

基本情報

プロフィール

項目	値
ハンドルネーム	つくぼし
最終学歴	早稲田大学 商学部卒
主要職種	クラウドインフラエンジニア

アカウント

- [DevelopersIO](#)
- [Zenn](#)
- [Github](#)
- [SpeakerDeck](#)
- [X](#)

業務スキル

クラウドインフラ設計

クラウドインフラの設計を専門としており、特にAWS のアーキテクチャ設計・構築が一番得意です。
モノリシック、マイクロサービス、静的ウェブサイトホスティング、CICD パイプライン、マルチアカウント、イベント駆動といった様々なパターンの設計・構築経験があります。
AWS アーキテクチャを構築するにあたり、CloudFormation、Terraform、CDK といった IaC の開発経験も多数あります。

生成AIの専門知識

LLMが流行し始めた頃から生成 AI、RAG、及び AI エージェントの分野に特化し、様々な情報を発信してきました。
生成 AI を活用したチャットボットアプリの自社開発や導入支援に携わった経験があります。
さらに、生成 AI インフラに特化した社内チームのリーダーを務めていたり、生成 AI コンピテンシーの取得に貢献した経験もあります。

サーバレス開発

きちんとコードを読んで様々な問題に対応できるエンジニアになりたいという思いから、インフラのみならずサーバレスのアプリケーション開発経験も積極的に積んできました。
アーキテクチャとしては、主に Lambda や ECS Fargate、Cloud Run 上でのサーバレスアプリケーションを設計・開発してきています。
言語としては主に Python 及び TypeScript を使用したバックエンド開発経験があります。

プロジェクトマネジメント

プロジェクトマネージャーを務めることが多く、チームマネジメント及び顧客折衝の経験が豊富です。
小規模チームで(3 - 5 人程度)でリーダーを務めた事が多数あります。
プロジェクトを成功に導くため、メンバーのモチベーション向上や業務効率化に繋がるような仕組みづくりを大切にしています。

アウトプット活動

多くの人と技術の面白さについて語り合いという思いから、ブログ執筆や勉強会登壇、個人開発等のアウトプット活動に積極的に取り組んできました。
DevelopersIO、Zennなどで100本以上の技術記事を執筆すると共に、技術雑誌への寄稿実績もあり、AWSや生成AIを中心とした最新技術を発信しています。
またJAWS-UGを主とした勉強会への登壇も20回以上実施しており、様々な形態での発信を心がけています。

技術スキル

実業務で使用した技術のみ列挙します。

AWS

IAM VPC ELB(ALB・NLB) EC2 EFS RDS(MySQL・Aurora) ElastiCache(Redis) S3 CloudFront Route 53 ACM WAF
ECS(Fargate) App Runner ECR API Gateway Lambda(Node.js・Python) DynamoDB Step Functions CodeCommit
CodeBuild CodeDeploy CodePipeline SQS SNS Data Firehose EventBridge CloudWatch (Alarm・Logs) Security
Hub GuardDuty Control Tower Organizations Identity Center Service Catalog DevOps Guru Bedrock Kendra

Google Cloud

Cloud Run Alloy DB

IaC

Terraform CloudFormation SAM CDK

OS

Linux Windows

Programming Language / Framework

Python TypeScript Node.js Fastify Prisma

SaaS

GitHub GitHub Actions

保有称号

表彰

表彰名	対象年度
Japan AWS Top Engineer	2024/2025
AWS Community Builder	2024/2025

資格

資格名	取得日
AWS DOP	2023/6/15
AWS SAP	2022/8/3
LPIC-2	2021/6/28
ORACLE MASTER Silver Oracle Database 12c	2021/3/29
基本情報 技術者試験	2019/11/20

経歴

株式会社ゼスト(業務委託：2025/3 -2025/6、正社員：2025/7 - 2025/8)

業務要約

役割

バックエンドエンジニア

プロジェクト一覧

- 介護事業者向けZEST SCHEDULE開発
- 関係者向けZEST SCHEDULE ADMIN開発

業務内容

- TypeScript/Fastify/Prismaを用いたバックエンド開発
- Cloud Runを用いたコンテナデプロイ
- Alloy DBを用いたSQLオペレーション作業

使用技術

- 言語・フレームワーク：TypeScript Node.js Fastify Prisma
- Google Cloudサービス：Cloud Run Alloy DB
- SaaS：GitHub GitHub Actions

クラスメソッド株式会社(正社員：2022/3 - 2025/6)

参画したプロジェクトが多いため、以下で主要なもののみ個別で記載します。

SaaSログ連携・保管コンテナシステム設計構築 (2024/9 - 2025/6)

概要

SaaS から出力されるログを定期的に AWS にアップロードし、特定用途で使用するためのログ形式に加工するコンテナアプリケーションをフルスクラッチで開発

役割

プロジェクトマネージャー/バックエンドエンジニア

業務内容

- 技術営業/顧客折衝/チームマネジメント(3 人)
- コンテナアプリ/インフラ設計書作成
- コンテナアプリ/インフラ構築作業

使用技術

- AWS サービス：VPC ECR ECS(Fargate) Lambda(Python) SQS SNS S3 CloudWatch (Alarm・Logs) EventBridge Step Functions
- IaC：CloudFormation SAM
- SaaS：GitHub

AI-Starter内製開発(2024/1 - 2025/6)

概要

生成 AI を活用した自社SaaSチャットボットアプリをフルスクラッチで開発

役割

生成AIエンジニア/インフラエンジニア

業務内容

1. 生成AIインフラ設計書作成
2. 生成AIインフラ構築
3. 生成AIインフラデプロイ手順整備

使用技術

- AWS サービス： WAF App Runner DynamoDB Secrets Manager Lambda(Node.js) Bedrock Kendra S3
- IaC： CDK(Typescript)

参考URL

- [生成AI環境構築サービス「AI-Starter」 | サービス | クラスメソッド株式会社](#)

AIチャットボット設計構築 (2024/9 - 2025/3)

概要

生成 AI を活用したOSSである[GenU](#)を用いたチャットボットを構築

役割

プロジェクトマネージャー/生成AIエンジニア

業務内容

1. 技術営業/顧客折衝
2. チャットボット設計書作成
3. チャットボット構築作業

使用技術

- AWS サービス： WAF DynamoDB Secrets Manager Lambda(Node.js) Bedrock Kendra S3
- IaC： CDK(Typescript)

社内生成AIインフラチーム活動(2024/1 - 2024/8)

概要

生成 AI 関連の案件に対応する体制作りのため、生成 AI に特化したインフラを専門に部署を跨いだ交流を促進する活動を実施

役割

チームリーダー/生成AIエンジニア

業務内容

1. 生成AI勉強会開催
2. 生成AI案件対応推進
3. 生成AIコンピテンシー取得

参考URL

- [40社の支援実績によりAWSの「生成AIコンピテンシー」を取得～業務システムへの組み込み、伴走型の人材育成支援など、技術力と問題解決力が評価～ | クラスメソッド株式会社](#)

RAG システム設計構築 (2023/11 - 2024/3)

概要

生成 AI を活用したチャットボットアプリで使用する、社内ドキュメントを検索するための RAG システムを構築

役割

生成AIエンジニア/インフラエンジニア

業務内容

1. RAG システム設計書作成
2. RAG システム構築構築

使用技術

- AWS サービス： WAF App Runner Bedrock Kendra S3
- IaC： CloudFormation

AWSマルチアカウント管理支援通知集約システム内製開発(2023/8 - 2024/2)

概要

複数の AWS アカウントに対する通知設定を一括で付与できる、通知集約システムを開発

役割

インフラエンジニア

業務内容

1. 通知集約システム設計書作成
2. 通知集約システム構築
3. 通知集約システムデプロイ手順整備

使用技術

- AWS サービス： Security Hub GuardDuty Step Functions EventBridge
- IaC： CDK(Typescript)

参考URL

- [AWSマルチアカウント管理支援 | クラスメソッド株式会社](#)

SaaSログ連携・保管サーバレスシステム設計構築(2023/5 - 2024/6)

概要

SaaS から出力されるログを定期的に AWS にアップロードし、特定用途で使用するためのログ形式に加工するサーバレスアプリ及びCICDシステムをフルスクラッチで開発

役割

プロジェクトマネージャー/バックエンドエンジニア

業務内容

1. 技術営業/顧客折衝/チームマネジメント(3 人)
2. サーバレスアプリ/CICD システム設計
3. サーバレスアプリ/CICD システム構築

使用技術

- ・ AWS サービス： EventBridge Lambda(Python) SQS SNS S3 CloudWatch (Alarm・Logs) DevOps Guru
- ・ IaC： CloudFormation SAM
- ・ SaaS： GitHub GitHub Actions

認証システム用コンテナ基盤技術支援(2022/11 - 2023/6)

概要

顧客情報利用のための認証システムにおいて稼働が必要な、アプリケーションのコンテナ化に伴う技術支援を実施。

役割

プロジェクトマネージャー/インフラエンジニア

業務内容

1. 顧客折衝/チームマネジメント(5 人)
2. コンテナインフラ/CICD システム設計
3. コンテナインフラ/CICD システム構築

使用技術

- ・ AWS サービス： VPC ELB(ALB・NLB) ECS EFS RDS(Aurora) ElastiCache(Redis) S3 CloudFront WAF ECS(Fargate) ECR API Gateway CodeCommit CodeBuild CodeDeploy CodePipeline EventBridge
- ・ IaC： CloudFormation

基幹システム用マルチアカウント基盤技術支援 (2022/7 - 2024/3)

概要

部門・プロジェクト単位での権限/環境の分離や、一元的に統一されたガバナンスの確保が必要な、社内システム用のマルチアカウント管理基盤の設計/構築に伴う技術支援を実施。

役割

プロジェクトマネージャー/インフラエンジニア

業務内容

1. 技術営業/顧客折衝
2. マルチアカウント基盤設計に必要な技術質問対応
3. マルチアカウント基盤構築に必要な動作検証

使用技術

- ・ AWS サービス： Control Tower Organizations Identity Center Service Catalog WAF Data Firehose CloudWatch (Logs)
- ・ IaC： CloudFormation
- ・ SaaS： GitHub

ECサイト用AWS環境構築(2022/6 - 2022/7)

概要

新規 EC サイトにおけるランディングページに必要な静的ウェブサイトホスティング基盤を構築。

役割

プロジェクトマネージャー/インフラエンジニア

業務内容

1. チームマネジメント(3 人)/顧客折衝
2. 静的ウェブサイトホスティングインフラ設計書作成
3. 静的ウェブサイトホスティングインフラ構築

使用技術

- ・ AWS サービス： IAM CloudFront S3 Lambda(Node.js) Route 53 ACM
- ・ IaC： Terraform

CMSインフラ用AWS環境構築(2022/5 - 2022/8)

概要

動画サイト構築 CMS 新規導入のためのウェブサイト基盤を構築。

役割

プロジェクトマネージャー/インフラエンジニア

業務内容

1. 顧客折衝
2. Webサイトインフラ設計書作成
3. Webサイトインフラ構築

使用技術

- ・ AWS サービス： IAM VPC ELB(ALB) EC2 RDS(MySQL) ElastiCache(Redis) S3 Route 53 ACM
- ・ IaC： Terraform

兼松エレクトロニクス株式会社(正社員：2019/4 - 2022/2)

業務要約

役割

インフラエンジニア

プロジェクト一覧

- ・ 基幹システム用vSphere基盤更改(2021/10 - 2022/2): Oracle19c DBサーバの設計・構築
- ・ 情報システム用vSphere基盤更改(2021/10 - 2021/12): RHEL8・Apache2.4環境の構築
- ・ 基幹システム用z/VM基盤更改(2021/6 - 2021/12): SLES12 SP5へのバージョンアップ
- ・ 基幹システム用VIOS基盤更改(2021/1 - 2021/7): WAS9.0.5 APサーバの設計・構築
- ・ 社内コンテナ検証チーム活動(2020/6 - 2022/1): Kubernetes・OpenShift検証環境構築

業務内容

- ・ オンプレミスインフラの設計・構築
- ・ コンテナ技術(Docker/Kubernetes)の検証・導入推進

使用技術: Linux(RHEL/SLES), VMware vSphere, Oracle, WebSphere, Apache, Docker, Kubernetes, OpenShift

業務外活動

副業

株式会社DiningX(業務委託：2025/1-現在)

概要

AWSを基盤とするWebアプリケーションの設計/開発を行う自社開発企業にて、アーキテクチャの相談役として参画

役割

- 技術顧問

業務内容

- AWS上でのインフラ設計/構築に関する相談対応
- AWS上でのトラブルシューティング

個人開発

以下では開発した自作リポジトリの内、主要なものを記載します。

リポジトリ名	言語・フレームワーク	概要
dotfiles	Bash	自身の PC セットアップ用の dotfiles
honox-resume-template	TypeScript HonoX Vite	Markdown職務経歴書をWebサイトとPDFに変換し自動デプロイするテンプレート
sam-notify-aws-billing	SAM Python	AWS 利用料金を通知するサーバレスシステムコード
sam-ai-cloud-engineer	SAM Python	Amazon Bedrock を活用し構成図を CloudFormation テンプレートとパラメータシートに変換するサーバレスシステムコード
nextjs-mastra-mcp-with-aws	CDK TypeScript	AWS とMastraで実装したMCP Serverを呼び出すAIエージェント
gas-count-schedule-time	Apps Script TypeScript	Googleカレンダーの予定から色ごとに工数をカウントし集計するアプリコード
sam-bedrock-kb-aurora-rag	SAM Python	Amazon Aurora と Amazon Bedrock Knowledge Bases を用いた RAG システムデプロイするためのコード
terraform-microservices-template	Terraform	AWS を用いたマイクロサービス構成用 Terraform テンプレート

ブログ投稿

以下では投稿したブログの内、主要なものを記載します。

DevelopersIO

2022/3から2025/6までに、100本以上の記事を作成していました。代表的な記事は以下の通りです。

- [AWS入門ブログリレー2024～Amazon Bedrock編～](#)
- [CloudFormation一撃で作るAWS料金通知ツール\(Email/Slack/LINE対応\)](#)
- [Knowledge Bases for Amazon Bedrock \(with OpenSearch Serverless\)をSAMで実装してみた](#)
- [S3にアップロードしたAWS構成図をCloudFormationに変換するシステムを作ってみた](#)
- [Kendraのアクセスコントロール設定をEntra IDのOpen ID Connectを用いて有効化してみた](#)
- [AWSの構成図をChatGPT\(GPT-4V\)に読み込ませてIaCコードを生成してみた](#)
- [ECSとCodePipelineのブルー/グリーンデプロイ構成をCDKで実装してみた](#)
- [AuroraとRDSの違いを一覧表でまとめてみた](#)
- [FireLens\(Fluent Bit\)におけるログルーティングの仕組みについて調査してみた](#)
- [TerraformでCloudFront Functionsを環境ごとに有効化/無効化してみた](#)

Zenn

2023/5から現在まで投稿している主要のブログサイトになります。代表的な記事は以下の通りです。

- [Macの環境をdotfilesでセットアップしてみた改](#)
- [Google Calenderの予定をTypeScriptを用いて色別で工数集計してみた](#)
- [GmailをTypeScriptとLINE Messaging APIを用いてLINEに自動転送してみた](#)
- [MCP Serverを呼び出すAIエージェントをMastra、Next.js、AWS CDKで実装してみた](#)
- [Markdownで書いた職務経歴書を良い感じのWebサイトとPDFに自動変換するツールを作ってみた](#)
- [Claude Codeとghコマンドで作るカスタムスラッシュコマンドn選](#)

イベント登壇

以下では登壇したイベントの内、主要な資料及び動画を記載します。

[Mita.ts #5](#)：Mastraに入門してみた ~~AWS-CDK~~を添えて

- [Mastraに入門してみた ~~AWS-CDK~~を添えて - Speaker Deck](#)

[Classmethod AI Talks\(CATs\) #14](#)：AWSエンジニアに捧ぐLangChainの歩き方

- [AWSエンジニアに捧ぐLangChainの歩き方 - Speaker Deck](#)
- [AWSエンジニアに捧ぐLangChainの歩き方 - YouTube](#)

[Toranomon Tech Hub #2](#)：世界の中心でApp Runnerを叫ぶ ~~Aurora-DSQL~~を添えて

- [世界の中心でApp Runnerを叫ぶ ~~Aurora-DSQL~~を添えて - Speaker Deck](#)

[Classmethod AI Talks\(CATs\) #5](#)：初めてのGPTs ～ネコ派を〇〇派に変える技術～

- [初めてのGPTs ～ネコ派を〇〇派に変える技術～ - Speaker Deck](#)
- [クラメソエンジニア陣が執筆！書籍『ビジネスのためのChatGPT活用ガイド』でChatGPTの業務活用を始めてみよう - YouTube](#)

[JAWS FESTA 2024 in 広島](#)：Amplify Gen 2ではじめる 生成AIアプリ開発入門

- [Amplify Gen 2ではじめる 生成AIアプリ開発入門 - Speaker Deck](#)

[Classmethod Odyssey](#)：AWSで構築するパターン別RAG構成解説

- [AWSで構築するパターン別RAG構成解説 - Speaker Deck](#)
- [AWSで実現するRAG構成！Retrieverの種類と最適なインフラパターン解説 - YouTube](#)

[Bedrock Claude Night 2](#)：AWS構成図からCloudFormationとパラメータシートを自動生成するシステムを作ってみた

- [AWS構成図からCloudFormationとパラメータシートを自動生成するシステムを作ってみた - Speaker Deck](#)

[JAWS-UG東京 ランチャタイムLT会 #11](#)：5分で分かる(かもしれない)Vector engine for OpenSearch Serverless

- [5分で分かる\(かもしれない\)Vector engine for OpenSearch Serverless - Speaker Deck](#)

[JAWS-UG朝会 #55](#)：君はApplication Composerというサービスを知っているか

- [君はApplication Composerというサービスを知っているか - Speaker Deck](#)

[JAWS-UG CDK支部 #12](#)：CDKをCloudFormationテンプレートとして利用する際の注意点n選

- [CDKをCloudFormationテンプレートとして利用する際の注意点n選 - Speaker Deck](#)

[JAWS-UG朝会 #51](#)：インフラエンジニアのためのLambda実践入門

- [インフラエンジニアのためのLambda実践入門 - Speaker Deck](#)

[DevelopersIO 2023](#)：AWSとGitHubを用いたパターン別CI/CD構成解説

- [AWSとGitHubを用いたパターン別CI/CD構成解説 - Speaker Deck](#)
- [AWS基盤でのCI/CD構成を徹底解説！CodeシリーズとGitHubの使い分けも解消 - YouTube](#)

[JAWS-UG朝会 #39](#)：(今更ながら)AWSのコンテナサービスについてざっくりまとめてみる

- [\(今更ながら\)AWSのコンテナサービスについてざっくりまとめてみる - Speaker Deck](#)

[JAWS-UG CLI専門支部 #273R](#)：LambdaとLine Messaging APIで湯婆婆botを作ってみた

- [LambdaとLine Messaging APIで湯婆婆botを作ってみた - Speaker Deck](#)

[DevelopersIO 2022](#)：AWS初心者に捧ぐコスト可視化のススメ

- [AWS初心者に捧ぐコスト可視化のススメ - Speaker Deck](#)
- [AWS初心者に捧ぐコスト可視化のススメ - YouTube](#)

[AKIBA.AWS ONLINE #08](#)：それ、t2.micro選んで大丈夫？

- [それ、t2.micro選んで大丈夫？ - Speaker Deck](#)

[JAWS-UG CLI専門支部 #256M](#)：CLI専門支部に参加したらAWSエンジニアに転職できた件について

- [CLI専門支部に参加したらAWSエンジニアに転職できた件について - Speaker Deck](#)

特筆事項

以下では特記すべき経歴について記載します。

[Software Design 2024年11月号](#)寄稿

「第1特集 新世代の開発スタイル はじめてのAI駆動開発」における「第4章：Infrastructure as Codeで生成AIを活用するアーキテクチャ図⇔IaCコードの変換も自由自在」を寄稿

[JAWS-UG 茨城運営](#)

JAWS-UG茨城の初期メンバーとして2024/11からの立ち上げに貢献し、運営メンバーとして活動中