How HabileLabs Modernized GradientLabs' Customer Support Operations with Salesforce and AI



Client: GradientLabs
Industry: AI-Powered Customer Support / Service Automation

SOLUTIONS
Salesforce Service Cloud, AI Case Resolution, Email-to-Case Automation, API Integration

Introduction

GradientLabs builds AI-powered products. The irony was that their own customer support operation was almost entirely manual.

Every inbound support email required a team member to read it, decide what it was, assign it to the right person, and respond. Common questions got the same answers written from scratch every time. Routine queries that could have been resolved in seconds were sitting in a queue waiting for a human to notice them. And as the customer base grew, the queue grew with it.

The support team was not struggling because of the bad process. They were struggling because the volume had outgrown a model built for a smaller business. HabileLabs rebuilt that model, replacing manual steps with automation and AI at every point where a human should not have been needed.

The Challenges

GradientLabs is experiencing the scaling problem that hits every growing support operation at some point. The tools that work at 50 customers a month do not work at 500. And the manual processes that hold things together at smaller volume become the reason things break at larger volume.

- **1. Email volume the team could not keep up with —** Inbound support emails were the primary support channel. Every one of them required manual handling, reading, classifying, assigning, and responding. As volume grew, response times stretched. Customers waited longer. The team worked harder. The gap between the two kept widening.
- **2. Manual routing that wasted the wrong people's time —** Cases were being assigned manually, which meant the person doing the assigning had to understand the case type and know which agent was best placed to handle it. That knowledge is not always consistent. Cases went to agents who were busy, to agents with the wrong skills, or sat unassigned while someone figured out where they should go.
- **3. Inconsistent response quality across the team —** Without a standardized process or AI-assisted response layer, the quality of support responses depended on which agent handled the case and how much time they had. Common queries got different answers on different days. The experience customers had was not consistent, and that inconsistency was becoming visible.
- **4. Routine queries that should never have needed a human —** A significant portion of GradientLabs' inbound case volume was made up of questions with known, consistent answers. Password resets, billing queries, standard product documentation requests, cases that could have been resolved automatically but were occupying agent time that should have been spent on genuinely complex issues.
- **5. Systems that did not talk to each other —** The support workflow touched multiple platforms, email, Salesforce, and internal tools. None of them were connected. Data was entered manually in more than one place. Information about a case in one system was not visible in another. Every agent's context switched constantly between tools, and nothing was ever in one place.

The Solution

HabileLabs built an end-to-end automated case resolution platform on Salesforce Service Cloud, connecting the inbound email channel to an AI resolution layer, a smart routing engine, and a clean human escalation path for the cases that genuinely need one.

The design principle was simple: remove humans from every step where their judgment is not actually required and make sure the cases that do need human attention arrive with full context and no wasted time.

Email-to-Case Automation

Automatic Case Creation: Every inbound support email now creates a Salesforce case automatically, no manual logging, no missed emails, no cases that exist in an inbox but nowhere else. Email threading keeps the full conversation history attached to the case record.

Attachment Handling: Files and screenshots sent by customers are captured and stored against the case automatically, accessible to any agent who picks it up without hunting through email threads.

Conversation Tracking: The full email conversation, every reply, every follow-up, is logged in Salesforce against the case. Agents see the complete history before they respond. Customers do not have to repeat themselves.

AI and Bot-Driven Case Resolution

Intelligent Case Analysis: As each case is created, the AI layer analyses the content, classifying the query type, identifying the likely resolution path, and determining whether the case can be resolved automatically or needs human handling.

Automated Response Generation: For cases within defined resolution parameters, the system generates and delivers a response automatically, accurate, consistent, and delivered in seconds rather than hours.

Self-Service Resolution: Common query types, password resets, billing questions, standard product queries, resolve end-to-end without agent involvement. The case is created, resolved, and closed automatically.

Natural Language Processing: The NLP layer interprets query intent rather than matching keywords, so the same question phrased in different ways by different customers is correctly categorized and routed to the same resolution path.

Intelligent Case Routing

Skill-Based Routing: Complex cases that require human handling are assigned to agents based on their skill set and the case type, not on whoever happens to be next in a generic queue.

Priority and Load Management: Cases are queued by priority. Agent load is balanced, so no single team member is overwhelmed while others are idle. SLA timers start the moment a case is created and escalate automatically when approaching breach.

Human Escalation Design: Cases that the AI cannot resolve, or that escalate beyond defined thresholds, reach a human agent with the full case history, classification, and prior resolution attempts already documented. Agents start with context, not a blank screen.

Integration Architecture

Multi-Platform API Integrations: Salesforce Service Cloud is connected to GradientLabs' existing systems via API, so data captured in one platform is available across all of them in real time, without manual entry.

Webhook-Based Event Handling: Real-time event triggers ensure that actions in external systems are reflected in Salesforce immediately, no sync delays, no stale data.

Secure Authentication: All integrations use role-based access and secure authentication protocols. The integration layer does not introduce a new security surface area.

Business Impact

- **Response times that dropped from hours to seconds**
  AI-driven case creation and automated responses cut first response time for qualifying cases from hours to under a minute. Customers stopped waiting for answers to questions the system already knew how to answer.
- **A significant share of cases resolved without human involvement**
  Common query types of the routine, repeatable requests that occupy a disproportionate share of agent time, are now resolved automatically. Agents work on the cases that actually need them.
- **Support agents spending time on the right work**
  With AI handling routine resolution, the support team's capacity shifted toward complex, high-value cases. Same headcount. Different allocation. Better outcomes for customers with genuinely difficult problems.
- **Consistent quality across every interaction**
  AI-generated responses for standard cases are accurate and consistent every time, not dependent on which agent is working that shift or how much time they have. Every customer gets the same quality of response.
- **One connected system instead of five disconnected ones**
  API integrations eliminated the multi-platform context-switching that was costing agents time on every case. All the information needed to handle a case is in Salesforce when the agent opens it.
- **A platform that scales with volume**
  The architecture was built for growth. Case volume can increase without a proportional increase in headcount, the AI layer absorbs routine volume, and only genuinely complex work reaches the team.

Technology Stack

PRIMARY PLATFORM	AI CAPABILITIES	INTEGRATION LAYER	AUTOMATION	SECURITY
Salesforce Service Cloud	AI/Bot-driven Case Resolution, Intelligent Routing, NLP	API Integrations, Email-to-Case, Webhook-based Event Handling	Salesforce Flows, Process Builder, Workflow Automation	Secure Authentication, Role-based Authorization

Why GradientLabs Chose HabileLabs

HabileLabs understood that the goal was not to automate its own sake, it was to put the right cases in front of human agents and keep everything else out of their queue. The architecture was designed around that principle from the start.

The integration architecture, connecting multiple external systems to Salesforce via API with real-time event handling, required a team that had built and maintained live integrations before. HabileLabs had done exactly that across prior enterprise engagements.

Building AI resolution on top of Salesforce Service Cloud requires both Salesforce platform expertise and an understanding of how AI layers interact with CRM data models. HabileLabs brought both to the same engagement.

HabileLabs' no-moonlighting, no-shadowing policy meant the architects and developers who designed the solution built it. GradientLabs dealt with the same team throughout, no handoff, no loss of context between design and delivery.

> "We build the automation infrastructure that lets support teams do the work that actually needs them, and keeps everything else out of their way."

The Conclusion

GradientLabs came to HabileLabs with a support operation that had outgrown the way it was running. Manual at every step, inconsistent in quality, and getting slower as the customer base grew.

What they have now is a case resolution platform where the inbound email channel feeds directly into an AI classification and resolution layer, routine cases close automatically, complex cases reach the right agent with full context, and all the systems involved are connected in real time.

The team is smaller than the problem used to require. The quality is higher than the team used to deliver. That is what a well-built automation layer actually does, it does not replace the support team, it makes what the support team does worth doing.