PineyWoods

AI AGENTS

AI Agents Without the Risk

A human-in-the-loop playbook for small teams.

pineywoods.co

512.234.5665

AI Agents Without the Risk

A human-in-the-loop playbook for small teams.

The real fear isn't that it won't work

Ask an owner what worries them about AI agents and you rarely hear, "I'm afraid it won't work." You hear something more specific and more honest: *I'm afraid it'll do something dumb.* It'll email the wrong customer the wrong thing. It'll promise a refund you never agreed to. It'll answer a furious client with a chirpy form letter and make a bad day worse. The technology working isn't the scary part. The technology working on its own, in front of a customer, with your name on it - that's the part that keeps the whole idea at arm's length.

That fear is reasonable. We'd feel it too. A business spends years earning a reputation, and the thought of handing the last word to software that occasionally gets confident and wrong is enough to make anyone close the tab and get back to work.

So here's the reframe this whole paper rests on. The risk you're picturing comes almost entirely from one design choice: letting the agent press *send*. Take that one thing away - keep a person on the final step - and most of the danger goes with it. The agent can still read, look things up, and write a reply. It just doesn't get to act on its own. A human reads what it drafted and decides. We call this *draft, don't send*, and it's the difference between AI that hands your team hours back and AI that costs you a customer.

That's the playbook. Not "never use agents." Not "trust the robot." Something in between, and a lot more useful: put the agent to work on the reading and the drafting, the parts it's genuinely good at, and keep a person exactly where judgment belongs - at the moment something leaves the building.

What an "agent" actually is

"Agent" sounds like a big word. The reality is plainer than the marketing.

Strip the jargon and an agent is software that does three things in a row. It *reads* something - an email, a form, a support ticket. It *looks things up* across the tools you already use - your help docs, past tickets, your order system, a customer's history. And then it *drafts an action* - usually a reply, sometimes a summary, sometimes a tidy version of a messy request. Read, look up, draft. That's the loop.

What makes an agent feel different from the simpler "automation" you might have seen is that middle step. It doesn't just follow one fixed rule. It can go find what it needs across a few places, the way a

sharp new hire would - "let me check the account, let me see how we answered this last time" - and then put together something that fits the situation in front of it.

Here's the part that matters most, and the part the hype usually skips. *Drafting is not the same as doing.* An agent writing a refund email is not the same as an agent issuing a refund. An agent composing a reply is not the same as an agent sending it. The drafting can be genuinely smart and the doing can still be zero - nothing sends, nothing gets charged, nothing changes - until a person says go. Those are two separate switches, and you decide whether the second one is even wired up.

Most of the fear about agents quietly assumes the two switches are the same switch. They're not. Once you see that, the whole thing gets less scary and a lot more practical. You can have all the help of the reading and the drafting, and keep all the safety of a person on the last click.

Three agents that are safe to start with

You don't have to start with something dramatic. The best first agents are almost boring - and boring, when your name is on the line, is exactly what you want. Here are three that earn their keep without ever acting on their own.

The support-draft agent. This is the one most teams should start with. A message comes in - a customer asking where their order is, how to change a booking, whether you do the thing they need. The agent reads it, looks through your help docs and your past tickets to find how you've answered that before, and writes a reply in your tone. Then it stops. The draft lands in front of a support person, who reads it, fixes anything that's off, and sends - or doesn't. What it touches: your inbox or help desk, your documentation, your ticket history. Where the human stays: every single send. The agent's job is to turn a blank reply box into a solid first draft. The person's job is to be the one who actually answers the customer. On a busy day that's the difference between replies going out before lunch and replies going out tomorrow - without a single message leaving the building unread by a human.

The pre-call research agent. Before a sales call or a client meeting, somebody on your team usually spends fifteen or twenty minutes pulling the picture together - who is this, what have we talked about, where did we leave it, what's still open. The pre-call research agent does that gathering for you. It reads across your CRM, your email history, your notes, maybe the company's public website, and hands your rep a short brief: here's who you're talking to, here's the history, here's what's outstanding, here's what to ask. What it touches: your CRM, your calendar, your email, public information. Where the human stays: the agent never contacts the prospect or writes to a soul - it only briefs your team. It's pure preparation. The worst case if it gets something wrong is your rep reads a slightly-off note and corrects it in their head, because they're a person who knows the account. Nothing the agent produces leaves your office. That makes it one of the safest places to start.

The intake agent. Every business has a front door where requests pile in - a contact form, a shared inbox, a stack of applications, work orders, referrals. Usually a person reads each one, figures out what it is, and routes it to the right place. The intake agent reads and sorts. It takes the messy incoming thing and structures it: what kind of request is this, who's it from, what do they need, how

urgent does it look, where should it go. Then a person picks up something already organized instead of a raw pile. What it touches: your forms, your inbox, whatever system you track the work in. Where the human stays: the person still decides what actually happens with each item - the agent has only sorted and labeled, not answered or committed to anything. It's the difference between starting your morning sorting the mail and starting it with the mail already sorted.

Notice what these three have in common. Each one does the reading and the drafting and the organizing - the tedious part - and each one hands a person something better than a blank page. And in all three, the agent stops before the moment that carries risk. It never sends, never spends, never commits. That's not an accident. That's the whole design.

The guardrail model

Once you've got an agent reading and drafting, the only question that really matters is this: what is it allowed to do without asking? Get that line right and the rest is detail.

Here's the model we use, and it fits on a napkin. There are things an agent may do on its own, and there are things that always wait for a human.

An agent may, on its own: **read, search, draft, and organize.** It can read an incoming message. It can search your docs and your records to find the answer. It can draft a reply or a summary. It can sort and label and route. All of these are reversible, and none of them touch the outside world. If the agent reads the wrong file or drafts a clumsy sentence, nothing has happened yet - a person is still going to look before anything moves. Low stakes, easy to check, safe to let it run.

What always waits for a human: **anything that sends, spends, or commits.** Anything that leaves your business - an email to a customer, a text, a public reply. Anything that moves money - a refund, a charge, a credit. Anything that makes a promise or changes a record someone else relies on. These are the actions you can't fully take back, and they're exactly the ones a person should approve. Not because the agent is dumb, but because the cost of being wrong is real and a human glance is cheap.

That's the *draft, don't send* default, and it's worth making it the literal setting: the agent prepares the action and parks it. Sending is a separate, human motion. One person, one look, one click. On most days that click takes a few seconds, because the draft is good. On the day the draft is wrong, those few seconds just saved you an apology.

It helps to make the no-go zones explicit rather than assumed. Write them down. "The agent never emails a customer directly. The agent never issues a refund. The agent never changes an account." When the line is written, it's something your team can see and rely on, instead of something you're hoping the software remembers. And when you eventually decide to let an agent do one small thing unattended - say, posting an internal note to your own team's channel - you move that one thing across the line on purpose, with your eyes open, after it's earned it. The line is allowed to move. It just never moves by accident.

What should never be on autopilot

Some things don't belong on autopilot at any point, no matter how well the agent has performed. It's worth being firm here, because this is where a small convenience can turn into a real problem.

Money movement. Refunds, charges, credits, discounts - anything that touches a balance. An agent can absolutely draft the refund email and line up the amount for someone to approve. It should never be the thing that actually moves the money. The gap between "drafted a refund" and "issued a refund" is the gap between a typo and a loss.

Account and password changes. Anything that alters who can get into what - resetting access, changing contact details, updating permissions. This is the territory where honest mistakes and outright fraud both live. A person approves these, every time. An agent that can change accounts on its own is a key left under the doormat.

Anything legal, medical, or financial. If a wrong answer could become advice someone acts on - a contract question, a health question, a tax or money question - it doesn't go out unread. These aren't places for a confident first draft to slip through. The agent can gather and organize the information. A qualified person gives the answer.

Anything going to an already-unhappy customer. This one gets missed, and it matters. When someone is already frustrated - a complaint, an angry email, a second message because the first didn't land - that is precisely the wrong moment for an automated-sounding reply. These should be flagged and routed to a person, not answered from a template. A good agent setup recognizes the upset customer and steps back on purpose: this one's too important, a human takes it. Handing your hardest moments to autopilot is how you turn a recoverable situation into a lost account and a one-star review.

The thread running through all four: the higher the cost of being wrong, the more certain you want a person on it. None of this makes the agent useless on these cases - it can still read, gather, and draft to save the prep time. It means the deciding and the sending stay human, full stop.

How to pilot one agent in about two weeks

You don't need a big project to find out whether an agent helps. You need one narrow job, a couple of weeks, and the discipline to keep a person on every send. Here's the shape of it.

Pick one narrow job. Not "handle support." One slice of support - say, drafting replies to order-status questions, the same kind that comes in over and over. The narrower the job, the faster you'll know if it works and the easier it is to watch. If you can't say the agent's job in one sentence, it's too big. "Draft a first reply to every order-status email so a person can edit and send it" is the right size.

Watch every action at first. For the first stretch, a person looks at everything the agent does - every draft, every bit of sorting - before it goes anywhere. You're not just checking quality; you're learning the agent's habits. Where is it sharp? Where does it get confused? Which cases does it handle cleanly,

and which should it never have touched? This is also where you confirm it's flagging the tricky ones instead of guessing. Run the agent's version right alongside how you do it now, so you've got an honest before-and-after on the same real work.

Keep a person on every send. Through the whole pilot, *draft, don't send* stays the rule. Nothing the agent writes reaches a customer without a human reading it first. The win you're testing for isn't "the agent replaced someone." It's "the assisted version is genuinely faster and at least as good." That's a lower bar and a more honest one - and it's the bar that actually saves you hours without risking your name.

Widen slowly, only as it earns trust. If after two weeks the drafts are reliably good and your team trusts them, you widen - carefully. Maybe the agent now handles two kinds of question instead of one. Maybe a person spot-checks instead of reading every single one. You move the line toward the agent one notch at a time, and only after it's earned that notch. Trust gets built first; then the line moves. Never the other way around.

And decide your exit criteria up front. Before you start, write down what would make you keep it and what would make you stop. Keep it if it saves real time, the quality holds, and your team trusts the drafts. Stop it if it needs more babysitting than it saves, or it keeps getting confident about things it should flag, or the number you're watching just doesn't move. Hitting the stop condition isn't failure. It's the pilot doing its only job: telling you the truth cheaply, before you built your week around it. A pilot you're allowed to stop is a pilot you can run honestly.

Do it this way and you find out for real, in about two weeks, whether this particular agent belongs in your business - with very little at stake either way.

A calm next step

If you're curious whether one of these agents fits your business - and a little wary, which is the right way to be - that's the conversation we like to have.

Book a free call. Thirty minutes, no pitch. We'll look at where your team's time actually goes, talk through whether an agent makes sense for any of it, and where it does, sketch what a safe first version looks like: one narrow job, a person on every send, *draft, don't send* built in from the start. If the honest answer is "not yet," we'll say so. You'll come away with a clear next step either way.

No hype, no shelfware - just a straight read on whether this is worth your time.

Call **512.234.5665** or book online. We'd be glad to help you think it through.

Appendix: The agent approval template

Before you switch on any agent, fill this out. One page, one agent. If you can't fill in a line, that's a sign the agent isn't ready to run yet. Keep the finished page somewhere the whole team can see it.

AGENT APPROVAL - ONE PAGE, ONE AGENT

Agent name / job (one sentence - what it does, for whom):

_____

Human owner (the person responsible for it):

_____

What it MAY READ (the inboxes, documents, and records it can look at):

- _____
- _____
- _____

What it MAY DRAFT (the replies, summaries, or sorting it can prepare):

- _____
- _____

What it MAY do UNATTENDED (often "nothing" to start - and that's fine):

- _____

What ALWAYS needs human sign-off (every send, spend, or commitment):

- _____
- _____
- _____

OFF-LIMITS - the agent never does these (check all that apply, add your own):

- [ ] Send anything to a customer without a person's approval
- [ ] Issue a refund, charge, credit, or discount
- [ ] Change an account, password, or permission
- [ ] Answer anything legal, medical, or financial
- [ ] Reply to an already-unhappy customer
- [ ] _____

STOP / ESCALATE RULE (when it must hand off to a person):

The agent stops and flags for a human when _____
_____ - and then it waits.

Reviewed by: _____ Date: _____ Review again on: _____