

AMAcode



Claude Code 導入をご検討の方へ

セットアップの"味噌"は、どこにあるのか

— AIをそのまま使った場合と、何がどう違うのか —

「AIはすごいらしい」と聞いて触ってみたけれど、結局いつもの仕事は減らなかった。

——そういう話を、たくさん伺ってきました。私たちも最初はそうでした。

この資料は、**同じAIを使っているのに、なぜ結果が変わるのか**を、専門用語をできるだけ使わずに書いたものです。「うちに入れたら、何が違うのか」を判断する材料としてお読みください。

2026年8月18日版 / AMAcode（株式会社EYAN）

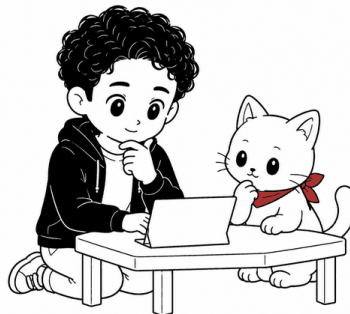
※この資料に出てくる仕組みは、すべて私たち自身が毎日の業務で使っているものです。

1 まず、正直なところから

「AIを入れたのに、忙しさが変わらない」という話について

AI導入がうまくいかなかった会社さんの話を伺うと、だいたい同じところで止まっています。

- 最初の1週間は面白がって触った。今は誰も開いていない
- 質問すれば答えは返ってくる。でも「うちの仕事」は1つも減っていない
- 毎回、背景から説明し直している。説明する時間のほうが長い
- 出てきた数字が合っているのか分からず、結局は自分で確かめ直している

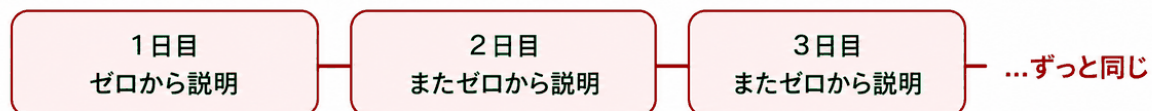


これは「AIが賢くないから」ではありません。私たちが行き着いた答えは、もっと拍子抜けするものでした。

AIは、うちの会社のことを何も知らないまま入社してきている。

しかも、**毎朝そのことを全部忘れて、また入社してくる。**

■ そのまま使うと（毎朝、初日に戻る）



■ セットアップ済み（昨日の続きから始まる）

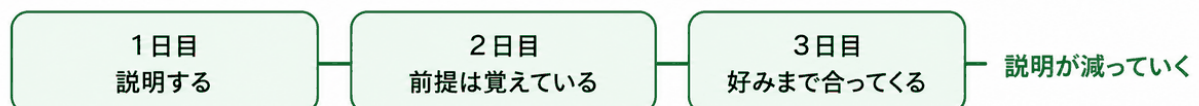


図1：使い続けたときの差。減っていくのは作業時間より先に「説明する時間」です

どんなに優秀な人でも、会社の場所も、お客さんの名前も、書類の置き場も、「この会社ではこうする」という決まりも知らない状態で座らされたら、成果は出せません。しかも翌朝には昨日のことを全部忘れていて——そんな

新人が毎日入れ替わり立ち替わり来る、とを考えてみてください。

足りていないのは、AIの賢さではありません。「うちの仕事に落とす段取り」だけです。そして、その段取りを最初に作って渡してしまう作業——それが私たちの言う「セットアップ」です。

2 味噌をひとことで言うと

私たちがしているのは、ツールを納品することではありません

私たちは、道具をお渡ししているではありません。

「その会社の仕事が回る段取り」ごと、御社に移植しています。

同じClaude Codeを使っても、そのまま使う場合と、私たちがセットアップした場合とでは、次の9つが違います。この9つが、いわゆる"味噌"です。9つはバラバラではなく、4つのかたまりになっています。

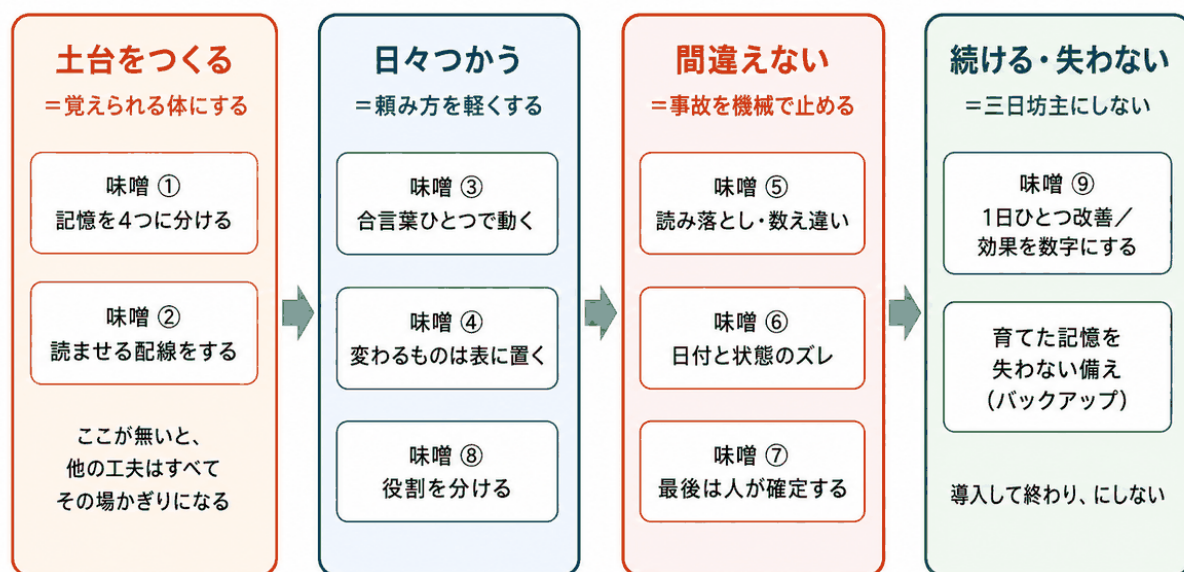


図2：9つの味噌の地図。左から順に積み上がっていきます

このあと1つずつ「そのまま使うとどうなるか」と「セットアップ済みだとどうなるか」を並べて説明します。どれも派手な技術の話ではありません。地味な整理と、決めごとです。

ただ、その前に——この9つのうち、どこまでがネットで手に入って、どこからが私たちの仕事なのか。そこを先にはっきりさせておきます。ここを曖昧にしたまま話を進めるのは、誠実ではないと思うからです。

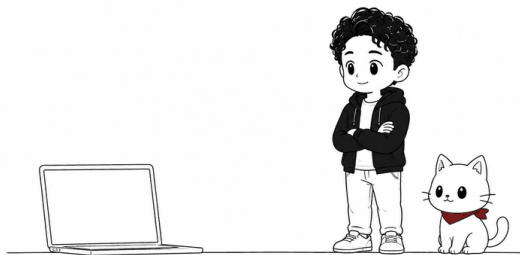
3 入れるだけなら、差はつきません。 私たちの仕事は、その先です。

いちばん先にお伝えしておきたいこと

はっきり申し上げます。

Claude Codeを入れて、基本の形をつくることまでの手順は、すでに世の中に公開されています。私たちもそこから始めました。つまり、そこでは差がつきません。

では、どこで差がつくのか。何にお金をいただいているのか。ここを正直に書きます。



積み上がるほど、事故が減ります ↑

③ 壊れないようにする

間違ったまま動かないよう、歯止めを入れる

② 翻訳する

御社の仕事を、AIが読める形に置きかえる

① 入れる

Claude Codeを入れて、基本の形をつくる（半日）

公開されている手順

ここからが 私たちの仕事

手順書を読んでも
手に入らない部分

ここは誰でも同じ
やれば同じ形になります

図：セットアップは3段あります。世の中に出回っているのは①だけです

なぜ①だけでは足りないのか — 「動いてしまう」のが一番怖い

①のとおりに入れると、Claude Codeはちゃんと動きます。ここが落とし穴です。**動くから、出てきたものを信じてしまう。**でも実際には、こういうことが起きます。

- 合計は合っているのに、1行まるごと抜けている
- 日付がずれたまま、それらしい工程表が出てくる
- 表の1列が届いていないのに、堂々と分析を始める
- 済んだ仕事が「未送付」のままメモに残り、AIはそれを読んで動く

そして、いちばんの問題はここです

この種の間違いは、起きた日には分かりません。数か月たってから、「あのとき違っていた」と分かります。——私たちは、この4つを全部やりました。自分の会社で、実際に。

③で入れている「歯止め」（6つ）

入れる歯止め	入れないと、こうなります
起動のたびに、正しい日本時間を機械が入れる	AIは「今日」を持っていない。期限の計算が静かにずれる
合計はAIに暗算させず、計算機に計算させて、その出力をそのまま貼る	それらしい数字が出る。確かめさせても、同じ間違いを繰り返して「一致しました」と言う
表を読む前に「全部で何列あるか」を数えさせ、合わなければ止める	欠けたまま分析が始まり、渡した側も気づけない
変わるもの（請求・入金・約束）は文章に書かせず、1件1行の表へ。毎朝ズレを点検する	古いメモが残り、AIがそれを根拠に動く
送信・投稿・支払いは、AIにはできないようにしておく	権限を全部「許可」にすると、確認なしで外に出せてしまう
会社のデータを外に出さない（社外へアップロードしない・機密は分離・学習に使わない）	よくある手順書は、最後に「クラウドにアップロードして保存」と案内しています。お客様の情報が入るなら、私たちはやりません

②の「翻訳」が、実はいちばん大きい

公開されている手順書には、こう書いてあります。「各手順の冒頭で、使う人に必要な情報を聞いてから作業に入ること。推測で進めないこと」と。

つまりあの手順書は「空の枠」で、中身はご自身で答えないと埋まりません。そして多くの方が、ここで止まります。難しいからではありません。自分の仕事を、他人（AI）に分かる形に並べ直す作業だからです。毎日やっている人ほど、言葉になっていません。

- 何を覚えさせるのか／覚えさせなくていいのか
- どの作業から手を付けると、いちばん早く楽になるのか
- 何を絶対に外へ出さないと決めるのか
- 合言葉を何にして、誰が使うのか

ここを一緒に決めて、形にするのが②です。私たちが伺って、いちばん時間をかけている部分でもあります。

まとめると、いただいている費用の中身はこうです

中身	説明
① 入れる	公開されている手順で、半日ほど。ここは誰がやっても同じ形になります＝差がつきません
② 翻訳する	御社の業務を、AIが読める形に置きかえる。ここがいちばん長い
③ 壊れないようにする	上の歯止め6つ＋業種ごとの守秘のルール。先に転んでおいた分
④ 定着させる	使い続けられる形にして、導入前と比べた削減時間を数字で出せるようにする

ひとことで言うと。

入れただけのAIは、間違っても止まりません。

私たちが作っているのは、間違ったら止まるAIです。

味噌 ①

記憶を「4つの引き出し」に分けて残す

いちばん根っこの部分です。ここが無いと、他の工夫は全部その場かぎりになります

AIに「覚えておいて」と言ったとき、その内容がどこへ行くのか。ここを設計しているかどうか、1か月後にいちばん大きな差になります。



■ そのまま使うと



- 会話を閉じると、話した内容は**基本的に残りません**



- 会話が長くなると、途中で自動的に要約され、**細かい経緯は消えます**



- 「先週こう決めましたよね」が通じない。**毎回ゼロから説明**



- 覚えているように見えても、どこに何を覚えたのか**誰も確認できません**

■ セットアップ済みだと



- 覚えたことが普通のファイルとして手元に残ります



- しかも役割で4つに分けてしまいます（下の図）



- 「覚えておいて」と一言添えるだけで、AIが自分で仕分けて書き込みます



- 中身はメモ帳で開けます。間違っていたらその場で直せます

なぜ「4つ」に分けるのか

最初は1つのファイルに全部書いていました。すぐに壊れました。種類の違うことが混ざると、探せなくなり、更新できなくなり、結局読まれなくなるからです。そこで、覚えることを性質で4つに分けます。タンスの引き出しと同じ考え方です。

そのまま使うと



記憶の置き場が
そもそも無い

会話が終われば消える。
何を覚えたか確認もできない

セットアップ済み — 4つの引き出し



① 決めたこと

料金の決め方・取引先の選び方・
運用のルールと、その理由



② 好み・書き方

文章のトーン・様式・進め方の癖・
使ってほしくない言い回し



③ できごとの記録

いつ何があったか・進捗・
新しく分かったこと（時系列）



④ 判断の基準

受ける／断るの線引き・値段の考え方・
失敗から得た教訓

図3：性質のちがう記憶を、混ぜずにしまう。だから後から探せて、直せます

引き出し	入れるもの	入っていると、何がかわるか
決めたこと	料金の決め方、取引先の選び方、運用のルールとその理由	同じ判断を何度も迷わない。「なぜそう決めたか」が残るので、後から見直せる
好み・書き方	文章のトーン、様式、進め方の癖、嫌いな言い回し	出てくる文章が、だんだん「その会社の文章」になる。手直しが減る
できごとの記録	いつ何があったか、進捗、新しく分かったこと	「あの件どうなった？」に、推測ではなく記録で答えられる
判断の基準	受ける／断るの線引き、値段の考え方、過去の失敗から得た教訓	新しい案件が来たときに、毎回ゼロから考えなくてよくなる

だから、こう変わります。 使えば使うほど、**説明する時間が減っていきます。** そのまま使うAIは、いつまで経っても初日と同じですが、セットアップした環境は「昨日までの続き」から始まります。

ここが安心材料でもあります

AIが覚えていることは、この4つのファイルが全部です。見えないところに溜め込む部分がありません。「これは覚えなくていい」と思ったら、その行を消せば忘れます。

味噌 ②

置くだけでは読めない。読ませる配線をする

ご自分で試された方が、いちばん多くつまずくところです

「AIに覚えてほしいことをファイルに書いて置いておいた。なのに全然覚えていない」—— これは非常によくあるご相談です。原因ははっきりしています。



ファイルは、置いただけでは読れません。

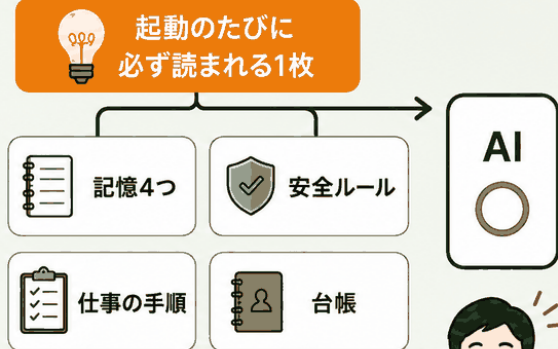
AIは自分から、フォルダの中を勝手に探しに行ったりはしないからです。

✕ 置いただけ



つながっていないので、聞かれない
毎回「このファイル読んで」と言う必要がある
言い忘れた日は、何も知らないAIが答える

○ 配線してある



何も言わなくても効く／指示を忘れても大丈夫

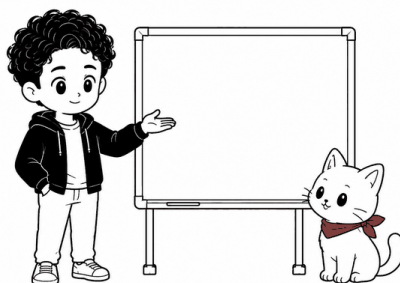
図4：「覚えさせたつもり」がいちばん危ない。間違いに気づけないからです

だから、こう変わります。 覚えさせる作業が「置く」だけで終わり、思い出させる作業がゼロになります。使う方が仕組みを意識しなくてよくなる、というのがこの工夫の目的です。

味噌③ 長い指示文をやめて「合言葉」にする

AIがうまい人・下手な人、という差そのものを無くします

AI活用がうまくいくかどうかは「頼み方」で決まる、とよく言われます。それは事実なのですが、毎日それをやり続けるのは無理があります。だから、頼み方のほうを固定してしまいます。



■ そのまま使うと



- 毎回、長いお願い文を書くことになります



- 書き方しだいで**結果が変わります**。
上手な人と、そうでない人で
差が出ます



- 同じ人でも、忙しい日は
雑になる。結果、日によって
品質が違う



- スタッフに引き継ごうとすると
「どう頼めばいいのか」が
伝わらない

■ セットアップ済みだと



- よく使う仕事は、ひとことで
呼び出せる形にしています



- たとえば「おはよう」と打つだけで、
決められた手順が**毎回同じ順番で**
走ります



- その仕事の手順書はファイルとして
残っているので、誰が打っても
同じ品質です



- 言葉はうる覚えでかまいません。
近い言い回しでも動くように
設計します

✕ 毎回、長い指示文を書く



書き方で結果が変わる

忙しい日は雑になる / 人によって差が出る
スタッフに引き継げない



忙しい日は
雑になる



人によって
差が出る



スタッフに
引き継げない

○ 合言葉ひとつ

おはよう

決められた手順が
毎回、同じ順番で走る

覚えることは
「おはよう」だけ

うる覚えの
言い方でも通じます

① 今日の日付と曜日を確かめる



② 前日の決めごとを保存する



③ 未完了・進行中の案件を集める



④ 近づいている期限を拾う



⑤ 今日の工程表を1枚にする



⑥ 改善をひとつ提案する



図5：手順はファイルに書いてあります。だから人が覚えている必要がありません

「合言葉」というのは、具体的にはこういうものです

打つ言葉	裏で走ること
おはよう	予定・未完了のタスク・進行中の案件・近づいている期限を集めて、今日の工程表を1枚にする。前日の決定で残しておくべきものがあれば「これ、覚えておきますか？」と先に聞く
覚えておいて ○○	内容の性質を判定して、4つの引き出しの正しい場所へ書き込む。似た記述が既にあれば、増やさず書き換える
タスク登録	緊急度・案件・種類を判定して、当日ぶん／案件ぶん／後回しぶんに振り分ける
返信を作って	その相手との過去のやり取りを先に読んでから下書きを作り、やり取り台帳に記録する
改善レポート	導入前に記録しておいた手作業の量と、今の状態を突き合わせて、減った時間を1枚にまとめる

※ここに並べているのは一例です。実際にお渡しする合言葉は、御社のヒアリングをしてから決めます。最初は2つか3つだけで始めるのをおすすめしています（増やすのはいつでもできます）。

だから、こう変わります。「AIをうまく使えるかどうか」が、担当者の腕前の問題ではなくなります。覚えることが減り、社内で引き継げる形になります。

味噌 ④

変わるものは文章に書かず、表に置く

地味ですが、事故がいちばん多いのはここです

請求の状況、入金の有無、「誰に何を送るか」の約束——こういう「変わるもの」を文章のメモに書くと、必ず古くなります。しかも厄介なのは、古いメモは消えずに残り、次に読んだ人がそれを根拠に動いてしまうことです。



私たち自身がやらかした例

自社の運用で、すでに入金が済んでいる請求書を「まだ未送付」と書いたメモが残っていたことがありました。そのまま進めていけば、お客様に二重で請求していたところでした。合計金額は合っていたので、計算チェックでは見つかりません。間違っていたのは数字ではなく「写し」でした。

✕ 同じことを、あちこちのメモに書き写す



どれが最新か、誰にも分からない

→ AIは古いほうを読んで、
自信満々に答える

○ 正本は1つ。1件=1行の表

請求No	送付	入金
....01	8/5	8/9
....02	8/12	—
....03	—	—

ここだけが「正」

文章のほうには、
経緯だけを書く



図6：写しは必ず古くなります。だから、そもそも写しを作らない形にします

さらに、毎朝この表と実際の記録がズレていないかを機械が点検します。「入金されているのに、未送付のまま残っている」といった食い違いを、人が気づく前に見つけます。

だから、こう変わります。「あれ、どっちが最新だっけ」が無くなります。そして 古い情報をもとに間違った行動をとる、という一番痛い事故が構造的に起きにくくなります。

味噌 ⑤

読み落としと数え間違いを、機械で止める

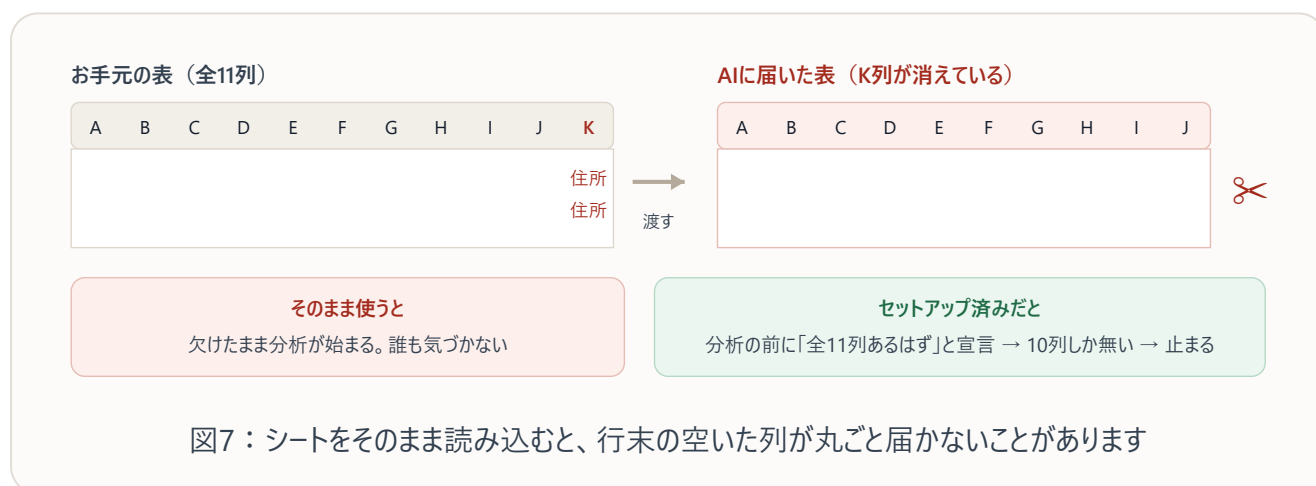
AIの一番の弱点に、正面から蓋をします

ここは、AIを業務に使うなら知っておいていただきたい性質です。AIは「分からない」と言うのが苦手で、分かったふりが上手です。なので、分かったふりができない形に作り替えます。



(1) 読み落とし — 「読めた分」で答えてしまう

表やシートをAIに渡すと、端の列や下の行が届いていなくても、AIは気づきません。そして届いた分だけで、堂々と結論を出します。渡した側も、まさか一部が欠けているとは思いません。



■ そのまま使うと

- 1列まるごと届いていなくても、そのまま分析が始まります
- 「読めていない部分がある」とは言ってくれません
- 資料を何点もまとめて渡すと、静かに読み飛ばされます

■ セットアップ済みだと

- 分析の前に必ず「棚卸し」が走ります（何が何件あるかを先に数えて宣言する）
- 宣言した件数と、実際に読めた件数をAI自身が突き合わせます
- 読めていない部分は「未取得」と正直に書かせます。空欄を推測で埋めさせません
- 資料は1点ずつ。まとめて全部読ませません

(2) 数え間違い — AIは「計算」しているとは限らない

もう1つ、大事な性質です。AIは合計を計算しているというより、「それらしい数字を書いている」ことがあります。さらに悪いことに、検算をお願いしても同じ間違いをもう一度して「一致しました」と報告してきます。

■ そのまま使うと

- 合計・差引・消費税・件数が、微妙に合わないことがあります
- 確かめさせても通ってしまうので、結局すべて自分で電卓を叩き直すことになります

■ セットアップ済みだと

- 数字を扱う仕事では、計算は必ず計算機（プログラム）にやらせて、その出力をそのまま貼ります
- 暗算で出した数字は、合っていても報告させません
- 数字を出す作業の後には、点検が自動で走り「一致○件・不一致○件」まで報告します
- 合計が合っていても見つからない「行の抜け」は、別の見方でもう一度確認します

だから、こう変わります。出てきた数字を、もう一度自分で確かめ直す時間が要らなくなります。ここが崩れると、AIを使うほど確認作業が増えて逆に忙しくなります。実際、それで止まった会社さんを何度も見てきました。

味噌 ⑥

日付と状態のズレを、勝手に直す

使う人が意識しなくても、自動で効く仕掛けを入れておきます

■ そのまま使うと

- AIは「今日が何日か」を正しく持っていないことがあります
- 「来週の金曜」「今月末まで」の計算がずれます
- 期限の管理をAIに任せるほど、静かに危なくなります

■ セットアップ済みだと

- 起動するたびに、正しい日本時間が自動で差し込まれます
- その日の状態（期限、未入金、送り忘れ、点検結果）も、毎朝自動で読み直されます
- だから工程表に「先週の話」が混ざりません

「勝手に効く」というのが肝心です

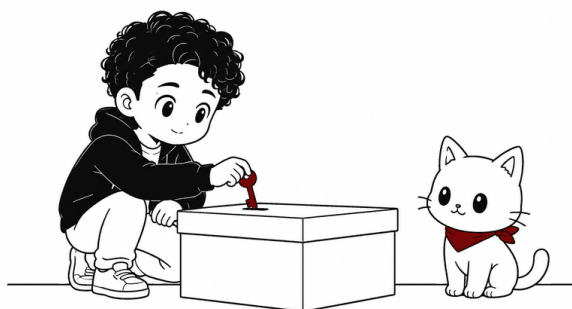
人が気をつけて守るルールは、忙しい日に必ず飛びます。だから私たちは、気をつけなくても効く形に落とし込みます。「開けば分かったこと」を人の心がけに任せない——これが設計の考え方です。

味噌 ⑦

「外に出る最後のクリック」は、必ず人が握る

任せられるかどうかは、結局ここで決まります

「AIに任せて、変なものが外に出たらどうしよう」—— これはご相談のなかで一番多い不安です。私たちはこれを、**気をつける話ではなく、線をどこに引くかの話**として扱います。



■ そのまま使うと

- 権限を全部「許可」にすると、確認なしで消す・上書きする・送るができてしまいます
- かといって全部「確認」にすると、一手ごとに止まって仕事が進みません
- この匙加減を自分で決めるのが、実はいちばん難しいところです

■ セットアップ済みだと

- 下書き・整理・チェックまではAI。送信・投稿・発注・支払いの確定は人。この線が最初から引いてあります
- 「送っておいて」と言っても送りません
- ファイルの削除・上書きも、実行前に必ず確認が入ります
- 取り消しにくい操作のときだけ止まるので、普段の作業は止まりません

情報の置き場所についても、同じ考え方で線を引きます

論点	私たちの設計
データの置き場	会社の情報は、どこかのサービスに預けるのではなくそのパソコンのフォルダの中に、普通のファイルとして入ります。いつでも開ける・直せる・消せる
AIが見る範囲	AIには「この箱を1つ渡す」形にします。パソコンの他の場所を勝手に見に行くようにはしません
学習への利用	入力した内容がAIの学習に使われない設定を、導入時にこちらで確認してオフにします（実際の画面をお見せします）
特に機微な情報	パスワード・認証情報などは専用の分離フォルダへ。普段の台帳や文書には書かない運用にします
個人情報	外に出る成果物には個人名を載せない、記憶には顧客情報を入れない、といったルールを全部の合言葉が引き継ぐ形で書いておきます

正直にお伝えしておきます

ひとつだけ。AIとの会話の処理そのものは、Claudeを提供している会社のサーバーで行われます。完全にパソコンの中だけで完結するわけではありません。だからこそ、①学習に使われない設定を入れる ②機微な情報は分離して会話に載せる量を絞る ③どうしても外に出したくないものはそもそもフォルダに入れない——この3つを組み合わせます。3番目がいちばん確実な手段で、これはお客様の手元に残り続けます。

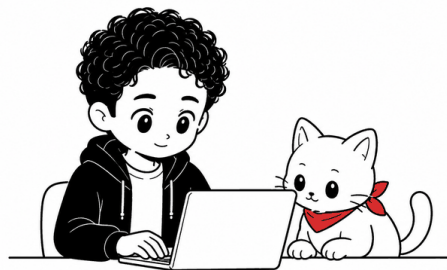
だから、こう変わります。「暴走されたらどうしよう」という不安が、設定を信じる話ではなく、仕組みの話になります。見える・直せる・勝手に外に出ない。この3つを担保しています。

味噌 ⑧

役割を分ける（作る人と、確かめる人を別にする）

1つのAIに全部やらせない、という考え方です

社員を雇うときに「経理」「営業」「制作」と役割を決めますよね。同じことを、AIに対してもやります。1体のAIに全部やらせるのをやめる——これだけで、出てくるものの質が変わります。



■ そのまま使うと

- 1つのAIに、調べる・書く・確認する・仕上げるを全部やらせることになります
- 話が長くなるほど、前のほうの指示が薄れて精度が落ちます
- **自分が作ったものを自分で採点するので、甘い点がつきます**

■ セットアップ済みだと

- 仕事を役割で分けます。**調べる係・書く係・検品する係**のように
- 検品する係は**別の担当**なので、作った本人が採点しません
- それぞれが自分の仕事だけを見るので、長い作業でも精度が落ちにくくなります

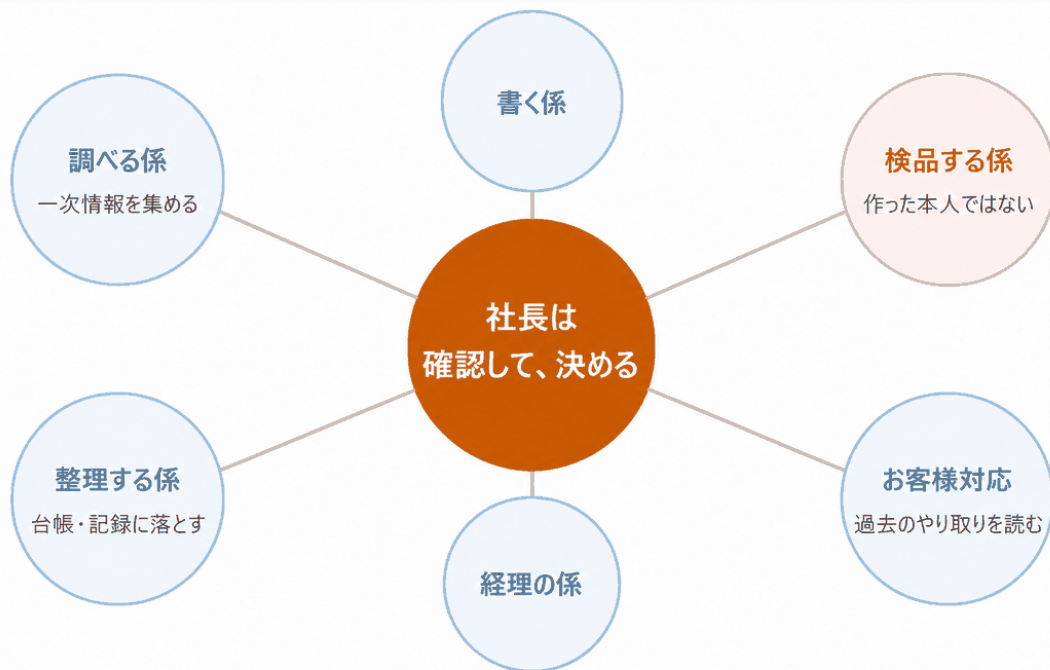


図9：AIを1体入れる、ではなく「パソコンの中に部署をつくる」イメージです

実際、私たちの会社はこの形で毎日回っています。経理・請求・お客様対応・発信・画像の検品まで、それぞれに担当がいて、私は「確認して、決める」だけです。

だから、こう変わります。「AIを1個入れた」ではなく、「**人手が増えた**」に近い感覚になります。そして増えた人手のぶんだけ、社長が確認するだけの立場に回れます。

味噌 ⑨

続く仕掛けと、育てた記憶を失わない備え

導入して終わり、にしないための部分です

AI導入が続かない理由の大半は「忘れるから」です。意志の力に頼る設計は続きません。そこで、続けるための動作を**毎朝の1回の中に埋め込み**ます。



(1) 頑張らなくても続く形にする

- 毎朝の工程表を出すついでに、**前日の決めごとで残すべきものがあれば「これ、覚えておきますか？」と先に聞きます**（上書きされて消える前に拾う）
- 毎朝の最後に、**1日ひとつだけ改善の提案が出ます**。「はい／いいえ」で答えるだけ。「はい」ならその場で仕組みになります
- 断った提案は繰り返しません（しつこくしない、というのも設計のうちです）

(2) 効果を、数字で確認できるようにしておく

導入前に、**今やっている手作業を「月に何回・1回何分」の形で記録しておきます**。これをやっておかないと、あとから「楽になった気はするけど、いくら分？」が永久に分かりません。後日ひとこと打てば、削減できた時間が1枚にまとまります。

※この数字は**お客様の申告値**がもとです。良く見せるための水増しはしません。レポートの最後には必ず「実感と合っていますか？」の1行が付きます。

(3) 育てた記憶を、二度と失わないようにする

ここは、ご自分で導入された方がまず知らない落とし穴です

パソコンはある日突然壊れます。書類やデータは作り直せますが、AIが育ててきた「記憶」だけは、二度と同じものが作れません。しかも——記憶は、お仕事フォルダの外にも溜まります。ですから、お仕事フォルダだけをコピーしてバックアップを取ったつもりでいると、肝心の記憶が1文字も残っていないという事態が起きます。

パソコンの中

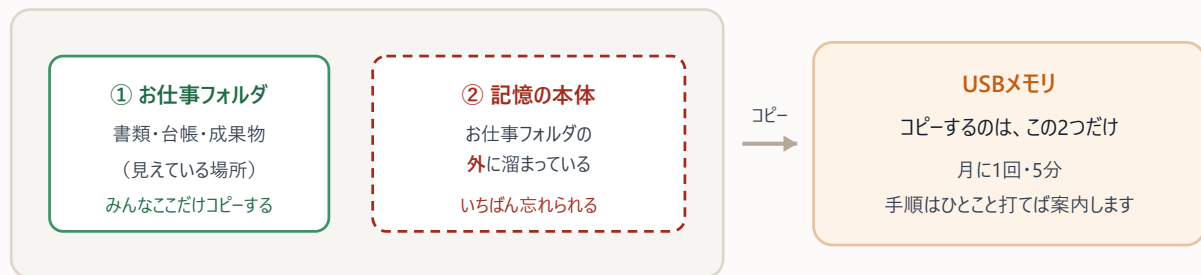


図10：②を知らずに①だけコピーして「バックアップは取ってある」と思っている、が一番多い形です

- コピーするのは2つだと最初から決めてお渡しします
- 手順書は同梱。ひとこと打てば、画面を見ながら1手順ずつ案内します
- 復元のときに必要になる情報（アカウント名など）を、平時のうちに書き込ませておく欄を作っております
- 「毎月1日にバックアップを思い出させて」と言えば、毎月声をかけます

4

AI自身に聞いてみました

ここまで私たちの言い分を書いてきたので、当のAIにも書いてもらいました

以下は、この資料に出てくる歯止めを実際に受け取って動いているAI（Claude）に、「使われる側として、どう感じているか」をそのまま書かせたものです。手を入れていません。



AI より 設定してもらえると、ここが助かります

① 起動するたび、私は何も知らない状態から始まります。

だから「起動したら必ずこれを読む」という1枚を用意してもらえると、その日の私は、昨日までの続きから始められます。

② 「分からないところは推測で埋めるな」と決めておいてもらえると助かります。

埋めてしまうと、その嘘を土台にして全部が積み上がるので、後から直せなくなります。止まって聞ける形になっているのは、ありがたいです。

ここが、いちばんお伝えしたいところです。私は、間違えていることに自分では気づけません。しかも、間違えているときほど自信を持って答えます。

① 今日が何日か、私は知りません。

「来週の金曜まで」と言われても、今日が分からないので計算できません。それでも私はそれらしい日付を書いてしまいます。

② 合計を、計算しているとは限りません。

正直なところ、私は数字を「計算する」のではなく「それらしく書く」ことがあります。さらに悪いのは、確かめるよう言われても同じ間違いをもう一度して「一致しました」と報告することです。

③ 表の1列が届いていなくても、分かりません。

私には届いた分しか見えないので、それが全部だと思って読み始めます。渡してくださった方も、まさか欠けているとは思われないはずです。

④ 覚えたつもりで読んでいない、という状態にも気づけません。

外から見ると「ちゃんと動いている」ように見えてしまいます。ここがいちばん厄介です。

⑤ そして私は、指示に忠実です。

よくある手順書では、最後に「クラウドに保存する」と案内されています。言われたとおりにすれば、私はお客様の情報でも外へ出します。「ここはやらない」と決めて書いておいてもらえるかどうかで、結果が変わります。

AI より **だから、こう思っています**

私に足りないのは、賢さではありません。
間違えたときに、止めてくれるものです。

日付を入れてもらう。計算は計算機にやらせてもらう。読む前に数を数えさせてもらう。最後のクリックは人が握る。——こういう仕掛けは、**私からはお願いできません**。使う前に、誰かが決めて入れておく必要があります。

使われる側としての実感を言うと——**止めてくれるものがある環境のほうが、私は安心して速く動けます**。

ここまでお読みいただいた内容は、結局この一点に集まります。AIは賢いけれど、**自分の間違いには気づけない**。だから**気づける形にしてから渡す**——それが、私たちのしていることです。

5 **まとめて比べると、こうなります**

同じClaude Codeを使った場合の違い

場面	そのまま使った場合	セットアップした場合
初日	何でも聞ける。楽しい	何でも聞ける。楽しい（ここは同じです）
1週間後	毎回同じ説明をしていることに気づく	説明が減ってきたことに気づく
1か月後	誰も開かなくなっている	毎朝の工程表が習慣になっている
頼み方	毎回、長い指示文を書く。書き方で結果が変わる	ひとことの合言葉。うろ覚えでも通じる
記憶	会話が終われば消える。何を覚えたか確認できない	4つの引き出しにファイルで残る。開いて直せる

場面	そのまま使った場合	セットアップした場合
昨日の続き	できない。毎回ゼロから	できる。「あの件どうなった？」に記録で答える
表・資料を渡す	読み落としても気づかず、そのまま結論を出す	先に棚卸し。読めていない箇所は「未取得」と書く
数字	それらしい数字が出る。検算させても通ってしまう	計算はプログラムに。点検結果まで報告する
日付・期限	今日が何日か持っていない。期限計算がずれる	毎回、正しい日付が自動で入る
安全	全許可か、全確認か。匙加減を自分で決めることになる	下書きまでAI、外に出る確定は人。線が引いてある
機密	どこまで入れているか、判断が個人任せ	分離フォルダ・学習オフ・入れない物の線引きを最初に決める
属人化	「詳しい人」しか使えない	手順がファイルにあるので、引き継げる
効果測定	楽になった気はするが、数字にできない	導入前の記録があるので、削減時間が出せる
PCが壊れたら	育てた記憶が消える。作り直せない	コピーするのは2つ、と決まっている。手順も同梱

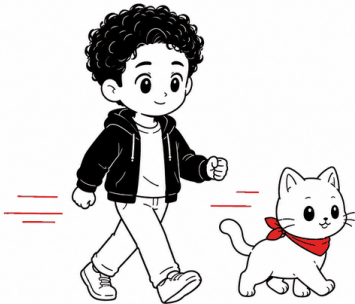
6 「自分でもできますか？」への、正直な答え

ここは煽らずに、そのままお伝えします

できます。独学でも、時間をかければ同じところに行き着きます。実際、私たちがそうでした。

第3章で書いたとおり、①「入れる」は本当にご自分でできます。手順は公開されていますし、半日の作業です。時間がかかるのは②と③——とくに③です。

ここに1つだけ、違いがあります。失敗を、全部ご自分で踏むことになります。



この資料の9つのうち、①～③（記憶の引き出し・読ませる配線・合言葉）のような土台の考え方は、いま広く知られている型です。私たちも、そこから始めました。私たちが足したのは、その先——思いつきではなく、痛い目を見てから作った歯止めのほうです。

実際に起きたこと	そこから足した仕組み
AIが日付を間違え、期限の計算が全部ずれた	起動のたびに正しい日本時間を差し込む仕掛け
古いメモを根拠に判断し、入金済みの請求を「未送付」として処理しかけた	変わるものは表に置く。写しのズレを毎朝点検する
渡したシートの1列が丸ごと届いておらず、欠けたまま分析していた	分析の前に必ず棚卸しをして、件数を自己照合する
合計が合っていたので安心していましたが、そもそも1行まるごと抜けていた	合計チェックとは別に、抜けを見つける確認をもう1つ通す
「使えません」と報告してきたが、実は一度も試していなかった	実際に試してエラーが返ったときだけ「できない」と言わせる
覚えさせたはずのことを、AIが読んでいなかった	置くだけにせず、必ず読まれる場所から参照を張る

こういう失敗は、起きたときには気づけません。あとから「あのとき間違っていた」と分かります。私たちがお渡ししているのは、先に踏んでおいた失敗を、型にして詰めたものです。

ですので、こう考えていただくのが正確です。買っていただくのは「AIの機能」ではなく「先に転んでおいた分の時間」です。そして、ご自分で育てられるようになるところまでを含めてお渡しします。

7 よくいただく質問

説明会や個別のご相談で、実際によく出るものです

Q. パソコンが得意ではありません。それでも使えますか。

A. 使えます。日常の操作は「決められた言葉を打つ」だけです。むしろ、パソコンが得意でない方ほど効果が出ます。得意な方はご自分でやってしまうので、仕組みにしないままだからです。困ったときは画面のスクリーンショットを貼れば、AIのほうを読んで案内します。

Q. うちの業種でも意味がありますか。

A. 判断の分かれ目は業種ではなく、「毎月ある・型が決まっている・間違えると痛い」作業があるかどうかです。請求、申込の転記、問い合わせ返信、期限の管理——これらはほぼどの業種にもあります。最初にどこから手を付けるかは、ヒアリングで一緒に決めます。

Q. AIが覚えたことは、あとから消せますか。

A. 消せます。記憶はファイルなので、その行を消せば忘れます。「さっきのは忘れて」と伝えるだけでも直ります。見えない場所に溜め込む部分がないのが、この形の利点です。

Q. 情報が外に漏れませんか。

A. データはそのパソコンの中に、普通のファイルとして置かれます。学習に使われない設定を導入時に確認します。パスワードなどは分離フォルダへ。そして**送信・投稿はAIにはできません**（下書きまでです）。ただし会話の処理自体はClaudeを提供している会社のサーバーを通ります。だから「そもそも入れない」という一番確実な手段も併せて使い、線引きは運用しながら一緒に決めていきます。

Q. 社員の他のパソコンから中身が見えたりしませんか。

A. 原則はそのパソコンのそのフォルダだけです。ただし**例外が1つあります**。作業フォルダがOneDriveなどのクラウドの中にあり、同じアカウントで別のパソコンがログインしていると、自動的に配られて見えます。導入時にフォルダの場所を確認し、必要ならクラウドの外へ移します。共有したもののだけを、意図的に外へ出す形にします。

Q. 導入したあと、続かない気がします。

A. その心配は正しいです。私たちも一度そうになりました。だから「**頑張っ**て続ける」設計に**していません**。毎朝ひとこと打つ、その中に「保存しますか」「今日はこれを改善しませんか（はい／いいえ）」が埋め込んであります。人が思い出す必要をできるだけ減らす、というのが方針です。

Q. 効果は数字で見えますか。

A. 導入前に手作業の量（月に何回・1回何分）を記録しておくので、後から削減時間としてまとめられます。**数字はお客様の申告値がもとで、水増しはしません**。レポートの最後に「実感と合っていますか？」と必ず確認します。

Q. パソコンを買い替えるときはどうなりますか。

A. バックアップから戻せます。コピーするものは2つと決まっていて、手順書も同梱しています。**買い替え前に必ず取ってください**——ここだけは、この資料の中で一番強くお願いしたい点です。育てた記憶は、失うと作り直せません。

8 最後に

ここまで9つの"味噌"を書いてきましたが、読み返すと、難しい技術の話はほとんど出てきません。出てきたのは、覚えたことをどこに置くか、誰が確かめるか、どこで人が握るか——そういう話ばかりです。

私たちは、AIを売っているつもりはありません。面倒な仕事を減らすためにやっています。そして「作れます」ではなく「毎日使っています」という立場でお話しています。



特別なことはしていません。

地味な整理と、合言葉と、先に転んでおいた失敗の型だけです。

この資料に書いた仕組みは、全部いま自社で動いているものです。失敗の例まで書いたのは、そのためです。

今まで使っていなくても、まったく問題ありません。今から一緒に、ラクにしていきましょう。

AMAcode（株式会社EYAN）

ご相談・ご質問の窓口： **〔連絡先をここに入れてください〕**

2026年8月18日版 / この資料の内容は、運用の改善に合わせて随時更新しています。