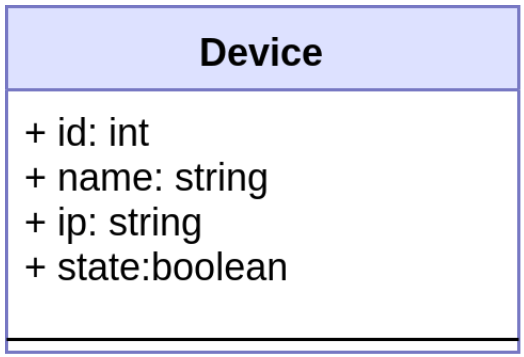
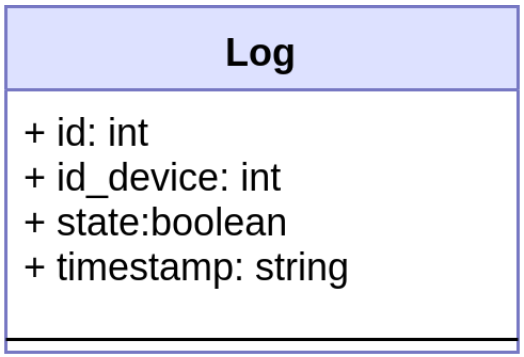
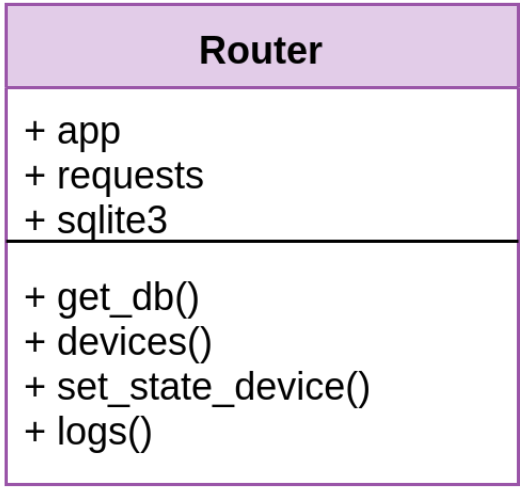
Análisis de los ejercicios

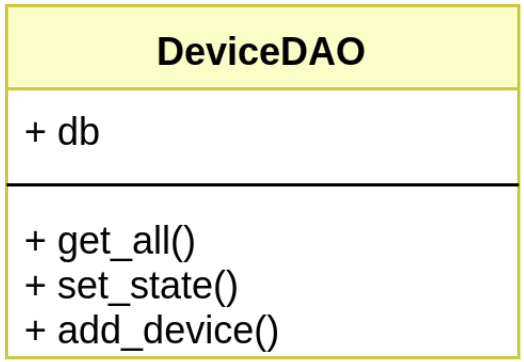
Autor: Esp. Ing. Ernesto Gigliotti. UTN-FRA

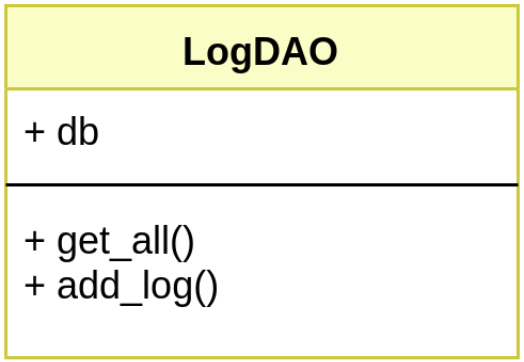
Diagrama de Clases de ejercicio 2 y 3

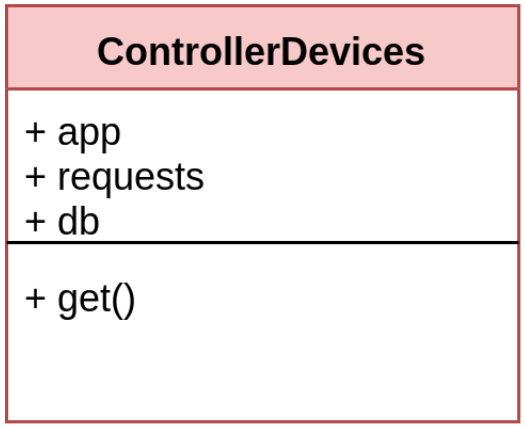

Device
+ id: int
+ name: string
+ ip: string
+ state:boolean

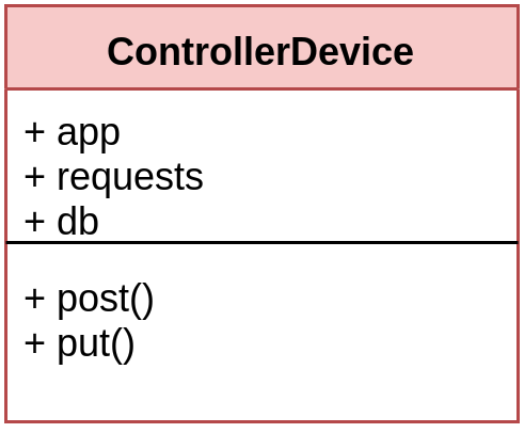

Log
+ id: int
+ id_device: int
+ state:boolean
+ timestamp: string

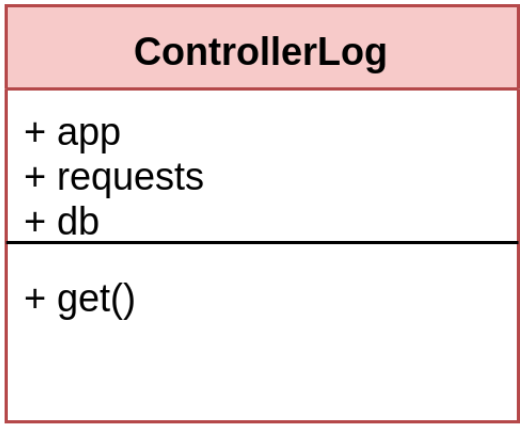

Router
+ app
+ requests
+ sqlite3
+ get_db()
+ devices()
+ set_state_device()
+ logs()

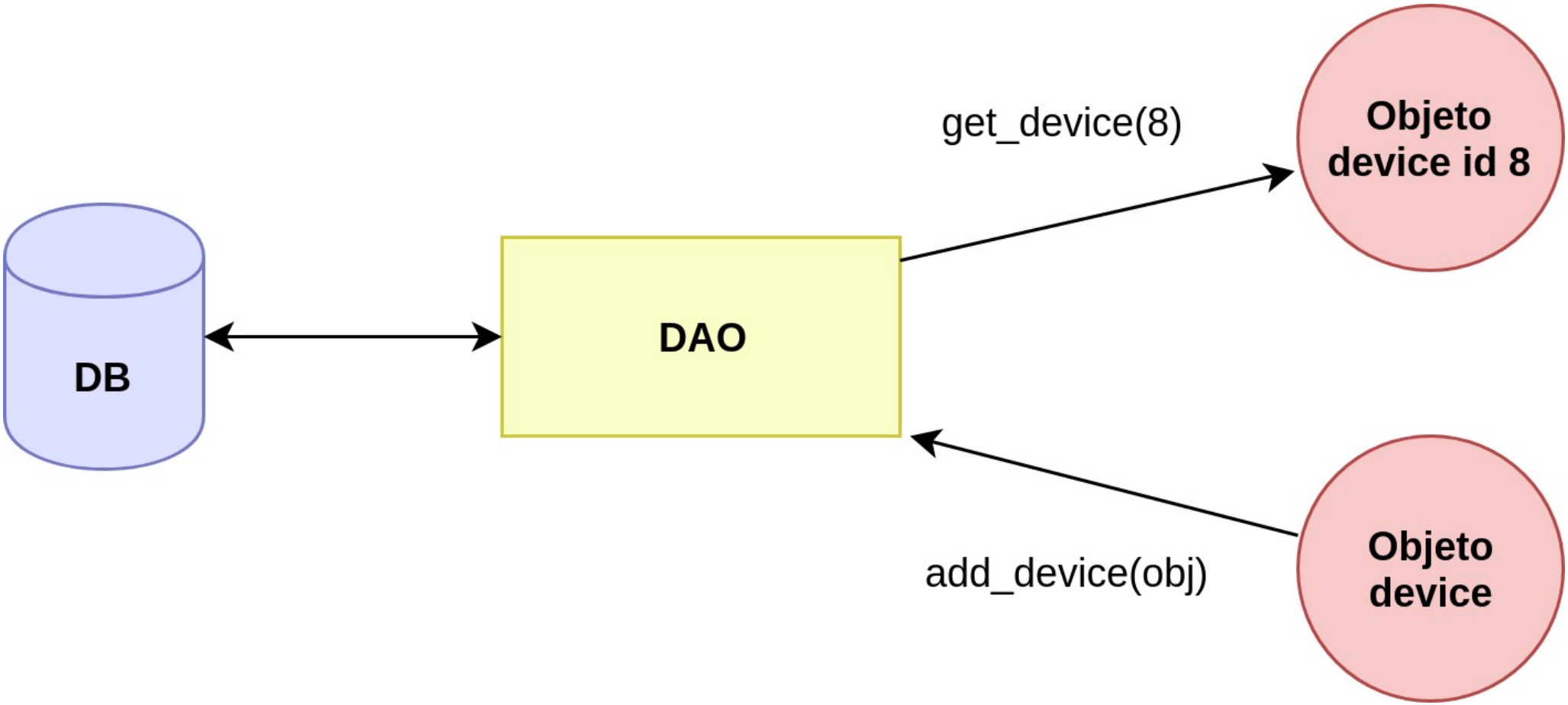

DeviceDAO
+ db
+ get_all()
+ set_state()
+ add_device()


LogDAO
+ db
+ get_all()
+ add_log()


ControllerDevices
+ app
+ requests
+ db
+ get()


ControllerDevice
+ app
+ requests
+ db
+ post()
+ put()


ControllerLog
+ app
+ requests
+ db
+ get()


get_device(8)
Objeto device id 8
DB
DAO
add_device(obj)
Objeto device

Estructura del ws: Router

```
app = Flask(__name__)

@app.route('/devices',methods=['GET'])
def devices():
    cont = ControllerDevices(app,request,get_db())
    return cont.get()
```

Estructura del ws: Controller

```
def __init__(self,app,request,db):
    self.app = app
    self.request = request
    self.db = db

def get(self):
        #devuelvo lista de devices
        dev_dao = DeviceDAO(self.db)

        response = self.app.response_class(
            response=json.dumps(dev_dao.get_all()),
            status=200,
            mimetype='application/json'
        )

        return response
```

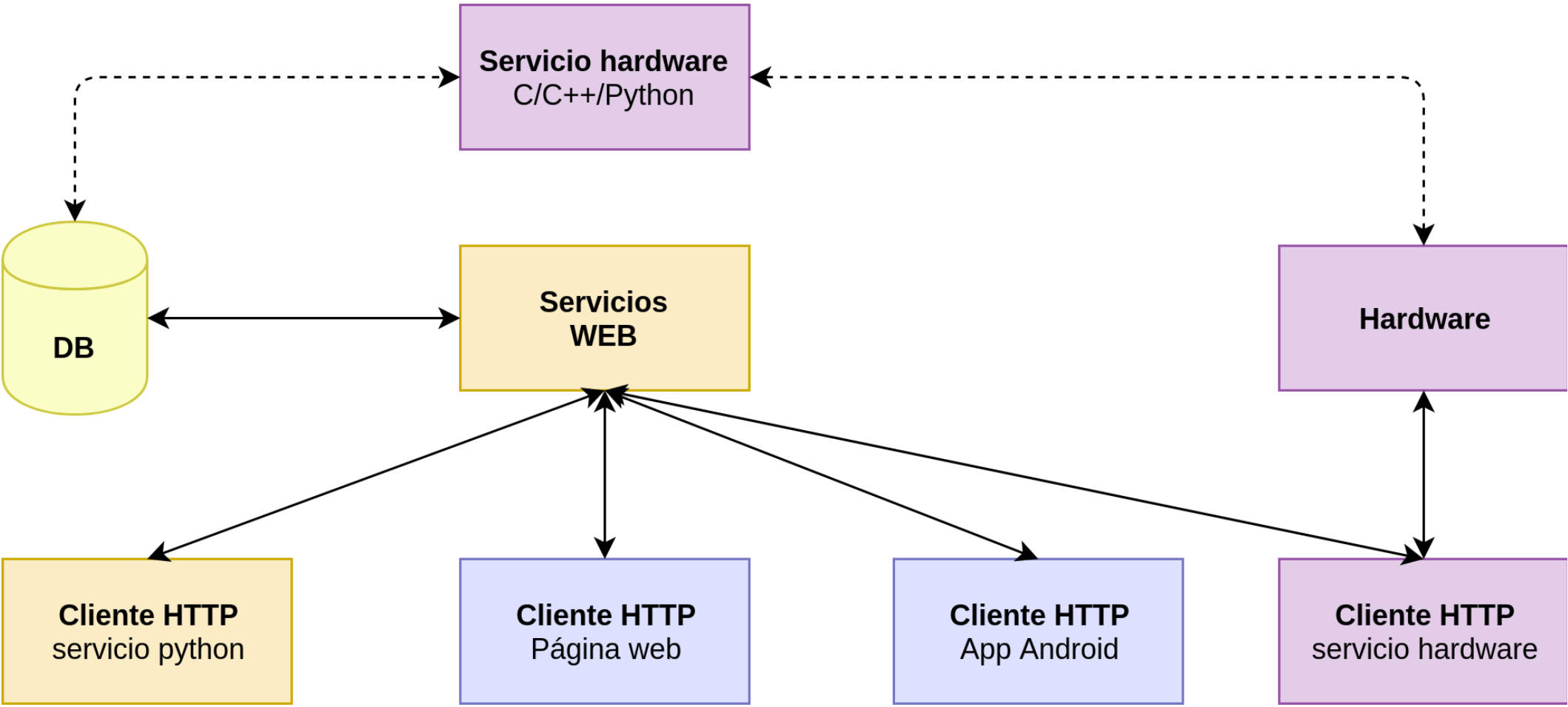
Estructura del ws: DeviceDAO

```
class DeviceDAO:

    def __init__(self,db):
        self.db = db
        self.c = self.db.cursor()

    def get_all(self):
        self.c.execute("SELECT * FROM devices")
        return self.c.fetchall()
```

Estructura del sistema. Ejemplo 1



Estructura del sistema. Ejemplo 2