

## 基本情報

**氏名**	沖 貴司 (OKI Takashi)
**生年月日**	1996/07/14
**居住地**	東京都
**メールアドレス**	ot07neoki@gmail.com

## 各種アカウント



## 保有スキル

- JavaScript / TypeScript + Reactを用いたフロントエンド開発
- Go / PHP (Laravel) を用いたバックエンド開発
- Gitを用いたバージョン管理
- テストコードの実装やCIの導入による品質を意識した開発
- コードレビュー文化の促進や勉強会の開催など、チーム開発の質を高めるための自発的な取り組み
- チーム内やプロジェクト全体での積極的な改善提案と、それに伴い、他のメンバーも自ら改善策を提案する文化の形成に貢献

## 技術スタック

### プログラミング言語



### フレームワーク・ライブラリ



## 職歴

---

### 株式会社Mt.SQUARE（2022/05～現在）

製造業向け基幹システムの開発に従事。コーディング面での品質向上に強い関心を持ち、勉強会の開催やNotionでの情報共有や、コードレビュー文化の形成、さまざまな改善提案を積極的に行っている。これらの取り組みが実を結び、チームリーダーやメンバーから日々実装方法に関する相談を受けている。さらには、CTOから担当外の箇所で起きている不具合の話題でメンションされ、原因究明と解決をしたこともあり、技術的な信頼を得ることができている。

- プロジェクト規模
  - プロジェクト全体は50名程度、所属チームは5名程度
- プロジェクト詳細
  - ReactでのSPA開発
    - 主要技術スタック: React 17, Redux 4, Redux-Form 8, React-Router 5, Tailwind CSS 2, Reactstrap 8, Storybook 6
  - LaravelでのREST API開発
    - 主要技術スタック: PHP 7.4, Laravel 8, MySQL 8
  - 画面単位でタスクを割り振られ、画面実装とその画面で使用するAPI・データベースの設計・実装を担当
  - 製造業務を担当する部署向けの機能開発（例：カレンダーライブラリを用いた製造スケジュール管理用の画面の開発）
- その他
  - 勉強会の開催やNotionでの情報共有
  - これまでチームリーダーのみが行っていたコードレビューを、チームメンバーも実施することを提案し、コードレビュー文化を根付かせることに貢献
  - バックエンド側にCIの導入（テストコードの実行、Swaggerドキュメント生成でのWarningやシンタックスエラーのチェック、PSR-4エラーチェックなど）
  - OPcacheの導入方法の調査と提案を行い、APIの高速化に貢献
  - DBのコードマスタのコードをまとめたJavaScriptオブジェクトを自動生成するスクリプトの実装（DBと実装内でのコードの二重管理を避けるため）

### 三菱電機インフォメーションネットワーク株式会社（2021/04～2022/04）

他部署が保有する自社クラウドサービスの構築・管理の自動化コードの開発や、所属部署が所有するvCenter Serverの運用・保守を担当。vCenter Serverには、他部署の開発環境として提供している仮想マシンもあり、新しい開発環境の構築や脆弱性対応などを行った。

- プロジェクト規模
  - 自動化コードの開発チームは3名、所属部署は10名程度
- プロジェクト詳細
  - Ansible 2.9での自動化コードの設計書作成・実装・テスト仕様書作成・テスト実施
  - vCenter Server内の仮想マシンの構築・管理
- その他
  - Ansible Moleculeによる自動化コードのテスト自動化の提案～導入（テストコードの実装方法の調査・実装・CI構築）
  - Log4j脆弱性対応のために、Ansibleで修正パッチ適用の自動化コードを実装し、他部署の開発環境も含めた複数の仮想マシンへの一括適用を実施

## 学歴

---

- 高知工科大学 大学院 工学研究科基盤工学専攻 情報学コース（2019/04 ～ 2021/03）
- 高知工科大学 情報学群 コンピュータサイエンス専攻（2015/04 ～ 2019/03）

## 業務外活動

---

### 個人開発

- [Next Bazaar](#)
  - Web開発の学習用に作成しているオンラインショッピングサイト
  - 技術スタック
    - フロントエンド: TypeScript, React, Next.js, Tanstack Query, Mantine, Jest, Storybook
    - バックエンド: Go, Fiber, Swagger, PostgreSQL, sqlc, swag
  - 大切にしていること
    - ドメイン駆動設計や[Bulletproof React](#)などのアーキテクチャを参考にした関心事の分離
    - 効果の高いテストコードの実装

- APIの正常系テストでは実際の環境に近いテスト用のDB（Dockerコンテナ）を使用し、異常系テストでは例外発生のためにモックDBを使用
- リファクタリングへの耐性を高めるために、外部から観察可能な情報のみ（APIのレスポンスなど）を使って振る舞いを検証
- コード自動生成による実装の効率化
  - [swag](#)というGoパッケージを使い、GoコードとコメントからOpenAPIドキュメントを自動生成
  - [Orval](#)というnpmパッケージを使い、OpenAPIドキュメントからデータ取得用のReactフックを自動生成
- [Myposts](#)
  - Markdown形式の記事を閲覧・投稿することができるプラットフォーム
  - 技術スタック
    - フロントエンド: TypeScript, React, Next.js, Tailwind CSS, Prisma
    - バックエンド: TypeScript, Next.js, tRPC, Prisma, PostgreSQL
- [Music Transformer Playground](#)
  - Music Transformerと呼ばれるAIを使って、簡単にピアノの演奏を生成することができるプレイグラウンド
  - 技術スタック
    - フロントエンド: TypeScript, React, Vite
    - バックエンド: Python, FastAPI, Music Transformer, Docker, Cloud Run
- [prettier-plugin-astro-organize-imports](#)
  - Astroファイルのimport文をフォーマットするためのPrettierプラグイン

## OSS活動

- Prettier
  - TypeScriptコードのフォーマットに関するバグ修正 ([#14830](#))
- Mantine
  - UIコンポーネントのスタイリングに関するバグ修正 ([#4281](#))

- その他（タイポ修正など）
  - [mantinedev/mantine#4290](#)
  - [withastro/docs#3308](#)
  - [withastro/docs#3310](#)
  - [withastro/docs#3321](#)

## 社内での情報共有活動

- 所属部署内での勉強会の実施
  - 多くのチームメンバーが同時期に入社し、React開発の経験がない人もいたため、JavaScript, React, Reduxの基本的な使い方や、Mac, VSCodeの便利なショートカットキー・拡張機能を紹介
  - チームでのテストコードの書き方に問題があると感じていたため、「単体テストの考え方/使い方」という本を読んだ上で重要だと思った内容を紹介（AAAパターンや、各テストケースが互いに影響しないように実装することなど）
  - ChatGPTを使い、開発業務の効率が上がったと感じたため、基本的な使い方・うまく使うためのコツ・注意点などを紹介
- Notionでの情報共有（これまでに約30ページを共有）
  - 開発環境構築の手順書作成
  - プロジェクトのディレクトリ構成や利用パッケージの紹介
  - 自分が実装する上で行き詰まった箇所の解決方法の共有

## その他

- 友人と輪読会を実施
  - [Clean Architecture 達人に学ぶソフトウェアの構造と設計](#)
  - [単体テストの考え方/使い方](#)（輪読中）

## 資格

---

- VMware認定資格 - VCP-DCV 2022（2022年1月）
- 応用情報技術者試験 合格（2018年春期）

## 意欲・興味

---

- 静的型付け言語を用いた開発・テストコードの実装・CIの導入など、間違いに素早く気付ける開発に興味があり、個人開発での実践を進めています
- ドメイン駆動設計や[Bulletproof React](#)などのアーキテクチャに基づく、変更に強いコードを書くことに興味があり、個人開発での実践を進めています
- チームで開発する上で、情報やスキルの共有が重要だと考えているため、コードレビュー・ペアプロ・勉強会の開催などに取り組みながら開発することに興味があります
- ある程度ゴールが見えている方が能力を発揮しやすいと考えているため、プロダクトの立ち上げフェーズよりも、それ以降の拡大・改善フェーズに興味があります

## 希望条件

---

- 開発効率を維持するには、既存の開発環境を改善し続けることが大事だと考えているため、改善に十分な時間を割ける環境だと嬉しいです
- コードレビューの実施やテストコードの実装など、コーディング面の品質を高めることも大切にしている環境だと嬉しいです
- ユーザの視点を重視し、使いやすいシステムを作ることを大事にしている環境だと嬉しいです
- 慣れているUS配列のMacBookで開発ができると嬉しいです