

# スキルシート

更新日: 2025 年 10 月 1 日

氏名	T.N	性別	男	年齢	32	最寄駅	池袋駅
学歴	専門高校卒						
保有技術	日商簿記3級, 日商簿記2級, Ruby技術者認定試験2.1シルバー, Ruby技術者認定試験2.1ゴールド, AWS CLF, AWS SAA-C03, AWS SOA-C02, AWS DVA-C02						
自己PR	ルーティン作業やタスクの自動化、業務に必要な知識の学習意欲あり、改善点があれば積極的に業務改善、多くのAWSサービス使用経験あり、キャッチアップの判断が早い、テストの重要性を理解、Ruby歴4年以上、自動テスト教材の販売歴あり						

No.	期間 (稼動)	プロジェクト名 業務概要	役割 規模	サーバ OS	DB	FW・MW ツール等	使用言語	作業工程					
								要件定義	基本設計	詳細設計	実装・単体	結合テスト	総合テスト
1-1	現在 ↑ 2025/4 (9ヶ月)	AIを活用したチャットアプリの自動テスト基盤の作成、運用メンテ	チーム 1名  全体 100名以上	mac	mysql	Google Meet API chatGPT playwright	TypeScript						
		[自動テスト基盤] Playwright, GoogleMeetAPI, テスト結果集計アプリ  [自身で作成した基盤系] - テスト自動化に必要な基盤の設計書を作成 - Playwright自動テスト基盤の作成 - テスト結果通知処理を作成 - テスト結果集計アプリを作成  [業務] - 作成したテストのメンテナンス - 定期実行 - Fail率の高いテストを安定化 - テスト振り返りと翌週にやることのタスク管理 - 別チームに依頼されたタスクのサポート - Playwrightによる負荷計測系のサポート - モバイルアプリの基盤作成						●	●				●
1-1	2025/3 ↑ 2021/5 (44ヶ月)	会計アプリの自動テスト作成、自動テスト基盤作成  [自動テスト実行ツール] 移行前: rspec, selenium, capybara, Jenkins 移行後: playwright, github actions, report portal  [自動テスト保守運用環境] staging, integration, local  [通常業務] - 本番デプロイ前のE2Eテスト実行(1日2回) - 新規E2Eテストの作成 - フレーキーテストの安定化 - 仕様変更による既存E2Eテストへの影響調査 - レビュー対応、問い合わせ対応 - Doc作成(google doc, スプレッドシート, kibela) - MTG系の司会進行 - seleniumからplaywrightへの移行 - google拡張機能のメンテナンス  [特別業務] - E2E実行環境のAWSアカウント移行 - SlackからEC2操作できるように新規作成 - ワンクリックで環境構築できるgithub workflow作成	チーム 8名  全体 1000名以上	mac EC2(ubunts)	AWS Aurora mysql	Jenkins AWS全般 slackAPI github action GAS Docker RSpec JIRA chatGPT localstack playwright report portal	Ruby TypeScript		●	●			●

2021/4 ↑ 2021/1 (3ヶ月)	学習支援サイトの実装  【業務内容】 ■プロジェクトA チーム人数:1人(自分のみ) 開発環境:rails(フロント、バック) 業務内容:運用しているアプリの保守、機能追加 ・rubyのバージョンアップ ・レスポンシブ対応 ・ボタン押下でPC画面用,SP画面用のデザインに切り替えができるように修正 ・既存機能を別ページに移行  ■プロジェクトB チーム人数:2人 開発環境:Vue(フロント),rails(api) 業務内容:運用しているアプリをベースに新規アプリの開発 ・デザイン周りや、メニュー構造の作成 ・CRUD周りの新規画面の作成 ・権限による表示、非表示の制御を実装  その他 ・レビューなし ・テストなし		チーム 2名	mac	mysql	gitbucket Backlog	Ruby2.3.1 Rails5 Vue2	
2								
2021/12 ↑ 2021/7 (6ヶ月)	データマーケティングツールの実装、テスト  ■連動テスト リリースされているアプリの保守、及び修正箇所をローカル環境、ステージング環境(AWS)でテスト実施  ■運用しているアプリを参考に0から新規アプリ開発 ・slimで書かれているファイルをVue.jsに置き換え ・カード式、テーブル式によるCRUD周りの実装(フロント、バックエンド) ・ソート、フィルター、検索、複製、複数削除などの実装(フロント、バックエンド) CircleCIがsuccess!になるようにeslint、rubocopにてコードを修正  ■単体テスト実施 ・RSpecにて実施(テスト項目書なし)	チーム 10名  開発 100名  全体 1000名以上	windows	MySQL5.7	docker vagrant github Rails6	Ruby2.7.1 Vue3	●	
3								
2021/6 ↑ 2021/1 (6ヶ月)	大手携帯会社のECサイトを実装、テスト  ■詳細設計書の作成(机上内部レビューあり) ・AmazonS3へのファイルアップロードに関する詳細設計(パッチ処理)  ■実装 ・AmazonS3ファイルアップロード(パッチ処理)を実装 ・File、 FileUtils(Rubyのライブラリ)を使用した圧縮、移動、削除を行う処理の実装 ・その他、詳細設計書をもとにパッチ処理に関するカラムの追加など  ■単体試験項目書の作成、テスト実施 ・実装担当箇所の単体テスト項目書の作成(机上内部レビューあり) ・テスト実施はRSpecか目視で確認(C1網羅、パターン網羅)  ■結合試験項目書の作成 ・基本設計書、詳細設計書、単体テスト項目書を参考に試験項目書を作成(レビュー、プロジェクトリーダーによる2度の内部レビューあり)  ■結合試験実施(AWS) ・MySQLにてテストデータを作成(表示条件に一致するデータなど) ・ログにてリクエストとレスポンスが想定通りであることの確認	チーム 22名  開発 22名  全体 100名以上	Windows10 ubuntu	MySQL5.7	Rails5 github	Ruby2.6 JavaScript jQuery	● ● ● ● ● ●	
4								
2020/12 ↑ 2020/10 (3ヶ月)	1.治験サイトの管理者画面実装(Ruby) 2.電気自動車用の運転効率画面の実装(PHP) 3.治験サイトの管理者画面実装(Ruby)  【業務内容】 1.治験アンケートアプリのロール画面実装(9月～10月末) 主に施設スタッフが患者の容体を確認するサイトとなります。 ・フロント側をNuxt.jsとElementalUIで作成し、バックをRailsでAPI実装しました。 ・ログイン画面の実装、パスワードの変更、一覧画面、検索画面、CSVダウンロード画面など  2.家庭用電気自動車の運転効率画面の実装(11月～12月末)主に一般のお客様が自動車の運転効率を確認するサイトとなります。 ・テンプレートファイルを参考に、jQueryにてグラフ絞り込み、CSVダウンロードなどを実装  3.治験アンケートアプリのロール画面実装(1月～2月末) 主に施設スタッフが患者の容体を確認するサイトとなります(1とは異なるアプリ) ・ReactJSで作られたフロント側が用意されており、バック側をRailsにて実装	チーム 2名	Windows10 ubuntu	MySQL5.7	Ruby on Rails5 Bootstrap4 Git docker ElementalUI	Ruby2.5 HAML JavaScript jQuery PHP nuxt.js	●	
4								
2020/9 ↑ 2019/11 (10ヶ月)	1.プロ野球ファンサイトとプロ野球ECサイトの動作テスト 2.プロ野球分析サイトの不具合修正 3.スポーツ用品ECサイトの実装と単体テスト  【業務内容】							

1.プロ野球ファンサイトとプロ野球ECサイトの動作テスト  
主に野球ファン会員様がチケットやグッズを購入するサイトとなります。  
・仕様書、ER図、テーブル定義書などを参照してシナリオテスト、及び単体テストの実行(全画面網羅)  
・SQLを使用したテストデータの作成

2.プロ野球分析サイトの不具合修正  
主に監督やコーチなどが打席打球分析の為に使用するサイトとなります。  
・Redmineで割り振られたチケットを元にバグ修正(画面遷移時のエラーが表示される不具合など)  
・分析されたデータの印刷、PDFレイアウトの実装

3.スポーツ用品ECサイトの実装  
販売会社様が商品管理などを行うバックエンド側を製造する開発プロジェクトとなります。  
既存サイトのリニューアルをプロ野球ECサイトを基盤に再開発を行いました。  
・仕様書の作成(機能概要、機能詳細など)  
・仕様書をもとに新規画面の実装(約15画面)、並びにレビュー指摘箇所の修正  
・単体テスト(Excelにてテスト項目書の作成し自身で打鍵テスト)  
・バグ回収(チケットを約30個ほど対応)

スポーツ用品ECサイトはOSSであるsolidusを基盤としておりgithubで機能を確認しながら実装を行いました。  
テンプレートエンジンはHAMLやHandlebarsなどを使用しております。  
ransackを使用した検索フォームの実装(1画面30フォーム)、CSVアップロード、CSVダウンロード、CSVフォーマットのダウンロード、CSVデータチェック、Excelアップロード、Excelダウンロード、一括更新、請求書レイアウトの作成(軽減税率対応)、などの機能も実装しており、controller周りが膨らみすぎないようmodelにscopeを追加するなどして可読性を意識しました。

チーム  
15名Windows10  
ubuntu

MySQL5.7

Ruby2.5  
HTML5  
HAML  
CSS  
JavaScript  
jQueryRuby on Rails5  
Bootstrap4  
Git  
docker