Helpfeel
このAIが、解決する

エンジニアの肌感覚を較正する



角田 徹 / Toru Tsunoda
株式会社Helpfeel　プロダクトエンジニア

Agile Japan 2026
2026年11月12日

「なんか怖い」は、どこから来る？

このコード、なんか怖い。

この設計、どうも座りが悪い。

うまく説明できる前に、異変に気づくことがある。

こうした「肌感覚」は、どう育つのだろう。

計算する前の「感覚」

この材料で、この太さなら危ない。
そのくらいは感覚で分からないと。

大学の工場実習で聞いた教授の言葉

直感、違和感、見当。呼び名はさまざまだが、
計算や説明より先に働くこれを、今日は「肌感覚」と呼ぶ。

計算が感覚と大きく違えば、「本当にそうか？」と立ち止まれる。

経験の裏には、大量の答え合わせがある

予想

「この変更なら30分くらい」
「サクッとかえってくるだろ」
「たぶんDBだろう」

現実

依存関係が複雑で、3時間。
描画が遅くてつかえない。
原因は、想像していない所。

変わるのは、知識だけではない。
どこを見て、何を疑うかが変わる。

自分の肌感覚を較正（Calibration）する

01

感覚で予想し、働きかける

02

結果を受け取る

03

期待との違いと条件から学ぶ

04

感覚と次の行動を調整する

計器の較正：条件をそろえ、基準と表示を照らす。

今日の較正：基準器はない。肌感覚を現実の結果に照らし、直し続ける。

任せ方や確かめ方も、エンジニアの肌感覚に含める。

島田 浩二「Let's Change Our World—Again」（スクラム祭り2026）より
「意思を持って働きかけ、結果を受け取り、期待との違いから学び、次の行動を調整する」から

経験則には、使える条件がある

前に危なかったから、今回も危ない？

前に大丈夫だったから、今回も大丈夫？

何を見て判断したか。

その手がかりが、今回も使えるか。

ここ数年で、AIに任せる範囲が広がった

最初は

コードの続きを提案してもらう

次に

複数ファイルの修正を頼む

さらに

調べて、作って、試すまで任せる

いまは

AI同士で、レビューや相談も

一つずつ指示する使い方から、
目的を渡して、試行錯誤ごと任せる使い方へ。

知識は外に置ける。アンテナは、自分の中に

「なんか変」を言葉にすると、人に話せる。
調べられる。AIにも聞ける。

この処理、二回動いても大丈夫だろうか。

でも、何も感じなければ、
質問そのものが生まれない。

仕事のなかで、自分は何を受け取ってきたか

自分で手を動かしていたとき

試した方法。
外れた予想。
条件を変えた理由。

AIに任せて、完了だけを受け取るとき

成果物は届く。
試行錯誤と学びは、
手元に届かない。

届いた結果を、最初の自分の予想とも照らせたか。

問題にしたいのは、AIではなく、完了だけを受け取る使い方。

全部、自分で触れる必要はない

役割	自分を較正してくれる現実
実装を担う人	**自分で書き、動かし、壊す**
運用を担う人	**本番のログや障害**
チームづくりを担う人	**チームの反応、会話の座りの悪さ**
プロダクトを担う人	**画面の前で30秒黙る、一人の利用者**

自分は、何によって較正されるのか。

生成されたコードは、全部を念入りに読んでいた

間違いを見抜けるか不安だった。
実行まで任せるようになり、次の問いで任せ方を見直した。

問題は、出てきたコードを誰がどう信じるかです。

橋本 翔「正気を疑う技術」

**実装の前に、AIと設計を調べ、話し合って、
検討すべき点とやらないことを決めた。**

対話ログをレビュー役のAIに渡すと、同じ観点でレビューしてくれた。

確認の観点そのものを、
レビューできるようになった

AIが、確認ポイントを先回りして提示してくれる。

「そこを見るべきか」

「ほかに見るべき所はないか」。

どこを確認するか、その観点からレビューできるようになった。

「動作を確認したい」とだけ頼んだら

新機能の確認には、検証用の環境とテストデータの準備が要る。
AIには、やり方を指定せずに頼んだ。

① 確認の手順を書いて返す

② 環境とテストデータを用意する

③ 動かして、確認まで終える

AIは、どの段まで進んだでしょう？

Helpfeel
このAIが、解決する

AIは、③まで進んだ

02 結果
表示と挙動を確認。私も同じ結果

①
確認の手順を
書いて返す

②
環境とテストデータを
用意する

③
動かして、
確認まで終える

Helpfeel
このAIが、解決する

私が狭く見ていたのは、任せられる範囲だった

01 予想
先に持っていた。書いてはいない

①
確認の手順を
書いて返す

②
環境とテストデータを
用意する

02 結果
表示と挙動を確認。私も同じ結果

③
動かして、
確認まで終える

03 違いと条件

内部ブラウザを操作して、
表示と挙動を確認できた。

04 調整

目的だけ与えて、
やり方は任せる選択肢を得た。

昔の経験は、今回も使えるか

設定の変更履歴で、特定の項目が変わった版だけを絞り込みたい。
それには、隣り合う版を比べる処理が要る。

比べる処理を、サーバーに置きたくない。

私の肌感覚。以前、一般の利用者向けのサービスで、アクセスが集中してサーバーが詰まった経験から。

AIに伝えた結論「比較は、クライアント側でする」

Helpfeel
このAIが、解決する

同じ「重い」で、違う負担を見ていた

履歴 → 読み出す → **サーバー**（比較を置きたくなかった） → 送るデータ量 → **ブラウザ**（既存と同じく、ここで比較）

私が気にしていた負担

サーバーで比較する計算量。

AIが気にしていた負担

ブラウザへ送るデータ量。
サーバーでの判定を勧めた。

AIの反論は、どこを確かめるかの手がかりだった。

持っていた感覚を、何に当て、何が変わったか

	持っていた感覚	当てた現実	肌感覚に返ったこと
レビュー	全部を念入りに 読むべきだ	AIが確認ポイントを 先回りして提示	確認の観点そのものを レビューできるようになった
確認	心の中の予想「①まで」 （書いてはいない）	AIは③まで進めた。 私も操作し、同じ結果	テストやドキュメントを たよりに検証も委譲可能
設計	比べる処理を、 サーバーに置きたくない	AIの反論（送るデータ量）。 想定する件数で確かめた	「重い」の中身が 分かれた

持っていた感覚を、現実に当てて、直した。

アジャイルは、何を較正していたのか

アジャイルの短いループは、作り手の「このくらいだろう」も直してきた。

成果物はスプリントレビューで、進め方はふりかえりで、検査と適応。
肌感覚も、その対象に。

加速したのは、試すこと

私自身、実装のコストは下がり、連動して調査やテストも
とても早くできるようになったと感じる。

大規模な調査でも、エージェント導入後にコードは約3割増えたが、
完了した課題の数は有意に変わらなかったと報告されている。

これも肌感覚。
提供までの期間で、答え合わせした。

大会テーマ「Agile Paradox -加速する世界と問い直す原点-」

AIのためのループと、人間のためのループ

AIのためには

テスト、型チェック、ログ。
必要な文脈を渡し、
失敗から直せる環境を作る。

人間のためには

最後の「できました」
だけを見ていないか。

AIのために設計するループを、
人間のためにも設計する。

作るのが速くなるほど、決める仕事が重くなる

速さ

試作がすぐ届き、
判断を求められる
回数が増える

難しさ

何でも作れるので、
何を作らないかを
決めにくい

変わりやすさ

前提が大きく変わり、
合わせた感覚も
すぐにずれる

**だから、チームの判断も、
較正と調整を続ける。**

人間という計器は、たやすく狂う

寝不足

「読んでるけど、
頭に入らない」

締切・焦り

「動いた。
もう出そう」

予想外の盛り上がり

「やべぇ、
楽しい」

AIの流暢な説明

「なるほど、
完全に理解した」

疲れるし、忘れるし、あっさり狂う。

チームの答え合わせを、仕組みにする

AIのためには	チームの判断のためには
テストを用意する	**予想を先に書き、見える所に置く** 計画、見立て、何を見たら当たりか
失敗から直せるようにする	**短い周期で、予想も答え合わせする** レビューとふりかえりで、成果物と一緒に。外れた予想は、リーダーから先に
ログを読めるようにする	**役割ごとに、現実に触れる場所を残す** 任せても、どこで触れるかは自分たちで決める。AIの要約で置き換えない

計器は狂う前提で。気合いではなく、仕組みで。

「なんか怖い」と言われたら

経験者に「なんか怖い」「十分に少ない」と言われたら、不慣れなメンバーも理由を聞ける。

「何を見て、そう判断しましたか？」

肌感覚がまだ育っていなくても、
答え合わせには参加できる。

聞く側は条件を知れる。答える側は根拠を点検できる。

自分の見立てを、検証の材料として渡す

任せる前に、確かめてほしい観点を添える。

- 「この変更では、一般ユーザーが閲覧できる範囲は変わらない」
- 「以前に保存したデータも、そのまま読み込める」
- 「通信に失敗しても、入力した内容は失われない」

自分で観点を挙げにくければ、詳しい人に聞く。

「どこを、なぜ確かめましたか？」
「その結果から、どう判断しましたか？」

この材料で、この太さなら危ない。
そのくらいは感覚で分からないと。

大学の工場実習で聞いた教授の言葉

計算しなくていい人になれ、ではなかった。
予想を持ち、結果と比べ、ずれから学び続けろ。

私の場合は、人の言葉で自分の死角に気づき、
手を動かして解像度を上げること。

AIが手を動かす世界でも、
自分を較正する現実は、手放さない。
エンジニアの肌感覚を、較正し続ける。

参考資料

島田 浩二
Let's Change Our World—Again（スクラム祭り2026）

橋本 翔
正気を疑う技術 - bugのないvibe codingを目指して

市谷 聡啓
AIが結果を生むほど、人間の「接地面」が必要になる

Addy Osmani
Critical thinking when agents do the work

Fiona Chen, James Stratton
AI Coding Agents Generate More Code, But Not More Software（報道）