FIAP

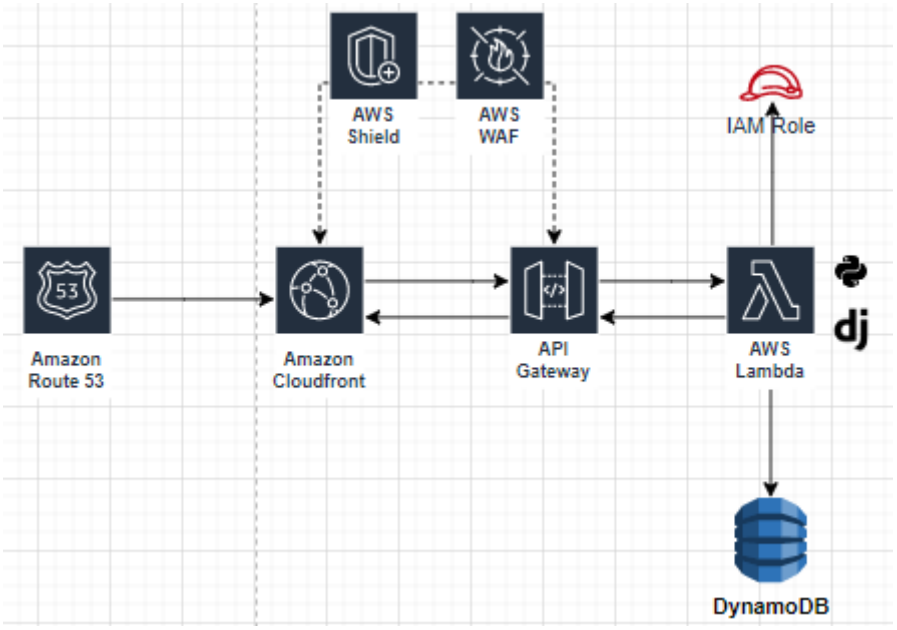
Hackathon

AWS

Grupo:
Alexandre Pires Martins
Guilherme Massaharu Hayashi de Almeida
Guilherme Viana Pereira
Matheus Lemes Tassara
Vitória Barbosa da Silva

São Paulo, 05 de maio de 2023

Neste documento, é possível encontrar toda documentação referente à implementação AWS que teve como referência o desenho de arquitetura abaixo.

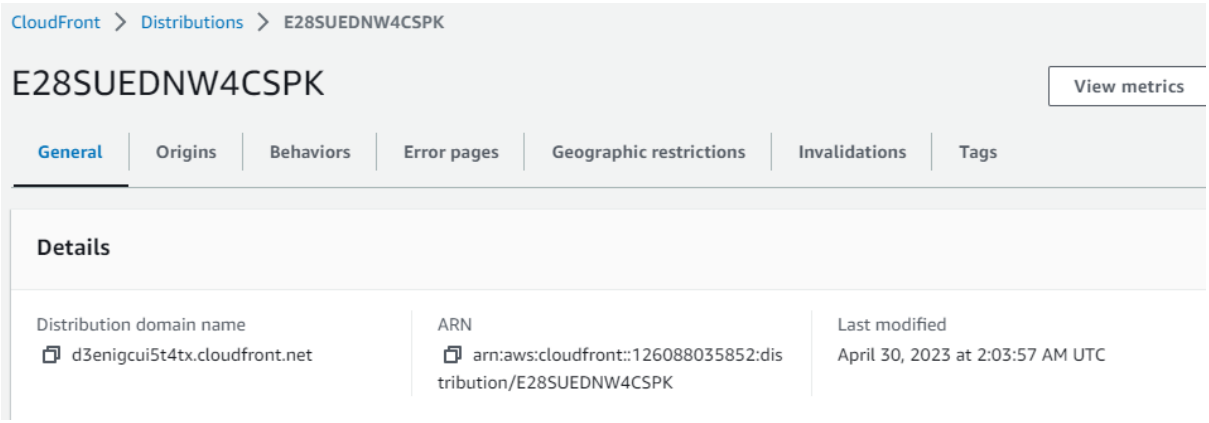


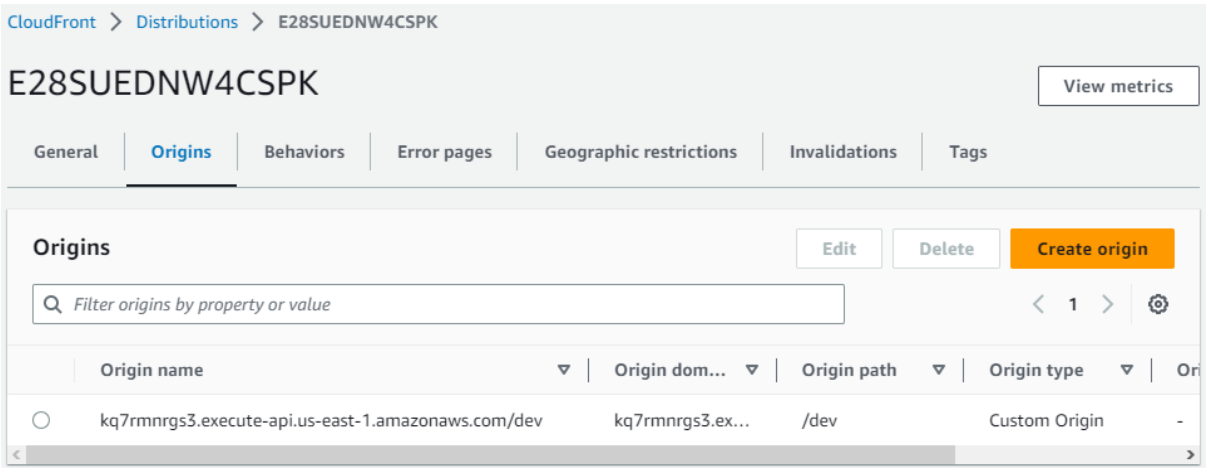
Para desenvolvimento desta arquitetura foram implementados no ambiente AWS Academy somente os principais serviços, de forma que a integração AWS Lambda x DynamoDB não pôde ser realizada tendo em vista a falta de permissão para criação/edição de IAM Roles para acesso:

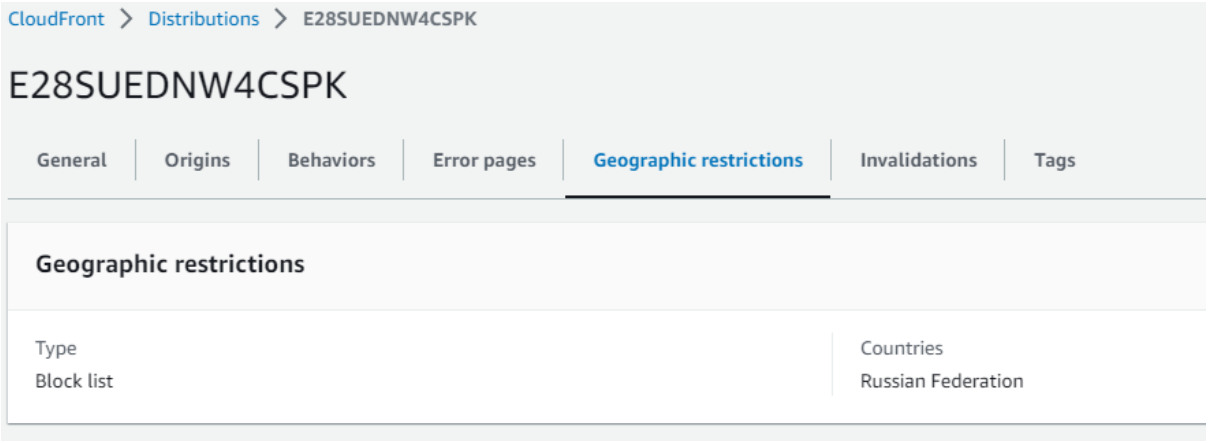
- ☑ **Amazon CloudFront**
- ☑ **API Gateway**
- ☑ **AWS Lambda**
- ☑ **DynamoDB**

Amazon CloudFront

O CloudFront foi criado com origem no API Gateway de destino, que está apontando ao path do endpoint default gerado pela AWS com a implementação do Gateway. Além disto, no CloudFront é possível fazer customizações de segurança tanto de restrição geográfica quanto a partir de regras que poderiam ser definidas pelo WAF para segurança da aplicação.



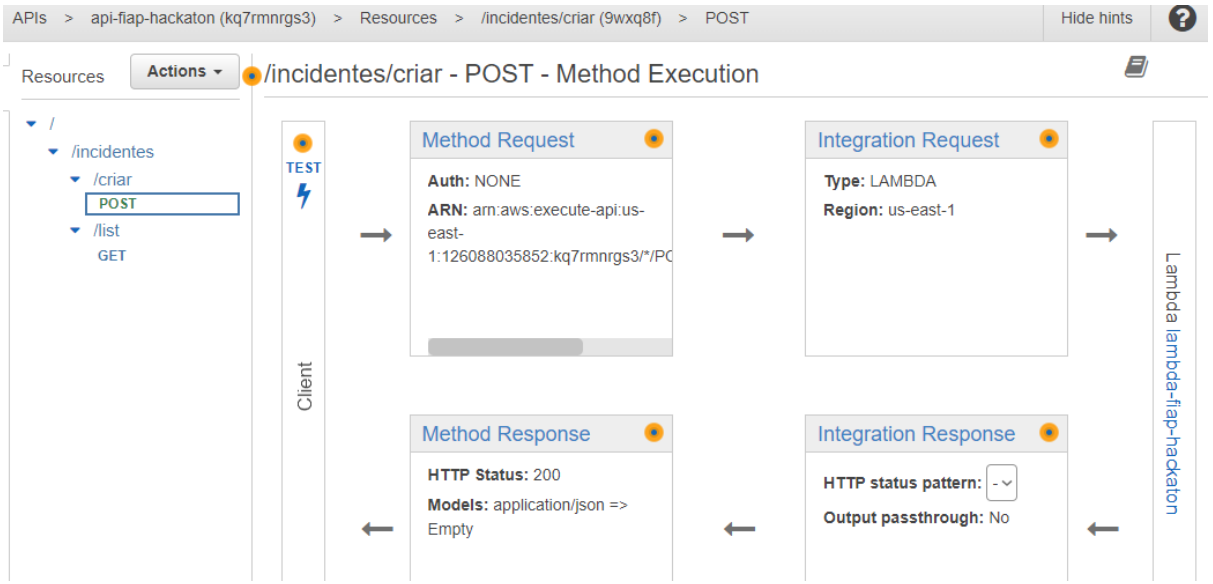




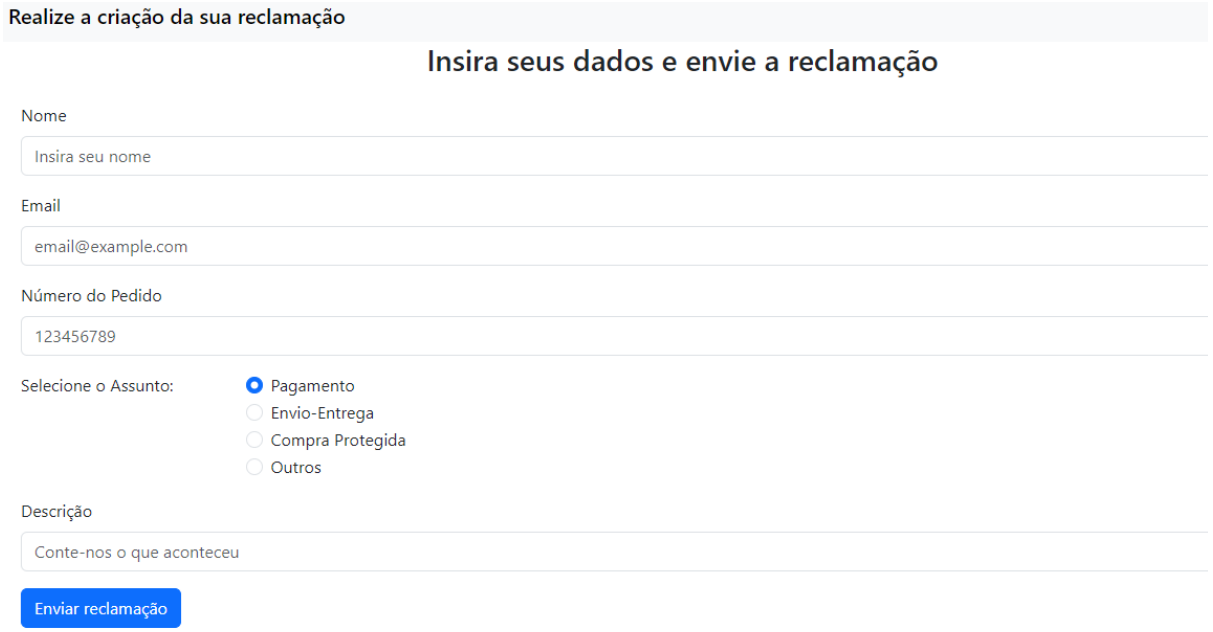
API Gateway

O API Gateway criado possui 2 rotas que executam a Lambda responsável pelo backend da aplicação:

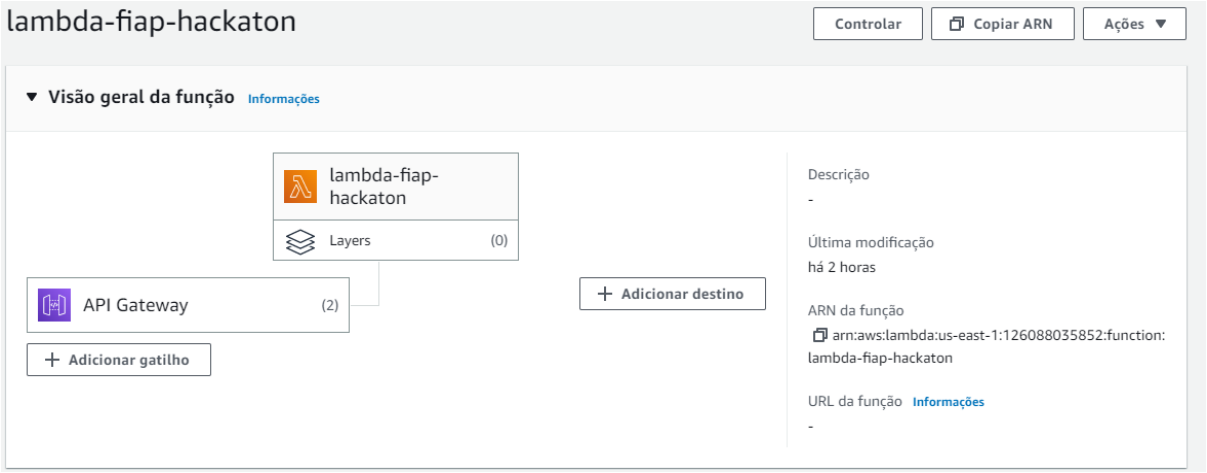
Rota	Método	Descrição
/incidentes/criar	POST	Criação de incidente
/incidentes/list	GET	Listagem de todos os incidentes criados



Acessando a rota /incidentes/criar, o seguinte formulário pode ser enviado:



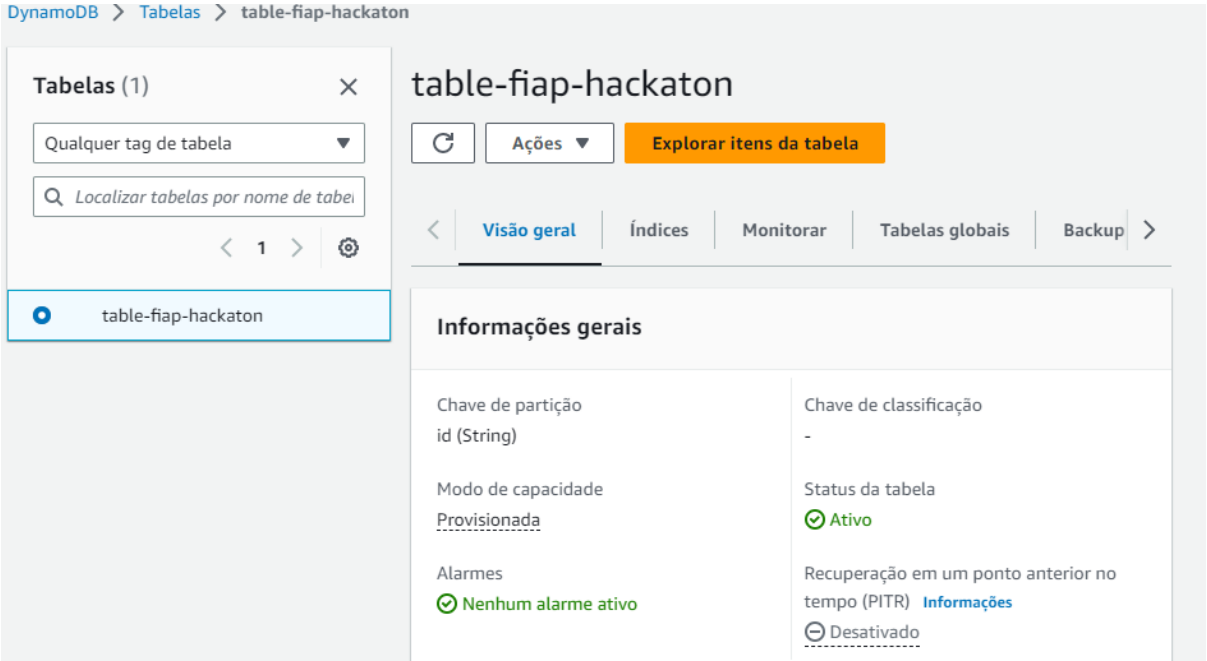
Lambda

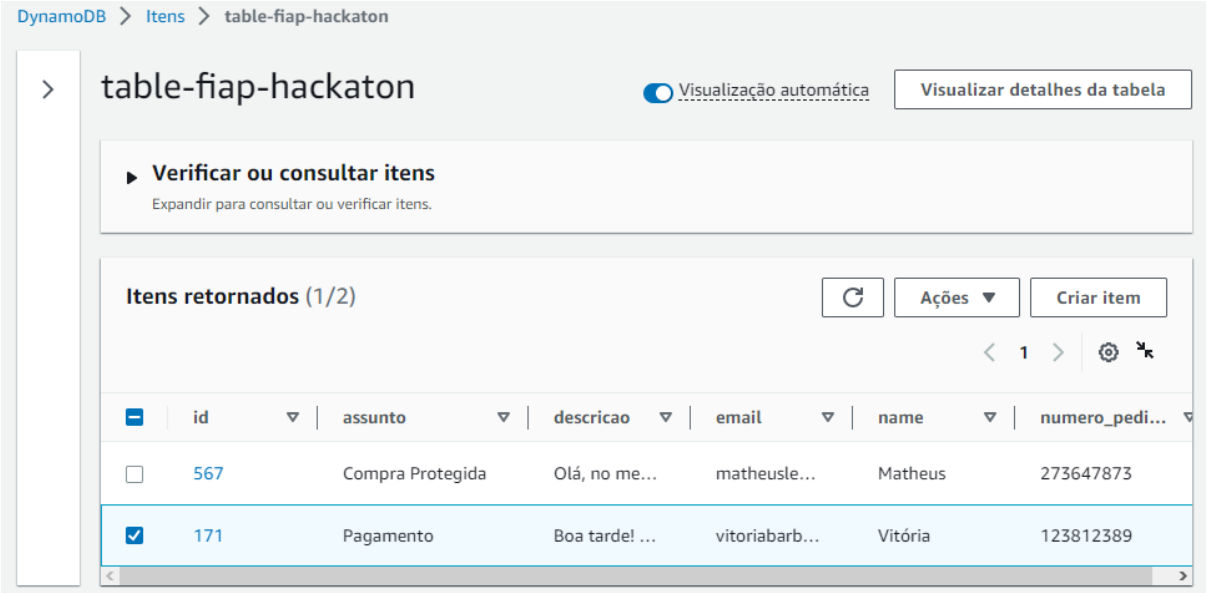


Para a Lambda, foi desenvolvido o código python com django framework conforme o repositório: https://github.com/vitoriabarbosas/pos/tree/main/hackaton-aws/app .

Dynamo

Embora a integração não pudesse ser realizada via automação tendo em vista a impossibilidade de criação de IAM Roles para acesso, foi criado uma Tabela manualmente para visualização dos dados. Para testes locais, foi utilizado o SQL dentro do repositório github.





Para que a Lambda realizasse o input de dados no DynamoDB, seria necessário a criação de uma IAM Role com a seguinte policy anexada:

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "SpecificTable",
         "Effect": "Allow",
         "Action": [
            "dynamodb:Get*",
            "dynamodb:List*",
            "dynamodb:Update*",
            "dynamodb:PutItem"
         ],
         "Resource": "arn:aws:dynamodb:*:*:table/table-fiap-hackaton"
      }
   ]
}
```

Componentes de Segurança

Para trabalharmos com uma arquitetura de segurança um pouco mais robusta, podemos enfatizar os seguintes serviços AWS:

- **CloudFront**
- **AWS Shield**
- **AWS WAF**
- **IAM Roles**

Com o CloudFront, podemos definir restrições geográficas para acesso quanto também atrelar regras de WAF para proteção da aplicação a nível de aplicação. O AWS Shield e WAF, nos permitem proteção para ataques DDoS, além de poder integrar regras WAF não só com CloudFront quanto com API Gateway. Por fim, IAM Roles com privilégio mínimo de acesso entre as aplicações.