



JBS 中級コース 案1 / 3日間オリエンテーション

AI 駆動開発で 要件定義から CI/CD まで一気通貫

Claude Code を主軸に、社内ヘルプデスクチケット管理 API を題材として、ヒアリングメモから動く本番準備済みコードまでを3日間で構築します。

日程

5/14・5/21・5/28

時間

各13:00~16:00

形式

オンライン(Zoom)

講師

Givery 安田 光喜

この研修で何を持ち帰るか

既存実装をなぞる研修ではなく、自分の業務で再現できる作り方を持ち帰ってもらうのが目的です。

3日間の到達目標

- 顧客ヒアリングから要件定義書・ER図・API仕様・シーケンス図を Claude Code で生成し、自分の目で精査できる
- FastAPI + SQLAlchemy 2.0で実装、pytest でカバレッジ80%以上を確保できる
- Docker マルチステージ build と GitHub Actions の CI を実機で動かして緑に保てる
- @claude メンションで Issue から自動 PR の流れを1周完走できる

この研修がカバーしないこと

- 本番認証(Entra ID OAuth)の実装 — JWT スタブで止めます
- Power BI / Teams 連携 — Phase 2のスコープ
- Kubernetes / 本番デプロイ — 研修後の各部署で対応
- フロントエンド開発 — 別途 Next.js で実装されている設定

日程と各日の重心

3日間は成果物を順に引き継ぐ構成です。Day1の設計が粗いと Day2の実装で詰まり、Day2のテストが薄いと Day3のCI で破綻します。

| Day | 日付 | テーマ | 主成果物 |
|------|---------|--------------|--|
| Day1 | 5/14(木) | 要件・設計 | 要件定義書、ER図、API仕様、シーケンス図 |
| Day2 | 5/21(木) | 実装・テスト | 動く FastAPI、ユニットテスト+結合テスト |
| Day3 | 5/28(木) | Docker・CI/CD | docker-compose、Actions 全ジョブ緑、@claude 自動 PR |

各日の時間配分

| 時間 | 内容 | 所要 |
|-------------|--------------|-----|
| 13:00~13:10 | オリエンと前日の振り返り | 10分 |
| 13:10~14:15 | 主要セッション 前半 | 65分 |
| 14:15~14:25 | 休憩 | 10分 |
| 14:25~15:50 | 主要セッション 後半 | 85分 |
| 15:50~16:00 | まとめと翌日への引き継ぎ | 10分 |

社内ヘルプデスクチケット管理 API

JBS 社内の IT 利用部門と情報システム部門の間で発生するヘルプデスク問い合わせを管理する API を題材にします。

受講者の立場設定

情報システム部門に異動初週の2年目エンジニア

部長から「3日でプロトタイプを動かしてほしい」と依頼を受けたところです。手元には顧客ヒアリングメモだけ。前任者が作った既存システムは動いてはいるものの、雑な箇所が多く、改善余地を含めた現状仕様の把握から始まります。

想定規模

- 社員500名(うち IT 部門エージェント15名)
- チケット起票件数 月800件規模
- ピーク同時50ユーザー(朝9:00-10:00)
- データ保持3年、監査ログ5年

公開デモ環境

Day1開始時にブラウザで触っていただく公開デモが用意されています。

- 管理UI: <https://0514-28cc.give-app.net/>
- Swagger UI: <https://0514-28cc.give-app.net/docs>

このデモは意図的に改善余地を含む雑な実装です。Day1で観察し、Day2で正しく実装し直します。

使うツールと役割分担

中級コースは Claude Code を主軸に、補助ツールを組み合わせます。

| ツール | 役割 | 使い所 |
|--------------------------------------|-----------------|-----------------------|
| Claude Code | 主要 AI エージェント | 全Day。設計生成、実装、テスト、レビュー |
| FastAPI 0.115 | Web フレームワーク | Day2実装 |
| SQLAlchemy 2.0 | ORM(async対応) | Day2実装 |
| pytest + pytest-asyncio + pytest-cov | テスト | Day2-3 |
| ruff + mypy | Lint と型チェック | 全Day |
| Docker / docker-compose | 環境のコード化 | Day3 |
| GitHub Actions | CI / CD | 全Day(緑を維持) |
| trivy | イメージ 脆弱性スキャン | Day3 |
| Vercel | サーバレスホスティング | Day3発展課題 |

なぜ Claude Code 主軸か

初級研修で扱った GitHub Copilot が補完中心の支援ツールなのに対し、Claude Code はファイル横断の編集と非同期エージェントが強みです。中級では複数ファイルにまたがる実装やリファクタリングが増えるため、こちらが適しています。

研修開始時に動く状態にしておくもの

下記が全て動く状態で研修に臨んでください。社内承認が必要なものは早めに着手をお願いします。

必須環境

- Python 3.11系インストール(`python --version` で3.11.x が返る)
- Node.js 20以上(Claude Code CLI が動く)
- Claude Code ログイン(`claude` コマンドでチャット起動を確認)
- VSCode と Docker Desktop(Day3で起動)
- Git のユーザー名・メール設定済み
- 受講者用リポジトリのクローン(事務局共有の EMU 上 URL を `git clone`)

推奨準備

- 配布物 `README.md` のクイックスタートを一度実行して、CI が緑になることを確認
- 公開デモ URL にブラウザでアクセスして表示を確認
- Claude Code で `claude` 起動後、配布物について簡単に質問してみる

EMU 環境関連の注意

個人 GitHub には push しないでください。clone 先のローカルフォルダ名は `0514-<EMUアカウント名>` です。ハンズオンガイド中の `0514-21-28_配布` という表記は読み替えてください。

研修中の心構え

3日間を有意義にするための前提を共有します。

手を動かす時間が中心

座学は最小限です。各セッションで Claude Code に指示を出し、生成された成果物を自分の目で読み返し、修正指示を返す、というループを回します。受け身では身につけません。

生成 AI を盲信しない

「動いた = 完成」と扱わない

Claude Code は数値要件を抜かしたり、ヒアリングにない要件を勝手に追加したりします。生成された成果物は必ず自分の目で読み返し、修正指示を出すのが基本動作です。

短く区切って確認する

Claude Code に長い指示を一度に投げるより、短く区切って1ステップずつ確認するほうが出力品質が安定します。「生成 → 読み返し → 修正指示」の往復回数で品質が決まります。

分からなければZoom チャットへ

質問は `[Q]` を頭に付けて投稿してください。エラーログ全文の貼り付け歓迎です。詰まった方には休憩時間も含めて個別フォローします。1人で5分以上動けなくなったら遠慮なく声を上げてください。

担当講師

講師紹介

安田 光喜

(生成AIエバンジェリスト)



プロフィール／経歴

生成AI活用普及（技術検証、研究開発、研修設計、研修講師、受託開発）を中心に活動中。
札幌大学において人工知能やデータサイエンスも指導しながら、AI社員と協働する次世代の働き方を実証中。

【経歴】

公立ほこだて未来大学 大学院（メディアデザイン領域）を卒業し、街づくり会社での複合商業ビルの立ち上げにおける情報インフラ整備やデザイン全般業務の責任として関与。その後デザイン事務所を立ち上げ、飲食店を中心としたデザイン業務や美術館展示のアートコンテンツ開発などを行う。また、地域ブランディングにも注力し、市主催の企画運営業務を行なった。2018年10月「小中学生向けのプログラミング教育」に注目し、函館市で初めての小中学生向けプログラミングスクール「D-SCHOOL北海道」を立ち上げて運営。その後北海道のドラッグストア「サッポロドラッグストア」からスクールの買収を受け、教育を専門とするグループ会社「株式会社シーラクス」にて技術開発責任者として運営。スクール設計・運営、20以上の自治体連携（イベントや出張教室運営）、大人向けプログラミングスクールメンターなどさまざまな「次世代IT教育」に取り組む。

Dify公式資料の日本語翻訳や、SecondMeアンバサダーなど生成AI最新トレンドに注力。

#ClaudeCode #Cursor <AI駆動開発・AI駆動PM・スペック駆動開発> #Antigravity #GithubCopilot #Dify #n8n #KamuiOS

得意分野／多言語経験

- ・生成AIをはじめとした世界最先端トレンド
- ・アプリケーション開発/自動化プログラム環境構築
- ・教育（大学講師、研修、スクール運営など）

資格／認定

- ・全国統一生成AI活用技能試験：S1（上位3%）
- ・Google Cloud - Generative AI Leader
- ・生成AIパスポート / 基本情報技術者 / ITパスポート
- ・AdobeIllustrator認定エキスパート / G検定
- ・修士（システム情報科学研究）

講演／研修実績など

- ・教育機関(小中高大)でのITトレンド講演多数
- ・自治体と連携した地域教育実績多数
- ・上場企業社員対象の生成AI研修多数

その他

私はこれまで何か1つの技術に特化するのではなく、ものすごくスピードの早い時代の流れにあわせ、さまざまな興味に対してマルチに取り組んでまいりました。ですので、1つの視点ではなく高い視点から広い視野での、“今後につながる”指導を心がけています。わかりやすくかつ 親身な指導を行ってまいりますので、ぜひお気軽にお声掛けください。

3日間の研修運営

- Day1～Day3共通で講師が進行
- JBS 側オブザーバーは Zoom 表示名に「[オブザーバー]」を付けて参加
- 録画あり(Givery 側 Zoom)。研修後に JBS 事務局経由で社内共有
- JBS 事務局からの連絡先は研修当日チャットで案内

研修前のよくある質問

Q. Claude Code を初めて使います。大丈夫ですか

はい、研修内で操作含めて扱います。事前にインストールとログインだけ済ませておけば、コマンド操作は Day1開始時に講師が画面共有で実演します。

Q. Python の経験が浅いです

FastAPI と SQLAlchemy は研修内で必要な範囲を扱います。型ヒントや async/await の基礎は事前に Python 公式ドキュメントで予習しておくくとスムーズです。実装は Claude Code が補助するので、構文を覚えていなくても進められます。

Q. Day1を欠席した場合は

Day1で作るべき設計書の完成版を講師から共有します。Day2開始時に `docs/` 配下に配置すれば追いつけます。Day1の録画は研修後に共有されるので、それも合わせて視聴してください。

Q. Docker Desktop が会社で使えません

Day3では Docker Desktop が前提ですが、Rancher Desktop や Colima などの代替でも基本的に動きます。ライセンス上の問題がなければそちらで進めてください。どうしても用意できない場合は、Day3講師実演の視聴のみでも構いません。

Q. Vercel アカウントが作れません

Day3の Vercel デプロイは発展課題なので必須ではありません。Vercel が作れない場合は CI が緑になる時点までを必須到達点としてください。

Q. 研修後にコードを業務利用してもよいか

受講者個人の業務利用、所属部門内での共有は可能です。社外配布や商用利用は事前に Givery への連絡が必要です。生成された AI コードの社内利用ガイドラインは各部門の方針に従ってください。



© 2026 Givery, Inc. All rights reserved.

制作 Givery / JBS 中級コース 案1オリエンテーション資料 / 2026-05-14配布