SALESFORCE CASE STUDY | SALES CLOUD

Accelerating Lead-to-Revenue for a Healthcare Technology Provider

Re-engineering a fragmented manual sales process into an automated, governed, and revenue-aware Sales Cloud engine.

Authored by: Parag Bhatt | Salesforce Technical Lead / Senior Developer | 8.5 years of CRM delivery experience

---

> *One-liner: We re-engineered a fragmented, manual sales process into an automated Sales Cloud engine with intelligent lead routing, opportunity governance, and a revenue-collection escalation system — cutting handoff delays and giving leadership real-time pipeline visibility.*

Engagement Snapshot

Client	MedPractice Solutions FZ-LLC (Dubai, UAE)
Industry	Healthcare Technology / Medical SaaS
Region	Middle East (Dubai) client, delivered from India
Salesforce Cloud	Sales Cloud
My Role	Salesforce Technical Lead + Senior Developer
Team Size	4 (1 Technical Lead, 2 Developers, 1 Admin / QA)
Duration	Approximately 6 months (implementation + hypercare)
Key Tools	Apex, Flows, Process Builder, SOQL / SOSL, Validation & Assignment Rules, Reports & Dashboards, Conga CLM

1. Project Context

MedPractice Solutions FZ-LLC is a Dubai-headquartered healthcare technology provider that sells clinical workflow and practice-management software to medical practices and small hospital groups across the GCC. Their sales motion is long, relationship-driven, and involves multiple stakeholders per account, combined with a recurring subscription revenue model where on-time collections matter as much as new bookings.

Before our engagement, the sales team operated across spreadsheets, scattered email threads, and a lightly configured CRM that was used as little more than a contact list. Leads arrived through multiple channels but with no consistent owner, no SLA, and no measurable conversion funnel. Renewal and collection cycles relied on individual reps remembering to follow up.

The client engaged our team to stand up Salesforce Sales Cloud as the single source of truth for the full lead-to-revenue lifecycle and to automate the high-friction manual steps that were costing them deals, slowing handoffs, and delaying recurring revenue.

2. Problem Statement

The business was losing pipeline value to internal process gaps rather than to competitive pressure. Our discovery surfaced five recurring failure modes:

- **Unowned, untracked leads.** Inbound leads sat in shared inboxes with no routing logic. Many were never actioned, and there was no audit trail to prove a lead had been worked.
- **Manual, error-prone handoffs.** Moving a qualified lead to an opportunity relied on memory and email. Deals routinely stalled at the conversion boundary, duplicate records were created, and account context was lost.
- **No revenue-collection discipline.** Account points-of-contact were not reminded when collection windows approached, and there was no escalation when they failed to act — directly delaying revenue recognition.
- **Zero pipeline visibility.** Leadership had no reliable forecast or stage-by-stage view. Weekly reporting was assembled by hand from spreadsheets and could not be trusted for board-level decisions.
- **Fragmented contact ownership.** A single account typically had several points-of-contact, but there was no clean mechanism to reassign responsibility when staff changed roles or left the company.

3. What We Built — The Solution

We delivered an end-to-end Sales Cloud implementation centred on automation, accountability, and real-time visibility. The solution included:

- **Structured lead management —** channel-aware lead capture, assignment rules that route by region, product line, and rep capacity, plus a qualification checklist so every lead lands with a clear owner and an SLA clock.
- **Governed opportunity pipeline —** a customised stage model aligned to the client's actual sales methodology, with validation rules and required fields enforcing data quality at each stage gate.
- **Custom revenue-collection notification engine —** a business-specific automation (internally referred to as the CPRD automation) that proactively notifies the relevant account point-of-contact as a collection window nears, and escalates to their manager if no action is taken within a defined SLA window.
- **Dynamic, criteria-driven notifications —** reminders fire on business-defined criteria rather than fixed schedules, so they adapt to each account's billing cycle and payment history.
- **Contact-reassignment capability —** a controlled UI that lets authorised users transfer responsibility for an account's points-of-contact to another user cleanly, with full audit history.
- **Document generation and e-signature —** sales agreements and order forms generated through Conga CLM, removing manual contract assembly and reducing turnaround time.
- **Reports and dashboards —** executive dashboards giving leadership live pipeline coverage, forecast accuracy, win-rate by stage, and collection-status visibility.

4. My Role on the Engagement

As the Salesforce Technical Lead I owned the engagement end to end, working directly with the client's business stakeholders and managing the offshore delivery pod of two developers and one admin / QA. My specific responsibilities included:

- **Requirement gathering —** direct workshops with sales leadership, account managers, the collections team, and end users, translating their day-to-day process into a Salesforce design.
- **Solution architecture —** designing the data model, automation architecture, sharing & security model, and the notification / escalation logic.
- **Hands-on development —** personally building the Apex classes, triggers, schedulers, batch jobs, and the more complex Flows; performing bulkification and governor-limit hardening.
- **Code review and mentoring —** reviewing all pull requests from the developers, enforcing Salesforce best practices, and coaching the team on bulk-safe and trigger-framework patterns.
- **Deployment and hypercare —** managing the release path from sandbox through UAT to production, running the cutover, and leading post-go-live hypercare and incident triage.

5. How We Achieved It — Delivery Approach

1. **Discovery and process mapping.** Ran a structured requirement-gathering phase with sales and collections stakeholders. Documented the as-is process, identified the highest-value pain points, and designed the target Salesforce process flow.
2. **Data model and security.** Modelled Accounts, Contacts, Leads, and Opportunities along with the custom objects required for the collection engine. Applied profile-based field-level security, record-type-driven page layouts, and built the contact-reassignment relational structure.
3. **Automation build.** Implemented assignment rules for lead routing, validation rules for stage-gate integrity, and the notification engine using Apex schedulers and batch jobs to dynamically evaluate collection criteria and dispatch reminders.
4. **Escalation logic.** Built the manager-escalation tier so that unactioned reminders within the SLA window automatically rolled up to the next level. Used carefully scoped SOQL / SOSL with selective filters and LIMIT clauses to keep the batches safely within governor limits.
5. **Document and signature flow.** Integrated Conga CLM to generate sales documents dynamically from opportunity data and route them through the e-signature workflow, with status updates flowing back into Salesforce.
6. **Reporting layer.** Built custom report types, executive reports, and dashboards for pipeline coverage, forecast, win-loss analysis, and collection-health KPIs.
7. **Test, deploy, stabilise.** Wrote Apex unit tests well beyond the 75% threshold (averaging 88% coverage), deployed through change sets and metadata API, and ran a 4-week hypercare phase to tune escalation timings and SLA thresholds against real production data.

6. Outcomes and Business Impact

The implementation moved the client from a reactive, spreadsheet-driven sales operation to a measurable, governed Sales Cloud engine. Indicative outcomes observed during hypercare and the first quarter post-go-live:

100%	~40%	~30%	Real-time
Lead ownership and routing (no more orphan leads)	Faster lead-to-opportunity conversion via stage gates	Reduction in overdue collection cycles	Pipeline & forecast visibility replacing manual weekly reports

- Lead response and ownership gaps effectively eliminated through automated routing and SLA enforcement.
- Faster, cleaner lead-to-opportunity conversion driven by enforced stage gates and validation rules.
- Proactive collections with automatic manager escalation, measurably reducing days-sales-outstanding on overdue invoices.
- Real-time pipeline and forecast visibility for leadership, replacing the previous manual weekly reporting cycle.
- A single, trusted source of truth for accounts, contacts, opportunities, and collection status across sales and finance.
- Codebase delivered with ~88% Apex test coverage and a bulk-safe trigger framework, leaving the client with a maintainable platform for future enhancements.

Note: percentages above are indicative portfolio figures and can be replaced with the client's measured KPIs before external use.

7. Tech Stack

Sales Cloud	**Apex (Triggers, Classes, Batch, Schedulers)**	**Flows**
Process Builder	**Validation & Assignment Rules**	**SOQL / SOSL**
Reports & Dashboards	**Conga CLM**	**VS Code**
Salesforce DX	**Change Sets / Metadata API**	

8. Architecture & Process Diagrams

The diagrams below illustrate the core process flows and architecture decisions behind the solution.

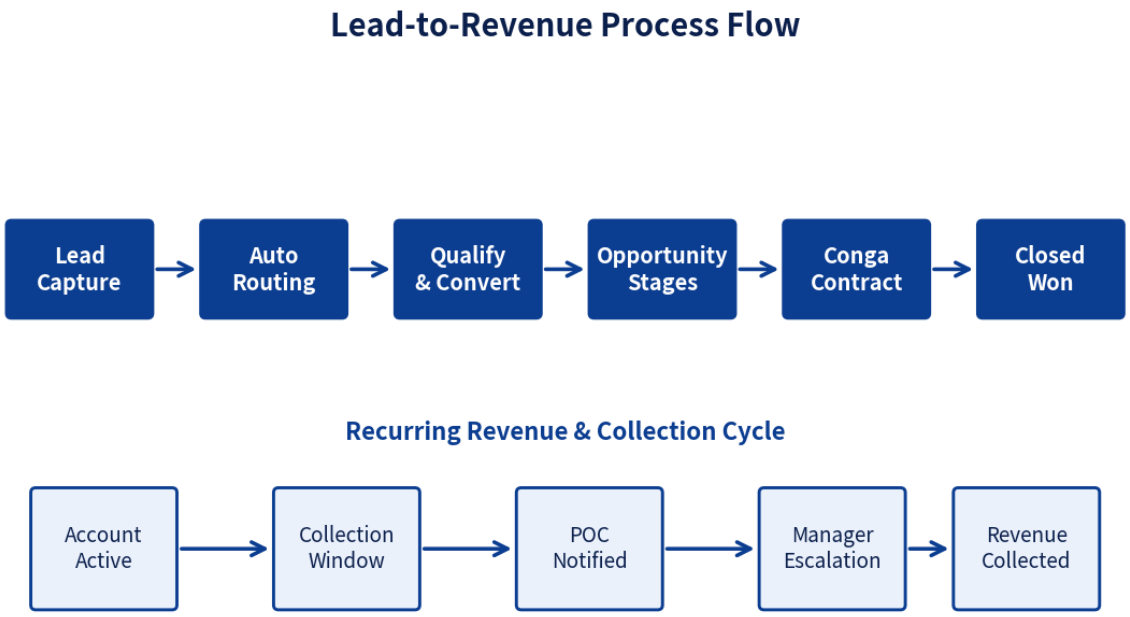

Figure 1 — End-to-end lead-to-revenue process, including the recurring collection cycle.

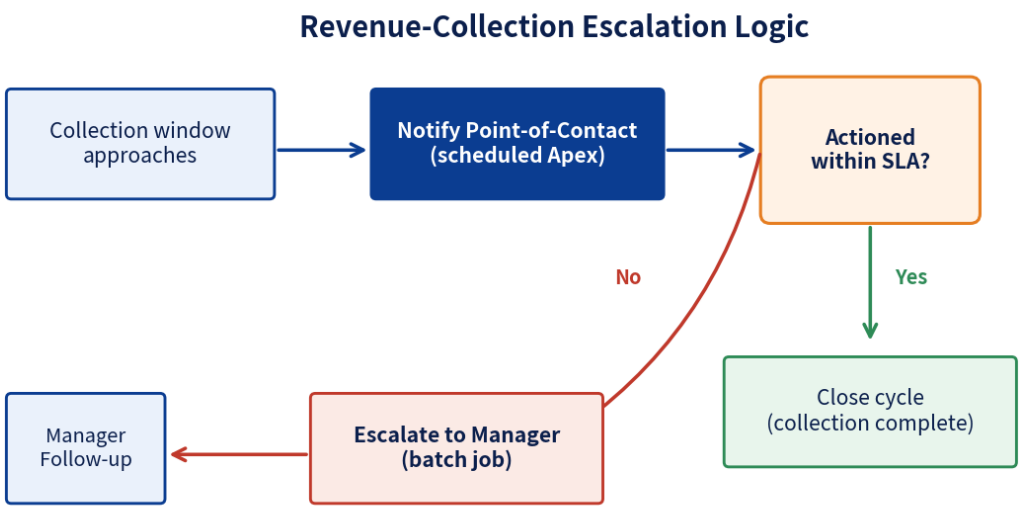

Figure 2 — Revenue-collection notification and manager-escalation logic (CPRD automation).

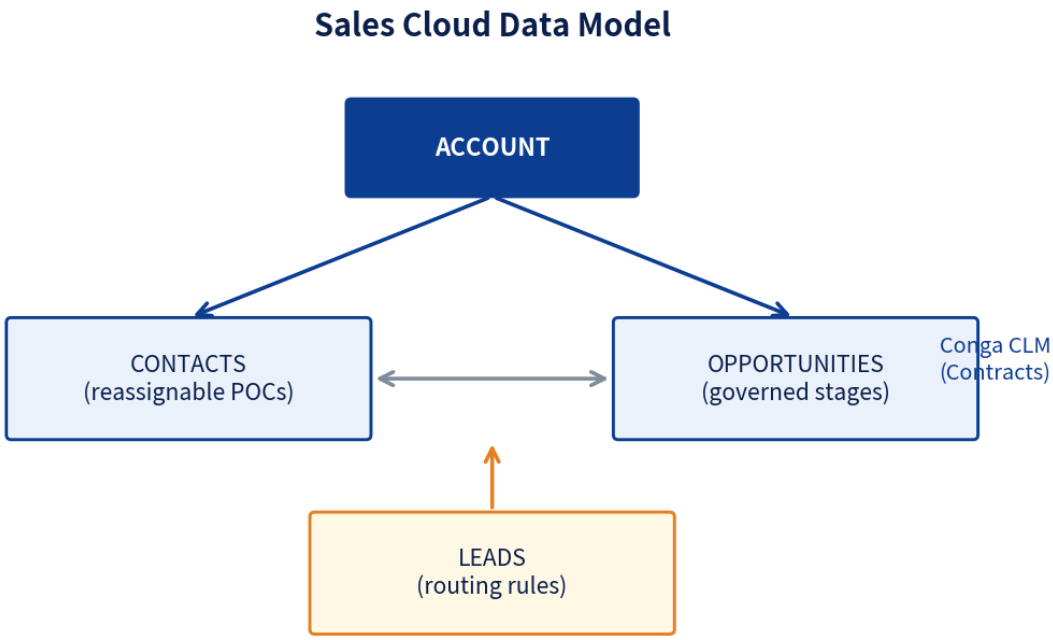

Figure 3 — Sales Cloud data model showing Accounts, reassignable Contacts, Leads, and Opportunities.

9. Key Takeaways

- **Automation only works on top of a clean process.** The biggest lift came from first agreeing the to-be process with the business, then automating it — not the other way round.
- **Governance beats heroics.** Validation rules, stage gates, and assignment rules removed an entire class of human error that no amount of training had previously fixed.
- **Bulk-safe Apex is non-negotiable.** Every scheduled job and batch in the collection engine was designed for governor limits from day one, so the platform scaled as the client's account base grew.
- **Hypercare is part of delivery.** The escalation timings and SLA thresholds were only truly validated under real production load; the 4-week hypercare window was where the solution moved from working to working well.

About the Author

Parag Bhatt is a Salesforce Technical Lead and Senior Developer with 8.5 years of experience delivering CRM transformation programmes across Sales Cloud, Service Cloud, Experience Cloud, CPQ, Revenue Cloud, and Marketing Cloud. He has led offshore and hybrid delivery teams for clients across the GCC, North America, and APAC, with strong hands-on expertise in Apex architecture, automation design, integrations, e-signature, quote-to-cash, and customer engagement platforms.