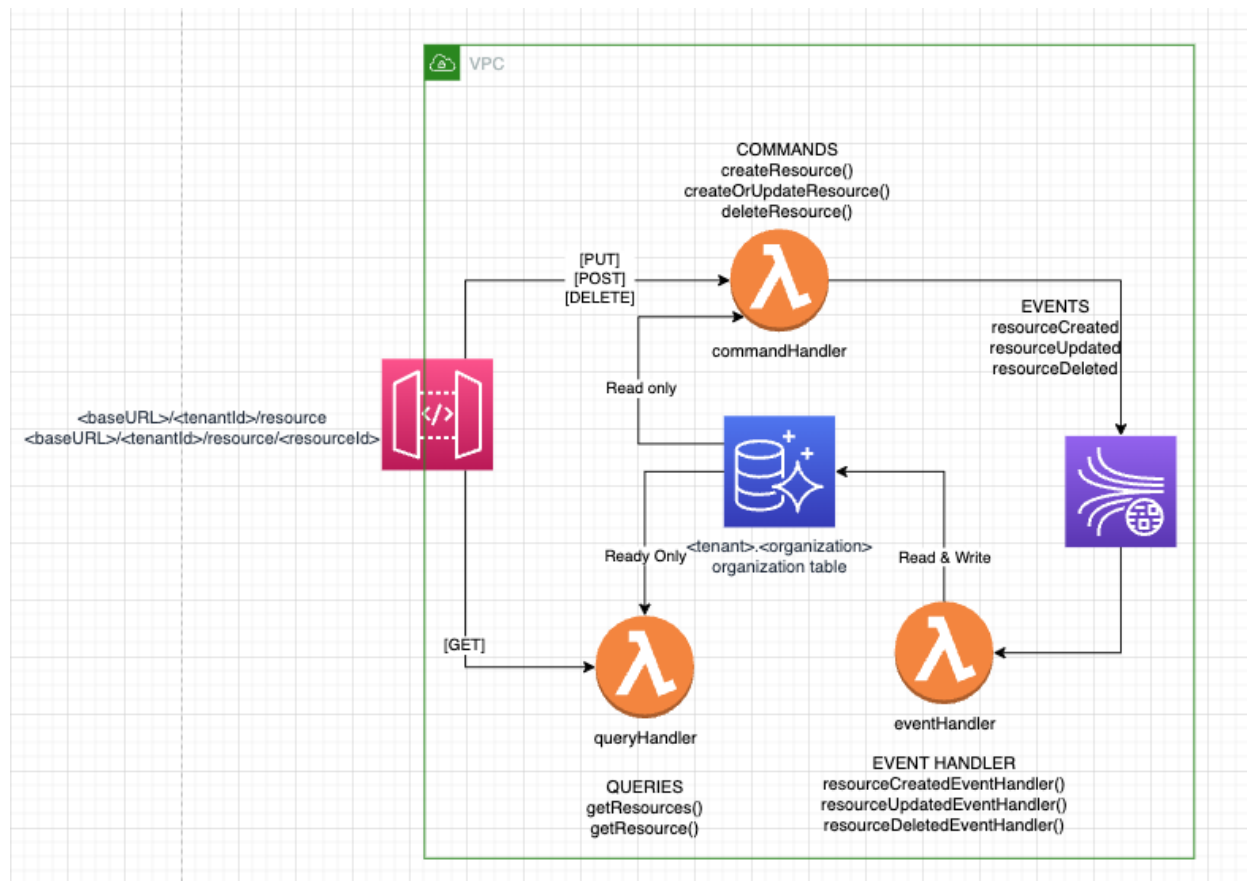


In this document, resource is called organization.

Programing language	Python preferred... Node JS is OK
Design Pattern	<a href="#"><u>CQRS - command query responsibility segregation</u></a>
Coding	<a href="#"><u>Hexagonal Design Pattern</u></a>

## High level solutions architecture



# APIs Requirements

Method	Resource	Meaning
--------	----------	---------

*POST*      *{baseUrl}/<tenantId>/resource*      Create a new record with provided parameters.

A successful api call must return:

- [200 OK - Successful](#) - {resourceId}
- [201 Created - Record created](#) - {resourceId}
- [202 Accepted - Record accepted](#), but not completed yet. - {resourceId}

*GET*      *{baseUrl}/<tenantId>/resource*      Search for a list of records based on the provided parameters.

A successful api call must return:

- [200 OK - Successful](#) - {resourceId}
- [204 No Content](#) - Successful, but no content meets the search criteria - []

*GET*      *{baseUrl}/<tenantId>/resource/{resourceId}*      Get a specific record for a provided resourceId.

A successful api call must return:

- [200 OK - Successful](#) - {resourceId}
- [204 No Content](#) - Successful, but no content meets the search criteria

*PUT*                    *{baseUrl}/<tenantId>/resource/{resourceId}*      Update the record if exists, else Create a new record with provided id

A successful api call must return:

- 200 OK - Successful - {resourceId}
- 201 Created - Record created - {resourceId}
- 202 Accepted - Record accepted, but not completed yet. - {resourceId}

*DELETE*                *{baseUrl}/<tenantId>/resource/{resourceId}*      Soft delete the record for the provided resource Id

A successful api call must return:

- 200 OK - Successful - {resourceId}
- 201 Created - Record created - {resourceId}
- 202 Accepted - Record accepted, but not completed yet. - {resourceId}

*PATCH*                *{baseUrl}/<tenantId>/resource/{resourceId}*      Update the record only if the provided id exists.

A successful api call must return:

- 200 OK - Successful - {resourceId}
- 201 Created - Record created - {resourceId}
- 202 Accepted - Record accepted, but not completed yet. - {resourceId}

# Database

Database Name: {TenantId}  
Schema Name: {ResourceName}  
Table Name: {Version}-{XXXX}

Database connection string will be provided as environment variable called "DATABASE"  
Kinesis stream name will be provided as environment variable called "STREAM"

## Domain Model

Domain Schema Attached with this project  
Event schema attached with this project

## Business Login

### Commands

#### **createResource()** :

On success: 200 OK - success : return the resourceID and generate *resourceCreated* event

#### **createOrUpdateResource()**

If resource does not exist: 200 OK - success - return new resourceID and generate *resourceCreated* event

If resource exists: 200 OK - success - return updated resourceID and generate *resourceUpdated* event

#### **deleteResource()**

On success: 200 OK - success : return the resourceID and generate *resourceDeleted* event

## Events

**resourceCreatedEvent**

- Create a new row in the resource table

#### **resourceUpdatedEvent**

- Update an existing resource at the resource id

#### **resourceDeletedEvent**

- Soft delete the existing resource

## Queries

#### **getResources()**

- Get an array of resources based on search parameters.

#### **getResource()**

- Get a resources based on ID provided in path parameter