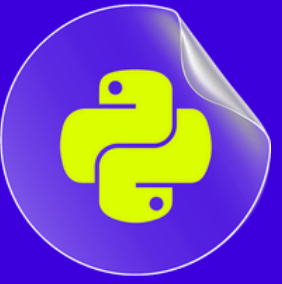


PYCON



COLOMBIA

2025

CLEAN ARCHITECTURE

with



FastAPI

SERGIO INFANTE



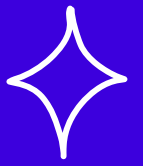


Who am I?

Sergio Infante

Software Developer

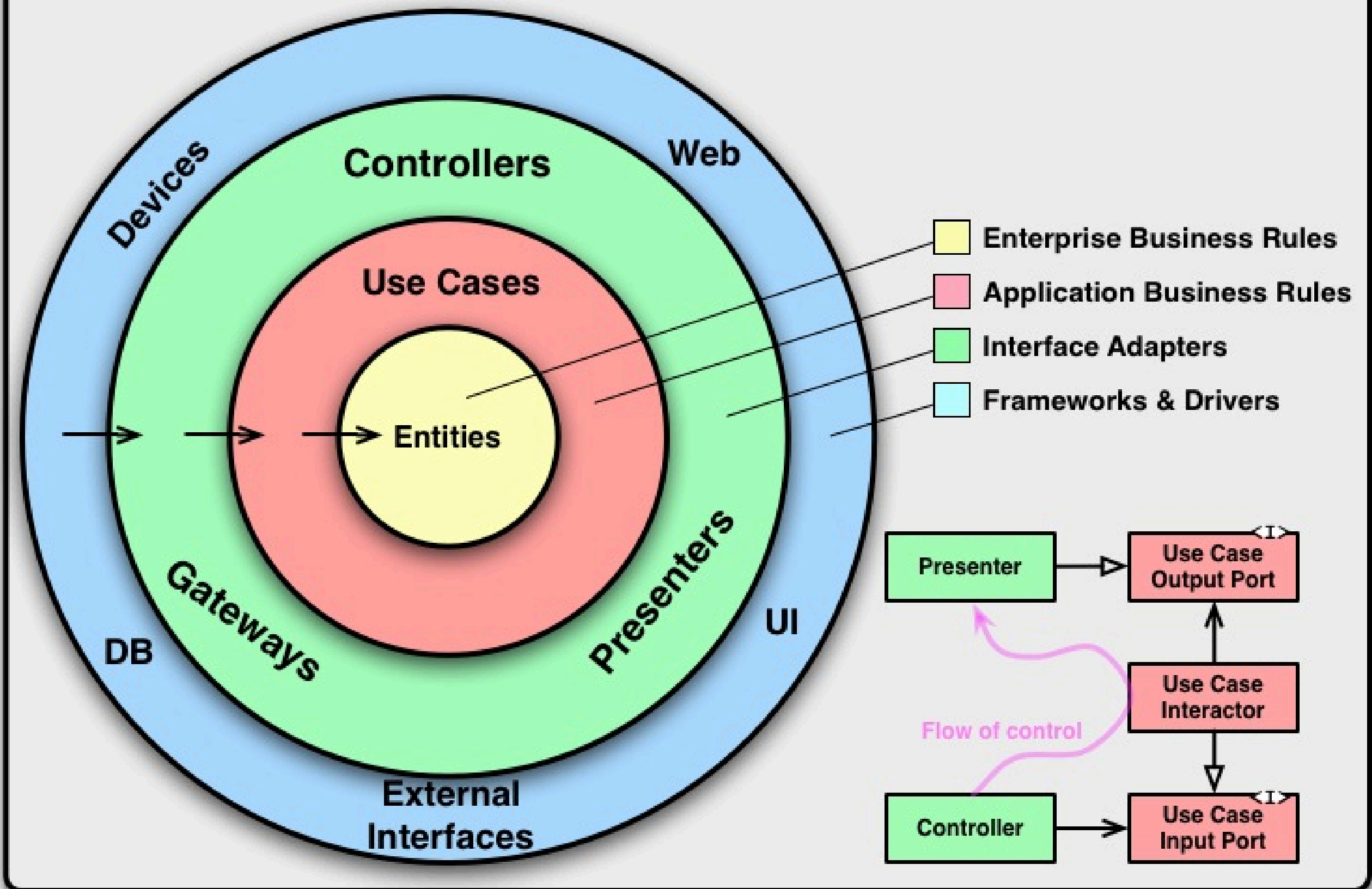
Remote. US companies



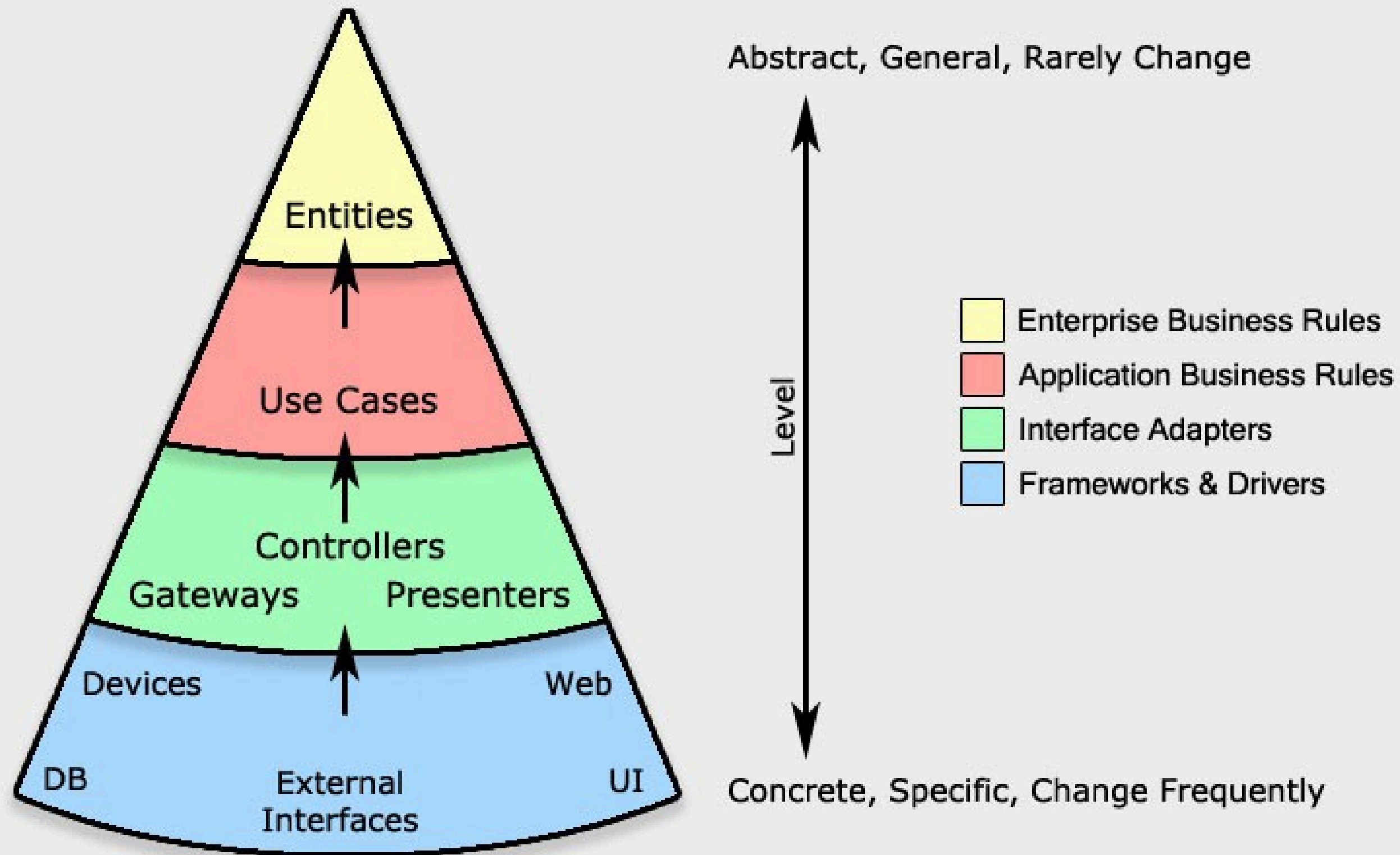
CLEAN ARCHITECTURE



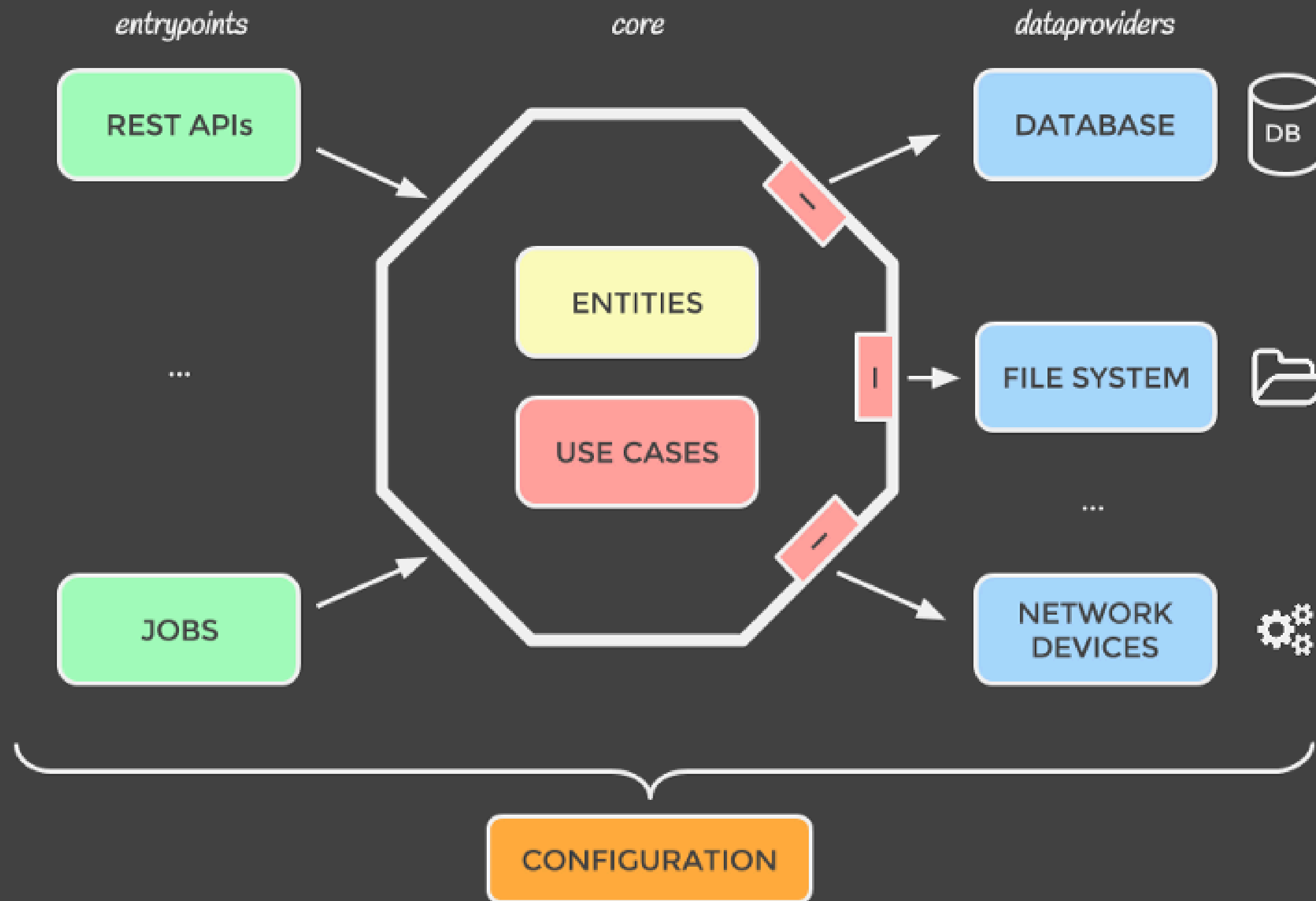
The Clean Architecture



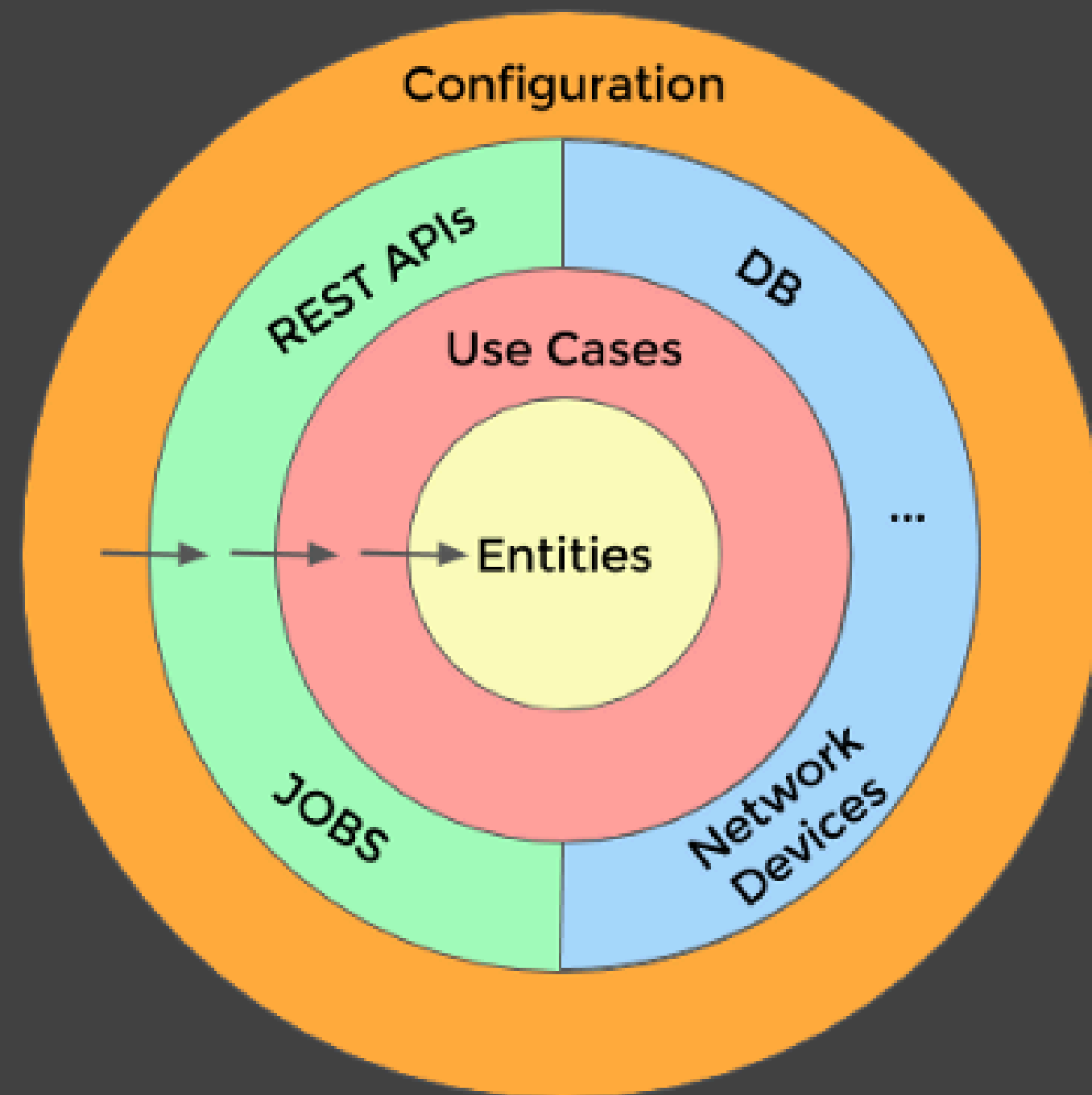
The Clean Architecture Cone



OUR CLEAN ARCHITECTURE



A.k.a (unclebob's style)



■ + ■ Core: business logic

■ Entrypoints: access to the application

■ Data Providers: retrieve and store information

■ Configuration: wires everything together

** arrows represent dependency*

SEPARACIÓN DE RESPONSABILIDADES



```
|— api/      # Rutas y controladores
|— core/    # Configuraciones y seguridad
|— models/  # Modelos de datos
|— repositories/ # Acceso a datos
|— services/ # Lógica de negocio
|— schemas/  # Esquemas de validación
|— db/      # Conexión a la base de datos
```

INDEPENDENCIA DE FRAMEWORKS



negocio



servicios



repositorios

 **FastAPI**

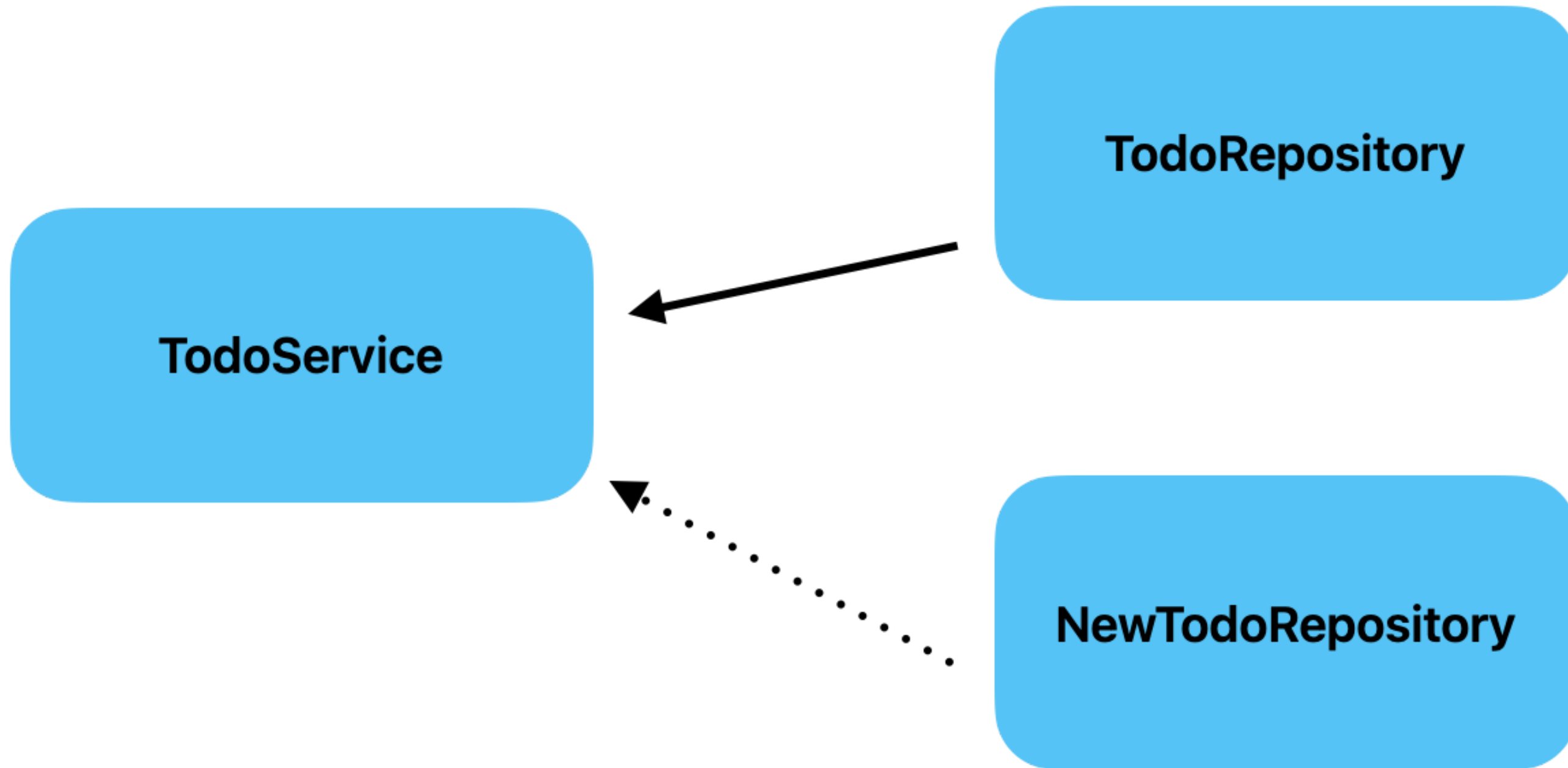

DynamoDB

TESTABILIDAD



```
def test_create_todo():  
    response = client.post(  
        "/api/v1/todos/",  
        json={  
            "title": "Comprar leche",  
            "description": "Ir al supermercado"})  
    assert response.status_code == 200  
    assert response.json()["title"] == "Comprar leche"
```

DESACOPLAMIENTO





**CASO:
TODO LIST**



```
|— api/
|   |— v1/
|   |   |— todo.py           # Rutas de la API
|— core/
|   |— config.py           # Configuración
|— models/
|   |— todo.py             # Modelos de datos
|— repositories/
|   |— todo_repository.py  # Acceso a datos
|— services/
|   |— todo_service.py    # Lógica de negocio
|— schemas/
|   |— todo.py             # Esquemas de validación
|— db/
|   |— session.py         # Conexión a la base de datos
```



```
from fastapi import APIRouter, HTTPException
from app.schemas.todo import Todo, TodoCreate, TodoUpdate
from app.services import todo_service

router = APIRouter()

@router.post("/", response_model=Todo)
def create_todo(todo_in: TodoCreate):
    todo = todo_service.create_todo(todo_in)
    if not todo:
        raise HTTPException(status_code=400, detail="Todo could not be created")
    return todo
```



```
from app.repositories import todo_repository
from app.schemas.todo import TodoCreate, TodoUpdate

def create_todo(todo: TodoCreate):
    return todo_repository.create_todo(todo)

def get_todo_by_id(todo_id: str):
    return todo_repository.get_todo_by_id(todo_id)

def update_todo(todo_id: str, todo: TodoUpdate):
    return todo_repository.update_todo(todo_id, todo)

def delete_todo_by_id(todo_id: str):
    return todo_repository.delete_todo_by_id(todo_id)
```



**CASO:
TODO LIST**



```
.
├── app/
│   ├── __init__.py
│   ├── main.py
│   ├── core/
│   │   ├── __init__.py
│   │   └── config.py
│   ├── models/
│   │   ├── __init__.py
│   │   └── todo.py
│   ├── repositories/
│   │   ├── __init__.py
│   │   └── todo_repository.py
│   ├── schemas/
│   │   ├── __init__.py
│   │   └── todo.py
│   ├── services/
│   │   ├── __init__.py
│   │   └── todo_service.py
│   └── api/
│       ├── __init__.py
│       ├── deps.py
│       └── v1/
│           ├── __init__.py
│           └── todo.py
├── db/
│   ├── __init__.py
│   └── session.py
├── tests/
│   ├── __init__.py
│   └── test_todo.py
├── .env
└── requirements.txt
```



ADICIONALMENTE...

INFRAESTRUCTURA

Archivos YML, IaC

ADAPTERS

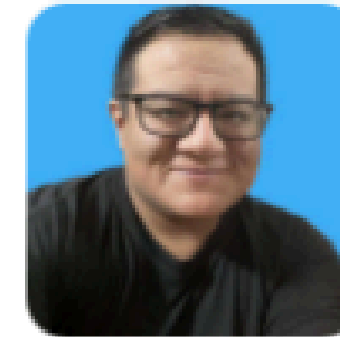
Integración con mas
servicios

TESTS

Otro tipo de tests



neosergio/fastapi-clean



A scalable To Do List application built with FastAPI, DynamoDB, and AWS CDK. Demonstrates a clean architecture approach, integration with...

 1 Contributor  0 Issues  0 Stars  0 Forks



neosergio/fastapi-clean: A scalable To Do List application built with FastAPI, DynamoDB, and AWS CDK....

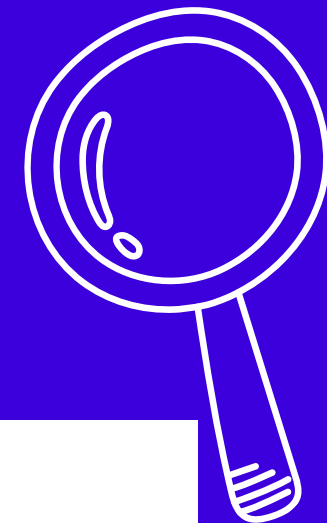
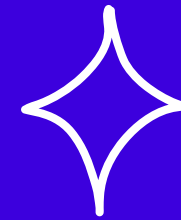
A scalable To Do List application built with FastAPI, DynamoDB, and AWS CDK. Demonstrates a clean architecture approach, integration with AWS services, and unit testing with moto. - neosergio/fasta...

 GitHub

<https://github.com/neosergio/fastapi-clean>



¿PREGUNTAS?



CLEAN
ARCHITECTURE

with

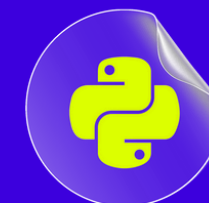


FastAPI



PYCON

COLOMBIA



2025

