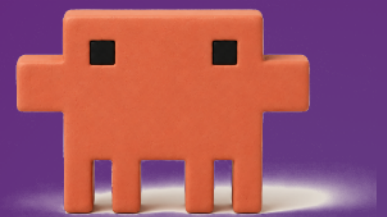


ONE

FREE GUIDE

Start With One

The one task to hand over first, and the four-week plan to do it without disrupting anything.



The mistake almost everyone makes

You start with the task that hurts most.

It makes sense — it is the one keeping you up, so it feels urgent. It is also, almost always, the task that **only you can do**. And a task only you can do runs on knowledge nobody ever wrote down.

So the project starts with an excavation: getting what is in your head into a form a system can use. That takes weeks, nothing works in the meantime, and by the time anything ships the team has quietly gone back to doing it by hand.

Start somewhere else.

Why the first one matters so much

It is not really about the task. It is about proving to your team that these projects finish. A first attempt that visibly works buys you permission for the second. A first attempt that drags kills the second before it starts.



The four tests

Take a candidate task and check it against these. You want four yeses.

1. Does it happen every day?

Frequency beats importance. A small task forty times a week beats a big one twice a month, because the savings compound and you find out fast whether it works.

2. Does the answer have the same shape?

Same shape, not same content. Quoting can pull twenty different numbers and still pass — as long as the rules for combining them are the same.

3. Does being late cost you a customer?

This is the one that turns hours into money. A reply eleven hours later is rarely just late — it is usually a customer who already called someone else.

4. Could someone else do it?

You want a yes. A task anyone could do given the information is a task a system can take over cleanly.

If a candidate fails two or more, **pick another one**. There is always another one, and forcing the wrong task is how first attempts die.

The four-week plan

Nothing here disrupts what you are already doing. The system runs alongside until it earns its place.

Week	What happens	Who does it
1 · Watch	Collect every instance of the task for a week, with the real input and the real output. Twenty examples is plenty.	Your team, 2 min a day
2 · Draft	The system produces answers and nobody sends them . Every output gets marked: correct, close, wrong.	You, 15 min a day
3 · Assist	The system drafts, a person approves, the approved answer goes out. Real customers.	Whoever handles it now
4 · Run	Routine cases handled alone. Anything unusual gets flagged and waits.	Nobody, until flagged

Week 1 is not wasted time

It looks like doing nothing. What it actually does is surface the exceptions — and everyone describes their process as if there are none. There always are, and finding them in week 1 costs nothing.

Week 2 decides everything

This is the week people skip, and it is the one that saves the project.

- Right eight times out of ten by Friday → it will be nineteen out of twenty within a month
- Right three times out of ten → the task failed test 2. You picked the wrong one
- You cannot tell → you are not collecting enough examples. Twenty minimum

Better to learn that now than after launch, in front of customers.

What should never be on this list

- The conversation with somebody who is angry
- The decision that changes what you charge
- The exception you make because you know the customer
- Anything where being wrong is expensive and being slow is not

If your candidate is one of these, it will fail test 2 or 3 anyway. The point of handing over the routine is to **have time for exactly these**.

After the first one

The second task takes about half as long. Not because the system is smarter, but because **you are**: you know what to collect in week 1, you know what «good enough» looks like in week 2, and your team already trusts the process.

50% less time on the second task, and less again on the third. That compounding is the real reason to start small.

How to know it worked

One number, measured before and after: **how many times that task reaches a person**. Not satisfaction, not time saved in theory. How many times somebody had to touch it.

If it does not fall by at least half in month one, stop and look at why. Usually the answer is that the task had more exceptions than anyone admitted, and the honest move is to narrow the scope rather than push harder.



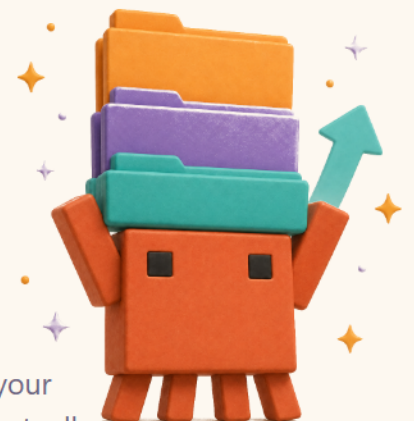


Want us to pick yours?

We do a free Process Diagnosis: we look at how your team actually works, run these four tests over your real processes, and tell you the one to start with and why. No cost, no commitment, nothing to install.

Comment

on any of our posts, or send us MAP by direct message.



Sinsajo Creators — Architects of Business. We build AI systems where your brand and your operations work as one: practical automation your team actually uses, always with a human in control.



WHO WE ARE

Sinsajo Creators

ARCHITECTS OF BUSINESS

We build AI systems where your brand and your operations work as one: practical automation your team actually uses, always with a human in control.

Process Diagnosis

We map how your team actually works and tell you the three processes worth automating first. Free.

AI Agents

Systems that answer, book, quote and follow up — you approve before anything ships.

Content & Brand

Video, carousels and guides that bring the right customers to the door.

Send us MAP by direct message for a free Process Diagnosis.

sinsajocreators.com [@sinsajocreators](https://twitter.com/sinsajocreators) [LinkedIn: Sinsajo Creators](https://www.linkedin.com/company/sinsajo-creators)