職務経歴書

hironeko

テックPdM / エンジニアリングマネージャー

職務要約

Webエンジニアとして10年、直近はPdM・テックリードとしてプロダクトの要求定義と技術基盤の双方を担当しています。利用者の業務、市場の需要、監査要件から達成すべきゴールと採らない方針を定義し、組織規模や運用体制から逆算したアーキテクチャ選定・実装まで一貫して担います。個社要望をドメイン課題として再定義した汎用機能化、IPO監査対応システムの0→1立ち上げ、本番Hotfixリリース年65件から2件、開発リードタイム5分の1への短縮など、プロダクトと開発プロセスの双方で成果を出してきました。開発フローを可視化するGitHub CLI拡張（gh-trends）を個人開発・公開しています。

職務経歴

O社

2025.08 — 現在

ヘルスケア系サービス / PdM・テックリード

技術環境	PHP · Laravel · JavaScript · AWS · Docker · Terraform · GitHub Actions

個社要望をドメイン課題として再定義し、汎用機能へ転換

- 特定顧客向けの個社対応として起票された案件に対し、既存機能を流用する短期対応を採らず、「システム上に代理店という概念が存在しない」というドメインモデルの欠落として課題を再定義。
- 複数の保険商品を扱う代理店の増加を前提に、解決された状態、やらないこと、実装対象機能を定義。特定顧客だけで閉じない汎用機能として設計・実装。
- 本番運用へ乗せ、後続の代理店には追加開発なしで対応可能な状態を実現。

理想フローを開発可能な要件へ再定義

- 業務側から理想とするフローのみが提示される案件では、必要な情報を収集し、解決された状態・課題・背景・業務上の必須要件を定義。理想を一括で実現する方針を採らず、リリース時点で必要な対応要件へスコープを絞った。
- エンジニアと設計方針・アーキテクチャを検討し、UIモックの先行実装を依頼。運用担当との認識合わせと合意形成を行い、実装可能な要件へ落とし込んだ。

期日直前の複数機能拡張に対する優先順位判断

- 新年度に展開予定の複数の機能拡張で、詳細が期日直前に共有される状況に対し、各機能の業務発生タイミング、依存関係、必須要件から優先順位とリリース範囲を判断。エンジニアとスケジュールを組み立てた。
- 請求処理が機能リリース後に発生するという業務上のラグを踏まえ、請求機能自体の改修か否かを基準に、請求関連対応を後続フェーズへ分離できるかを案件ごとに判断した。
- これらの判断に基づき、複数の機能拡張を期日までに提供した。

品質・リリース体制の再構築

- 既知のバグを報告後にトリアージし、影響度と緊急度からHotfixまたは次回リリースへの組み込みを判断。参画後の本番リバートは0件。
- 品質管理とリリース判断を見直し、本番Hotfixリリース数を前年度の65件から2件へ削減。
- 開発リードタイムを最長だった時期の5分の1へ短縮し、定期的にリリースできる流れを確立。

設計・技術基盤とチームづくり

- 既存プロダクトの設計、アーキテクチャ検討、コードレビュー、品質改善を担当。
- PHPUnitのFeatureテスト、脆弱性対応、インシデントの原因分析と恒久対応、AWS・IaC・権限設計の見直しを推進。
- 業務委託メンバーが自律して開発できるよう、担当範囲・判断基準・レビュー体制を設計。採用要件、候補者評価、必要な組織体制も整理。

T社

2024.03 — 2025.07

不動産系SaaS / エンジニア

技術環境	PHP 8.3 · TypeScript · Laravel · Nuxt · Vue · MySQL · Docker · GitHub Actions

メインプロダクトのチーム開発・リリースプロセス改善

- チーム開発を意識した進め方を整備し、リファインメントを含むスクラムベースのイベントを導入。
- 開発メンバー間の連携を強めることでリリース後の不具合を減らし、QAチームと円滑に検証・リリースを進められる状態を構築。

IPO監査対応を目的とした社内請求管理システムの0→1開発

- 当初提示されたUI・データ構造をそのまま実装する方針を採らず、経理業務とIPO監査への対応に必要な観点が不足していると判断。達成すべきゴールを再定義し、組閣されたチーム内で共通認識を形成。
- 少人数で開発・運用し、将来も大幅な増員を前提としない社内システムと捉え、フロントエンドとバックエンドを1リポジトリで扱うモノレポを選択。作業と変更管理の効率を優先。
- 過度なクリーンアーキテクチャやDDDの全面採用は避け、必要な考え方だけを取り入れたレイヤードアーキテクチャを設計。ControllerからUseCaseを介してServiceを呼ぶ依存ルールを定め、担当者が入れ替わっても処理フローが崩れにくい構成とした。
- 監査対応に加え、経理担当へ日々の業務課題をヒアリング。簡易UIと解決後の状態を共有して合意を取り、仕様と実装へ落とし込んだ。CI/CD、テスト方針、オンボーディングも整備し、重大な不具合なくリリース。

F社

2022.03 — 2024.02

建築系SaaS / エンジニア・スクラムマスター・SRE

技術環境	PHP · TypeScript · Go · Laravel · Vue · React · Next.js · AWS · Terraform · Docker

需要と期日から優先順位を決めた新規機能の段階リリース

- 複数の様式への対応が求められる一方、実装に必要な情報・定義が揃っておらず、パターン数も多く、外部の受託会社との連携も必要なため、期日までの全量実装は不可能と判断。すべてを同時に進める方針を採らなかった。
- 利用需要が高い様式から順に対象を決定。初回の対象と後続対応を切り分け、段階的にリリースした。
- 既存のVue実装は共通化を前提とした構成ではなかったため、個別の様式ごとに実装を増やす設計を避け、共通部分を再利用しながら新しい様式を追加しやすい構造へ設計・実装した。

テスト文化の導入

- PHPUnitのハンズオン形式の勉強会と、アジャイル開発に関する輪読会を、非エンジニアも巻き込みながら継続開催。
- 実際の開発でPHPUnitを利用する文化を定着させ、ほぼすべてのPRにテストが付く状態へ改善。

スクラムマスター・SREの立ち上げと基盤改善

- 認定スクラムマスターとして1on1やリファインメントを設計し、PMとチームが自律して進められる体制づくりを支援。
- SREの立ち上げを経験し、Datadog導入、Terraform Cloudの検証、Laravel 6から9への更新、CIとTerraformの整備を推進。

R社

2019.10 — 2022.02

人事系SaaS / エンジニア

技術環境	PHP · JavaScript · Laravel · Vue · Nuxt · Aurora · GitHub Actions

機能開発・品質課題の改善

- Laravel / NuxtによるAPI・画面開発を担当。モノレポ化に伴うコード移行ではPHPUnitのFeatureテストを整備し、監査ログ・承認機能はDB設計から実装。既存画面のUXも改善。
- Sentryに多数のエラーが記録される状態に対し、Sentry当番を設置。即日修正できるものは迅速にリリースし、原因が深いものは調査結果と対応方針をチームで共有して優先度を判断する運用へ変更。
- エラーの原因と再発防止策をリファインメント・プランニングへ反映し、考慮漏れを減らす進め方を定着。チーム発足から2か月でエラー発生率を80%削減。
- ビジネスサイドへのヒアリングと社内外のユーザーインタビューからストーリーを作成し、事業課題に対する解決案をエンジニアリングの観点から提案。

大手他社との協業プロジェクト

- 明確な役職はなかったものの社内リーダー相当として、仕様とストーリーの作成、実装対象とスケジュールの判断、他社・社内他部署との合意形成を担当。
- メイン開発チームへタスクと仕様を移譲しながら、自らも必要機能を実装してメンバーを支援。上長不在時にはスコープ判断と代替案の提示を担い、リリースまで完走。

G社

2015.05 — 2019.09

SES・受託開発 / エンジニア

技術環境	PHP · JavaScript · Ruby · Laravel · Symfony · Ruby on Rails · React · Vue · Vagrant

社内研修・受託開発・複数のSES案件

- 新卒・未経験者向けの研修カリキュラムをゼロから設計し、社内勤怠システムの開発を外部PMと進行。
- レガシーシステムのリプレイスでRepositoryパターンやレビュー文化を導入。
- マッチングサービスのSQLチューニング・クラス設計・障害対応、Laravel / Vueによる管理画面刷新、テスト・Git Flow導入を担当。

個人開発

gh-trends

GitHub CLI Extension / Go

GitHub上のPR、リードタイム、リリース頻度、コードレビュー、GitHub Actionsを月次・年次で分析し、開発フローの変化を可視化するGitHub CLI拡張。

技術・公開情報	Go · 日本語・英語対応 · MIT License · https://github.com/hironeko/gh-trends