamiento

### Flujo Principal del Sistema

1. **Acceso Inicial**: El usuario accede al dashboard principal con estadísticas generales del sistema
2. **Gestión de Productos**:
    - Creación y mantenimiento del catálogo de productos
    - Control automático de stock con historial de cambios
    - Alertas visuales para productos con stock bajo
3. **Configuración de Impuestos**:
    - Definición de tipos de impuestos con porcentajes configurables
    - Activación/desactivación según necesidades operativas
4. **Proceso de Facturación**:
    - Selección de cliente y productos
    - Cálculo automático de impuestos por línea
    - Generación de factura con numeración secuencial
    - Descuento automático de stock del inventario
5. **Gestión Post-Facturación**:
    - Cambio de estados de factura (emitida → pagada)
    - Anulación con restauración automática de stock
    - Generación de documentos PDF para impresión

## Sistema de Integridad de Datos

### Protección contra Eliminación Accidental

El sistema implementa restricciones de integridad que previenen la eliminación de registros críticos:

- **Productos en Facturas**: No se pueden eliminar productos que han sido incluidos en facturas existentes
- **Tipos de Impuesto en Uso**: No se pueden eliminar tipos de impuestos que están siendo utilizados en facturas
- **Facturas Activas**: Solo se pueden eliminar facturas que previamente han sido anuladas

### Gestión de Stock

- **Descuento Automático**: Al crear una factura, el stock se descuenta automáticamente
- **Validación de Disponibilidad**: El sistema verifica stock disponible antes de permitir la facturación
- **Restauración por Anulación**: Al anular una factura, el stock se restaura automáticamente
- **Historial Completo**: Todos los cambios de stock se registran con fecha y hora

## Base de Datos PostgreSQL

El sistema utiliza PostgreSQL como motor de base de datos, proporcionando:

- **Transacciones ACID**: Garantía de consistencia en operaciones críticas
- **Relaciones Referencial**: Integridad referencial entre entidades
- **Índices Optimizados**: Búsquedas eficientes en tablas principales

- **Restricciones de Datos**: Validaciones a nivel de base de datos

## Consideraciones de Seguridad y Operación

- **Validación de Datos**: Validaciones tanto en frontend como backend
- **Prevención de Pérdida de Datos**: Confirmaciones para operaciones destructivas
- **Auditabilidad**: Registro de cambios críticos como movimientos de stock
- **Recuperación de Errores**: Manejo de excepciones con mensajes informativos para el usuario

Este sistema proporciona una solución completa y robusta para la gestión empresarial, manteniendo la integridad de los datos mientras ofrece una experiencia de usuario intuitiva y eficiente.