# **Project Documentation: AWS VPC Management API using Lambda, API Gateway, and DynamoDB**

## ****1. Project Overview****

This project deploys an AWS-based API for **VPC creation and retrieval** using **AWS Lambda, API Gateway, and DynamoDB**. The API supports:

* Creating a **VPC** with multiple subnets
* Storing VPC details in **DynamoDB**
* Retrieving VPC details using a **GET request**
* Using a **Lambda Authorizer** for authentication

## ****2. Architecture Diagram****

**Components:**

* **Lambda Functions:** Handles VPC creation and retrieval
* **DynamoDB Table:** Stores VPC details
* **API Gateway:** Exposes the Lambda functions as REST API endpoints
* **IAM Roles:** Defines permissions for Lambda
* **Lambda Authorizer:** Ensures secure access via token authentication

## ****3. Deployment Instructions****

### ****3.1 Prerequisites****

Ensure the following are installed and configured:

* AWS CLI
* AWS IAM User with permissions to create Lambda, API Gateway, and DynamoDB resources
* Terraform (if using Terraform version)
* AWS CloudFormation (if using the provided YAML template)

### ****3.2 Deploy Using AWS CloudFormation****

#### **Step 1: Navigate to AWS CloudFormation Console**

1. Go to **AWS Console** > **CloudFormation**
2. Click **Create Stack** → **With new resources**
3. Upload the provided **YAML file**
4. Click **Next** → Set stack name (e.g., VPCManagementStack)
5. Click **Next** → **Create Stack**

#### **Step 2: Wait for Deployment**

Once the stack is successfully created, go to **Outputs** in CloudFormation to find the **API Gateway Invoke URL**.

### ****3.3 Deploy Using AWS CLI****

aws cloudformation create-stack --stack-name VPCManagementStack \

--template-body file://vpc\_management.yaml \

--capabilities CAPABILITY\_NAMED\_IAM