# Sistema de creación y seguimiento de cotizaciones BI-Tecnología

Análisis y especificación

*Última edición: Oct 13, 2025*

## 0) Propósito y alcance del proyecto

Diseñar, planear y desarrollar un sistema web que permita a los integrantes de **BI-Tecnología** crear, enviar y dar seguimiento a cotizaciones de clientes de forma segura, estructurada y eficiente. El sistema debe garantizar la calidad y consistencia de la información entregada al cliente, además de proporcionar herramientas de seguimiento y reportes para evaluar el desempeño del equipo de ventas.

El objetivo principal es contar con una herramienta **fácil de usar**, que **agilice el proceso comercial** sin comprometer la calidad de la información, y que apoye directamente el **cierre de ventas** mediante una mejor trazabilidad y control de cotizaciones.

**Alcance (Minimum Viable Product)**:

El sistema incluirá las siguientes funciones principales:

*Usuarios*

* Inicio de sesión de usuarios obligatorio.
* Control de permisos basado en roles (administrador, gerente y vendedor).

*Clientes*

* Módulo de clientes para la creación y/o selección del cliente a cotizar.
* Se podrá dar de alta y registrar clientes desde el frontend. Cualquier usuario puede crear un cliente.
* Un cliente debe tener al menos un contacto.

*Productos*

* Catálogo de productos con categorías (ej: plotters, consumibles, papel, etc.).
* Asociación de **productos relacionados** que se agregan automáticamente junto con el producto principal (por ejemplo, al cotizar un plotter se añaden sus consumibles), con la opción de excluirlos manualmente.
* Módulo de **servicios** complementarios, como instalación y envío.
* Descuentos por línea de cotización
* Descuentos mayores al **10 %** requerirán autorización de un **Gerente**.
* Envío automático de **documentos relacionados** (fichas técnicas, catálogos) junto con la cotización.
* Generación de **documentos PDF** con formato profesional y datos de la empresa.
* Envío de cotizaciones por **correo electrónico** directamente desde el sistema.
* Seguimiento del ciclo de vida de la cotización con los estados:  
  **Borrador → Pendiente por Abrobar →Enviada → (Aceptada – Rechazada – Expirada) → Cerrada**
* Módulo de **reportes** con filtros por rango de fechas, vendedor, cliente y estado (abiertas, cerradas, perdidas).
* Uso de una plantilla visual profesional consistente con la identidad de BI-Tecnología.

**Supuestos**

* Mercado: Nacional (México).
* Moneda MXN con IVA 16%.
* Las cotizaciones mostrarán precios **antes de IVA**, desglosando el impuesto en el total.
* Python 3.13; Django 5.2

## 1) Roles y permisos

**Roles** (define nombres exactos):

* **Vendedor**: Puede crear y editar sus propias cotizaciones, ver cotizaciones abiertas y cerradas, enviar cotizaciones por correo, solicitar aprobación para descuentos especiales, y crear o editar clientes.
* **CSR**: Posee todas las funciones del Vendedor, además de poder editar cotizaciones de otros vendedores con el fin de responder de inmediato al cliente en caso de ausencia del responsable. También puede crear o editar clientes.
* **Gerente**: Incluye todas las funciones del Vendedor, con la capacidad adicional de **aprobar cotizaciones con descuentos especiales** y consultar un listado de cotizaciones pendientes de aprobación.
* **Admin**: Tiene privilegios de configuración: alta, edición y eliminación de productos, usuarios, categorías, y parámetros globales del sistema.

Permisos clave

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Acción | Vendedor | CSR | Gerente | Admin |
| Crear cliente |  | X |  |  |
| Editar cliente |  | X |  |  |
| Crear cotización | X | X | X |  |
| Editar cotización propia | X | X | X |  |
| Editar cotización del equipo |  | X | X |  |
| Enviar a aprobación | X | X |  |  |
| Abrobación de cotización |  |  | X |  |
| Enviar cotización por correo | X | X | X |  |
| Cancelar cotización propia | X |  |  |  |
| Cancelar cotización del equipo |  | X | X |  |
| Gestionar catálogo de productos |  |  |  | X |
| Ver reportes propios | X |  |  |  |
| Cambiar status de cotización | X | X | X |  |
| Ver reportes de toda la organización |  |  | X |  |
| Ver reportes de equipo |  | X |  |  |

## **Reglas de aprobación:**

## **Umbral de descuento:** Descuentos mayores al 10 % deben ser aprobados por un Gerente.

## **Umbral de monto:** Podrá definirse en una futura fase según políticas internas.

## **Autorización propia:** Un vendedor no puede aprobar sus propias cotizaciones.

## **Registro:** Todas las aprobaciones y rechazos quedarán registradas en el historial de cambios con usuario, fecha y comentarios.

## 2) Flujos Principales

### 2.1 Autenticación y Acceso

1. El usuario inicia sesión con **correo/nombre de usuario y contraseña**.
2. El **rol** asignado determina el alcance de funciones y datos disponibles (ver 1) Roles y permisos).

### 2.2 Captura/Selección de Cliente

1. Búsqueda de cliente por razón social, nombre de contacto o RFC
2. **Creación/edición** de cliente con campos mínimos:
   * Razón social
   * RFC
3. **Contactos** (un cliente puede tener varios):
   * Nombre
   * Teléfono
   * Extensión (opcional)
   * Correo

### 2.3 Productos

1. Categorías de productos.
2. Productos (obligatorio que tengan una categoría asignada).
   * Número de parte
   * Nombre
   * Descripción
   * Garantía
   * Precio de lista
   * Pecio venta
3. Productos relacionados (ej. consumibles de un equipo)
4. Productos de servicio (pólizas de mantenimiento, garantía extendida, servicio técnico, etc.).
5. Servicios complementarios (ej. envío, instalación, capacitación, etc.).

### 2.4 Creación de Cotización

1. Iniciar **borrador de cotización**.
2. Seleccionar **cliente** y **contacto**.
3. Agregar **líneas de producto/servicio** (producto + cantidad).
4. Calcular **automáticamente** el **subtotal por línea**.
5. (Opcional por línea) **Tiempo de entrega** y **garantía**.
6. Calcular automáticamente **subtotal, IVA** y **total** de la cotización.
7. Si el producto tiene **relacionados**, agregarlos automáticamente en una **sección separada**, con opción de excluirlos.
8. Indicar descuento global
9. Indicar **términos de pago** y **vigencia** de la cotización.
10. Adjuntar **archivos** (fichas técnicas, catálogos, etc.).
11. **Guardar** cotización.
12. Si el **descuento > 10 %**, **enviar a aprobación** (flujo de Gerente).
13. **Previsualizar PDF**.
14. **Aprobación/Rechazo** por Gerente (cuando aplique).
15. Guardar cotización.
16. **Enviar cotización por correo** al contacto.

### 2.5 Estatus de cotización

Borrador → Pendiente de aprobación → Aprobada → Enviada → (Aceptada | Rechazada | Vencida) → Cerrada

* Cada transición registra **usuario, fecha y comentario** en Historial.
* No se puede editar una cotización después de enviada. Se debe de realizar nueva cotización para agregar cambios.

## 3) Modelo de Datos (Conceptual)

**Entidades y sus campos**

**App:** users

**Comentario:** se va a utilizar el **User** model default de Django y se va a extender con el modelo **Profile** con el cual va a existir una relación one-to-one

**Model name:** Profile

class Roles(models.TextChoices):

SALES = "S", "Vendedor"

CSR = "C", "CSR (Servicio a Clientes)"

MANAGER = "M", "Gerente"

ADMIN = "A", "Administrador"

|  |  |  |  |
| --- | --- | --- | --- |
| Field name | type | Related table | Comment |
| user | OneToOneField | settings.AUTH\_USER\_MODEL |  |
| role | Charfield(1) |  | choices(roles) |
| phone | Charfield(10) |  | validator |
| position\_title | Charfield(25) |  |  |

**App:** customers

**Comentario:** Esta aplicación administrará todo lo relacionado con clientes. Un cliente puede tener varios contactos

**Model name:** Customers

|  |  |  |  |
| --- | --- | --- | --- |
| Field name | type | Related table | Comment |
| name | Charfield(50) |  | Nombre del cliente |
| rfc | Charfield(13) |  | Opcional, validador |
| created | DateTimeField |  | auto\_now\_add=True |
| updated | DateTimeField |  | auto\_now=True |
| sales\_rep | ForeignKeyField | users.User |  |

**Model name:** Contacts

|  |  |  |  |
| --- | --- | --- | --- |
| Field name | type | Related model | Comment |
| name | Charfield(50) |  | Nombre del cliente |
| phone | Charfield(10) |  | Opcional, validador |
| extension | Charfield(5) |  | Opcional, validador |
| email | EmailField |  | blank and null=False |
| customer | ForeignKeyField | Customer |  |
| created | DateTimeField |  | auto\_now\_add |
| updated | DateTimeField |  | auto\_now |

**App:** catalog

**Comentario:** La información de los productos se contiene en esta aplicación. Tendremos el modelo principal del producto, otro para productos relacionados, uno más para documentos relacionados

**Model name:** Product

|  |  |  |  |
| --- | --- | --- | --- |
| Field name | type | Related model | Comment |
| sku | Charfield(10) |  | número de parte |
| name | Charfield(100) |  |  |
| description | TextField() |  |  |
| list\_price | FloatField |  |  |
| sale\_price | FloatField |  |  |
| created | DateTimeField |  | auto\_now\_add |
| updated | DateTimeField |  | auto\_now |

**User**: id, name, email, role, is\_active, team, last\_login. - **Customer**: id, business\_name, rfc, trade\_name, type (Prospect/Client), default\_terms, owner\_user, tags, created\_at. - **Address**: id, customer\_id, type (billing/shipping), street, ext\_no, int\_no, neighborhood, city, state, country, postal\_code. - **Contact**: id, customer\_id, name, email, phone, position, is\_primary. - **Product**: id, sku, name, category, unit (ea, box), base\_price, currency, tax\_rate\_id, active, attributes (e.g., width, color), brand. - **Service**: id, code, name, base\_price, pricing\_type (fixed/hourly), default\_tax\_rate. - **TaxRate**: id, name (IVA 16%), rate, region (MX), active\_from/active\_to. - **PriceList**: id, name, currency, priority; **PriceListItem**: product\_id/service\_id, unit\_price, start/end. - **Quote**: id, number (human-friendly), version, customer\_id, contact\_id, owner\_user\_id, status, subtotal, discount\_total, tax\_total, total, currency, validity\_days, payment\_terms, delivery\_estimate, notes\_internal, notes\_customer, expires\_at. - **QuoteLine**: id, quote\_id, line\_no, item\_type (product/service), item\_id, description, qty, unit, unit\_price, discount\_pct, tax\_rate, line\_subtotal, line\_tax, line\_total. - **Attachment**: id, quote\_id, file\_path, filename, mime\_type, size. - **EmailLog**: id, quote\_id, to, cc, bcc, subject, sent\_by\_user\_id, sent\_at, message\_id, status. - **StatusHistory**: id, quote\_id, from\_status, to\_status, changed\_by\_user\_id, reason, changed\_at. - **AuditLog**: id, actor\_user\_id, entity\_type, entity\_id, action, payload, created\_at. - **ReportSnapshot** (optional for performance): date, metrics json.

**Relationships** - Customer 1–N Addresses, 1–N Contacts, 1–N Quotes. - Quote 1–N QuoteLines; Quote 1–N Attachments; Quote 1–N StatusHistory. - Product/Service prices can be overridden by PriceList or by per-line negotiated price (captured as unit\_price on line).

**Identifiers** - Sequential human-readable **Quote Number**: e.g., BIT-2025-001234 with reset rule per year.

## 4) Business Rules

* **Currency**: default MXN; store amounts as decimals with currency; ready to add USD later.
* **Taxes**: calculate per line; show tax breakdown in PDF (e.g., IVA 16%).
* **Discounts**: per-line and per-quote; max discount bounded by role and approval rules.
* **Validity**: expires\_at = created\_at + validity\_days.
* **Rounding**: define rounding mode (bankers vs commercial); apply consistently at line & totals.
* **Cloning**: allow duplicating quotes (new id/number, reset status to Draft).
* **Versions**: if revising a sent quote, create a new version linked to the original (e.g., …-V2).
* **Soft Deletes**: archival instead of hard delete; retain audit trail.

## 5) PDF & Email Requirements

**PDF** - Company branding (logo, colors), header with quote number/date, customer details. - Line table: item code, description, qty, unit, unit price, discount, tax, total. - Totals section: subtotal, discounts, taxes, grand total; currency label. - Terms & Conditions, validity, delivery, warranty, signatures (optional). - Footer: legal info, contact details. - QR code linking to secure **Quote Viewer** page.

**Email** - Templates with variables (customer name, quote number, total, expiration date). - Attach PDF; include secure link to online quote view. - BCC a central mailbox (e.g., cotizaciones@bit.mx). - Track **delivery status** (best-effort) and **open events** (pixel), if provider supports.

## 6) Reporting & Analytics (MVP)

* **Pipeline**: count and total by status (Draft/Pending/Approved/Sent/Accepted/Rejected/Expired) per user/team.
* **Quote Aging**: days since created and since sent; identify stale quotes.
* **Win Rate**: Accepted vs Sent count & value.
* **Top Products/Services**: by quoted value/qty.
* **Discount Analysis**: average discount by rep/team.
* **Conversion by Customer**: new vs repeat customers.
* Export CSV for all reports.

Future: margin analysis (requires cost data), targets vs actuals, cohort analysis.

## 7) Non‑Functional Requirements

**Security** - HTTPS-only, secure session cookies, CSRF protection, rate limiting on auth. - Password policy; optional 2FA for privileged roles. - Field-level encryption for sensitive data (e.g., RFC, phone/email hashes for search?). - Access logs and **AuditLog** for critical actions.

**Performance & Scalability** - Typical user count: ~5–20 concurrent; quotes up to hundreds of lines (rare). - DB indexing on customers, quotes, status, owner\_user\_id, created\_at.

**Reliability** - Nightly backups; point-in-time recovery goal (if using Postgres WAL). - Error monitoring and alerting (Sentry-like).

**Usability** - Responsive UI; keyboard-first line editing; autosave on drafts. - Fast search (customer, product) with typeahead.

**Internationalization** - UI in Spanish; currency/number formatting for MX.

**Compliance** - Basic data retention policy; ability to export/delete a customer on request.

## 8) Integrations (Now vs Later)

**Now (MVP)** - Email: choose provider (SMTP/Google Workspace vs SendGrid/SES). - File storage: S3-compatible for PDFs and attachments.

**Later** - CRM sync (HubSpot/Pipedrive) or simple CSV import/export. - ERP/accounting (costs, inventory) to enable margin & availability. - Shipping quotes (DHL/Estafeta) via API; tax service for non-standard items.

## 9) Admin Back Office

* Manage users & roles.
* Manage products/services, categories, price lists, tax rates.
* Global settings: company info (logo, colors), default terms, email templates, approval thresholds, currency.
* Numbering scheme configuration.

## 10) UI Sketch (Functional Outline)

* **Top Nav**: Quotes · Customers · Products · Reports · Admin.
* **Quotes List**: filters (status, owner, customer, date range); bulk actions (export, assign).
* **Quote Editor**: tabs: Details · Line Items · Terms · Attachments · Preview · History.
* **Customer Profile**: overview, contacts, addresses, quotes, notes.
* **Product Catalog**: list, categories, quick edit.
* **Reports**: cards + tables; export.

## 11) Testing Strategy & Acceptance Criteria

**Testing** - Unit tests: pricing, taxes, discounts, rounding, validity/expiration. - Integration tests: quote create → approve → send flow; email + PDF generated. - E2E (Playwright): happy paths for each role. - Security tests: auth, RBAC, CSRF, rate limiting.

**Acceptance (MVP must)** 1. A Sales Rep can create a draft quote with products & services and save it. 2. Approval is enforced when thresholds are exceeded; Manager can approve/reject. 3. Approved quote can be emailed with a properly formatted PDF attached. 4. Quotes are searchable by customer, status, date, owner; pagination works. 5. Reports show pipeline counts/values and export to CSV. 6. All state changes and sends appear in History/AuditLog.

## 12) Risks & Mitigations

* **Pricing complexity creep** → Hold scope to one active price list + per-line override in MVP; document future rules.
* **Email deliverability** → Use verified domain, SPF/DKIM, provider with metrics.
* **Data quality** → Required fields, basic validation, duplicate detection on customers.
* **Change management** → Provide short training and in-app help/tooltips.

## 13) Roadmap & Milestones (8–10 weeks, part-time)

**Milestone 1 — Foundations (Week 1–2)** - Project setup, RBAC, base models (Customer, Product, Quote, QuoteLine, TaxRate). - Admin screens for products/taxes; seed data; file storage config.

**Milestone 2 — Quote Editor (Week 3–4)** - Draft creation, line item UX, pricing & tax calculations, autosave; PDF preview.

**Milestone 3 — Workflow (Week 5)** - Approval rules, status transitions, history, cloning, versioning basics.

**Milestone 4 — Communications (Week 6)** - Email templates, send with PDF, EmailLog, secure Quote Viewer.

**Milestone 5 — Reporting (Week 7)** - Pipeline/aging dashboards, CSV export, basic filters.

**Milestone 6 — Polish & Hardening (Week 8–9)** - Performance tuning, validation, security checks, backups, error monitoring.

**Milestone 7 — UAT & Launch (Week 10)** - User testing with your team, feedback fixes, documentation, go-live.

## 14) Future Extensions (Post‑MVP)

* Multiple currencies and tax regimes.
* Cost tracking & margin; vendor quotes comparison; bundles/kits.
* Customer portal for accepting quotes online and requesting changes.
* Conversion to Sales Order and eventual integration with Django Oscar catalog.
* Inventory visibility and availability estimates.

## 15) Open Questions / Decisions Needed

* Email provider (SMTP vs SendGrid/SES)?
* Where to host (e.g., AWS, DigitalOcean)?
* Do quotes require approval by default or only above thresholds?
* Default validity days and terms for BIT?
* Will you need multi-currency (MXN/USD) in MVP?
* Do you want per-customer price lists in MVP or only global + per-line override?
* Any legal text your quotes must include?

## 16) Glossary

* **CSR**: Customer Service Representative.
* **RBAC**: Role-Based Access Control.
* **MVP**: Minimum Viable Product.
* **CFDI**: Mexican electronic invoice (out of scope here).