

BEFORE YOU BUILD READINESS CHECKLIST

Ten questions to answer honestly before you open a builder.

→ [Read this online](#)



How to use this: Check each box as you answer honestly. A no isn't a dead end, it just tells you what to go handle first.

Most automations that fail were doomed before anyone opened a builder, so run through these ten before you start.

CLARITY

Do you know what you're building?

A no here means you're not ready to design the thing yet. You'll end up automating the wrong steps, or steps that don't actually repeat.



QUESTION 1

Can you describe the task you want to automate as the exact steps you do by hand today?

If you can't write the steps down, a tool can't follow them, and it will break the first time reality doesn't match your head.



QUESTION 2

Does this task happen often enough that automating it is worth the build time?

Automating something you do twice a year costs more than it saves, because automation pays off on repetition.



QUESTION 3

Can you describe exactly what a successful run looks like?

If you can't picture the finished result, you'll either keep tweaking a workflow that's already working, or ship one that looks done but isn't checking the right thing.



QUESTION 4

Does this task follow the same steps every time, or does it change often?

Automation is literal, so if the steps change every week you'll rebuild it more often than you ever ran the task by hand.



Can you actually build it?

A no here means you can design it perfectly and still not be able to finish building it. This is where projects stall for weeks waiting on a login.

☐ QUESTION 5

Do you have admin access to every tool this automation will touch?

A read-only login or a limited seat stops you halfway through, so find out before you design around access you don't have.

☐ QUESTION 6

Do the tools you want to connect offer a real way in, like an API, a webhook, or a native integration?

If a tool offers no supported connection, no amount of building gets the data out, and you need a different plan.

☐ QUESTION 7

Do you have the API keys, credentials, and connected accounts gathered before you start building?

It's easy to confirm a key exists and forget to actually go get it. Stopping mid-build to track one down breaks your momentum and usually takes longer than you'd think.



Will it still be working in a month?

A no here doesn't stop you from building it. It means the thing will fail quietly later, and you won't be the one who catches it.

☐ **QUESTION 8****Do you know who is responsible for this automation once it is running?**

An automation with no owner is the one nobody notices when it fails, so name the person now, even if it's you.

☐ **QUESTION 9****Do you have a plan for what happens when it breaks?**

It will break eventually, usually when a connected tool changes, so know where to look and how to run it by hand meanwhile.

☐ **QUESTION 10****Will you know if it stops working quietly?**

The failure that hurts is the quiet one that stops while you assume it's still running, so build in a way to catch it.

What a no means

CLARITY

Go define the task more before you touch a builder.

ACCESS

Go get the login, the key, or the integration sorted before you design around access you don't actually have.

MAINTENANCE

Doesn't block you from starting, but it means you're building something that will eventually fail with nobody watching. Decide now who owns it and how you'll know if it stops.

Three or more no's across the board usually means the idea is right but the timing is early. That's not a bad outcome. Better to know now than after three days of building.

