WHITE PAPER

Ready before you build

Most of what makes a CRM or Salesforce program hard is decided before anyone configures anything. This paper sets out the fourteen readiness questions we have learned to ask, across the four areas that matter, and ends with a checklist you can score in an afternoon.

01 Why programs struggle before they start

02 People and ownership

03 Data and reporting

04 Systems and integration

05 Process and automation

06 The readiness checklist, with a self-score

For
Anyone planning, building or owning a new system

Time to read
About 25 minutes

Use it
Before you choose a platform or a partner

A NOTE BEFORE YOU START

The questions that decide a systems program are rarely technical.

When a new membership or fundraising system disappoints, the post-mortem usually blames the software, the partner or the timeline. Look closer and the cause is almost always further back.

Nobody owned the data. Nobody could approve a change of scope without a three-week wait. Two departments used the same word to mean different things. A connected system wrote to fields nobody knew it touched. The first error reached the organization as a complaint from a member.

None of those is a configuration problem. Each one was a decision the organization had not made yet, and the build simply exposed it. The good news is that every one of them can be made before the build starts, by the people who already run the place, at little or no cost.

This paper comes from more than 80 Salesforce implementations for associations and nonprofits, and in particular from a multi-year Salesforce program at a US national association with local member programs. I have written the lessons as patterns, because that is what they are. You will recognize some of them.

Read it with the person who will own the system after go-live. Score the checklist together. The conversation it starts is worth more than the score.

Sunil Dua
Executive Director and Principal Consultant, Kloudserv

The short version

- Name an owner for your data before you name a platform.
- Write down what your words mean: member, lapsed, gift, paid.
- Map every system that will connect, and who owns each side.
- Decide who may change the live system, and how it is recorded.
- Decide who says "done", and give them time to test.

How to use it

Each of the four areas ends in plain questions. Page 9 gathers all fourteen into one checklist. Score it honestly. A low score is not bad news; it is a to-do list you found early.

Worth remembering. Readiness is not a gate you pass once. It is a set of decisions you make on purpose, so the build does not make them for you by accident.

01 · THE OPENING ARGUMENT

The program usually struggles before it starts

A systems program has two halves. The visible half is the build: screens, automations, reports, training. The invisible half is everything the build assumes is already settled.

Who owns the data. Who decides when two teams disagree. What a word means. Which system is the source of truth for an address, a payment, a chapter. Who watches for errors. Who signs off. When those are settled, the build is mostly a matter of craft. When they are not, every one of them gets decided anyway, in the middle of the project, under deadline, by whoever happens to be in the room.

That is where programs lose their months. Not in the configuration, but in the meeting where nobody can say whether two records are the same organization, or the week spent waiting for someone with authority to approve a change, or the go-live that slips because nobody on the client side was given time to test.

Associations and nonprofits are especially exposed, for reasons that have nothing to do with competence. Chapters and affiliates make "who is this record" a real question. Dues, gifts, grants, event fees and certification credits each come with their own rules and their own finance expectations. Staff are lean, so the person who knows how renewals really work is also the person running renewals this month. And the knowledge of how things work tends to sit in two or three heads.

So readiness is not about being technical. It is about making a handful of organizational decisions on purpose, and writing them down. We group them into four areas.

AREA 1

People and ownership

Who owns the data, holds the knowledge, decides scope and runs the system afterwards.

AREA 2

Data and reporting

What counts as a duplicate, who keeps the data clean every week, and what your words mean.

AREA 3

Systems and integration

What each connected system reads and writes, how money reconciles, and what you build versus use.

AREA 4

Process and automation

How changes reach the live system, how failures surface, who says done, and who sees what.

02 AREA 1 People and ownership

Every other area depends on this one. Technology does not own anything. People do, and when nobody has been asked to, ownership falls to whoever is closest to the problem when it breaks.

LESSON 1

Data without an owner turns cleanup into a dispute

Legacy data always needs someone on your side to extract it, judge it and sign off that it is right. If that job is not named early, it lands late, on a person who already has a full-time role, and every question about a record becomes a negotiation between departments.

Ask: who on our staff will extract, clean and sign off on our existing data, and have we given them the time?

LESSON 2

Knowledge held by one person becomes reverse-engineering

When how renewals or gift processing really work lives in one admin's head, the project spends its early weeks rebuilding that knowledge from the system itself. Slow, and risky: the system records what happens, not why.

Ask: if our key admin left tomorrow, what would be written down?

LESSON 3

Train people on decisions, not on powerful buttons

Not everyone needs the merge button. The pattern that holds up is three tiers: everyone spots and reports problems, a few data stewards decide, and one or two admins act on the system.

Ask: who will be our stewards and our admins after handover?

LESSON 4

Decisions stall without someone who can say yes

Every program meets questions of scope and budget mid-flight. If the only person who can approve them meets the project once a month, the project waits a month. Name the person, and agree how fast they will answer.

Ask: who can approve a change of scope or budget, and how quickly?

What ready looks like

A named data owner with protected hours, the way the core processes work written in plain English, a steward and admin plan for after go-live, and one decision-maker who answers within days.

03 AREA 2 Data and reporting

Board reports, renewal notices and funder reports are only as good as the records underneath them. The decisions here are judgments only your organization can make. A partner can propose rules; it cannot know which of two near-identical records is the real member.

LESSON 5

Only you can decide what counts as a duplicate

Two records with the same name may be one organization, a parent and its chapter, or two genuinely separate bodies. With chapters and affiliates in the picture, matching rules are a policy question, not a technical one. Without a written rule and a person who rules on edge cases, the same argument happens record by record.

Ask: have we written down how national and local entities relate, and who rules on edge cases?

LESSON 6

Clean data gets dirty again unless someone owns it every week

A cleanup is a moment. Quality is a habit. Duplicate prevention and validation rules need to be switched on, and a named steward needs a standing slot, typically 30 to 60 minutes a week, to review what the rules flag. Without that, the cleanup slowly undoes itself, and nobody notices until a board number looks wrong.

Ask: who owns data quality every week after go-live, and is it in their job description?

LESSON 7

Inconsistent vocabulary fails silently

Two spellings of one payment type. Amounts in dollars in one system and cents in another. "Lapsed" meaning 30 days past due to membership and a full year to finance. None of these throws an error. Each one quietly sends a record down the wrong path, and the report at the end is wrong in a way nobody can see.

Ask: do we have one agreed glossary, signed off by the teams that use it, before configuration starts?

What ready looks like

A one-page rule for how organizations, chapters and individuals relate; a weekly stewardship slot with a name on it; and a glossary of the twenty or so words your reports depend on, with one meaning each.

04 AREA 3 Systems and integration

Very few organizations run on one system. The website, member portal, events platform, learning system, email tool and finance package all touch the same people. The CRM is only as reliable as the agreements between them.

LESSON 8

Every connected system needs a written contract

Not a legal one. A plain list of which fields each system reads, which it writes, how often, and who owns it on each side. When that list does not exist, a field gets overwritten overnight and two teams spend a week proving it was not them.

Ask: do we have that map for our website, portal, events and finance systems, with an owner on both sides?

LESSON 9

Money needs a verified handshake, defined by finance

Dues, donations, event fees and refunds pass between the payment processor, the CRM and the finance system. If "paid" and "refunded" are defined by whoever builds the integration, finance will not trust the numbers, and they will be right not to. Finance should write the definitions and the reconciliation check, and the build should follow them.

Ask: what do "paid" and "refunded" mean to finance, and how will the systems reconcile each month?

LESSON 10

Challenge every custom requirement

Many requirements arrive as a description of how the old system did something. Rebuilt faithfully, they become custom code that someone must maintain forever. The better question is what the work needs, and whether the platform already does it in a standard way. Custom work should be the exception you chose, not the default you drifted into.

Ask: for each custom request, are we rebuilding something the platform already does?

What ready looks like

A one-page integration map with fields, direction, frequency and owners; finance-written definitions of every money state; and a short list of genuinely custom needs, each with a reason.

05 AREA 4 Process and automation

Automation multiplies whatever it is given. We have seen a single membership save set off eight separate automations. That is fine when each is known, tested and watched. It is a quiet risk when nobody can list them.

LESSON 11

Every change goes through a tracked release

The fastest way to break a live system is a well-meant fix made directly in it. Few people should be able to change production, and every change should be recorded: what changed, why, who approved it, and how to undo it.

Ask: who can change our live system today, and where is each change recorded?

LESSON 12

Failures should arrive as alerts, not as member complaints

Integrations fail, automations error and data drifts. The question is who finds out first. If it is a member whose renewal did not go through, the organization learned about its own system from the outside.

Ask: who watches errors and data health, and how often?

LESSON 13

"Done" needs a definition and an owner on your side

Acceptance testing is the organization checking that the system does its real work, with real scenarios. It cannot be delegated to the people who built it. It needs a named owner and time set aside in the calendars of the people who know the work.

Ask: who runs acceptance testing, and when is that time blocked?

LESSON 14

Design access from the work, not from an old profile

Copying permissions from the old system copies its accidents. We have seen an access review find hundreds of missing permissions and hundreds more granted that nobody needed. Start from what each team does, and grant that.

Ask: what does each role actually need to see and change?

06 · BEFORE YOU SCORE

How to use the checklist honestly

The checklist on the next page is the fourteen questions in one place. It takes an afternoon if you do it properly, and ten minutes if you do not. Do it properly.

Score each question 0, 1 or 2. The scale is deliberately strict, because the gap between "we know this" and "this is written down and someone owns it" is exactly where programs lose time.

SCORE	WHAT IT MEANS	A TEST YOU CAN APPLY
0	Not decided, or nobody owns it.	The answer starts with "I think" or "it depends who you ask".
1	Known, but held in someone's head or only partly owned.	One person could answer it, but could not hand you a page.
2	Decided, written down, and owned by a named person with time to do it.	You could give the page to a new hire and they would act on it.

Who should score it

Not one person alone. Ideally the executive sponsor, the person who will own the system, and someone from finance. Score separately first, then compare. Where your scores differ by two points, you have found a question the organization has not answered, only individuals have.

Common traps

- Scoring 2 because a vendor or partner will "handle it". They cannot own your decisions.
- Scoring the system you hope to have, not the one you run today.
- Counting a document nobody has opened in two years as written down.

A note on scope. These questions apply whatever platform you choose, and whether you are replacing a system, rebuilding one, or improving the one you have. They are about your organization, not about the software.

COPY THIS

The readiness checklist

Circle 0, 1 or 2 for each question. Add up each area, then the total out of 28.

#	QUESTION	SCORE
	PEOPLE AND OWNERSHIP	
1	A named person will extract, clean and sign off on our existing data, with time set aside.	0 1 2
2	If our key admin left tomorrow, how our core processes work is written down.	0 1 2
3	We have named our data stewards and admins for after handover.	0 1 2
4	One person can approve scope and budget changes, and will answer within days.	0 1 2
	DATA AND REPORTING	
5	We have written down how national, chapter and individual records relate, and who rules on edge cases.	0 1 2
6	A named steward owns data quality every week after go-live.	0 1 2
7	We have one agreed glossary for the words our reports depend on.	0 1 2
	SYSTEMS AND INTEGRATION	
8	We have a map of what each connected system reads and writes, with an owner on both sides.	0 1 2
9	Finance has defined "paid" and "refunded" and how the systems will reconcile.	0 1 2
10	Each custom requirement has been checked against what the platform already does.	0 1 2
	PROCESS AND AUTOMATION	
11	Few people can change the live system, and every change is recorded.	0 1 2
12	Someone watches errors and data health on a set schedule.	0 1 2
13	A named person on our side runs acceptance testing, with time blocked for it.	0 1 2
14	Access will be designed from what each role does, not copied from the old system.	0 1 2
	Total (out of 28)	_____

READING YOUR SCORE

What the number tells you, and what it does not

Look at the areas before the total. One area at zero matters more than a respectable total.

0 TO 10

Not ready to build yet

That is useful to know now. Starting a build here means making these decisions under deadline. Spend the next month on ownership and vocabulary before you talk to vendors.

11 TO 20

Ready to plan, not to build

The foundations are partly there. Turn every 1 into a 2 by writing it down and naming an owner. Most of that is staff time, not spend.

21 TO 28

Ready to build

You will get more from any partner, because they can spend your budget on the build, not on discovering decisions. Keep the pages current.

If your score is low, start with two things

Ownership first. Questions 1, 4 and 6 are about names, not systems. Put a name and a weekly hour next to each. That one meeting changes the rest of the checklist, because every other gap now has someone responsible for closing it.

Vocabulary second. Question 7 costs nothing but an afternoon with the people who build your reports. Agree what member, lapsed, active donor, gift, paid and renewal mean. Write each in one sentence. You will be surprised how many of your board's questions this answers on its own.

Both cost staff time rather than money, and both make every later decision cheaper. They are also the two we have most often seen skipped, because neither looks like progress. They are progress.

A pattern worth checking

Teams often feel most worried about integration and least worried about ownership. In practice it tends to be the other way round: the integrations are the visible problem, and the ownership gaps are what make them hard to fix. If your people and ownership score is lower than your systems score, start there.

WHAT TO DO NEXT

Two steps for this month

STEP 1

Score it together

Book ninety minutes with your sponsor, the future system owner and someone from finance. Score separately, then compare. Write down every question where you disagreed. That list is your real starting point.

STEP 2

Close the cheapest gaps first

Name the owners. Write the glossary. Draft the one-page rule for how organizations and chapters relate. None of it needs a vendor, and all of it will be asked for the day a build begins.

The organizations that get the most from a new system are rarely the ones with the biggest budget. They are the ones that knew what they had, decided what they meant, and named who would own it, before anyone configured anything.

That work is unglamorous. It does not show up in a demo. It is also the part of the program the organization keeps forever, whatever platform it runs and whichever partner it chooses.

IF YOU WANT A SECOND VIEW

The readiness scorecard and the Digital Readiness Blueprint

The scorecard on our website covers the same four areas in twelve questions and about three minutes. The Blueprint is the full version: four to six weeks, vendor-neutral, ending in a plan in stages your team owns. Either is a reasonable next step; so is doing the checklist above on your own.