🔧 Stockly - Manual Técnico v1.3.0 - ACTUALIZADO

Última Actualización: 23 de Octubre, 2025
Versión: 1.3.0 - Completamente Actualizado
Estado: ☑ Producción - Documentación Completa

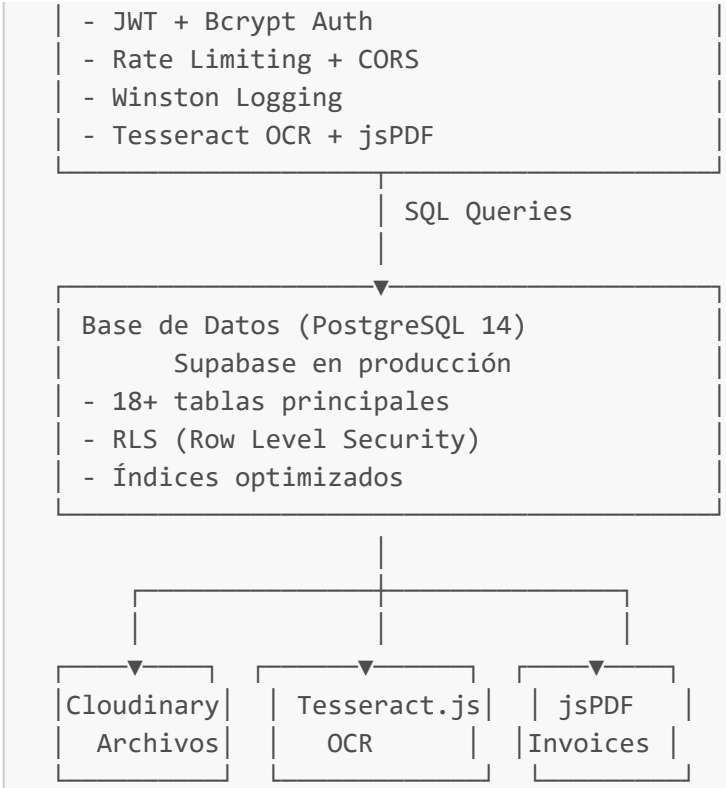
📑 Tabla de Contenidos

📋 Descripción General

Stockly v1.3.0 es un SaaS multi-tenant completo, listo para producción, con:

Stack Backend: Node.js 20 + Express 4.19 + PostgreSQL 14
Stack Frontend: React 19 + TypeScript 5.6 + Vite 5.4 + TailwindCSS 4.1
Características: Invoices PDF, OCR, Profit tracking, Garantías, Reportes, RBAC (4 roles)

🏗 Arquitectura del Sistema



Stack Tecnológico Completo

Backend

Layer	Componente	Tecnología	Versión
Runtime	Node.js	Node.js	20+
Framework	HTTP Server	Express.js	4.19.2
BD - RDBMS	Database	PostgreSQL	14+
BD - Hosting	Cloud DB	Supabase	Latest
BD - Driver	SQL Client	node-postgres (pg)	8.16.3
Auth	JWT	jsonwebtoken	9.0.2
Auth	Password Hash	bcryptjs	3.0.2
Validación	Schemas	Zod	3.23.8
OCR	Text Extraction	Tesseract.js	5.1.1
PDF	Generation	jsPDF	3.0.3
PDF	Tables	jsPDF-autoTable	5.0.2
Excel	Export	ExcelJS	4.4.0
Storage	CDN	Cloudinary	2.7.0

Layer	Componente	Tecnología	Versión
Images	Processing	Sharp	0.34.4
Logging	Logger	Winston	3.15.0
Rate Limit	Protection	express-rate-limit	7.4.1
Security	HTTP Headers	Helmet	8.0.0
CORS	Cross Origin	cors	2.8.5
Upload	File Handling	Multer	2.0 RC4
Scheduler	Tasks	node-cron	3.0.3
ENV	Config	dotenv	16.4.5

Frontend

Layer	Componente	Tecnología	Versión
UI Framework	Library	React	19.1.1
Language	TypeScript	TypeScript	5.6+
Build Tool	Bundler	Vite	5.4+
Styling	Framework	TailwindCSS	4.1.15
Styling	CSS Processor	PostCSS	8.5.6
HTTP	Client	Axios	1.12.2
Routing	SPA Router	React Router	v7.9.4
State	Global State	Redux Toolkit	2.9.1
State	Redux React	react-redux	9.2.0
Data Fetch	Query Client	React Query	5.90.5
Forms	Form Library	React Hook Form	7.65.0
Validation	Schemas	Zod	4.1.12
Charts	Visualization	Recharts	3.3.0
Charts Alt	Charts Library	Chart.js	4.5.1
Charts 2	React Charts	react-chartjs-2	5.3.0
UI	Notifications	React Hot Toast	2.6.0
OCR	Browser OCR	Tesseract.js	6.0.1
Camera	Webcam	react-webcam	7.2.0
PDF	Reader	react-pdf	10.2.0

Layer	Componente	Tecnología	Versión
Upload	Drag & Drop	react-dropzone	14.3.8
Icons	Icon Set	Lucide React	0.546
Icons 2	More Icons	React Icons	5.5.0
Icons 3	Heroicons	@heroicons/react	2.2.0
Components	Headless UI	@headlessui/react	2.2.9
Animation	Motion Library	Framer Motion	12.23.24
JWT	Decode	jwt-decode	4.0.0
CSS	Autoprefixer	autoprefixer	10.4.21
Forms	Styling	@tailwindcss/forms	0.5.10

Estructura de Backend

Archivos Backend

```
Backend/
├── src/
│   ├── server.js                    # Clase StocklyServer + setup
│   │
│   ├── config/                      # Configuraciones
│   │   ├── database.js              # Pool PostgreSQL
│   │   ├── logger.js                # Winston logger
│   │   ├── cloudinary.js            # Cloudinary API
│   │   ├── tesseract.js             # Tesseract OCR
│   │   └── jwt.js                   # JWT tokens
│   │
│   ├── controllers/                 # Lógica de negocio (14 archivos)
│   │   ├── auth.controller.js       # Register, login, refresh, logout, change-
password
│   │   ├── user.controller.js       # Profile, companies, switch company, search
│   │   ├── company.controller.js    # CRUD, members, roles, logo upload
│   │   ├── invitation.controller.js # Create, validate, accept invitation
│   │   ├── category.controller.js   # CRUD jerárquico, tree view
│   │   ├── product.controller.js    # CRUD, stock, search, filters
│   │   ├── productAttribute.controller.js # Dynamic attributes
│   │   ├── purchase.controller.js   # CRUD, profit calculation, stats
│   │   ├── sale.controller.js       # CRUD, OCR processing, warranties
│   │   ├── invoice.controller.js    # CRUD, PDF generation, finalize
│   │   ├── warranty.controller.js   # CRUD, expiration, search
│   │   ├── serviceHistory.controller.js # Technical service history
│   │   ├── supplier.controller.js   # CRUD suppliers
│   │   └── report.controller.js     # Reports, analytics, export
│   │
│   ├── models/                      # Data access layer
```

```
│   │   ├── user.model.js
│   │   ├── company.model.js
│   │   ├── product.model.js
│   │   ├── category.model.js
│   │   ├── sale.model.js
│   │   ├── purchase.model.js
│   │   ├── invoice.model.js
│   │   ├── warranty.model.js
│   │   ├── serviceHistory.model.js
│   │   ├── supplier.model.js
│   │   └── [otros]
│   │
│   ├── routes/                      # HTTP routes (14 routers)
│   │   ├── auth.routes.js
│   │   ├── user.routes.js
│   │   ├── company.routes.js
│   │   ├── invitation.routes.js
│   │   ├── category.routes.js
│   │   ├── product.routes.js
│   │   ├── sale.routes.js
│   │   ├── purchase.routes.js
│   │   ├── invoice.routes.js
│   │   ├── warranty.routes.js
│   │   ├── service.routes.js
│   │   ├── supplier.routes.js
│   │   ├── report.routes.js
│   │   ├── health.routes.js
│   │   └── index.js               # Agrupa todas
│   │
│   ├── middlewares/                 # Express middleware
│   │   ├── auth.middleware.js      # JWT validation
│   │   ├── error.middleware.js     # Error handling
│   │   ├── role.middleware.js      # Role-based access
│   │   ├── validation.middleware.js # Zod validation
│   │   └── request.middleware.js   # Request logging
│   │
│   ├── services/                    # Business logic
│   │   ├── cloudinaryStorage.service.js # Cloudinary uploads
│   │   ├── firebaseStorage.service.js  # Firebase alternative
│   │   ├── ocr.service.js             # Tesseract OCR
│   │   └── report.service.js          # Report generation
│   │
│   ├── utils/                       # Utilities
│   │   ├── formatters.js           # Data formatting
│   │   ├── validators.js           # Zod schemas
│   │   ├── errorHandler.js         # Error handling
│   │   └── responseHandler.js      # Response standardization
│   │
│   └── validations/                 # Zod schemas
│       ├── auth.validation.js
│       ├── product.validation.js
│       ├── invoice.validation.js
│       └── [otros]
│
```

```
├── database/
│   └── init.sql                    # Schema initial
│
├── migrations/                      # Version control BD
│   ├── add-invoices-system.sql
│   ├── add-product-improvements.sql
│   ├── add-purchase-profit-tracking.sql
│   └── [otros]
│
├── docs/
│   ├── API_REFERENCE.md
│   ├── ARCHITECTURE.md
│   ├── SETUP.md
│   └── [otros]
│
├── .env.example
├── .env                            # Variables (NO en git)
├── package.json
├── package-lock.json
├── Dockerfile
├── Procfile                        # Para Railway/Heroku
├── railway.json
└── README.md
```

Controladores Backend

Controller	Responsabilidad	Métodos Principales
AuthController	Autenticación	register, login, refresh, logout, changePassword
UserController	Usuarios	profile, companies, switchCompany, search
CompanyController	Empresas	getCrud, members, roles, logo, fiscal
InvitationController	Invitaciones	create, validate, accept, getByCode
CategoryController	Categorías	getCrud, tree, hierarchy
ProductController	Productos	getCrud, stock, attributes, filters
PurchaseController	Compras	getCrud, profit, stats
SaleController	Ventas	getCrud, OCR, warranty
InvoiceController	Invoices	getCrud, PDF, finalize, stats
WarrantyController	Garantías	getCrud, expiration, service
ServiceHistoryController	Servicio	getCrud, history
SupplierController	Proveedores	getCrud
ReportController	Reportes	costVsRevenue, profit, stats

🗀 Estructura de Frontend

Archivos Frontend

```
Frontend/src/
├── App.tsx                          # Root app with routes
├── main.tsx                         # React entry point
├── App.css & index.css             # Global styles
│
├── pages/                           # Page components
│   ├── Auth/
│   │   ├── Login.tsx               # Login form
│   │   └── Register.tsx            # Registration form
│   ├── Dashboard.tsx               # KPIs dashboard
│   ├── Inventory/
│   │   ├── ProductList.tsx         # Product listing
│   │   ├── ProductDetail.tsx       # Product view
│   │   └── ProductAdd.tsx          # Create/Edit
│   ├── Sales/
│   │   ├── SalesList.tsx           # Sales history
│   │   └── NewSale.tsx             # Create sale + OCR
│   ├── Purchases/
│   │   ├── PurchaseList.tsx        # Purchases history
│   │   └── NewPurchase.tsx         # Create + profit
│   ├── Invoices/
│   │   ├── InvoiceList.tsx         # Invoice listing
│   │   └── CreateInvoice.tsx       # Create + PDF
│   ├── Services/
│   │   └── ServiceBoard.tsx        # Service history
│   ├── Warranties/
│   │   └── WarrantyList.tsx        # Warranties list
│   ├── Suppliers/
│   │   └── SupplierList.tsx        # Suppliers CRUD
│   ├── Users/
│   │   └── UserList.tsx            # Users CRUD
│   ├── Invitations/
│   │   └── InvitationList.tsx      # Pending invitations
│   ├── LoginPage.tsx               # Alternative login
│   └── Settings/
│       └── CompanySettings.tsx     # Company config
│
├── components/                      # Reusable components
│   ├── Header.tsx                  # Top navbar
│   ├── Sidebar.tsx                 # Left navigation
│   ├── ProtectedRoute.tsx          # Auth guard
│   ├── SerialNumberInput.tsx       # OCR serial input
│   ├── OCRRegionSelector.tsx       # OCR region select
│   ├── ImageCropper.tsx            # Image cropping
│   ├── PdfPreview.tsx              # PDF preview
│   ├── LoadingSpinner.tsx          # Spinner
│   ├── Modal.tsx                   # Modal dialog
│   ├── Button.tsx                  # Button component
│   ├── FormInput.tsx               # Form input
│   ├── Table.tsx                   # Generic table
```

```
│   ├── Pagination.tsx              # Pagination
│   ├── NoData.tsx                  # Empty state
│   └── index.ts                    # Exports
│
├── services/                        # API services (Axios)
│   ├── api.ts                      # Axios base config
│   ├── authService.ts              # Auth endpoints
│   ├── productService.ts           # Products CRUD
│   ├── categoryService.ts          # Categories CRUD
│   ├── saleService.ts              # Sales CRUD
│   ├── purchaseService.ts          # Purchases CRUD
│   ├── invoiceService.ts           # Invoices CRUD + PDF
│   ├── warrantyService.ts          # Warranties CRUD
│   ├── serviceService.ts           # Service history
│   ├── supplierService.ts          # Suppliers CRUD
│   ├── reportService.ts            # Reports
│   ├── companyService.ts           # Company CRUD
│   ├── userService.ts              # Users CRUD
│   └── invitationService.ts        # Invitations
│
├── context/                         # React Context
│   ├── ThemeContext.tsx            # Dark/Light mode
│   ├── AuthContext.tsx             # Auth state
│   └── CompanyContext.tsx          # Current company
│
├── store/                           # Redux Toolkit
│   ├── index.ts                    # Store config
│   ├── authSlice.ts                # Auth reducer
│   ├── companySlice.ts             # Company reducer
│   └── appSlice.ts                 # App general
│
├── hooks/                           # Custom hooks
│   ├── useAuth.ts                  # Auth hook
│   ├── useApi.ts                   # API hook
│   ├── usePagination.ts            # Pagination
│   ├── useLocalStorage.ts          # LocalStorage
│   ├── useDebounce.ts              # Debounce
│   └── useNotification.ts          # Notifications
│
├── types/                           # TypeScript types
│   ├── index.ts
│   ├── api.ts                      # API responses
│   ├── models.ts                   # Domain models
│   ├── forms.ts                    # Form types
│   └── enums.ts                    # Enumerations
│
├── utils/                           # Utilities
│   ├── formatters.ts               # Data formatting
│   ├── validators.ts               # Validations
│   ├── helpers.ts                  # Helper functions
│   ├── dateUtils.ts                # Date utilities
│   ├── numberUtils.ts              # Number utilities
│   └── pdfUtils.ts                 # PDF utilities
│
```

```
├── layouts/                          # Layout components
│   ├── AuthenticatedLayout.tsx      # Authenticated layout
│   ├── GuestLayout.tsx              # Guest layout
│   └── MainLayout.tsx               # Main layout
│
├── config/                           # Configurations
│   ├── api.config.ts                # Axios config
│   ├── routes.config.ts             # Routes mapping
│   └── constants.ts                 # App constants
│
└── assets/                           # Static resources
    ├── images/
    ├── icons/
    └── [otros]

[Root Config Files]
├── vite.config.ts                   # Vite: port 3000 (puede cambiar)
├── tsconfig.json
├── tsconfig.app.json
├── tsconfig.node.json
├── tailwind.config.js               # TailwindCSS
├── eslint.config.js
├── postcss.config.js
├── .env.example
├── index.html
├── nginx.conf                        # For containerization
├── Dockerfile
├── Procfile
├── railway.json
├── package.json
└── README.md
```

Base de Datos - 18+ Tablas

Modelo ER

```
┌──────────────────────────────────────────────┐
│        AUTENTICACIÓN Y MULTI-TENANT          │
├──────────────────────────────────────────────┤

users (PK: id)
├── id (UUID)
├── email (UNIQUE)
├── password_hash
├── name
├── phone
├── created_at
└── updated_at

companies (PK: id)
```

```
├── id (UUID)
├── name
├── email
├── phone
├── logo_url (Cloudinary)
├── fiscal_data (JSON)
├── created_by (FK → users.id)
└── created_at

user_company (PK: user_id, company_id)
├── user_id (FK → users.id)
├── company_id (FK → companies.id)
├── role (owner|admin|seller|inventory)
└── joined_at

┌─────────────────────────────────────────┐
│          INVENTARIO Y PRODUCTOS         │
├─────────────────────────────────────────┤

categories (PK: id)
├── id (UUID)
├── company_id (FK → companies.id)
├── name
├── description
├── parent_category_id (FK → categories.id) [Jerarquía]
├── created_at
└── updated_at

products (PK: id)
├── id (UUID)
├── company_id (FK → companies.id)
├── name
├── sku (UNIQUE per company)
├── category_id (FK → categories.id)
├── description
├── price_buy
├── price_sell
├── created_at
└── updated_at

product_attributes (PK: id)
├── id (UUID)
├── product_id (FK → products.id)
├── attribute_name
├── attribute_value
└── created_at

product_stock (PK: id)
├── id (UUID)
├── product_id (FK → products.id)
├── state (new|used|open_box)
├── quantity
├── created_at
└── updated_at
```

```
┌─────────────────────────────────────────┐
│          COMPRAS Y PROVEEDORES          │
├─────────────────────────────────────────┤

suppliers (PK: id)
├── id (UUID)
├── company_id (FK → companies.id)
├── name
├── email
├── phone
├── address
└── created_at

purchases (PK: id)
├── id (UUID)
├── company_id (FK → companies.id)
├── supplier_id (FK → suppliers.id)
├── purchase_date
├── invoice_number
├── shipping_cost
├── discount
├── tax
├── total_cost
└── created_at

purchase_items (PK: id)
├── id (UUID)
├── purchase_id (FK → purchases.id)
├── product_id (FK → products.id)
├── quantity
├── unit_price
├── state (new|used|open_box)
├── profit_potential (calculated)
└── created_at

┌─────────────────────────────────────────┐
│          VENTAS E INVOICES              │
├─────────────────────────────────────────┤

sales (PK: id)
├── id (UUID)
├── company_id (FK → companies.id)
├── customer_name
├── customer_email
├── customer_phone
├── sale_date
├── total_amount
└── created_at

sale_items (PK: id)
├── id (UUID)
├── sale_id (FK → sales.id)
├── product_id (FK → products.id)
```

```
├── quantity
├── unit_price
├── state (new|used|open_box)
├── serial_number (OCR extracted)
└── created_at

invoices (PK: id)
├── id (UUID)
├── company_id (FK → companies.id)
├── sale_id (FK → sales.id, optional)
├── invoice_number (INV-YYYY-00001)
├── customer_name
├── customer_email
├── customer_address
├── subtotal
├── tax_amount
├── total_amount
├── status (draft|pending|paid|cancelled)
├── pdf_url (Cloudinary)
└── created_at

invoice_line_items (PK: id)
├── id (UUID)
├── invoice_id (FK → invoices.id)
├── product_id (FK → products.id, optional)
├── description
├── quantity
├── unit_price
├── line_total
├── item_type (product|shipping|fee|discount)
└── created_at

┌─────────────────────────────────────────┐
│       GARANTÍAS Y SERVICIO TÉCNICO      │
├─────────────────────────────────────────┤

warranties (PK: id)
├── id (UUID)
├── company_id (FK → companies.id)
├── sale_id (FK → sales.id)
├── product_id (FK → products.id)
├── customer_name
├── start_date
├── end_date
├── status (active|expired|claimed)
└── created_at

service_histories (PK: id)
├── id (UUID)
├── warranty_id (FK → warranties.id)
├── issue_description
├── solution_applied
├── technician_name
├── time_spent_hours
```

```
├── service_date
└── created_at

┌─────────────────────────────────────────┐
│          INVITACIONES Y TOKENS          │
├─────────────────────────────────────────┤

invitations (PK: id)
├── id (UUID)
├── company_id (FK → companies.id)
├── code (UNIQUE)
├── created_by (FK → users.id)
├── expires_at
├── used_at
└── created_at

refresh_tokens (PK: id)
├── id (UUID)
├── user_id (FK → users.id)
├── token (hash)
├── expires_at
└── created_at
```

Características:

- ☑ RLS (Row Level Security) - Multi-tenant automático
- ☑ Índices en SKU, email, company_id para queries rápidas
- ☑ Foreign Keys con integridad referencial
- ☑ Triggers para auditoría (created_at, updated_at)

⚙ Instalación y Configuración

Continúa en la próxima sección...

Versión: 1.3.0 | **Estado:** ☑ Actualizado | **Fecha:** 23 Octubre 2025