

職務経歴書

担当工程

上流工程から下流工程までを一貫して担当しています。

基本的なチーム構成は営業、デザイナー、フロント CSS、バックエンド、QA の 5 名体制になります。

担当	技術	内容	目標/課題
要件定義		WEB 会議に開発担当として同席します	説明・提案力
詳細設計	スライド	画面遷移図を作成して確認を進めます テキストベースで質問・回答を行います	効率化
DB 設計		視認性・拡張性を意識して ER 図を作成します 規模感: 最大 30 テーブル、70 万レコード 【その他】 インデックス, view, trigger	DB チューニング パーティショニング レプリケーション
インフラ	VPS, AWS ロードバランサ スケールアップ	OS から LAMP 環境の構築 WAF や Fail2ban でのセキュリティ対策 Zabbix でのサービス監視 【その他】 LDAP, postfix, AWS-EC2	CI/CD AWS, GCP Github Action
バックエンド	PHP/CakePHP Python	規模感: 見積もり 400 万円 / 3 ヶ月	チーム開発 OSS 開発/貢献 Node.js/Java
フロントエンド	React	使いやすさ・速度を重視します	SSG 量 結合テスト
アプリ開発	ReactNative	スムーズに動作するようメモリを意識します	

サーバー構築

VPS 構築をメインに、AWS-EC2 の構築や、LDAP サーバー構築、ロードバランサ構築等、幅広く携わりました。
基本的にはマニュアル通りに LAMP 環境+メールサーバー構築+セキュリティ対策を行います。
CentOS6 から CentOS7、AlmaLinux への移行を担当し、脆弱性対策とともにマニュアルの更新を行いました。
サーバー構築に半日~夕方まで時間がかかっていたところ、スクリプトを作成し 30 分以下に短縮しました。
直近ではビルド構築されていた WEB サーバーを改修し、PHP モジュール等を手動で導入しました。

開発

自社 CMS の改修- 汎用的関数のライブラリ化

こだわりは、応答時間を 0.8 秒以下にすることです。

処理を最適化した上で、改善が見込めない/難しい場合は他言語でも解決できないか模索します。

その結果としてバックエンド部署に所属しながらもフロントエンドにも携わりました。

- 時間のかかるサブコンテンツは、React を導入し非同期に
- ダイアログ処理が多いサイトには、React を導入し手軽に
- 動画を流す際に負荷が高い場合は、Python を利用し編集処理

プロジェクトまとめ

2ヶ月以上かかったプロジェクトの概要をまとめています。

その他にも1週間~2ヶ月未満の案件(HP作成,サーバ構築)に数多く携わっております。

わずかな待機時間には、自社開発、自社CMSの改修、スクリプト作成等に従事しております。

プロジェクト名	期間(目安)	金額(目安)	主な技術
サイネージアプリ	2022年12月~2ヶ月		CakePHP, ReactNative, Python
教育支援システム	2022年06月~3ヶ月	400万円	CakePHP, React, 言語解析, AWS, Laravel
内覧受付システム	2021年10月~2ヶ月	200万円	WordPress
販促システム	2021年07月~2ヶ月	200万円	CakePHP, SMS, OCR, ロードバランサ
学内システム	2021年03月~3ヶ月	300万円	CakePHP, LDAPS
見積システム	2020年06月~3ヶ月		CakePHP, Backlog

サイネージアプリ(2022-12~ 2 ヶ月)

PHP 管理画面でプレイリストを作成し、Android 端末からモニタに出力するアプリ。

特徴

自社開発, 5,000 円の Android 端末, モバイル開発

作業内容

内容	期間	技術	詳細
詳細設計	約 3 日		- プレイリストコンテンツ内容の確認 (動画,画像,url,テキスト,,) - コンテンツ表示方法の確認 (表示間隔, モニタ向き,,) - 端末環境の確認 (端末スペック,通信環境,,) - 編集権限の確認 (本社と支店とそれぞれの関係)
DB 設計	約 1 日		柔軟性を維持しながら管理者と店舗の権限を明確にすること
技術選定	約 1 日		- 端末の選定 (候補: シングルボードコンピュータか、Android 端末) - Android 開発言語の選定 (候補: Kotlin,ReactNative,Cordova) - 動画編集言語 (Python)
サーバー構築	約 30 分	Apache,Bash	自作スクリプトを利用し VPS 構築
開発-管理画面	約 8 日	CakePHP, Shell	DB に基づいた登録が可能であること アプリの要求に正しく API を返すこと 動画処理の進捗状況が視覚的に確認できること
開発-アプリ開発	約 5 日	ReactNative	管理画面で登録したデータと連携ができること プレイリストを正しく再生すること
開発-m3u8 拡張子導入	約 3 日	PHP, FFmpeg	ライブ配信で使われている拡張子(m3u8)を利用する クライアント側の読み込み速度を安定させること
開発-オフライン対応	約 5 日	ReactNative	プレイリストデータを外部ストレージに保存すること
開発-動画編集ツール	約 4 日	Python, MoviePy	プレイリストデータを 1 つの動画に変換すること

業務での実績

Android/ReactNative の開発経験

ストレージ操作の知識が深まり、同時に Android 端末への苦手意識もなくなりました。

m3u8 の活用

1 秒単位で動画を分割することで、サーバー参照型の読み込みを安定化することができました。
使用する場面は限定的ですが、貴重な経験を積むことができました。

Python の開発経験

サーバー側での動画編集処理を提案し、実現することができました。

教育支援システム(2022-06~ 4 ヶ月)

課題に対する生徒の意見を形態素解析し、集計することが主な目的のシステム

特徴

自然言語処理, React 導入, アクセス権限

作業内容

内容	期間	技術	詳細
詳細設計			- クライアント提供の仕様書を読み認識をすり合わせる - 権限の確認(管理者, 学校, 先生, 生徒) - 形態素解析の方法(MeCab)
DB 設計	約 3 日		クライアント提供の仕様書から DB を設計すること 不明な点は旧システムを参考にする
サーバー構築	約半日	AWS	VPS と同じ要領で構築する
開発-API 開発	約 24 日	CakePHP	DB に基づいた登録が可能であること アプリの要求に正しく API を返すこと
開発-形態素解析	約 3 日	MeCab	旧システムから処理方法が変わらないように構築すること システム側から辞書の追加を行うこと
開発-フロント開発	約 15 日	React	管理画面で登録したデータと連携ができること
開発-関数の共通化	約 5 日	React	再利用可能なページネーション 編集後に戻る時の Confirm 機能

業務での実績

形態素解析

形態素解析を学ぶことで、データ分析の実用性について再認識しました。
お問い合わせから検索方法など、応用が効く機能だと考えています。

内覧受付システム(2021-10~ 2 ヶ月)

スムーズな内覧受付機能を重視したホームページ

特徴

WordPress, ブロックエディタの拡張, API 開発

作業内容

内容	期間	技術	詳細
設計	約 1 日		物件情報、部屋情報、お知らせ等 カスタムフィールドによるリレーション対応
サーバー構築	約 30 分	Bash	Bash スクリプトを利用し構築する
開発-管理画面	約 3 日	WordPress	DB に基づいた登録が可能であること 登録のしやすい登録画面であること
開発-API 開発	約 3 日	WordPress	フロント側の要求に正しく API を返すこと
開発-多言語対応	約 1 日	WordPress	英語用の登録を可能にする
開発-ブロックエディタの拡張	約 2 日	プラグイン	ブロックエディタのブロックを追加
開発-カスタマイズ	約 3 日	プラグイン	お問い合わせとカスタムフィールドを連携させる 物件等の情報 CSV をインポート・エクスポートする

業務での実績

これまでも WordPress を実装してきましたが、
データ登録の大変さ、集計方法、スピード面など、多くの面で悔いが残ってしまいました。

多言語化

1 つの物件に対する項目量が多いため、日本語の枠の隣に英語用の枠を用意するのではなく、
英語用のサイトとして、サイトごと複製することが最善の策だと判断し実装しました。
今思えば、DeepL の API で自動翻訳する策も視野に入れるべきだったと思います。

CSV インポート・エクスポート

WP の制限が多いものの、通常の CMS と同じようにインポート対応しました。
実装時は CSV が前提にあったので画像のインポートは実用的ではないと考えていましたが、
ZIP 形式なら対応できるのではないかと思います。

API 連携

REST API を使う際のフロントエンド部署との連携の難しさと、実装した時の使いやすさを再認識しました。
フロントエンドとの連携を深めるためにも、可読性の高い API 設計書(Swagger)を作成することと、自分自身 React
を学ぶことを決めました。

メーカー向けの販促システム(2021-07~2ヶ月)

SMS 認証を行い景品を掲示、レシート画像を元に購入確認を行い、承認後に景品を送るシステム。
新規システム開発を担当(詳細設計、DB 設計、実装、テスト、保守)

規模

LB

SMS は計〇〇円分、1 万人への送信

特徴

・SMS を導入し、簡易的に不正応募を防止 ・OCR を導入し、目視で行う購入確認作業の効率化 ・LB 構築を行い、
負荷の分散(rsync)

作業内容

内容	期間	技術	詳細
詳細設計	約 4 日		キャンペーンごとに複製することを考慮する
DB 設計	約 2 日		キャンペーンごとに複製することを考慮する
選定	約 2 日		SMS 会社 OCR 会社
サーバー構築	約 1 日	Bash	Bash スクリプトを利用し構築する 2 つのサーバーをロードバランサを組む 振り分け先の 2 つサーバーを rsync 同期させる
開発-管理画面	約 20 日	CakePHP, crontab	DB に基づいた登録が可能であること 登録のしやすい登録画面であること crontab によるスケジュール処理
開発-SMS	約 3 日	SMS	応募時・登録時・結果発表全てに SMS を用いる 管理画面からの CSV 送信にも対応すること 未着時にコールバックを行う
開発-OCR	約 3 日	OCR	レシートを OCR で読み込み、目視での作業の簡略化を図る
開発-フロント	約 3 日		外部フロントエンジニアとの連携

学内ファイル共有システム(2021-03~ 4 ヶ月)

ユーザー(教職員と学生)がページ(複数ファイルやテキスト)を登録し公開申請をする。学部毎に存在する担当者が承認することで、学内に共有されるシステム。

新規システム開発を担当(詳細設計、DB 設計、実装、テスト、保守)

規模

70 万レコード, 20 テーブル

特徴

- 既存 LDAP サーバとの連携(ログイン処理、アカウント情報の取得)
- 更新前のデータを残すことで履歴表示に対応
- ブロック型エディタの作成(ファイル、テキスト、画像)
- 操作ログの監視
- 通知処理(承認依頼、承認結果、公開予定)

作業内容

内容	期間	技術	詳細
詳細設計	約 3 日		用意された仕様書を読み認識をすり合わせる
DB 設計	約 3 日		リッチエディタ登録ができること あらゆるテーブルにユーザー権限を付与すること あらゆる操作のログを残すこと 複数のページ種別/スラッグ対応
サーバー構築	約 30 分	Bash	Bash スクリプトを利用し構築する
開発-管理画面	約 15 日	CakePHP	DB に基づいた登録が可能であること 登録のしやすい登録画面であること
開発-記事入力画面	約 8 日	JS	WP のようなリッチエディタを実装すること それを HTML として出力すること
開発-LDAP 連携	約 5 日	LDAP	アカウントは LDAP から取得すること テスト用の LDAP サーバーを構築すること
開発-通知処理	約 3 日	CakePHP	ページ追加・更新・削除時に通知を行うこと
開発-フロント	約 10 日		登録した内容を正しく表示すること

社内向け見積システム(2020-06~3ヶ月)

案件・見積・請求先・サイト・担当者・仕入を一元的に管理するシステム。
新規システム開発及び、旧システムからの DB 移行を担当。

特徴

- 登録情報から見積書・請求書の自動生成(Excel, PDF)
 - タスク管理ツール(Backlog)と連携し、案件ごとの実働工数を算出
 - SSL 監視, React 導入
-