

## ☁ UD07 · Terraform + LocalStack

Cheat Sheet de simulación de AWS en local usando LocalStack y Terraform. Incluye teoría y caso práctico completo.

### 💡 1 · ¿Qué es LocalStack?

- 💡 Emula servicios AWS como S3, Lambda, DynamoDB... en local
- 🚫 Sin costes ni necesidad de cuenta AWS
- 💻 Compatible con AWS CLI, SDKs y Terraform
- 🔗 Endpoint universal: <http://localhost:4566>

### ✖ 2 · Servicios AWS emulados

- **Almacenamiento:** S3, EBS, SQS, SNS
- **Base de datos:** DynamoDB, RDS
- **Cómputo:** Lambda, EC2, ECS
- **Infraestructura:** CloudFormation, IAM, CloudWatch
- **Otros:** Secrets Manager, API Gateway, Step Functions

### ⬇ 3 · Instalación

#### [-] Requisitos

- Python 3.8+
- Docker
- Terraform

#### ⬇ Instalar LocalStack

```
pip install localstack  
pip install localstack-client
```

#### ⌚ Arrancar LocalStack

```
localstack start -d
```

#### ⌚ Verificar servicios

```
curl http://localhost:4566/health
```

### 🔗 4 · Terraform + LocalStack

Configura el provider AWS para que apunte a LocalStack:

```
provider "aws" {  
  region = "us-east-1"  
  access_key = "test"  
  secret_key = "test"  
  skip_credentials_validation = true  
  skip_metadata_api_check = true  
  skip_requesting_account_id = true  
  s3_use_path_style = true  
  
  endpoints {  
    s3 = "http://localhost:4566"  
  }  
}
```

### 🚀 5 · Caso práctico

#### 🕒 Estructura

```
UD07-CasoPractico01/  
|--- main.tf
```

```
|-- variables.tf  
└-- outputs.tf
```

## Recursos simulados

- Bucket S3
- Archivo `hola.txt` con contenido simulado

## main.tf

```
resource "aws_s3_bucket" "demo_bucket" {  
  bucket = var.bucket_name  
}  
  
resource "aws_s3_object" "demo_file" {  
  bucket = aws_s3_bucket.demo_bucket.id  
  key    = "hola.txt"  
  content = "¡Hola desde Terraform con LocalStack!"  
}
```

## variables.tf

```
variable "bucket_name" {  
  default = "mi-bucket-localstack"  
}
```

## outputs.tf

```
output "archivo_subido" {  
  value = aws_s3_object.demo_file.key  
}  
  
output "endpoint_localstack" {  
  value = "http://localhost:4566"  
}
```

## 6 · Comandos clave

- `terraform init` → Inicializa el proyecto
- `terraform apply` → Aplica la infraestructura
- `aws --endpoint-url=http://localhost:4566 s3 ls` → Lista buckets
- `aws --endpoint-url=http://localhost:4566 s3 cp s3://mi-bucket-localstack/hola.txt -` → Muestra archivo
- `terraform destroy` → Limpia recursos
- `localstack stop` → Detiene servicios

## 7 · Beneficios de usar LocalStack

- Pruebas sin riesgo
- Cero costes AWS
- Laboratorios educativos
- CI/CD offline o con mínimo impacto
- 100% compatible con Terraform IaC