

10 AI Pipelines That Pay Back in 3-5 Months

Checklist for businesses with €1M–10M revenue

AUTHOR

VERSION · YEAR

FORMAT

**Mikhail Zelenin · AI
Potential**

v1.0 · 2026

A4 · 16 PAGES

§ ГЛАВА

10 AI Pipelines That Pay Back in 3-5 Months

Checklist for owners of €1M–10M revenue businesses

This collection is a map — it helps you understand *where to start* with AI and *how to count* the payback. For each pipeline: typical pain / architecture / stack / payback hypothesis / 30/60/90 control points.

Author: Mikhail Zelenin — coach and AI architect. Limassol, Cyprus. 4+ active pilots and live deals in 2026.

Contact: t.me/metaoptimist · mzelenin.com

§ ГЛАВА

How to read this checklist

Each pipeline is described in the same structure:

1. **Typical pain** — how it sounds from the owner's mouth
2. **Architecture** — what steps are in the process and where AI fits in
3. **Stack** — concrete tools (Claude / Codex / MCP / ERP / CRM / Ollama)
4. **Payback hypothesis** — how many months to recover the investment
5. **30/60/90** — control points in the first 90 days
6. **Are you ready** — short checklist «should I start with this pipeline»

The goal is *not to pick everything*, but to pick **one first pipeline**. After that, the foundation is in place, and the next pipelines stack on it in weeks, not months.

§ ГЛАВА

1. Contract↔Payment control

Typical pain. «Two customers haven't paid, products are sitting in the warehouse, no one tracks the contract↔payment link systemically. At the quarterly retro, no one remembers who's at what stage.»

Architecture. An agent pulls data from CRM (Bitrix24/Pipedrive/HubSpot) and the ERP, reconciles contract dates with actual payments, raises alerts on shipments without coverage and contacts without confirmation. Weekly — a traffic-light report to the owner.

Stack. Claude Agent SDK · MCP connectors for ERP and CRM · Postgres journal · Telegram for alerts.

Payback hypothesis. A single recovered shipment pays for the implementation. Typical payback — 2–4 months.

30/60/90.

- 30 days: data connectivity, manual reconciliation, alpha of the weekly report
- 60 days: automatic alerts on uncovered shipments, owner dashboard
- 90 days: linkage to the sales team, SLA escalation

Are you ready. CRM + ERP with up-to-date data. One person on the client side as the process owner.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini. 20-30 minutes of dialogue — and we save the first 30 minutes of discovery on our call.

I'm the owner of a B2B distribution / manufacturing company. Help me prepare for a call with an AI consultant on the «Contract↔Payment control» pipeline. Ask me sequential questions so I walk in knowing: (1) how many contracts we sign per month and what's the average ticket, (2) what percentage of shipments go out without confirmed payment, (3) where contracts physically live — CRM, ERP, folders, paper, (4) who currently tracks payments and how much time it takes, (5) the 2-3 biggest non-payments of the last year and what they cost. After each answer, clarify if anything is unclear. End with a one-page summary.

Ask me on the call:

- How many recovered shipments do I need for the €2 500 pipeline to pay back in my niche?
- Can we start without replacing the CRM/ERP — on the current stack?
- Where should alerts go — Telegram, email, straight into the CRM?
- Who on my team should own the pipeline?
- What about clients who are already not paying — is there a recovery scenario?

§ ГЛАВА

2. AI sales rep assistant

Typical pain. «Sales reps spend 30 minutes on each email, forget about follow-ups, lose warm leads, fail to transcribe calls.»

Architecture. Agent listens to calls (Zoom/Twilio), prepares meeting summaries, drafts follow-up emails, updates CRM. Before sending — owner or sales head validates.

Stack. Claude Sonnet · Whisper transcription · MCP in CRM · n8n/Make for the pipeline.

Payback hypothesis. +20-40% of sales reps' time on new leads (by reducing admin work).
Payback — 3-4 months for a team of 5+ reps.

30/60/90.

- 30 days: call transcription, follow-up drafts in Notion
- 60 days: automatic CRM field updates, post-meeting brief
- 90 days: next-step suggestions + sales head dashboard

Are you ready. Call recording (legally clean). 5+ reps. Sales head willing to implement.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm the owner of a company with a sales team. Help me prepare for a call with an AI consultant on the «AI sales rep assistant» pipeline. Ask me sequential questions so I walk in knowing: (1) how many reps I have and how many calls they make per week, (2) which 3 recurring tasks eat the most of their time, (3) where the CRM is and where calls are recorded (Zoom, telephony, mobile), (4) whether clients are legally informed about call recording, (5) how much time per rep goes to admin (emails, notes, CRM updates) vs actual selling. After each answer, clarify.

Ask me on the call:

- Should I start with one champion rep or roll out to the whole team at once?
- Can we plug into my existing CRM (Bitrix24/HubSpot/Pipedrive) without a swap?
- What about mobile-only calls — is there a workable solution?
- How long until reps start improving the prompt templates themselves?
- What do I do with reps who'll sabotage — is resistance inevitable?

§ ГЛАВА

3. Project documentation verification

Typical pain. «Every project — 200+ pages of technical documentation. The chief engineer spends 2 weeks reading and cross-checking against standards. Half the typos get caught on site.»

Architecture. Agent compares documents to the standards base (national / EU / industry), finds inconsistencies and disputed clauses. Output — a checklist for the engineer: what is OK, what to verify, what is non-compliant.

Stack. Claude Opus + RAG over standards · vector store · OCR (for scans).

Payback hypothesis. Reduces verification from 2 weeks to 2 days. Payback — 1-2 projects.

30/60/90.

- 30 days: load the standards base, alpha on one document type
- 60 days: expansion to 3-5 document types, validation on historical projects
- 90 days: integration into current workflow, «missed errors before/after» metrics

Are you ready. Electronic standards archive. At least 3-5 projects per year.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm the owner of a design / engineering firm. Help me prepare for a call with an AI consultant on the «Project documentation verification» pipeline. Ask me sequential questions so I walk in knowing: (1) how many projects per year and what's the average documentation volume, (2) which standards we use (national, ISO, industry, foreign), (3) who verifies today and how many hours per project, (4) whether we have an electronic standards archive or work with paper/scans, (5) how many typical errors slip through to the site each year and what they cost. After each answer, clarify.

Ask me on the call:

- Can we upload scans or only native PDFs?
- Who validates AI findings — chief engineer, project lead, or a third party?
- What happens when the standards base is updated — does the system retrain?
- Can we give clients an AI-annotated checklist as part of the deliverable?
- What accuracy is realistically achievable — 95%? 99%? And what about the residual?

§ ГЛАВА

4. AI accountant

Typical pain. «The accountant spends a week per month reconciling source documents, exporting from the bank, classifying payments. Month-end close is always stressful.»

Architecture. Agent pulls bank statements, reconciles with ERP, classifies payments, drafts entries. Chief accountant validates and posts. Monthly — a discrepancy report.

Stack. Claude Sonnet · MCP in ERP · bank API · local LLM (Ollama) for sensitive data.

Payback hypothesis. Frees up 30-50% of the accountant's time. Payback — 4-6 months.

30/60/90.

- 30 days: connecting banks and ERP, classification on historical data
- 60 days: daily reconciliation, escalation of disputed payments
- 90 days: automation of draft entries, accuracy metrics

Are you ready. Electronic accounting. Open banking or electronic statements. Chief accountant as process owner.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm the owner of a company with bookkeeping. Help me prepare for a call with an AI consultant on the «AI accountant» pipeline. Ask me sequential questions so I walk in knowing: (1) how many bank transactions per month and how many operation types, (2) which accounting system (QuickBooks, Xero, 1C, local), (3) is the bank connected via API or only via electronic statements, (4) how many hours per week the accountant spends on reconciliation and classification, (5) what the chief accountant categorically will NOT delegate to automation (month-end close? reporting?). After each answer, clarify.

Ask me on the call:

- Which data categorically should NOT go to cloud LLMs — what counts as sensitive?
- Can we start with payment classification only, without auto-posting entries?
- What about discrepancies — auto-fix or escalation to chief accountant?
- How is access controlled — who sees bank flows?
- What happens when the chief accountant changes — does the agent need re-training?

§ ГЛАВА

5. CRM↔ERP synchronization

Typical pain. «Between the CRM and the ERP — a gap. Reps create a deal in the CRM, accounting duplicates it manually into the ERP. Discrepancies surface at the retro.»

Architecture. MCP connector between CRM and ERP with sync rules: deals in CRM → invoices in ERP, payments in ERP → status in CRM. Discrepancies → alerts + dashboard.

Stack. Claude Agent SDK · custom MCP servers for the CRM and ERP · n8n for orchestration · Postgres journal.

Payback hypothesis. Frees up 8-15 hours per week of double entry. Payback — 3-5 months.

30/60/90.

- 30 days: CRM → ERP connector (one deal type)
- 60 days: two-way sync, discrepancy alerts
- 90 days: coverage of all deal types, owner discrepancy dashboard

Are you ready. CRM (Bitrix24/HubSpot/Pipedrive) + ERP. IT person in the team or contractor.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm the owner of a trading / manufacturing company. Help me prepare for a call with an AI consultant on the «CRM↔ERP synchronization» pipeline. Ask me sequential questions so I walk in knowing: (1) which CRM I use and what data lives in it, (2) which ERP I use and what customizations sit on top, (3) who and how syncs them today — manually, scripts, scheduled exports, (4) how many hours per week double entry eats up and where discrepancies surface, (5) which 2-3 deal types are most painful and should be synced first. After each answer, clarify.

Ask me on the call:

- Can we start with one-way sync (CRM → ERP) and add two-way later?
- What about conflicts — does CRM win or ERP win?
- Can we do this without my IT team — or is one of mine required?
- What about historical data — migrate or only new records?
- What load does this add to the ERP — will it slow down peak sales hours?

§ ГЛАВА

6. Competitor watcher

Typical pain. «I can't keep up with what competitors are doing. By the time I find out — it's too late to react. Sitting in reactive mode, not ahead.»

Architecture. Every 1-2 weeks the agent pulls fresh posts, articles, product updates from competitors (channels, sites, GitHub if any). Extracts 3-5 main signals and writes a short briefing — what is changing, what it means for us.

Stack. Claude Sonnet · social/Telegram MCP · web scraper · local embed (Ollama) for semantic search.

Payback hypothesis. Not direct money — but one right strategic bet from a digest pays for the pipeline per quarter.

30/60/90.

- 30 days: watchlist setup (5-15 sources)
- 60 days: first bi-weekly report with 3-5 key signals
- 90 days: routine in cron, direct delivery to owner

Are you ready. Clarity on who your Tier 1-3 competitors are. Willingness to read the digest regularly.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm the owner of a company in [your niche]. Help me prepare for a call with an AI consultant on the «Competitor watcher» pipeline. Ask me sequential questions so I walk in knowing: (1) who my 5-10 key competitors are (Tier 1 — direct, Tier 2 — alternative, Tier 3 — adjacent), (2) where they're publicly active (X, LinkedIn, GitHub, YouTube, industry press, newsletters), (3) what I want to monitor — product releases, marketing, key hires, fundraising, (4) how often I'll actually read the digest — daily, weekly, biweekly, (5) what 3 surprising findings from last year I wish I'd had earlier. After each answer, clarify.

Ask me on the call:

- What format is best for the digest — PDF, email, Slack, dashboard?
- Can we monitor closed sources (paid communities, my subscriptions)?
- Where's the «react now» vs «just know» threshold?
- Can I add competitors on the fly?
- What if a competitor goes silent for 3 months — skip or dig deeper?

§ ГЛАВА

7. Owner's Personal OS

Typical pain. «Decisions are made impulsively. A quarter later I don't remember why I chose this and not that. I can't see what paid off. The retro is formal, in reality nothing changes.»

Architecture. Personal decision store (Obsidian + MCP) + an agent that captures each owner's «bet» (money/time/risk/opportunity) with metadata and links. At the quarterly retro — a dashboard: which bets paid off, which were a waste.

Stack. Claude Code · Obsidian · MCP on the personal vault · manual entry via TG bot.

Payback hypothesis. Not direct money. Decision quality + visible memory — the main return. Visible over 6-12 months.

30/60/90.

- 30 days: install Obsidian + basic Claude Code skills, capture last 30 decisions
- 60 days: automatic bet capture from calendar and email
- 90 days: first data-driven quarterly retro

Are you ready. Willingness to maintain the vault 10-15 minutes a day. One project as the testing ground.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm the owner of multiple businesses / projects. Help me prepare for a call with an AI consultant on the «Owner's Personal OS» pipeline. Ask me sequential questions so I walk in knowing: (1) how many major decisions (with risk or \$-stakes) I make per week, (2) how many projects I run in parallel and where they physically live, (3) where my notes, decisions, and commitments are now (Telegram saved, Apple Notes, my head, scattered), (4) what usually prevents me at the quarterly retro from remembering «why I chose this», (5) how much time per day I can realistically spend on the system — 5 min, 15 min, 30 min. After each answer, clarify.

Ask me on the call:

- If I already use Notion / Obsidian — migrate or start from scratch?
- Can I start without uprooting my current systems?
- When will I see first results — 3 months, 6, a year?
- What happens if I drop the vault for a week — disaster?
- Can I bring my team in, or is this strictly my pipeline?

§ ГЛАВА

8. Devil's advocate before decisions

Typical pain. «I commit major bets in 24-48 hours, no second opinion. Then regret it. The team doesn't push back — hierarchy.»

Architecture. Before each major decision, the owner writes a brief in Claude. The agent asks 5 hard questions + runs a premortem («what destroys this decision in 3 months»). Verdict: GO / GO-WITH-CONDITIONS / PAUSE / ROADMAP.

Stack. Claude Opus · `/devil` skill · prompt-engineered for the owner's business context.

Payback hypothesis. One avoided mistake per quarter pays for the pipeline. Often — orders of magnitude more.

30/60/90.

- 30 days: install the skill, run on 5-10 historical decisions
- 60 days: integrate into pre-commit workflow
- 90 days: accumulate patterns of «where I usually err», personalization

Are you ready. 2-3 serious decisions per month. Willingness to take hard feedback.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm an owner who makes major decisions solo. Help me prepare for a call with an AI consultant on the «Devil's advocate before decisions» pipeline. Ask me sequential questions so I walk in knowing: (1) how many \$-stakes decisions (>€5K or significant risk) I make per month, (2) name 3 decisions from last year I now regret — what went wrong, (3) what usually throws me off — emotion, urgency, fear, perfectionism, authority, (4) who currently pushes back on my decisions — partner, mentor, family, no one, (5) what feedback format I take best — text, voice, video. After each answer, clarify.

Ask me on the call:

- Can I hide from my team that I'm consulting with AI?
- What if devil says «PAUSE» but deadlines are burning?
- How much context about my business does it need for quality feedback?
- What if the AI says what I don't want to hear?
- Will it remember my patterns after 6 months — or start fresh every time?

§ ГЛАВА

9. Decision OS dashboard

Typical pain. «At the quarterly retro I can't remember which bets paid off. Information is fragmented: part in CRM, part in email, part in my head.»

Architecture. Extension of Personal OS — all the owner's bets are aggregated in one dashboard with metrics (invested / returned / horizon / status). Cuttable by bet type, business, quarterly cycle.

Stack. Claude Code · Obsidian DataView · Personal OS as the foundation · web dashboard (optional).

Payback hypothesis. Frees up 3-5 hours per quarter on retro, raises quality of subsequent decisions. Visible over 2-3 quarters.

30/60/90.

- 30 days: bet format, history tagging
- 60 days: dashboard, first data-driven quarterly retro
- 90 days: automate entry from CRM/calendar

Are you ready. Personal OS already in place (pipeline №7). Willingness to work with the dashboard each quarter.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm an owner who places many bets (decisions with financial or time risk). Help me prepare for a call with an AI consultant on the «Decision OS dashboard» pipeline. Ask me sequential questions so I walk in knowing: (1) how many bets I make per quarter — hires, investments, product calls, partnerships, (2) what's the average bet size in money and/or team-hours, (3) how my quarterly retro currently works — formal, in my head, not at all, (4) which «did this bet pay off» metrics I want to see — money returned, time freed up, market position, (5) do I want bets segmented by business or pooled together. After each answer, clarify.

Ask me on the call:

- How do we formally define «did this bet pay off» — per bet or globally?
- Can we split the dashboard by business or must we pool?
- Access for the team or only for the owner?
- What about «blind» bets where outcome is unclear for 2+ years?
- Can we go backwards — export historical decisions from email / messengers?

§ ГЛАВА

10. Cash gap forecast

Typical pain. «I learn about a cash gap a week before it hits. Firefighting. No early indicators.»

Architecture. Agent pulls data from ERP (payments, receivables, obligations), calendar (payouts, taxes), CRM (expected deals) and builds a liquidity forecast for 30/60/90 days. Alerts at thresholds.

Stack. Claude Sonnet · MCP in ERP · Postgres journal · Telegram alerts.

Payback hypothesis. One avoided cash gap (cost of an emergency loan or late-payment penalty) pays for the pipeline. Usually — 2-3 months.

30/60/90.

- 30 days: ERP integration, manual liquidity model for 30 days
- 60 days: 60-day forecast, threshold alerts
- 90 days: integrate CRM data into the forecast, 90-day horizon

Are you ready. ERP system. Obligations and expected revenue properly recorded.

Prep with your AI. Drop this prompt into Claude / ChatGPT / Gemini.

I'm the owner of a company with operational cashflow. Help me prepare for a call with an AI consultant on the «Cash gap forecast» pipeline. Ask me sequential questions so I walk in knowing: (1) how many cash gaps happened last year and what they cost (penalties, emergency loans, downtime), (2) where obligations data lives — ERP, bank, separate Excel, CRM, (3) who's responsible for cashflow forecast today — CFO, accountant, me, (4) what horizon I plan today and want to plan — 30 / 60 / 90 / 180 days, (5) is there seasonality and how predictable is it. After each answer, clarify.

Ask me on the call:

- What thresholds for alerts — subjective or by norm (e.g., <10% of monthly obligations)?
- What happens at «red light» — manual review or auto-escalation to CFO?
- Can we factor in seasonality and special events (holidays, tenders)?
- What accuracy is realistic at 30/60/90 days?
- What if the forecast is wrong — retrain or escalate manually?

§ ГЛАВА

Is my business ready for AI pipelines — quick check

Before taking on your first pipeline, verify:

- ☐ Revenue €1M–€10M. Smaller — AI is often overkill, better basic automation first.
- ☐ Have electronic systems (CRM, ERP, some bookkeeping). If everything is «on paper» — start with basic digitization.
- ☐ Have a process owner. Not CEO — someone working with this daily.
- ☐ Have a «let's try» budget of €2.5K–€5K for the first pipeline.
- ☐ Ready for 4-6 weeks where 4-8 hours of your time go to analysis and validation.

If 4 of 5 — **mature case**, we go into the first pipeline.

If 2-3 of 5 — **early case**, start with the €700 teardown (90 min), which will show what to do first.

If 0-1 of 5 — **too early**. Basic automation without AI first.

§ ГЛАВА

What's next

This checklist is a map. Which pipeline is yours — it's a question for your business. To figure it out:

1-hour AI session · €99 — one hour, one question. You leave with 3 working scenarios.

AI Business Teardown · €700 — 90 minutes with prep. You leave with 2-3 vectors and roll-out priority.

Booking: mzelenin.com or t.me/metaoptimist

© 2026 Mikhail Zelenin · Limassol, Cyprus · Meeting confidentiality is absolute.