# Roteiro para o desenvolvimento da atividade prática do DIO Live Coding do dia 17/11/2021

## Serviços AWS utilizados

- Amazon Cognito

- Amazon DynamoDB

- Amazon API Gateway

- AWS Lambda

## Etapas do desenvolvimento

### Criando uma API REST no Amazon API Gateway

- API Gateway Dashboard -> Create API -> REST API -> Build

- Protocol - REST -> Create new API -> API name [dio\_live\_api] -> Endpoint Type - Regional -> Create API

- Resources -> Actions -> Create Resource -> Resource Name [Items] -> Create Resource

### No Amazon DynamoDB

- DynamoDB Dashboard -> Tables -> Create table -> Table name [Items] -> Partition key [id] -> Create table

### No AWS Lambda

#### Função para inserir item

- Lambda Dashboard -> Create function -> Name [put\_item\_function] -> Create function

- Inserir código da função ```put\_item\_function.js``` disponível na pasta ```/src``` -> Deploy

- Configuration -> Execution role -> Abrir a Role no console do IAM

- IAM -> Roles -> Role criada no passo anterior -> Permissions -> Add inline policy

- Service - DynamoDB -> Manual actions -> add actions -> putItem

- Resources -> Add arn -> Selecionar o arn da tabela criada no DynamoDB -> Add

- Review policy -> Name [lambda\_dynamodb\_putItem\_policy] -> Create policy

### Integrando o API Gateway com o Lambda backend

- API Gateway Dashboard -> Selecionar a API criada -> Resources -> Selecionar o resource criado -> Action -> Create method - POST

- Integration type -> Lambda function -> Use Lambda Proxy Integration -> Lambda function -> Selecionar a função Lambda criada -> Save

- Actions -> Deploy API -> Deployment Stage -> New Stage [dev] -> Deploy

### No POSTMAN

- Add Request -> Method POST -> Copiar o endpoint gerado no API Gateway

- Body -> Raw -> JSON -> Adicionar o seguinte body

```

{

"id": "003",

"price": 600

}

```

- Send

### No Amazon Cognito

- Cognito Dashboard -> Manage User Pools -> Create a User Pool -> Pool name [TestPool]

- How do you want your end users to sign in? - Email address or phone number -> Next Step

- What password strength do you want to require?

- Do you want to enable Multi-Factor Authentication (MFA)? Off -> Next Step

- Do you want to customize your email verification messages? -> Verification type - Link -> Next Step

- Which app clients will have access to this user pool? -> App client name [TestClient] -> Create App Client -> Next Step

- Create Pool

- App integration -> App client settings -> Enabled Identity Providers - Cognito User Pool

- Callback URL(s) [https://example.com/logout]

- OAuth 2.0 -> Allowed OAuth Flows - Authorization code grant -Implicit grant

- Allowed OAuth Scopes - email - openid

- Save Changes

- Domain name -> Domain prefix [diolive] -> Save

### Criando um autorizador do Amazon Cognito para uma API REST no Amazon API Gateway

- API Gateway Dashboard -> Selecionar a API criada -> Authorizers -> Create New Authorizer

- Name [CognitoAuth] -> Type - Cognito -> Cognito User Pool [pool criada anteriormente] -> Token Source [Authorization]

- Resources -> selecionar o resource criado -> selecionar o método criado -> Method Request -> Authorization - Selecionar o autorizador criado

### No POSTMAN

- Add request -> Authorization

- Type - OAuth 2.0

- Callback URL [https://example.com/logout]

- Auth URL [https://diolive.auth.sa-east-1.amazoncognito.com/login]

- Client ID - obter o Client ID do Cognito em App clients

- Scope [email - openid]

- Client Authentication [Send client credentials in body]

- Get New Acces Token

- Copiar o token gerado

- Selecionar a request para inserir item criada -> Authorization -> Type - Bearer Token -> Inserir o token copiado

- Send