



Givery, Inc.

INSTRUCTOR OPERATIONS MANUAL

# 運営マニュアル(講師向け)

生成AI実務研修(7時間 × 2日間)

講師用 運営マニュアル

---

提供	Givery 株式会社
配布先	協栄産業株式会社 御中(講師・運営内部資料)
研修形式	VS Code × Claude Code CLI/オンライン(Zoom)・オンサイト両対応
版数	v1.0(2026年6月11日)

---

本書は講師・運営スタッフ向けの内部資料です。受講者への配布は行いません。時間表記は分数のみ([Nmin])で記載しています。

# 1. 案件サマリと役割分担

---

## 1.1 案件サマリ

項目	内容
顧客	協栄産業株式会社 様
研修時間	7 時間 × 2 日 = 14 時間(各日 [420min] + 昼休憩 [60min])
受講人数	1~15 名想定(現時点で 7 名規模)
開発環境	VS Code × Claude Code CLI(統合ターミナル経由)。プラグイン版は紹介のみでメインにしない
主軸	Spec Kit による仕様駆動開発。比較視点として Kiro 式(3 ファイル固定)を体験
実施形態	オンライン(Zoom)/オンサイト両対応。オンサイトは会場 1 拠点のみのため事前確定必須
Day2 題材	メイン「部品在庫・納期照会ダッシュボード」(ブラウザで動くWebアプリ)。営業寄り別トラック「技術営業 提案アシスタント」。学習用に「在庫アラート通知ツール」を Spec Kit と Kiro 式の両方で作る

## 1.2 役割分担

役割	担当	当日の主な責務
講師	安田 光喜(Givery)	全進行、講義、デモ、受講者対応、ヒント提供
営業	加藤 智也(Givery)	開始時の挨拶、チャット監視補助、Day 終了後のフォローアップ、議事録

---

## 2. 受講者背景とレベル差対応

---

### 2.1 受講者の現在地

- 協栄産業様は Claude Code × Spec Kit での開発効率化を社内検証中。研修はその加速が目的
- 中核 3 名(湯浅様・滝田様・青柳様)は CLI ベースの実機運用を開始済み。質問の解像度が高い
- 若手 4 名はプロジェクト途中参加で、前回提案資料も未読。Claude Code 自体がほぼ初見
- つまり同じ教室に「すでに毎日触っている層」と「これから触る層」が同居する。レベル差吸収が進行上の最重要ポイント

### 2.2 レベル差の吸収方針

場面	経験者(中核 3 名)への対応	初学者(若手 4 名)への対応
座学・基礎演習(Day1 S01~S03)	「検証中に困った点」を随時拾い、講義に織り込む	用語が出るたびに 1 行定義を口頭で添える
ハンズオン共通	本編を早く終えたら hints/ の発展課題へ誘導	各 Step の冒頭で「いま打つコマンド」を画面に出したまま進める
巡回優先度	原則放任、質問が来たら深い議論に付き合う	巡回・チャット確認は初学者を優先。Step 完了チェックは名指しでなく全体に投げる
詰まりの検知	-	各 Step 終わりの「詰まった点チャット投下」を必ず実施し、無言の人ほど注視

経験者を「教える側」に回さないこと。本研修は個人ワーク原則(§ 8 参照)であり、経験者の学習時間も等しく確保します。

---

## 3. Day 1 進行詳細

### 3.1 分単位進行表

合計 [420min] + 昼休憩 [60min]。小休憩 [10min] は S02 後・S04 後に挿入します(下表は本編 [420min] の内訳)。

枠	区分	講師アクション
[60 min ]	S01 Claude Code 基礎と AI 最新動向 ●	挨拶 [5min] → 技術動向 [20min](Claude/GPT/Gemini の違い・公式情報源・Agentic Coding)→ Claude Code 基礎 [25min](モデル/Effort・5部品・設定階層、座学編を投影)→ 情報キャッチアップ [10min]。Spec Kit 最新リリース要約 ◆デモ
[60 min ]	S02 Claude Code 基礎演習 ○	TRY IT 01~03。/init /context /compact /clear・モデル/Effort 切替・ユーザー/プロジェクト設定の優先・5部品の地図(notes/parts_map.md)。巡回中心、初学者優先
[60 min ]	S03 環境セットアップ確認 + コマンド演習 ○	環境確認 [15min](claude/python/pytest/uvx)→ TRY IT 04 既存コマンド → TRY IT 05 自作コマンド。詰まりはこの枠で申告させる
[60 min ]	昼休憩	午後の Skills/Subagents デモ環境を確認
[70 min ]	S04 Skills 基礎演習(3段階) ○	TRY IT 06 単機能(meeting-summary)→ 07 Subagent並列(multi-review)→ 08 画像読込(ui-from-mock)。段階の違いを観察させる
[50 min ]	S05 Subagents 基礎演習 ○	TRY IT 09 単体(code-reviewer)→ 10 コマンド起点3体並列(/triple-review)。逐次との待ち時間差をメモさせる
[90 min ]	S06 仕様駆動開発の基礎 + Spec Kit ○	SDD概念 [15min](Spec Kit/Kiro思想)→ HANDS-ON 11 [75min]:speckit-inventory で /specify→/plan→/tasks(在庫アラート通知ツール、3ファイルまで)。各STEP終わりにチャット確認
[30 min ]	S07 質疑応答・振り返り・閉会 ●	key takeaways → 全員1行チャット → 質疑 → 宿題(自分の業務で AI に任せたい定型作業1つ持参)→ アンケート

### 3.2 チャット確認タイミング

- S05 の Step 1~6 各終了時:「ここまでで詰まった点」を全員チャット投下。投下がない人へ口頭で状況確認
- S01 終了時(キャッチアップするソース)、S06/S07 終了時(明日試すこと 1 行)もチャット回収。オンサイト時はシート記入に読み替え
- オンライン時は Zoom チャットを講師サブモニタに常時表示。デモ中は加藤がチャット監視を代行

### 3.3 デモ切替の判断

- S05 の Spec Kit ハンズオンで uvx が動かない受講者が 3 名以上出た場合、Step 2 を講師画面共有のデモに切り替え、受講者は spec.md の中身を手元 VS Code で写経する形に変更
- S04 Skills / S05 Subagents のテンプレが不調なら、配布済みの完成テンプレ(.claude 同梱)をそのまま動かして観察に切り替える

## 4. Day 2 進行詳細

### 4.1 分単位進行表

合計 [420min] + 昼休憩 [60min]。小休憩 [10min] は S01 の中間と S05 後に挿入します。

枠	区分	講師アクション
[90 min]	S01 復習 + Spec Kit 完走 ○	復習 [20min](/context で Day1到達点・自作コマンド・triple-review)→ HANDS-ON 06 [70min]: テスト先行→/implement で完走→追加機能3ステップ。宿題(業務課題)をチャット投下させる
[80 min]	S02 Kiro 式ハンズオン ○	HANDS-ON 07.kiro-inventory で同一要件を別プロジェクト: requirements(EARS)→design→tasks→テスト先行で実装。kiro-spec Skill 同梱
[30 min]	S03 Spec Kit と Kiro 式の比較 ●○	2プロジェクトを並べ5観点でメモ [20min] → 使い分けマトリクス提示 [10min]。判断基準を1つ決めさせチャット投下
[60 min]	昼休憩	午後の TDD・本題材の環境を確認
[40 min]	S04 テスト駆動開発の基礎 ●	Red→Green→Refactor [25min] → 配布フォルダの TDD 設定確認 [15min](day2/tdd_setup の pytest が通ること)
[50 min]	S05 テスト駆動開発の実践 ○	HANDS-ON 08.day2/tryit_tdd の parse_daily_report を Red→Green→Refactor で2周
[90 min]	S06 メイン題材開発 ○	HANDS-ON 09.apps/parts-dashboard(別トラック proposal-assistant)に SDD×TDD で機能を1つ追加。docs先読み→spec→Red→Green→ブラウザで動作確認。巡回中心
[40 min]	S07 発展課題 + 振り返り・閉会 ●	発展課題 [20min](早く終わった人向け追加課題)→ 成果物整理(README/commit/tag)→ 全員1行 → 質疑 → 教材サイトURL配布 → アンケート

### 4.2 チャット確認タイミング

- S04 の各 Phase 終了時: 「Red 確認できた」「Green になった」を短文で投下させ、進度を可視化
- S01/S05/S06 で Green に届かない人が出たら、全体を 5 分まで待ち、それ以上は hints/(答え合わせ)を案内して合流させる
- S07 の発展課題・各自業務への適用は個別性が高いため、チャットではなく巡回(オンラインは個別 DM)で拾う

## 5. デモ準備チェックリスト

研修前日までに講師環境で全デモを通して再生し、所要時間を実測しておきます。当日朝はチェックのみ。

デモ	使用枠	事前準備	所要 (本 番)
/triple-review 並列レビュー + Skills 3段階(meeting-summary/multi-review/ui-from-mock)	Day1 S04	3 ツールのログイン・最新版更新。同一お題「議事メモから REST API 1 本」の議事メモを 3 ウィンドウに配置。比較結果サマリ表のスライドを準備。失敗時用の録画を撮っておく	[15 min]
Kiro 式デモ(3 ファイル生成 → コード)	Day1 S05	Kiro 風プロンプト(requirements / design / tasks 固定)を .claude/commands/ に登録済みの状態にし、空リポジトリを用意	[20 min]
Spec Kit デモ( <code>uvx specify init</code> → /specify → /plan → /tasks → /implement)	Day1 S05	<code>uv</code> 導入済み確認、 <code>specify init</code> の初回キャッシュを済ませる。議事メモ素材を手元に展開。ネットワーク断時に <code>init</code> 済みリポジトリのコピーを保持	[20 min]
CLAUDE.md Before/After	Day1 S03 / Day2 S01	「規約なし」「規約あり」の 2 リポジトリを用意し、同一プロンプトの出力ログを事前取得(ライブが崩れた時の差し替え用)	[10 min]
Skill ライブコーディング(会議メモ → spec.md)	Day2 S01	SKILL.md の骨子(frontmatter まで)を手元メモに準備。10 分で組み切れるよう description 文面は事前に推敲しておく	[10 min]
Hook ライブ実装(Write 直後に lint、失敗でブロック)	Day2 S01	<code>ruff</code> 導入済みのサンプルリポジトリと、わざと lint エラーになるコード片を用意。settings.json の記述例を手元に	[10 min]
パイプラインデモ(PR URL → 設計 / セキュリティ / パフォーマンスの並列レビュー → 統合レポート)	Day2 S01	レビュー対象のサンプル PR を講師リポジトリに事前作成。Subagent 3 体 + 集約 Skill を組んだ <code>.claude/</code> を完成状態で保持。実行に数分かかるため、開始したら待ち時間に構成図を解説する段取り	[15 min]

## 6. 配布物一覧

受講者配布は `kyos-handson.zip` に集約します(Windows / Mac 両対応、Git 操作なしで展開して使える構成)。事前セットアップ案内とともに研修 1 週間前に送付します。

同梱物	内容	使用枠
<code>README.md</code>	展開手順、動作確認コマンド、フォルダ案内	事前 + 全日
<code>hints/</code>	各ハンズオンの段階別ヒントと完成形コード(Day1 Step 別 / Day2 Phase 別)。詰まった受講者の合流用	Day1 S05 / Day2 S04
議事メモ素材	学習用「在庫アラート通知ツール」議事メモ(speckit-inventory / kiro-inventory の docs)、本題材の docs(部品ダッシュボードの社内メモ・会社情報・parts.csv、提案アシスタントのヒアリングメモ・架空案件「製造業向け予知保全」。いずれも架空・ダミーデータ	Day1 S05 / Day2 S02
<code>_reference_agent/</code>	各題材の参考解(在庫アラート通知ツールの完成形、ダッシュボード/提案アシスタントの追加機能実装例)。救済・答え合わせ用。受講者には「研修中は開かない」と案内	Day2 S04~S05
<code>requirements.md</code> テンプレ	ユーザーストーリー × 受け入れ基準(Given-When-Then)の記入枠付きテンプレート	Day2 S03
ガイドライン雛形	協栄産業様向け生成AIガイドラインたたき台(A4 5 ページ)	Day1 S02
<code>.claude/</code> 雛形	CLAUDE.md ひな形、Skill / Command / Hook の最小サンプル	Day1 S03 / Day2 S01
メモシート類	5部品の地図(parts_map)、キャッチアップするソース、各自の業務課題メモ	Day1 S01・S02 / Day2 S02

運営・講師向けの記載や md / html 元データ、変換スクリプトは ZIP に混入させないこと。最終 ZIP は送付前に Windows / Mac 双方で展開テストを実施します。

## 7. 詰まりポイント先回り

詰まりポイント	想定原因	介入方針
Claude Code 認証エラー	初回ログイン未完了、組織アカウントの切替ミス、トークン期限切れ	<code>/login</code> をやり直させる。ブラウザ既定が社内 SSO に飛ぶ場合はプライベートウィンドウで認証。事前セットアップ案内で <code>claude --version</code> と簡単な 1 往復まで済ませてもらい、当日発生を最小化する
uvx 未導入で Spec Kit が始められない	uv 本体が未インストール、PATH 未反映	公式インストールスクリプトを案内し、VS Code のターミナルを開き直させる。5 分で復旧しなければ <code>init</code> 済みリポジトリ(hints/ 同梱)を展開して合流
Spec Kit init 失敗	社内プロキシで GitHub テンプレート取得がブロック、既存ディレクトリとの衝突	空ディレクトリで再実行。通信起因なら初回成功者の <code>.specify/</code> をコピー配布。それも難しければ講師デモ + 写経に切替 (§ 3.3)
pytest 未導入でテストが走らない	Python 環境はあるが pytest が入っていない、仮想環境の有効化漏れ	<code>pip install pytest</code> (または <code>uv pip install pytest</code> )。Python 以外の言語の受講者には言語別の同等フレームワーク(Jest 等)を許容し、Red → Green の往復だけ守らせる
レベル差で進度が乖離する	経験者が先行し、初学者が Step 途中で停滞	先行者は hints/ の発展課題へ。停滞者は完成形コードで合流させ、「全 Step を自力で」より「全員が一気通貫を体験」を優先。Phase / Step の開始は全体同期、終了は非同期とする

## 8. 進め方と個人ワーク原則

---

### 8.1 Spec Kit と Kiro は「全員が順番に両方」

- 新フローではグループ分けはしない。全員が Day1 S06 + Day2 S01 で Spec Kit、Day2 S02 で Kiro 式を、同一要件「在庫アラート通知ツール」で順番に作る
- 同じ要件を別プロジェクト(speckit-inventory / kiro-inventory)で2回作るので、全員が両方式を必ず1度ずつ手で体験する
- S03 で2つを並べて5観点で比較し、講師が使い分けマトリクスを提示する

### 8.2 レベル差の吸収

- 経験者: 各演習を早く終えたら exercise の発展課題・S07 の追加課題へ誘導(放任気味、質問が来たら深く付き合う)
- 初学者: 各演習の冒頭で「いま打つコマンド」を画面に出したまま進める。巡回・チャット確認は初学者優先
- 演習の開始は全体同期、終了は非同期。Green に届かない人は hints/ で合流させ「全員が一気通貫を体験」を優先する

### 8.3 個人ワーク原則

- ペアプログラミング、グループディスカッションは実施しない(運営ルール)
  - 共有は「講師が指名して画面共有」「全員チャット投下」の2形式のみ。受講者同士の相互レビューはさせない
  - S03 の比較も、受講者同士の議論ではなく、講師が成果物を見せながら使い分けマトリクスを提示する形で進める
-

## 9. タイムオーバー時の調整弁

---

### 9.1 削ってよいもの

対象	削り方	回収できる時間
Day1 S01 情報キャッチアップ	デモを1本に絞り、ソース一覧は座学編の参照に切替	[10min]
Day1 S04 Skills 段階3	ui-from-mock は講師デモの観察のみにし、手元実行は任意化	[15min]
Day2 S01 追加機能3ステップ	追加機能を1つに縮小(完走の下限は満たせる)	[15min]
Day2 S07 発展課題	発展課題の枠を縮め、振り返り・閉会を優先	[10min]~[20min]

### 9.2 絶対に削らないもの

- Day1 S06 Spec Kit で /specify→/plan→/tasks(在庫アラート)。Day2 開発の前提になる仕様化体験
  - Day2 S01~S02 Spec Kit 完走 と Kiro 式(同一要件を別プロジェクト)。両方式を全員が手で触る
  - Day2 S03 使い分けマトリクス提示(Spec Kit / Kiro 比較は商談時の明示要望)
  - Day2 S05~S06 テスト駆動の Red→Green 往復と、本題材へのSDD×TDD適用(本研修の背骨)
  - S07 のアンケートと、Day2 の教材サイト URL 配布
-

## 10. アンケートと研修後フォロー

---

### 10.1 アンケート設問(Google Form、5 問)

1. 本日の内容の難易度はいかがでしたか(易しすぎる～難しすぎるの5段階)
2. 明日からの業務で最初に試したいことを1つ教えてください(自由記述)
3. Spec Kit と Kiro 式のどちらが自分の業務に合いそうですか(Spec Kit/Kiro 式/使い分けたい/まだ判断できない)
4. ハンズオンの時間配分はいかがでしたか(短い～長い の5段階)
5. 追加で学びたいテーマがあれば教えてください(自由記述)

Day1 / Day2 とも同一設問で実施し、回答の変化を研修後フォローの材料にします。回答時間は各日 [2min]～[3min]。

### 10.2 教材サイト閲覧案内

- Givery のチュートリアル教材サイトを研修料金内で閲覧可。期間は研修翌月末まで
- 受講者専用 URL は Day2 S06 のアンケート案内と同時にチャットで配布する。口頭だけで終わらせない
- 「閲覧期限」と「期限後はアクセス不可」を明言しておく(問い合わせ防止)

### 10.3 研修後フォロー

- Day2 終了後 1 週間以内: 加藤から窓口(湯浅様)へ実施報告とアンケート集計の共有
- 1 か月後(教材サイト閲覧期限の前): 加藤から閲覧期限のリマインドと、次フェーズ(伴走支援)の打診
- 伴走支援は今回スコープ外。研修中に伴走の相談が出た場合は「別途ご提案できます」とだけ答え、講師は深入りしない

---

文書末尾。



協栄産業 生成AI実務研修

© Givery, Inc. All Rights Reserved.