## クライアント企業様

「生産部品の在庫管理システム」

AWS環境構築手順書

| 作成者 | 三好　順樹 |
| --- | --- |
| 作成日 | 2024年12月11日 |
| 最終更新日 | 2024年12月11日 |

## 目次

[**本書の目的と概要**](#_br92nkwb2mxp) **3**

[環境情報](#_8m8t9liy8de5) 3

[AWSリソース構成図](#_pep9ohidpn7p) 4

[**業務要件**](#_hc5iohcyl9xt) **4**

[機能要件](#_stq8dwoy969e) 4

[非機能要件](#_oseurqhstyxx) 5

[**構築実施日・作業者・作業場所・条件などの当日の作業情報（※架空）**](#_x9ka75xztdc6) **6**

[構築実施日](#_7ezja01yox5t) 6

[作業者](#_jo9oiziw0q8l) 6

[作業場所](#_dv7lqxdwm7d3) 6

[条件](#_xgmpzdq5esxk) 6

[**App Runner構築手順**](#_sbr745m34ail) **6**

[**今後の予定**](#_v77tht4ij4ny) **10**

### 本書の目的と概要

本手順書では「生産部品の在庫管理システム」をAWS App Runner上に構築し、ネロギング・モニタリングの設定を実施し、またコスト管理の設定することを目的としています。

#### 環境情報

構築作業に必要な環境を示します。

・Git/Github

・Docker

・AWS

AWSリージョン

App Runner

CloudTrail

CloudWatch

SNS

Budgets

RDS

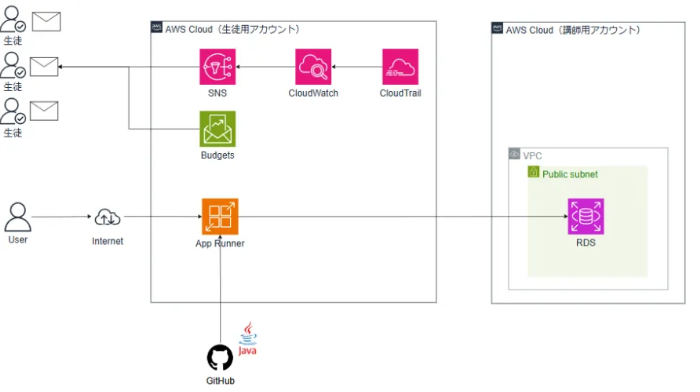
IAM

VPC

・デプロイ対象のリポジトリ

https://github.com/digital-ojt/infra-course-drone/

#### AWSリソース構成図



### 業務要件

クライアント企業様の生産部品の在庫管理システム新規開発プロジェクトの機能要件と非機能要件を明記します。

#### 機能要件

・部品情報管理

部品の一覧表示と詳細情報の閲覧

部品の追加、編集、削除機能

部品カテゴリーの設定と管理

・在庫管理

部品の入庫および出庫の記録

在庫数量の管理

在庫の閲覧と検索機能（カテゴリー、部品番号、在庫状況などでのフィルタリング）

・注文処理

部品の注文と受け取りのトラッキング

注文履歴の表示と管理

注文ステータスの更新

在庫が一定数以下になった場合の自動発注

・ユーザー管理とアクセス制御

管理者と一般ユーザーの役割の設定と管理

ユーザーごとのアクセス権限の設定（閲覧、編集、削除など）

#### 非機能要件

・パフォーマンス要件

システムの応答時間：ユーザーの要求に対するシステムの応答時間は2秒以内であること。

同時アクセスのサポート：システムは最大100人の同時アクセスをサポートすること。

データ処理速度：在庫データの更新や検索などのデータ処理は高速かつ効率的に行われること。

・セキュリティ要件

アクセス制御：ロールベースのアクセス制御（RBAC）を実装し、ユーザーごとに適切なアクセス権を付与すること。

データの暗号化：重要なデータはトランジットおよびアットレストで暗号化すること（AES256など）。

ログと監査：システムへのアクセス、変更、操作などのアクティビティをログとして記録し、適切に監査可能な形式で保持すること。

・可用性と耐障害性

システムの可用性：システムは99.9％の可用性を維持すること。

バックアップと復元：定期的なデータバックアップと災害復旧計画を実施し、データの損失を最小限に抑えること。

・拡張性と保守性

システムの拡張性：将来的なシステムの拡張性を考慮し、新しい機能やユーザーの追加が容易に行えるアーキテクチャを採用すること。

コード品質とドキュメント：コードは適切にコメントされ、保守性が高く、新しい開発者が迅速に理解できるようにすること。

・ユーザビリティ

インターフェースの直感性：ユーザーが簡単に操作できる直感的なインターフェースを提供すること。

エラーハンドリング：エラーが発生した場合には、ユーザーに分かりやすいエラーメッセージを表示し、適切な対処方法を提供すること。

・コスト最適化

インフラコストの最適化：インフラのコストを最適化することによりサービスの持続可能性を高めること。

これらの非機能要件を満たすことで、生産部品の在庫管理システムが高品質で信頼性の高いものとなり、ビジネスの運用を円滑に支援できるようになります。

### 

### 構築作業情報

#### 構築実施日

2024年11月１日

#### 作業者

三好 順樹

増本 辰哉

向井 莉菜

愛甲 ありな

松井 聖士

岸本 敬太

福 俊也

#### 作業場所

クライアント企業様オフィスビル23F 執務エリア

#### 条件

？？？？？

### 

### App Runner構築手順

#### 命名規則

本プロジェクトでは以下のような命名規則に従って構築します。

**リソースに対する命名規則**

| **要素** | **詳細** |
| --- | --- |
| sysname | 固定値としてdroneを使用 |
| env | 固定値としてdevを使用 |
| user | (個別) I AM user名 |
| service | (個別) 対象サービス |
| 予備 | 一意にできない場合に使用 |

例. drone-dev-yourname-apprunner

**タグに対する命名規則**

| **キー** | **値** |
| --- | --- |
| Cost | drone\_IAM user名 |
| Project | (固定)infra-course-drone |
| Name | リソース名 |
| createdBy | (個別)IAM user名 |

例.Cost:yourname

#### AWS App Runnerへのデプロイと各種設定

・開発環境

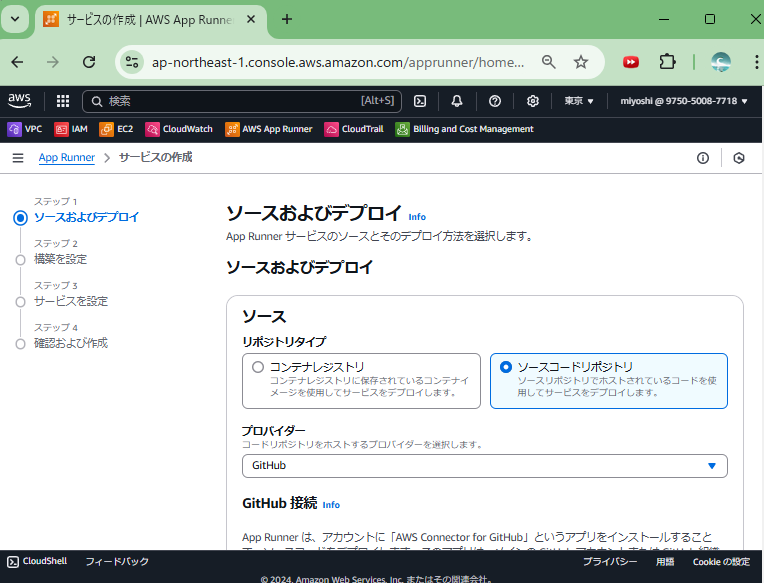
Java 11

Spring 2.7.15

・ ECS Fargate by App Runner の構築

App Runnerはフルマネージド型のコンテナアプリケーションサービスであり、簡単に ECS・Fargate で Web アプリを構築できる。 AWSの管理画面を開いて、App Runnerで「サービスの作成」ボタンを押下する。

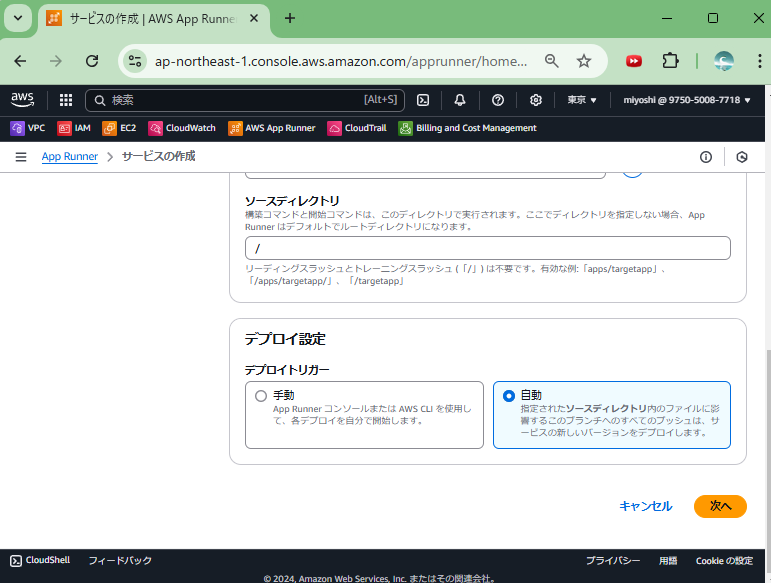
「ソースコードリポジトリ」を選択してGitHubと連携する



GitHub連携の「ソースディレクトリ」は「/dev」を指定する

デプロイ設定では「自動」を選択する





ランタイムは「Corretto 11」を選択する

AppRunner 上ではビルドコマンドとスタートコマンドを設定する。

$ mvn clean package

$ java -Xms256m -jar target/dev-0.0.1.jar



#### ロギングとモニタリングの設定

本設定で使用するツールはCloudWatch,Cloud Watch Alerm, Cloud Tail

・CloudWatchにてダッシュボードを作成

AppRunner>メトリクス>ダッシュボード追加から下記項目をダッシュボードに追加する。

システムメトリクス

CPU

Mem

ログメトリクス

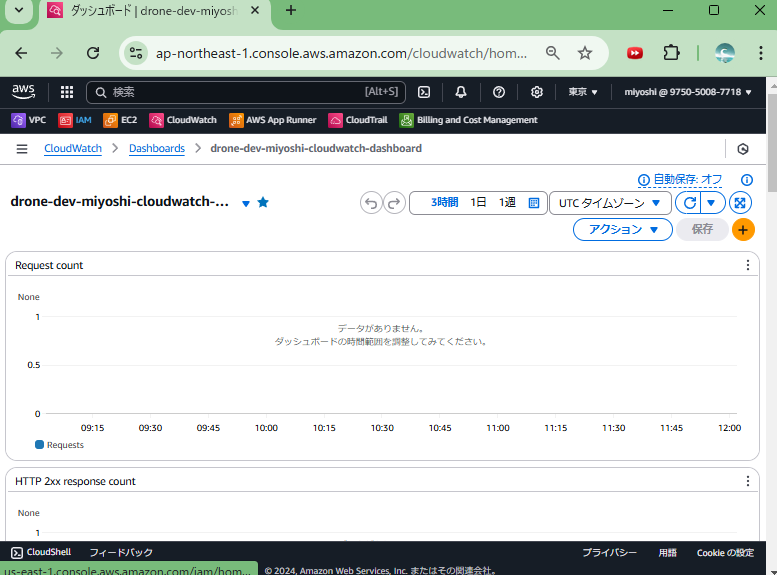
ログカウント

サービスメトリクス

HTTPリクエスト数

HTTPステータスコード別レスポンス数

レイテンシー



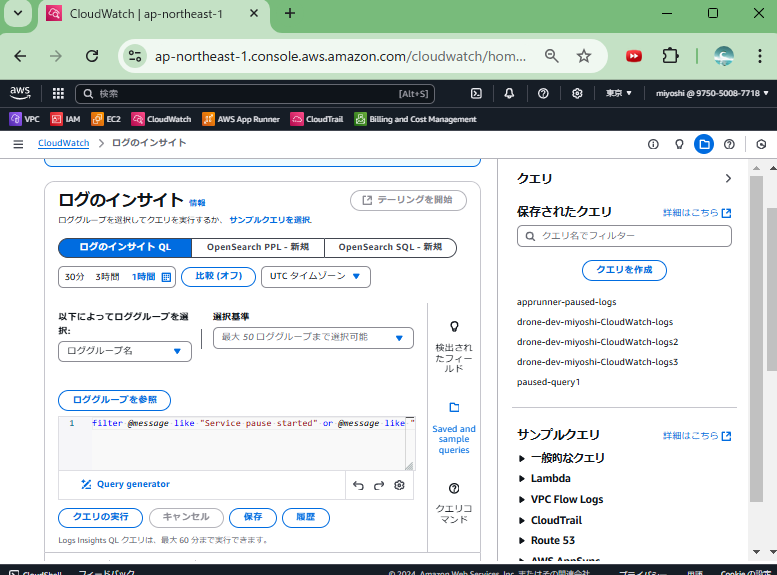
・一時停止ログの設定

一時停止した際のログを取得するためのウィジェットを追加する。

ログのインサイト画面まで遷移し、下記のログクエリを登録する。

filter @message like "Service pause started" or @message like "Service pause completed successfully"

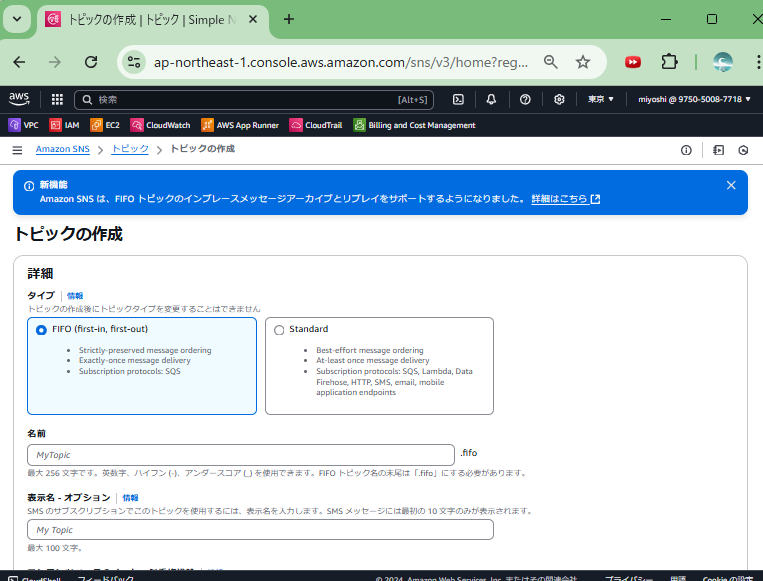




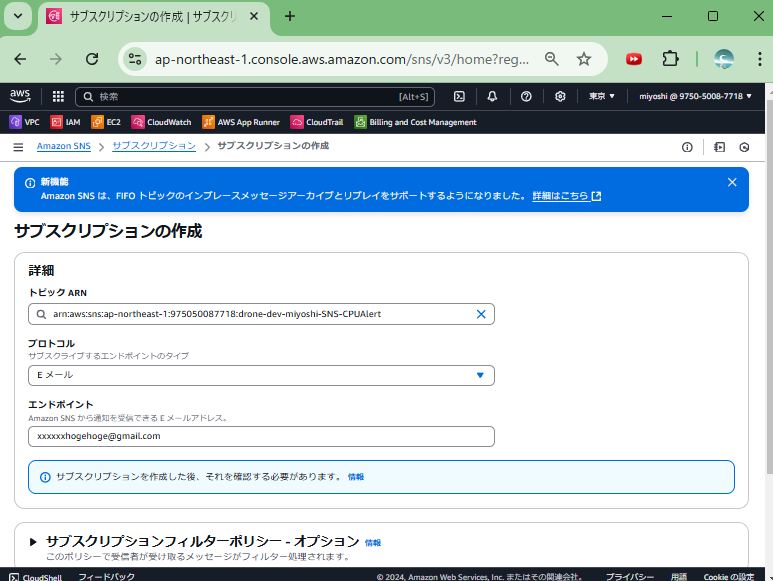
・SNSの設定

CPUを基準にしたアラームを設定するために、SNS（通知サービス）の設定を行う

SNS > トピック　よりトピックを作成する。



サブスクリプションの画面より、作成したトピックを選択し、プロトコル「Eメール」を選択、エンドポイントに通知を受信したいEメールを指定してサブスクリプションを作成する。



※設定後にメールに届いたAWS notificationのリンクをクリックしてサブスクリプションの有効化をしておく必要あり。

・Cloud Watch Alermの設定

下記の条件でアラームの設定をする。

・メトリクス名：CPUUtilization

・インスタンス：App runnerのものを選択

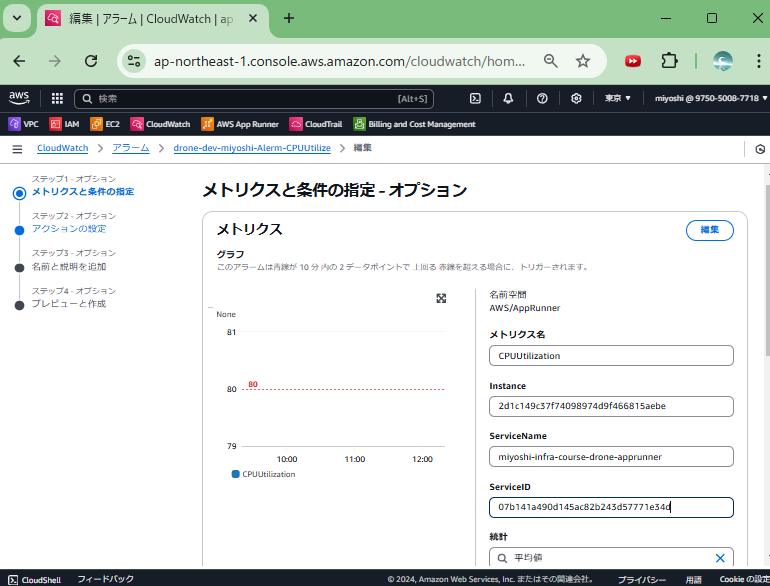
・サービス名：（命名規則に従う）

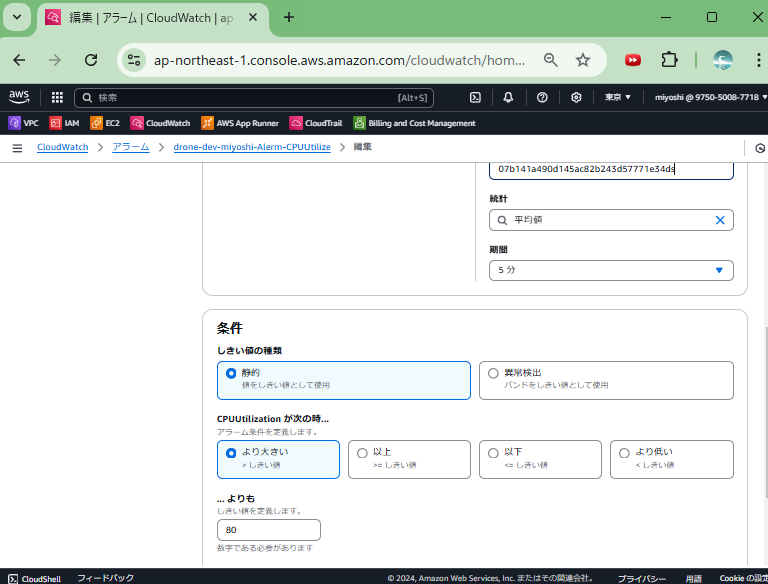
・統計：平均値

・間隔：５分

・しきい値：静的

・アラーム条件：80より大きい

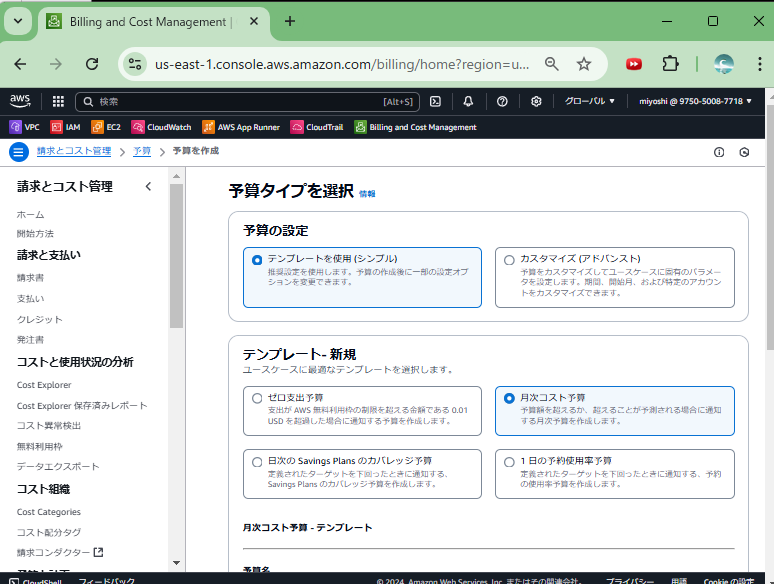




インフラコスト管理の設定

・予算の作成

予算を作成して予算超過した場合に、設定したアドレス宛にアラートが通知される。請求コストと管理＞予算＞予算作成より、月次コスト予算を選択し、予算額のしきい値を設定して作成を完了する。



### 

### 今後の予定

環境構築完了しましたので、下記の作業を実施予定です。

※詳細なスケジュールは別紙WBSをご参照ください。

・セキュリティテストの実施と報告

・運用テストの実施と報告

・「生産部品の在庫管理システム」のECSへの構築

・ECS環境の構築と設定

・開発完了 / リリース