STEELE PROCESS PARTNERS

PORTFOLIO CASE STUDY · PROGRESS RESIDENTIAL

Fourteen teams had been left out of the design. The build was already underway.

I stepped into a Salesforce case management program that had stalled mid-flight. Development was already underway, but 14 of the teams who would live in the system had never been part of the design.

WHAT THE PROGRAM DELIVERED

29.8%

Lower cost-to-serve, including 37% lower cost per contact and 30% fewer contacts

82% to under 29%

Employee attrition

63% to over 80%

First contact resolution

25%

Higher productivity through centralized knowledge management and AI-automated notes

Figures as reported by the program's executive sponsor. They came from a lot of people doing their part well, and I am proud of the part I drove.

WHERE IT STARTED

The build was following requirements that covered only part of the business.

- A 40 plus page requirements document the build was following
- 14 functional teams on the Case object, each with its own record type
- Most of those teams never went through discovery
- Leadership had been told discovery was complete

THE REAL PROBLEM

The build was not the problem. The build was faithfully following requirements that described only part of the business.

HOW IT TURNED AROUND

14 teams missing from the requirements → **Discovery run mid-build** → **Findings folded in, release date held** → **All 14 teams signed off**

WHY THE PROGRAM EXISTED

A fragmented, customer-intensive operation losing trust.

The resident experience organization was fragmented and customer-intensive. Operating costs were climbing and customer trust was slipping. Leadership could not see what was happening in the contact center, the right data was not being captured, and most people were measured on activity numbers that had little to do with whether the resident's problem got solved.

The goal was to consolidate 14 separate issue resolution pathways, and the customer-facing functions and technology around them, into a single standardized model in Salesforce for more than 1,400 users.

WHAT WE WALKED INTO

1. A requirements document of more than 40 pages, already in development
2. Teams that customized their piece of Salesforce on their own
3. Many teams learning about the project only after development started
4. VPs and directors who said they had no real idea what was going on

If development had kept going, the program would have shipped a system that most of the teams using it had never agreed to. **That is the kind of go-live that turns into months of rework.**

HOW WE GOT IT BACK ON TRACK

We folded what discovery found back into the build without moving the date.

Ran the missing discovery, together

Our Senior Process Improvement Architect and I ran every discovery session with all 14 teams, confirmed current state, mapped a future state, and got sign-off from each team's leader.

Showed every team its future state

Each team saw exactly how the system would be configured for them before go-live. No surprises on day one.

Turned findings into buildable work

I wrote all of the user stories and acceptance criteria. We worked refinement with the development firm and folded new findings into the build.

Held the date

The timeline was tight and the release date held. New information kept surfacing, so it was constant adjustment and confirmation.

Ran QA and UAT as a team

I worked with QA and wrote the UAT scripts. We trained the teams and set them up in a sandbox. Every UAT test was signed off by all 14 teams.

Prepared the teams for launch

I produced the documentation the training team used to build curriculum for all 14 teams.

Put answers on the case page

I set up Salesforce Knowledge with the knowledge management manager, with expert review before any article went live, served on the case page.

Gave leadership the view it was missing

After go-live, I worked with the teams to build reports and dashboards aligned with what leadership needed to see and manage.

WHO MADE IT WORK

No one fixes a program like this alone.

It took the executive sponsor, who championed the program. It took the functional team leaders, who gave their time to discovery and sign-off. It took the development firm, which adjusted mid-build, along with the QA team, the training team and the knowledge management manager. And it took a process improvement partner I worked beside from the first discovery session to the last UAT sign-off.

How I used AI

This was the first project where I used AI heavily, to analyze discovery findings, draft process maps, write user stories and acceptance criteria, and build out UAT. The judgment stayed with the people in the room.

What came next

The work on this program led to the forming of a root cause analysis team at Progress. I was invited to join it, which is when I moved into my Operational Efficiency Analyst role.

Salesforce Service Cloud | Case object and record types | Salesforce Knowledge | Reports and dashboards | User stories and acceptance criteria | QA and UAT | Current and future state mapping | AI-assisted analysis