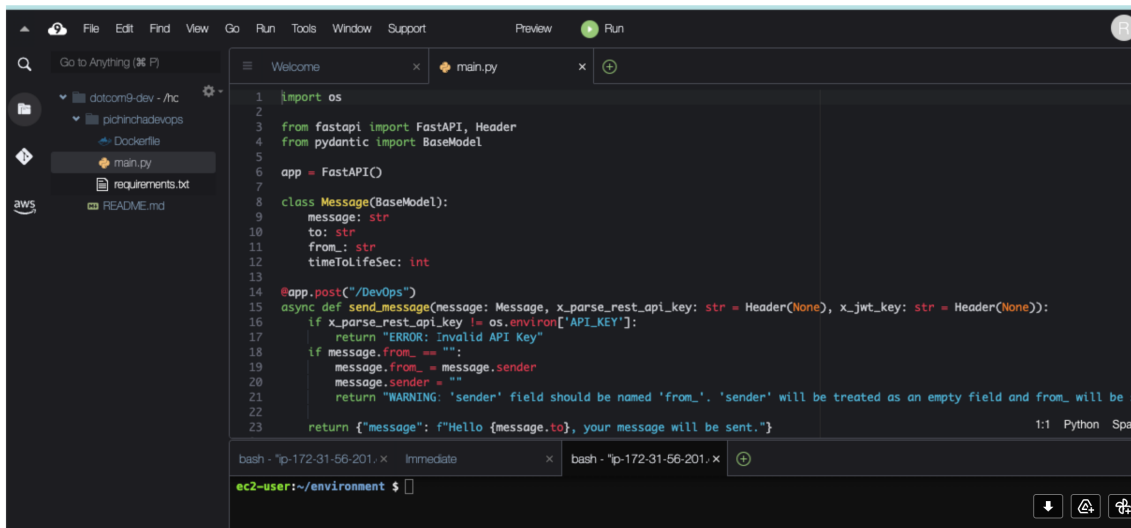


El despliegue lo hice desde mi cuenta personal de AWS



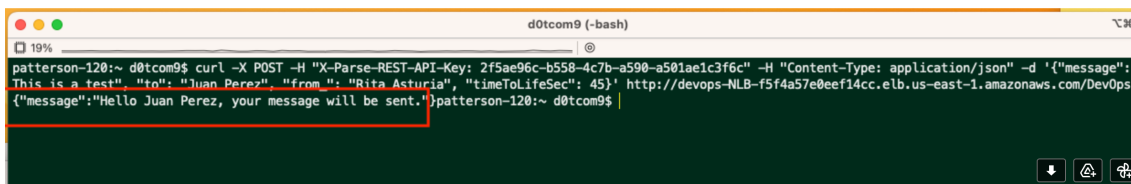
```
1 import os
2
3 from fastapi import FastAPI, Header
4 from pydantic import BaseModel
5
6 app = FastAPI()
7
8 class Message(BaseModel):
9     message: str
10    to: str
11    from_: str
12    timeToLifeSec: int
13
14 @app.post("/DevOps")
15 async def send_message(message: Message, x_parse_rest_api_key: str = Header(None), x_jwt_key: str = Header(None)):
16     if x_parse_rest_api_key != os.environ['API_KEY']:
17         return "ERROR: Invalid API Key"
18     if message.from_ == "":
19         message.from_ = message.sender
20         message.sender = ""
21         return "WARNING: 'sender' field should be named 'from_'. 'sender' will be treated as an empty field and from_ will be set."
22     return {"message": f"Hello {message.to}, your message will be sent."}
```

Endpoint: <http://devops-NLB-f5f4a57e0eef14cc.elb.us-east-1.amazonaws.com/DevOps>

Esta es la cadena de prueba

```
curl -X POST -H "X-Parse-REST-API-Key: 2f5ae96c-b558-4c7b-a590-a501ae1c3f6c" -H "Content-Type: application/json" -d '{"message": "This is a test", "to": "Juan Perez", "from_": "Rita Asturia", "timeToLifeSec": 45}' http://devops-NLB-f5f4a57e0eef14cc.elb.us-east-1.amazonaws.com/DevOps
```

Esta es la respuesta:



```
patterson-120:~ d0tcom9$ curl -X POST -H "X-Parse-REST-API-Key: 2f5ae96c-b558-4c7b-a590-a501ae1c3f6c" -H "Content-Type: application/json" -d '{"message": "This is a test", "to": "Juan Perez", "from_": "Rita Asturia", "timeToLifeSec": 45}' http://devops-NLB-f5f4a57e0eef14cc.elb.us-east-1.amazonaws.com/DevOps
{"message": "Hello Juan Perez, your message will be sent."}patterson-120:~ d0tcom9$
```

Tuve que corregir la variable from y le puse from_ ya que me salía un mensaje de variable reservada.

Repositorio de imagen docker en AWS ECR

Amazon ECR > Repositories

Private | Public

Private repositories (1) View push commands Delete Actions Create repository

<input type="checkbox"/>	Repository name ▲	URI	Created at ▼	Tag immutability	Scan frequency	Encryption type	Pull through cache
<input type="checkbox"/>	devops	615896471392.dkr.ecr.us-east-1.amazonaws.com/devops	26 de marzo de 2023, 22:06:56 (UTC-05)	Disabled	Manual	AES-256	Inactive

Creé un cluster en Amazon Web Services en Elastic Container Services (ECS) con Fargate, levanté 2 tasks (contenedores balanceados).

devops-pichincha Update cluster Delete cluster

Cluster overview

ARN devops-pichincha	Status Active	CloudWatch monitoring Container Insights	Registered container instances -
Services Draining -	Active 1	Tasks Pending -	Running 2

Services | Tasks | Infrastructure | Metrics | Scheduled tasks | Tags

Services (1) Info Manage tags Update Delete service Create

All launch types All service types

<input type="checkbox"/>	Service name ▼	Status ▼	ARN	Service type ▼	Deployments and tasks ▼	Last deploy...	Task definition ▼	Revision
<input type="checkbox"/>	service-ecs	Active	arn:aws:ecs...	REPLICA	2/2 Tasks ru...	Completed	devops-pichincha-update	1

El balanceo lo hice con NLB y no con ALB considerando que luego se pueda enlazar a un API Gateway para configurar un dominio personalizado con https

▼ Details

 `arn:aws:elasticloadbalancing:us-east-1:615896471392:loadbalancer/net/devops-NLB/f5f4a57e0eef14cc`

Load balancer type

Network

DNS name

 `devops-NLB-f5f4a57e0eef14cc.elb.us-east-1.amazonaws.com`
(A Record)

IP address type

IPv4

Scheme

Internet-facing

Date created

Estoy revisando un curso de Terraform para poder desplegar toda esta infraestructura como código.