

# 職務経歴書

2022年11月1日 古山 蒼二郎

## 経歴

- ・ 2020年12月1日 ~ 現在 株式会社Vitalize

社内での受託開発、及びSESとして3社へ派遣され、開発に従事しました

## 経歴詳細

### 1. 人材紹介サービスの開発

Ruby on Rails 6 を採用したtoB向けの人材マッチングWebアプリケーション開発

#### チーム構成と役割

6人のチームで3人目のバックエンドDevとして参画。小さなバグ修正からとりかかり、感覚をつかんだところで並行して新規機能の要件定義、DB設計、開発を担当しました。

#### 実績

##### 課題

当該サービスの抱える問題点として、新規登録者数は徐々に増えているものの、アクティブユーザー数が伸び悩んでいました。

##### 解決

そこで、直近一週間で一度もログインを行わなかったユーザに対してログインを促すメールを送信するバッチ処理を追加開発することとなり、開発の担当者となりました。

依頼者と話し合いを重ねた結果、cronjobとrakeタスクのみで実現できることがわかったので、wheneverを採用しました。

また、rakeタスクのログはagentを通してcloudWatchへ出力し、更にバッチ処理の死活監視にはhealthcheck.ioを採用しました。

##### 追加調査

メール配信サービスに関しては既に開発初期段階でAWS SESが既に選定、運用されていました。しかし、開発を進めるに連れて、送信したメールが迷惑メールに判定される事象が多発することを確認し、追加調査を行いました。

##### 成果

調査の結果、DKIMやSPFが設定されていなかったことが判明したので、開発の傍らにSESの設定の見直しを行いDKIM,SPFおよびDMARCを導入し、メール機能の改善に寄与しました。

---

### 2. キャッシュレス決済サービス開発

クレジットカード決済、コード決済、電子マネー決済等、様々な決済方法の導入を 契約会社に包括的に提供する決済代行事業サービスシステムの開発

#### チーム構成と役割

システム全体では200人程度の開発者が在籍し、10人程度のスクラムチームに別れ開発していました。

そのうちの一つのスクラムチームにバックエンド開発者として参画し、gRPC,golang,kafkaを用いた決済中継マイクロサービスの開発を担当しました。

また、2022年下期からは

- ・新規顧客登録マイクロサービス
- ・顧客審査マイクロサービス

を担当していたチームが解散となり新規チームが結成されることとなったため新規チームのリードエンジニアとして開発を担当し、

上記マイクロサービスに加えて、Koaを採用したWeb BFFやGraphQL、Vue.jsを採用したフロントエンドの開発も担当しました。

## 実績

### 1. 設計、コードレビューを通じた品質の向上。DDDに則ったコーディングスタイルの周知

バックエンド開発メンバーとアプリケーション設計やDB設計について以下のようなことを議論しました。

- ・どこまでを1集約として管理するのか。その場合のメリットデメリットは何か。
- ・ validation層で不正requestとしてするのか。validationの責務ではないdomain固有のビジネスロジックエラーなのでdomain生成時のconstructorでerrorとするのか。
- ・ 他マイクロサービスのgRPC Clientはどの層に置くのが正解か。
- ・ MySQL(5.7,InnoDB,REPEATABLE READ)上でこの処理はANOMALYを起こさないのか。
- ・ ローカルのmysqlコンテナ上だけではなく、Cloud SQL上でもちゃんと動作するDB設計になっているのか。

### 2. wireの導入によるボイラープレートコードの削減

usecaseとrepository間のように依存性があるレイヤーのDependency Injectionのコードを手書きで書いていましたが、開発規模が大きくなり手作業に限界を感じていました。

そこでDIの負担を軽減する方法を模索し、最終的にwireを導入することでボイラープレートの削減に成功しました。

### 3. フロントエンド、バックエンド間の仲介

ヘルプチーム内でgolang,Typescriptの実務経験があるのが自分だけだったので、フロントエンド担当チーム、バックエンド担当チームの仲介に力をいれました。

具体的には、以下のようなことを行いました。

- ・ フロントエンドチームから「backendで500エラーがでているが原因が分からない」という報告が上がった際に、サーバ側のエラーなのか フロント側がバックエンド実装者の意図しないrequestの投げ方をしているのかを調査し問題を解決
- ・ バックエンド開発でprotobufに変更が生じてしまった場合、フロントエンドの画面やBFFに影響があるかの判断。影響がある場合フロント側への修正内容、修正箇所の案内

## 開発担当機能

- ・ 決済中継マイクロサービス開発 (Kafka,Golang1.17)
- ・ 新規顧客情報登録、審査機能開発 (Front: Vue.js 2系, BFF: Koa(Node.js), Backend: Golang1.17)
- ・ 決済中継マイクロサービスと新規マイクロサービスの連携 (Golang1.17)