

目次

- [目次](#)
- [プロフィール](#)
- [バリューを発揮しやすい環境・業務](#)
- [スキル](#)
 - [開発手法](#)
 - [言語](#)
 - [フレームワーク・ライブラリ](#)
 - [RDB/NoSQL](#)
 - [AWS](#)
 - [Azure](#)
 - [SaaS/PaaS](#)
 - [プロジェクト管理/チャット](#)
 - [その他](#)
- [職務経歴詳細](#)
 - [インターネット広告・コンシューマー向けサービスを展開するIT企業](#)
 - [ふるさと納税支援サイトの開発](#)
 - [コンテンツ・ヘルスケア系IT企業](#)
 - [航空会社向け航空気象サービスの開発](#)
 - [ヘリコプター事業者向け 医療用ヘリ（ドクヘリ） 離発着場管理サービス](#)
 - [Web to Printプラットフォームを提供する スタートアップ企業](#)
 - [学習用ドリル原稿作成業務支援webシステム開発](#)
 - [デジタル印刷プラットフォーム構築PJ](#)
 - [印刷製造業務 生産管理パッケージ開発](#)
 - [医療機器メーカー マニュアル作成フロー改善PJ](#)
 - [大手コンビニチェーン 開店販促システム構築](#)
 - [ゴルフコンペ記念品生成支援システムの構築](#)
 - [ナンバープレートデザイン公募サイト構築](#)

プロフィール

安達 清慈

1991年5月生まれ

東京都在住

大学卒業後、Web to Printプラットフォームを提供するスタートアップ企業にて、パッケージソフトおよびSaaSサービスの開発に従事。

2022年よりフリーランスエンジニアとして独立。

主に.NET系のバックエンド開発を得意としながら、React・TypeScriptを用いたフロントエンド開発やその他の技術領域もキャッチアップし、幅広いWebアプリケーション・業務システム・ToC向けサービスの開発を手がけています。

バリューを発揮しやすい環境・業務

- How (解決手段) にとらわれることなく、本質的なユーザーの課題解決を考慮したうえでの設計・開発
- 仕事の領域を制限せずユーザーストーリー達成のために必要なアクションを行えるチーム
- 気軽に雑談できる心理的安全性の高いチーム
- 少人数で密なコミュニケーションを取りながら意思決定・開発を進められる環境
- 詳細な設計書通りの実装ではなく、目的に応じて設計・実装の裁量を持てる環境

スキル

開発手法

スクラム開発・チケット駆動開発

言語

C#・Ruby・PHP・JavaScript・TypeScript

フレームワーク・ライブラリ

ASP.NET (MVC、Web API、Web Forms)・React・Vue.js・Ruby on Rails・CakePHP・jQuery

RDB/NoSQL

SQL Server・MySQL・PostgreSQL・Oracle・MongoDB・Amazon DynamoDB・Azure Cosmos DB

AWS

RDS・ECR・ECS・EC2・Lambda・SES・SQS・S3・CloudFront・Secrets Manager

Azure

Kubernetes Service・Container Instances・Container Registry・Container Apps・Web Apps・API Management・SQL Database・Blob Storage・Functions・Key Vault

SaaS/PaaS

GitHub・GitHub Actions・Bitbucket

プロジェクト管理/チャット

Backlog・Asana・Notion・Redmine・Azure DevOps・Slack・Microsoft Teams・Chatwork

その他

Docker・OpenAPI・Jenkins・Apache Airflow

職務経歴詳細

インターネット広告・コンシューマー向けサービスを展開するIT企業

【雇用形態】

業務委託

【在籍期間】

2023年12月 ~ 2025年6月

ふるさと納税支援サイトの開発

【チーム編成】

- バックエンドエンジニア 20名程度
- フロントエンドエンジニア 20名程度

【言語・フレームワーク】

ASP.NET Web API・ASP.NET MVC・ASP.NET Web Forms・Vue.js

【インフラ】

IIS（EC2上のWindows Server）・RDS(SQL Server)・S3・SQS・CloudFront

【ツール】

Visual Studio・Visual Studio Code・GitHub・Git Bucket・Docker・Asana・Chatwork

【プロダクト概要】

全国の自治体と寄附希望者をマッチングするふるさと納税ポータルサイト。

寄附者はWebサイト上で返礼品の検索、寄附申込み、決済までを一括で行うことができ、寄附額に応じた独自ポイントの付与・利用も可能。

自治体職員向けには、返礼品の登録・在庫管理・寄附受付状況・寄附者情報の確認などを行う管理画面を提供。

本システムの導入により、寄附者にとってはふるさと納税の手続きをスムーズかつ利便性高く行える環境を実現し、自治体側は返礼品・寄附受付・寄附者管理業務の効率化と情報管理の精度向上を実現した。

【担当業務概要】

ふるさと納税サービスのバックエンドエンジニアとして参画。

歴史あるサービスのため、レガシーな設計やドキュメント不備、運用ルールの人件化といった課題が存在し、それらを整理・改善しながら新旧機能の開発・改修等の業務を担当し、バックエンドを中心に一部フロントエンドの実装も対応。

【担当業務詳細】

- **ASP.NET Web API を使用したREST APIの設計・実装**
- **xUnitを用いた単体テスト・結合テストの実装**
 - テストカバレッジ向上と不具合検知の早期化を目的とした自動テストの作成・実行
- **DB設計**
- **新規OTA（Online Travel Agency）サイト立ち上げに伴う各種機能開発**
 - ポイント付与・消費機能のAPI開発（DB設計、API設計～実装）
 - 予約空室カレンダー機能のAPI開発・外部予約管理サービスとのAPI連携仕様調査・設計・実装

- **Vue.js + TypeScriptを使用したフロントエンド開発**
 - 予約空室カレンダー表示機能の画面実装
- **社内管理機能の設計・実装**
 - 大量請求書作成処理のアーキテクチャ刷新
 - 大量請求書処理によりプロセスが落ちるという課題があり、既存実装の問題点を整理し、SQSを活用した非同期処理方式を提案・採用。バックエンド実装を担当し、安定運用を実現
 - 外部サービスのメンテナンススケジュール設定機能の改善
 - 保守性が低く属人的に運用されていた独自DSL(json)形式を、Cron形式へ改修する案を提案・採用。柔軟な設定と保守コスト削減を実現
- **ポイント付与・決済機能を備えたAPIの新規構築**
 - アプリケーションディレクトリ構成の設計、README・環境構築手順の整備を実施
 - 共通基盤ロジックの設計・実装
 - Pollyを用いたレジリエンスの高いDynamoDBクライアントの開発
 - ポイント付与・決済機能のAPI実装

コンテンツ・ヘルスケア系IT企業

【雇用形態】

業務委託

【在籍期間】

2022年3月 ~ 2023年12月

航空会社向け航空気象サービスの開発

【チーム編成】

- バックエンドエンジニア 3名
- フロントエンドエンジニア 5名

※全体で約15名

【言語・フレームワーク】

ASP.NET Web API・Python

【インフラ】

Azure Kubernetes Service・Azure Container Instances・Azure Container Registry・Azure Web Apps・Azure Blob Storage・Azure Functions・Azure SQL Database

【ツール】

Visual Studio Code・Azure DevOps・Microsoft Teams・Docker・Apache Airflow

【プロダクト概要】

航空機の運航管理者を対象としたWebシステム。

運航に影響を及ぼす気象・災害情報と飛行ルート情報を1画面で集約し、リアルタイムに確認・比較できる機

能を提供する。

これにより、運航管理者が気象状況を踏まえた最適な飛行ルートを選定や、安全性を考慮したフライトプランの迅速な策定を支援。

従来、複数のシステムや紙資料を突き合わせて行っていた情報確認作業を効率化し、意思決定の迅速化と運航リスクの低減を実現。

【担当業務概要】

1stリリース直後の保守・追加開発フェーズからバックエンドエンジニアとしてプロジェクトに参画。

2週間スプリントのスクラム開発体制のもと、既存機能の保守運用、新規機能の設計・実装、一部のインフラ移行対応等を担当。

航空運航管理の専門ドメイン理解や、初めての本格的なAzure基盤での開発にも短期間でキャッチアップし、開発に貢献。

【担当業務詳細】

- **ASP.NET Web API + Swashbuckleを使用したREST APIの設計・実装**
 - クリーンアーキテクチャをベースにロジックを実装
- **xUnitを用いた単体テスト・結合テストの実装**
 - テストカバレッジ向上と不具合検知の早期化を目的とした自動テストの作成・実行
- **DB設計**
 - 気象情報、運航ルート、運航状況管理用のテーブル設計
- **Apache Airflowによる気象データ取込ETLジョブの作成**
- **Azure Web App から AKS (Azure Kubernetes Service) への移行**
 - バックエンドAPIプロジェクトのコンテナ化対応、Azure PipelineでのCI/CDパイプラインの構築
- **航空運航管理特有の空域表示機能の開発**
 - 空域情報の座標データが未公開のため、仕様書の自然言語記述をもとにQGISを用いて地図上にプロットし画像化
 - 仕様書のパターンを抽出し、Web上の情報をスクレイピングすることで一部空域画像の自動生成を実現
- **クラウド基盤・ドメイン知識のキャッチアップ**
 - Azure開発が初経験だったが、Microsoft Learnやドキュメントを活用し短期間で環境構築・開発に対応
 - 航空用語や運航ドメイン特有の専門知識についてはプロジェクト内の用語集を活用しつつ、独自に資料化・整理を行い理解を促進

ヘリコプター事業者向け 医療用ヘリ（ドクヘリ） 離発着場管理サービス

【チーム編成】

- バックエンドエンジニア 3名
- フロントエンドエンジニア 3名

※全体で約10名

【言語・フレームワーク】

ASP.NET Web API・Python

【インフラ】

Azure Container Apps ・ Azure Container Instances ・ Azure Container Registry ・ Azure API Management ・ Azure Blob Storage ・ Azure SQL Database

【ツール】

Visual Studio Code ・ Azure DevOps ・ Microsoft Teams ・ Docker ・ WSL2 ・ Apache Airflow

【プロダクト概要】

ドクターヘリの運航業務において、ランデブーポイント（離発着場）や飛行ルート（フライトメモ）情報を多拠点間で一元管理・共有するWebシステム。

地上オペレーターと機上の機長・整備士といった運航関係者間で、リアルタイムな情報共有と連携を支援することを目的とし、従来Excelで個別管理されていた運航関連情報の集約・システム化を実施。

本システム導入により、情報共有の迅速化、運航判断の精度向上、ヒューマンエラーの防止、運航業務全体の効率化を実現。

【担当業務概要】

PJのキックオフからバックエンドエンジニアとして参画し、主に初期リリースまでの開発を担当。

2週間スプリントのスクラム開発体制のもと、開発基盤の構築、REST APIの設計・実装、ETLジョブ開発、外部API連携の検証など幅広い業務を担当。

また、法人向けマルチテナントSaaSの特性を踏まえたデータ分離設計や、ローカル開発環境・CI/CDパイプラインの整備による開発効率化にも貢献。

新技術の採用に伴うトラブル対応なども行い、プロジェクトの安定稼働に寄与。

【担当業務詳細】

- **ASP.NET Web API + Swashbuckleを使用したREST APIの設計・実装**
 - クリーンアーキテクチャをベースにロジックを実装
- **Azure PipelinesによるCI/CDパイプラインの構築**
 - ビルド・テスト・デプロイの自動化を実現し、開発フローの効率化と品質向上に貢献
 - GA直後のAzure Container Apps採用に伴う破壊的変更発生時にも、迅速な修正方法を調査・実装しプロジェクトを支援
- **開発基盤の整備**
 - プロジェクトのディレクトリ構成策定、READMEテンプレートや環境構築手順書の整備により、開発体制の立ち上げを主導
- **DB設計**
 - マルチテナント環境におけるデータ混在防止のため、行レベルセキュリティ（RLS）による分離設計を実施
 - RLS実現のための基盤コードも併せて整備
- **xUnitを使用した単体テスト・結合テストの実装**
 - 安定稼働を目指し、テストコードの作成・カバレッジの向上を推進
- **外部API連携のフィジビリティ検証**
 - 地図データ、ランデブーポイントデータ、住所データ取得用APIの調査・検証を実施
 - 地図データAPIへのアクセス最適化のため、プロキシサーバーとnginx.confを作成し、リクエストの負荷分散・スループット改善を実現
- **Apache Airflowを用いた住所データ取込用ETLジョブの開発**
- **作業フロー・開発効率化への取り組み**

- ローカル開発環境の整備やCI/CDパイプラインの構築を通じて、開発体制全体の生産性向上に貢献

Web to Printプラットフォームを提供する スタートアップ企業

【雇用形態】

正社員

【在籍期間】

2016年5月 ~ 2022年2月 退職後は副業としてスポット参画

学習用ドリル原稿作成業務支援webシステム開発

【チーム編成】

- エンジニア 2名

※全体で3名

【言語・フレームワーク】

ASP.NET Web API ・ React ・ TypeScript ・ Ant Design

【インフラ】

IIS（EC2上のWindows Server） ・ SQL Server

【ツール】

Visual Studio ・ Docker ・ WSL2 ・ Microsoft Teams ・ Backlog

【プロダクト概要】

学習用ドリルの原稿作成業務を効率化・支援する社内向けWebシステム。

Excel形式で管理されている製造依頼データを取り込み、印刷用の学習ドリル原稿（PDF形式）を自動生成する。

従来手作業で行っていたPDF原稿作成プロセスをシステム化することで、作業工数削減とヒューマンエラー防止を実現。

【担当業務概要】

要件定義フェーズから担当営業とともにエンドユーザとの仕様検討・要件調整に参加し、設計・開発以降の工程をほぼ1人で担当。

システムの設計・実装・テスト・運用設計まで一貫して行い、短期間でのプロダクトローンチと安定稼働を実現。

また、直近のチーム課題であった「運用設計不備による開発者の手離れ不可」問題について、本PJでは運用設計を事前に十分検討し、運用開始後の問い合わせ対応をCSへ完全委譲することに成功。

【担当業務詳細】

- エンドユーザ・営業と連携したシステム要件の検討・調整

- 要件定義フェーズから参画し、エンドユーザ・営業と直接調整し仕様策定を実施
- **各種設計・ドキュメント作成**
 - 基本設計書、画面設計書、DB設計書、テスト仕様書の作成
- **DB設計**
- **ASP.NET Web API + Swashbuckleを使用したREST APIの設計・実装**
 - クリーンアーキテクチャをベースに、保守性・拡張性を意識したAPIロジックの実装
 - OpenAPI仕様の自動ドキュメント生成による開発効率化
- **React + TypeScriptを使用したフロントエンド開発**
 - 管理画面の画面設計・実装、API連携機能の開発
- **xUnitを使用した単体テスト・結合テストの実装**
 - 継続的な品質担保のためのテストコード作成
- **サードパーティ製パッケージのフィジビリティ検証**
 - 要件実現に必要な外部パッケージの選定・ドキュメント読解・動作検証
 - 検証結果を元に実装可否を判断し、最適な設計を提案

デジタル印刷プラットフォーム構築PJ

【チーム編成】

- エンジニア 4名（社員2名+パートナー2名）

※全体で約10名

【言語・フレームワーク】

ASP.NET Web API・Windows Service(.NET Core Generic Host)・React・Remix（SPA）

【インフラ】

IIS（EC2上のWindows Server）・SQL Server・S3

【ツール】

Visual Studio・Visual Studio Code・Docker・WSL2・Microsoft Teams（社内用チャット）・Slack（開発パートナー用チャット）・Backlog

【プロダクト概要】

「印刷発注者」と「印刷工場」をつなぐデジタル印刷プラットフォーム。
印刷発注者からの印刷物制作の受注、印刷工場への発注手配、製造結果のステータス返却までを一元管理し、印刷物制作のオペレーション全体を包括的に支援。
本システムは、入稿された原稿データを印刷仕様に最適化し、自動で印刷データを生成。
発注ルールに基づき、印刷工場の選定から発注、納品状況の管理までを効率化。

これにより、印刷発注者は煩雑な印刷発注業務の効率化、進行状況の可視化、発注ミスの防止が可能であり、印刷工場は新規受注機会の獲得、製造指示の標準化による業務効率化が可能となる。

GEMiNX

【担当業務概要】

PJのキックオフから新規開発チームの主要メンバーとして参画し、アーキテクチャ設計、開発基盤の構築、新規機能の設計・実装、オフショア開発管理まで幅広く担当。

2週間スプリントでのスクラム開発体制のもと、開発プロセスの改善とプロダクト品質の向上に貢献。

【担当業務詳細】

- **ASP.NET Web API + Swashbuckleを使用したREST APIの設計・実装**
 - クリーンアーキテクチャをベースに、保守性・拡張性を意識したAPIロジックの実装
- **Jenkinsを使用したCI/CDパイプラインの構築**
 - ビルド・テスト・デプロイの自動化による開發生産性と品質の向上
- **Hangfireを使用したバックグラウンド処理基盤の構築**
 - Windows Serviceとして定期実行・非同期ジョブ管理を実装
- **React + TypeScriptを使用したフロントエンド開発**
 - 印刷発注者・印刷工場・社内向けの各種管理画面の設計・実装
 - 画面ごとの要件整理・UI設計・API連携機能の開発
- **xUnitを使用した単体テスト・結合テストの実装**
 - 継続的な品質担保のためのテストコード作成
- **入稿データのプリフライト処理用API調査・連携モジュールの実装**
 - 海外製パッケージのAPI仕様調査と.NETアプリケーションとの連携モジュール開発
- **開発基盤の整備**
 - プロジェクトディレクトリ構成、READMEテンプレート、環境構築手順書の整備
- **DB設計**
- **オフショアパートナーへの開発指示・コードレビュー・タスク管理**
 - 設計書・開発指示書の作成、コードレビュー対応、進捗管理などチームマネジメント

印刷製造業務 生産管理パッケージ開発

【チーム編成】

- エンジニア 4名（社員2名+パートナー2名）

※全体で約10名

【言語・フレームワーク】

ASP.NET Core MVC・ASP.NET Web API・React・TypeScript

【インフラ】

IIS（Windows Server）・SQL Server

【ツール】

Visual Studio・Visual Studio Code・Microsoft Teams（社内用チャット）・Slack（開発パートナー用チャット）・Backlog

【プロダクト概要】

印刷会社のデジタル印刷製造業務を支援する生産管理パッケージシステム。

印刷機器・用紙種などの事前登録情報に加え、都度発生する納期・出荷先などのオーダー情報をもとに、独自のグルーピングロジックで多様な印刷オーダーの生産計画を最適化する。

上記のようなプリプレス作業および生産管理業務を自動化することで、印刷工程全体の効率化・最適化を支援する。

OneFlow

【担当業務概要】

新規プロダクト開発プロジェクトのキックオフから参画し、アーキテクチャ設計、開発基盤の構築、API/フロントエンド実装、バックグラウンド処理基盤の構築、オフショア開発管理など幅広く担当。
2週間スプリントのスクラム開発体制のもと、開発プロセス改善・品質向上・外部仕様変更に強いアーキテクチャの構築を推進し、プロダクトの安定運用と開發生産性向上に貢献。

【担当業務】

- **ASP.NET Web API + Swashbuckleを使用したREST APIの設計・実装**
 - クリーンアーキテクチャをベースに、保守性・拡張性を意識したAPIロジックの実装
 - 外部APIの仕様変更に対する影響を抑えるため、腐敗防止層を設計・実装し、外部依存の影響範囲を局所化
- **Jenkinsを使用したCI/CDパイプラインの構築**
 - ビルド・テスト・デプロイの自動化による開發生産性と品質の向上
- **React + TypeScriptを使用したフロントエンド開発**
 - 画面ごとの要件整理・UI設計・API連携機能の開発
- **xUnitを使用した単体テスト・結合テストの実装**
 - 継続的な品質担保のためのテストコード作成
- **入稿データのプリフライト処理用API調査・連携モジュールの実装**
 - 海外製パッケージのAPI仕様調査と.NETアプリケーションとの連携モジュール開発
- **開発基盤の整備**
 - プロジェクトディレクトリ構成、READMEテンプレート、環境構築手順書の整備
- **DB設計**
- **オフショアパートナーへの開発指示・コードレビュー・タスク管理**
 - 設計書・開発指示書の作成、コードレビュー対応、進捗管理などチームマネジメント
- **フィジビリティ検証**
 - 組版機能実現のため、サードパーティパッケージの仕様調査・プロトタイプ実装を実施
 - 各種実機（印刷機・製本機・検査機等）との連携仕様（JDF）の調査・実機検証を行い、仕様書の誤り・相違点の早期発見と修正提案を実施
- **現地でのパッケージ導入支援**
 - 海外工場現地に於てパッケージ導入・設定・動作確認を担当
 - 現場オペレーションとシステム仕様のすり合わせを実施し、運用課題の早期解決を支援

医療機器メーカー マニュアル作成フロー改善PJ

【チーム編成】

- エンジニア 4名（社員2名+パートナー2名）

※全体で約10名

【言語・フレームワーク】

ASP.NET Web API・React・JavaScript・Python

【インフラ】

IIS (Windows Server) ・ SQL Server ・ Oracle

【ツール】

Visual Studio ・ Visual Studio Code ・ Microsoft Teams (社内用チャット) ・ Slack (開発パートナー用チャット) ・ Backlog

【プロダクト概要】

医療機器メーカー向けの医療機器マニュアル原稿作成業務を支援する社内向けWebシステム。

マニュアル表紙のカスタマイズ作成機能、外部プリフライトモジュールによる品質チェック機能、承認機能を備え、マニュアル製造工程全体の最適化と業務効率化を実現。

従来手作業で行っていたマニュアル作成・確認・承認のプロセスをシステム化することで、作業工数の大幅削減と品質の均一化、ヒューマンエラー防止に寄与。

【担当業務概要】

リリース直前に前任SEが離脱し、急遽プロジェクトのマネジメントを引き継ぎ、品質管理体制の立て直しと開発進行のリカバリを実施。

自らも開発に携わりながらチームを牽引し、無事リリースまで完遂。

リリース後は、既存システムの追加開発および機能改修を行い、エンドユーザとの要件調整から新規機能設計、オフショア開発管理、コードレビューまで幅広く担当。

【担当業務詳細】

- **エンドユーザとのシステム要件検討・調整**
 - 追加開発・既存機能改修における要件ヒアリング、調整、仕様確定の推進
- **新規機能の設計・既存機能の改修**
 - 画面・サーバサイドの設計、外部プリフライトモジュールとの連携設計
- **オフショア開発チームの管理**
 - 設計書・開発指示書の作成
 - コードレビュー、進捗・課題管理、日次定例による開発状況のフォローアップ
- **品質改善・開発体制の立て直し**
 - リリース直前に発生した設計・内部品質課題に対し、設計レビュー・コードレビュー体制を再構築
 - ブレインガーマネージャとして自らも開発に参画し、品質確保と開発スケジュールの両立を実現
- **プロジェクトマネジメント補佐**
 - タスク管理、進捗報告、ユーザ調整のサポート

大手コンビニチェーン 開店販促システム構築

【チーム編成】

- エンジニア 4名 (社員2名 + パートナー2名)

※全体で約10名

【言語・フレームワーク】

ASP.NET MVC ・ ASP.NET Web API ・ JavaScript

【インフラ】

IIS（EC2上のWindows Server）・SQL Server・S3

【ツール】

Visual Studio・Visual Studio Code・Microsoft Teams（社内用チャット）・Slack（開発パートナー用チャット）・Backlog

【プロダクト概要】

大手コンビニチェーン向けの店頭販促チラシ作成業務を支援するWebシステム。

各店舗担当者が、販促内容（クーポン情報や商品情報など）を店舗ごとに自由にカスタマイズできる機能を提供し、作成したチラシの内容を即時プレビュー・承認できるワークフローを実装。

煩雑な確認・承認作業をシステム化することで、作業工数の削減・情報伝達の迅速化・誤掲載防止を実現。さらに、完成イメージを確度高く確認できることで、販促施策の柔軟な調整も可能となった。

【担当業務概要】

新規プロダクト開発プロジェクトのキックオフから参画し、開発リーダーのもとシステム全体の機能設計・実装を担当。

運用管理サイトの新規開発や組版サードパーティパッケージと連携するラッパーAPIサーバの構築などを行い、プロダクトの開発・運用基盤を整備。

社内初のReact + TypeScriptを採用した管理サイト開発も手がけ、社内開発環境の技術チャレンジにも貢献。

【担当業務詳細】

・ フロントサイト（店頭チラシ作成サイト）承認機能の実装

- ASP.NET MVCを用いて、店舗担当者・本部担当者が作成したチラシ内容をプレビュー・承認・差し戻しできる承認機能を実装
- 承認状況のステータス管理、承認履歴の記録、メール通知を含むワークフローを担当

・ 運用管理サイトの新規開発

- 人事異動に伴うマスタ情報の管理・更新機能を設計・実装
- フロントエンドにReact + TypeScriptを採用し、バックエンドとAPI連携する構成を社内プロダクトで初導入。
- UI設計・状態管理・API通信部分の開発を担当

・ 組版サードパーティパッケージ用ラッパーAPIサーバの構築

- 組版パッケージとの連携処理をREST API化し、他案件でも再利用可能なAPI基盤を整備
- ASP.NET Web APIでのAPI実装

・ 組版用スクリプトの作成・移行

- 既存VBスクリプトをNode.jsへ移行し、保守性向上を実現

ゴルフコンペ記念品生成支援システムの構築

【チーム編成】

- ・ エンジニア 2名

※全体で3名

【言語・フレームワーク】

Ruby on Rails

【インフラ】

Nginx

【ツール】

Visual Studio Code ・ Microsoft Teams ・ Backlog

【プロダクト概要】

ゴルフコンペの参加者向けに配布する記念品作成を支援する業務システム。

コンペ当日に撮影した参加者の写真データを取り込み、オリジナル記念品へ印刷するためのデータ作成を効率化。

従来、手作業で行っていた画像の加工・印刷用データ作成をシステム化することで、記念品作成業務の迅速化とミス削減を実現し、参加者への満足度向上にも貢献。

【担当業務概要】

PJの途中から開発メンバーとして参画し、新規機能の設計、Ruby on Railsを用いた実装および既存機能の改修を担当。

【担当業務詳細】

- **Ruby on Railsを用いたAPIエンドポイントの設計・実装**
 - 記念品データ作成用の画像アップロード・データ取得・記念品印刷用データ出力機能のAPI開発
- **RSpecを用いた単体テストの実装**
 - 開発機能のテストコード作成と既存テストの改善を実施し、テストカバレッジと品質向上を推進
- **既存機能の改修・バグ修正**
 - 不具合対応および仕様変更に伴う既存API・機能の改修を担当

ナンバープレートデザイン公募サイト構築

【チーム編成】

- エンジニア 4名

※全体で6名

【言語・フレームワーク】

Cake PHP

【インフラ】

CentOS ・ MySQL

【ツール】

Atom ・ Bitbucket ・ Dropbox(ドキュメント管理) ・ Skype(社内用チャット)

【プロダクト概要】

オリンピック特別仕様ナンバープレートのデザイン公募を受け付けるWebサイトの構築。
一般ユーザーを対象に、ナンバープレートのデザイン応募および投票をオンライン上で受け付ける仕組みを提供。

【担当業務概要】

オリンピック特別仕様ナンバープレートのデザイン公募Webサイト開発プロジェクトにおいて、バックエンド開発・設計書の整備・テスト業務を担当。

【担当業務詳細】

- **CakePHPを使用した応募フォームのバックエンド機能実装**
 - 応募データの登録・バリデーション処理・管理画面との連携機能を実装
 - フロントエンドと連携するAPIインターフェースの実装
- **既存設計ドキュメントの改訂**
 - 要件変更に伴うDB設計書・画面設計書の修正、改版履歴管理の実施
- **テスト仕様書の作成およびテスト実施**
 - 応募フォーム機能を中心とした機能単体テスト・結合テストの仕様書作成