



DRUTEK INSIGHTS · WHITE PAPER

The Harvest Framework

Your teams are already vibe coding. A practical method for turning that grassroots AI experimentation into production value — safely.

PUBLISHED

October 2026

FROM

Drutek Insights

FOR

Operations & technology
leaders

CONTENTS

- 01 Executive summary
- 02 The candy problem
- 03 The legitimate concerns
- 04 Risk, governance and ownership
- 05 The Harvest Framework
- 06 Identify: problems worth solving
- 07 Enable, Evaluate, Integrate
- 08 Worked example: the Monday report
- 09 Running your first harvest
- 10 The Harvest Workshop

01 Executive summary

Your teams are already using tools such as Claude Code, Cursor and Bolt. This isn't a problem to stamp out. It's a signal of unmet needs and untapped innovation.

“Vibe coding” means describing what you want to an AI tool in plain English and letting it write the software. With a little training, most staff can now build in days what once took a full team months. But software built this way is rarely fit to hold client or patient data — and if nobody understands how it works, nobody can fix it when it breaks.

The Harvest Framework gives that energy a sanctioned route: surface the real problems staff are solving, let them prototype safely, and have your experts rebuild what works on approved platforms — inside your existing governance.



Building is cheap

Non-specialists can now prototype in days what used to take a team months.



Understanding isn't

Someone accountable must understand anything that goes live — not just the AI that wrote it.



Harvest the idea

The prototype proves the need. Your experts build the real thing.

THE AIM

More people contributing ideas and prototypes. Your experts still owning what goes live. And AI never having the final say.

02 The candy problem

Keep the kids away from the candy and they'll find it anyway. The difference is that what employees build with unsanctioned AI tools can be catastrophic — or something rather wonderful.

Building software used to sit behind bureaucracy and layers of process. Now it sits behind a chat box, and everyone from the newest associate to the CEO can become a creator. The look on a designer's face when they build a working application from scratch, by themselves, is a joy — and that enthusiasm is worth channelling rather than suppressing.

These “shadow” experiments are valuable because they start from real friction. People are finding the gaps in their working day, turning ideas into something that runs, and creating genuine solutions — often to problems the organisation didn't know it had.

So the question isn't whether to allow it. It's where. Experimentation should be encouraged — in a safe space, without customer or patient data, in areas that aren't business-critical.

What matters next is what happens to the output. Is a bundle of generated code solid enough to pass legal and security review, or is it a throwaway prototype? The truth is usually somewhere in the middle — and that middle is where the value is.

What it looks like in practice

FINANCE

An analyst builds a small app to reconcile two exports that never quite match.

OPERATIONS

A coordinator automates the weekly status pack they used to assemble by hand.

CLIENT TEAMS

A manager prototypes an intake form because the official one takes three systems.

Typical patterns, not specific clients.

ENCOURAGE

- Sandboxed experimentation with AI coding tools
- Non-critical internal problems
- Sharing what was built, and why

KEEP OUT OF THE SANDBOX

- Customer and patient data
- Financial transactions
- Anything that goes live without human review

03 The legitimate concerns

None of this argues against using AI to build software. It argues for being clear-eyed about four risks that no single request to an AI tool — however detailed — will resolve.



Understanding

Who can fix it?

AI can produce thousands of lines of code that nobody in the business fully understands. When it breaks — or an auditor asks how it works — someone accountable needs to have the answer.



Security

Who keeps it safe?

Established software is secure because experienced people maintain it and patch flaws quickly when they're found. An unreviewed tool built by AI has nobody in that role.



Lock-in

Can you take it with you?

Some AI building tools let you export what's been built; others keep it locked inside their platform. If you can't take it with you, you can't hand it to your own team to finish.



Scale

Will it hold up?

Tools built from a single request often miss the unglamorous requirements — volume, performance, unusual cases — that only show up in real use, with real people.

RULE OF THUMB

Prefer tools that let you export what's been built. Anything you can't export should be treated as a prototype to learn from — never as the finished product.

04 Risk, governance and ownership

Harvest works inside your existing controls, not around them. It gives people a sanctioned route to act on their ideas — which is safer than the unsanctioned one they’re already using.

Tools already in use

Start with an amnesty, not a crackdown. Ask people to declare what they’ve built and what data it touches. For each tool, decide to harvest it, move it onto approved platforms, or retire it. Anything that has touched personal or client data goes straight to your security team and DPO.

The sandbox

Run by IT and security, not by the people using it: approved accounts on enterprise terms, synthetic or anonymised data only, no connection to live systems, tools that let you export, and activity that’s logged.

Data protection

No personal, client or patient data enters the sandbox. When a harvested idea will process personal data in production, assess whether a data protection impact assessment is needed, as you would for any new system. For most AI-supported processing, it will be.

Your existing policy

Harvest doesn’t replace your AI or acceptable-use policy. It gives people a way to act within it. Update the policy to name the sandbox and the route from prototype to production, so the safe path is also the obvious one.

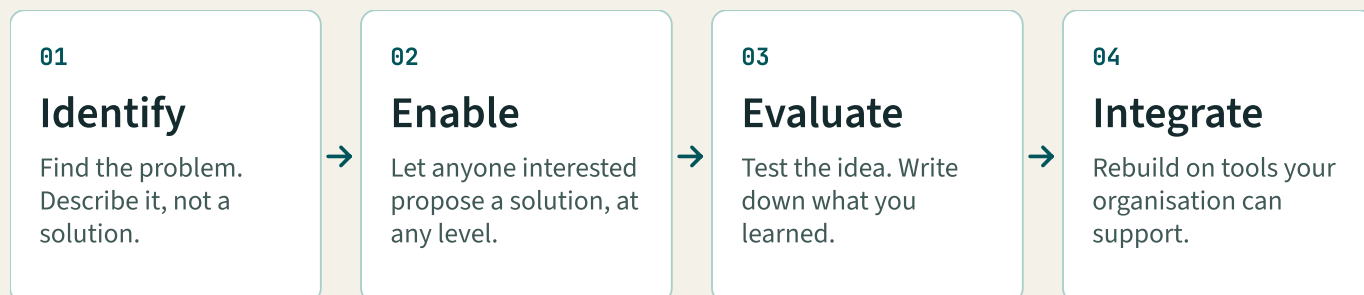
Who owns what

Role	Responsible for
Executive sponsor	Sets priorities, removes blockers, reports progress to the board
Problem owner	Owens the Problem Canvas and the business outcome
IT and security	Runs the sandbox, approves tools, reviews anything heading for production
DPO and compliance	Sets the data rules, decides on impact assessments, checks regulatory fit
Delivery team	Builds, runs and supports the production version
Contributors	Anyone who proposes an idea or builds a prototype in the sandbox

This is a framework, not legal advice. Involve your DPO and follow your sector regulator’s guidance.

05 The Harvest Framework

A method for capturing the innovation already bubbling up across your organisation — and turning it into solutions that are safe to run.



THROUGHOUT

Use AI to help craft every stage — but AI never has the final say on what goes live.

Who's involved, and what each stage produces

Stage	Who's involved	What you get
Identify	The team that feels the pain, with a facilitator	A Problem Canvas: who's affected, what it costs, what "solved" looks like
Enable	Anyone interested — working in the sandbox	Ideas, process changes and working prototypes
Evaluate	Problem owner, delivery team, security and DPO	A go / change / stop decision, with the reasoning recorded
Integrate	The delivery team that will run it long-term	A supported service on approved platforms, with a named owner

Why "harvest"?

Because the crop is already growing. You don't need to plant innovation; you need a way to gather it, sort it, and keep what's good — then rebuild it with proper support, security review and accountability behind it.

Gradual and moderated

Use of AI should grow gradually and under control. Even if your people don't write every line, someone must understand every line — because the only thing worse than a 4am security call is the one where nobody knows how to fix it.

06 Identify: problems worth solving

The hardest part of identifying problems is resisting the urge to solve them. The moment you attach a solution, you narrow who can contribute.

“We need a dashboard that shows X” is already a solution. The real problem might be “I don’t know if our campaigns are working”, or “I spend three hours every Monday pulling numbers for my boss”. Those are different problems — and different people across the organisation might solve them in radically different ways.

Where problems hide

Spreadsheet graveyards

An elaborate model that takes an hour to update each week is often the most honest documentation of how the business actually works.

Apology emails

“Sorry this took so long” is a breadcrumb. Someone is consistently blocked, and they’ve normalised it.

Workarounds

“We just do it this way because the system doesn’t let us” — they’ve already found the gap between need and tool.

Sync meetings

Recurring meetings that only move information between systems or teams point to integration problems ripe for automation.

Four techniques for surfacing them

- 01 The five-minute friction log.** For one week, team members note every moment of waiting, copying, pasting, reformatting or context-switching. No judgement, no solutions — just observations.
- 02 “Explain it to me like I’m new.”** Someone from outside the team shadows the work and asks naive questions. “Why do you do that step?” uncovers processes that exist only because they always have.
- 03 The cost-of-waiting audit.** For any decision that needs approval or information from elsewhere, measure the real wait. A data request that takes three days because someone pulls a report by hand is a quantifiable problem.
- 04 The tool inventory.** Compare the tools people actually use every day with the ones that are officially sanctioned. The gap between the two lists is fertile ground.

Document the problem, not the build

Directs the answer

“We need to build a Python script that automatically pulls data from Salesforce and formats it for the weekly report.”

Invites contribution

“The sales team spends about four hours every Monday morning compiling data from several sources for the weekly leadership report. It delays other work and invites human error.”

The second framing lets the junior marketing coordinator ask “what if we changed what leadership actually needs?” just as validly as the senior engineer proposing an API integration. Both are legitimate responses to the problem as stated.

THE PROBLEM CANVAS

Capture these five things for every problem you take forward.

Who feels this pain?

Not just the job title — how many people, and how often.

What’s the current workaround?

There’s always a workaround.

What does the status quo cost?

Time, money, frustration, risk.

What would “solved” look like?

Outcomes, not outputs.

What are the constraints?

Compliance, integration requirements, budget.

How?

Leave this blank. That’s for the Enable stage.

What makes a good harvest candidate?

- ✓ **Repeated friction** — it happens regularly, not as a one-off annoyance.
- ✓ **Broad impact** — it affects several people or touches several processes.
- ✓ **Clear boundaries** — you can say where it starts and where it ends.
- ✓ **Non-critical path** — at first, failure shouldn’t cause real harm. Patient data and financial transactions can wait until the process has proved itself.
- ✓ **Measurable improvement** — you’ll be able to tell whether it’s better afterwards.

THE UNCOMFORTABLE TRUTH

The best problems often come from the people closest to the work — the ones who’ve been dismissed before. Listen when the person who processes invoices says the system is broken, even when it looks fine from the outside. And sunk cost shouldn’t protect bad solutions from scrutiny.

07 Enable, Evaluate, Integrate

02 Enable

Open the solution space to everyone — from the CEO’s “what if we just...” to the intern’s fresh perspective.

- Anyone interested in the problem can respond, at whatever level they’re comfortable with: a process change, a sketch, a spreadsheet, or a working prototype.
- AI tools make prototyping accessible to anyone willing to try. Train people to use them well.
- Prototype in the sandbox — no customer or patient data — and prefer tools whose code you can export.

03 Evaluate

Judge the premise, not the polish. A rough prototype that proves an idea is worth more than a perfect one that doesn’t.

- Test each response against the canvas: does it deliver what “solved” looks like, within the constraints?
- Probe what a quick prototype misses: unusual cases, security, volume.
- Write down what was decided and why, with a simple process diagram, so the knowledge outlives the prototype.

04 Integrate

Build the production version on the platforms and skills your organisation can support for the long term.

- Follow your organisation’s skills, not the prototype’s: rebuild on the platforms and languages your teams already support.
- Experienced engineers own the result and understand every line, even where AI helped write it.
- Nothing goes live on AI’s say-so. A human makes the final call.

THROUGHOUT EVERY STAGE

Use AI to help you craft it — but AI must never have the final say on what goes live. Knowledge has to live in your people, not only in an AI tool.

08 Worked example

Illustrative scenario

A sales team loses a morning every week to a report. Here's how a harvest turns that into something better than anyone would have specified up front.

Identify

A friction log shows the sales operations team spends about four hours every Monday pulling numbers from the CRM, a finance export and two spreadsheets into the weekly leadership report. The Problem Canvas records the cost — a lost morning, late numbers, copy-paste errors — and one constraint: the CRM holds client personal data.

Enable

Three very different responses come back:

- A sales coordinator builds a working prototype with an AI app builder, using dummy data, that merges the exports and lays out the report.
- A marketing coordinator asks whether leadership reads all fourteen metrics. They don't — five drive every decision.
- An engineer suggests connecting the CRM directly to the company's approved reporting platform.

Evaluate

The prototype proves the layout and the calculations leadership actually wants. The question cuts the report from fourteen metrics to five. The decision — rebuild on the approved platform, retire the prototype — is written down with the reasoning.

Integrate

The data team builds the five-metric dashboard on the approved platform, using the prototype as a working specification. The DPO signs off the data access, and the coordinator who built the prototype checks the result before it goes live.

WHAT THIS SHOWS

The best answer was combined

A process change (fewer metrics) and a technical one (a direct connection) — from people who'd never normally be in the same meeting.

Retiring a prototype is success

It did its job: it proved what was needed. No client data ever touched it.

The pain shaped the fix

The person who felt the problem helped design the solution — and checked it worked.

09 Running your first harvest

You don't need a programme to begin. A first harvest can start with one team and one week of honest observation.

Identify

Run a friction log

Ask one team to keep a five-minute friction log for a week, and run a tool inventory alongside it. Write the strongest candidate up on a Problem Canvas — with the “how” left blank.

Enable

Open it up

Share the canvas widely and invite responses of any kind. Give contributors a safe sandbox, basic training in AI coding tools, and a clear rule: no customer or patient data.

Evaluate

Test the premise

Review responses against the canvas. Keep what proves the idea, and write down what was decided and why.

Integrate

Build it properly

Hand the validated idea to the team that will own it, on approved tooling. Use AI to accelerate the build — and keep a human as the final say on what goes live.

How you'll know it's working

- ✓ **The problem moved** — the cost on the Problem Canvas (hours, delays, errors) is measurably lower than the baseline.
- ✓ **Problems are surfacing** — people are bringing real friction forward, not just tool requests.
- ✓ **Prototypes reach a decision** — every one ends in go, change or stop, with the reasoning written down.
- ✓ **Data stayed where it should** — no personal or client data found in the sandbox.

ASK THIS WEEK

“What tools are you actually using every day that aren't on the official list — and what problem are they solving for you?”

10 The Harvest Workshop

One day with your technical and operations leaders in the same room — leaving with real problems on the table, the guardrails agreed, and a first harvest ready to start.

FORMAT

One day, in person

WHO

Technical and operations leaders, up to 12

BEFORE THE DAY

A short tool survey and a one-week friction log

INVESTMENT

From £3,500 + VAT

Morning

Find the problems

Review what the survey and friction logs surfaced, including the tools already in use. Shortlist the strongest candidates and write them up as Problem Canvases.

Midday

Agree the guardrails

Set the sandbox rules, data boundaries and owners — with technology, operations and risk perspectives in the same conversation.

Afternoon

Prototype and plan

Prototype one problem live with AI tools, judge it against its canvas, and plan your first harvest from Identify to Integrate.

You leave with

- ✓ A shortlist of problems worth solving, each on a Problem Canvas
- ✓ Named owners for each role in the framework
- ✓ A working prototype of one problem — built without client data
- ✓ Draft sandbox and data rules, ready for IT, security and your DPO
- ✓ A first-harvest plan your teams can start the following week
- ✓ A written summary of the day and its decisions

WHO RUNS IT

Drutek embeds with enterprise teams and trains developers and non-technical colleagues in modern tools such as Cursor, Claude Code, Copilot Studio, Bolt.new and Lovable. Our experience includes work delivered for Pfizer through an agency partner.



Book a Harvest Workshop

One day. Your technical and operations leaders. Real problems, agreed guardrails, and a first harvest ready to run.

Your teams are already experimenting. We help you capture the ideas, document the patterns, and turn them into production-ready solutions on approved platforms.

✉ hello@drutek.com

🌐 drutek.com/contact