

# 0.構成まとめ

## 構築作業の全体像と構造別まとめ

高可用性とスケーラビリティを確保した三層構造のWebアプリケーション環境

### 1. ネットワーク/負荷分散層 (Network / Load Balancing)

カテゴリ	要件	構成	実施した作業
VPC/サブネット	高可用性とセキュリティ分離。	<b>Public Subnet 2つ</b> (ALB配置用) と <b>Private Subnet 2つ</b> (EC2/DB配置用) を複数AZに分散。	VPC、サブネット、インターネットゲートウェイ、NAT Gateway、ルートテーブルの設定。
負荷分散	EC2インスタンスへのトラフィック分散と冗長化。	<b>ALB (Application Load Balancer)</b> を Public Subnetに配置。	ALBの作成、ターゲットグループの作成とEC2の登録。 <b>HTTPSリスナー</b> と <b>HTTPリダイレクト</b> の設定。
DNS/SSL	ドメイン名でのアクセスとHTTPS接続。	<b>Route 53</b> でドメインを管理し、 <b>ACM</b> でSSL証明書を取得。	ドメイン登録、ホストゾーン作成、ACM証明書取得、 <b>ALBへのAレコード登録</b> 。
監視	サービス稼働状況の監視。	<b>CloudWatch</b> でALBのメトリクスを監視。	<b>HealthyHostCount</b> 、 <b>5XX</b> 、 <b>ConnectionErrorCount</b> のアラーム設定。

### 2. アプリケーション/Web層 (Web / AP Layer)

カテゴリ	要件	構成	実施した作業
Web/APサーバー	Webアプリケーションの実行環境と冗長化。	<b>EC2 (Amazon Linux 2023)</b> 2台を Private Subnetに配置。	EC2の構築、基本的なパッケージ ( <b>git</b> , <b>php-fpm</b> など) のインストール。

ミドルウェア	PHPアプリケーション (Laravel) の実行環境。	<b>Nginx</b> (Webサーバー) と <b>PHP-FPM</b> (PHP実行環境) の連携構成。	Nginx設定 (リバースプロキシ設定)、PHP-FPM設定、連携動作確認 ( <b>Hello GEMY!</b> 表示)。
ログ/監視	APサーバーの異常検知とトラブルシューティング。	<b>CloudWatch</b> でEC2メトリクスを監視し、 <b>S3</b> にALBログを保存。	<b>CPU使用率 (&gt; 80%)</b> アラーム設定。ALBアクセスログのS3バケット作成とライフサイクルルール設定。

### 3. データベース層 (Database Layer)

カテゴリ	要件	構成	実施した作業
DBサービス	高可用性とマネージドサービスによる運用負荷軽減。	<b>RDS (MySQL)</b> をマルチAZ構成で Private Subnetに配置。	RDSインスタンスの作成、パラメータグループ設定、DBクライアントのインストール ( <b>mariadb105</b> ) と疎通確認。
監視	DBの性能ボトルネックと接続飽和の検知。	<b>CloudWatch</b> でRDSメトリクスを監視。	<b>DatabaseConnections</b> や <b>CPUUtilization</b> のアラーム設定 ( <b>DBInstanceIdentifier</b> 別)。