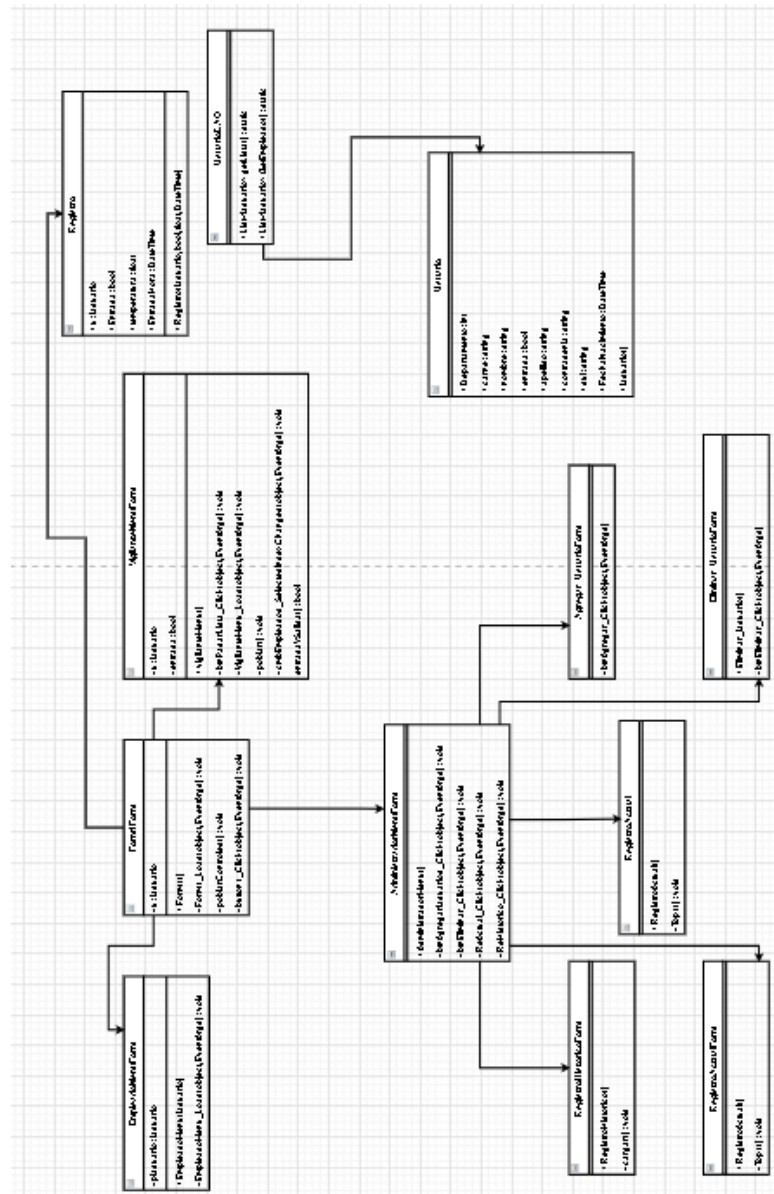


DIAGRAMA UML



Acceso directo al UML de clases:

https://app.diagrams.net/#G1Jin2VseTp9gOZYyM7ntqCzq0tH1sv_tK

PATRON DE DISEÑO DATA ACCES OBJECT (DAO)

Hemos ocupado el patrón de diseño Data Acces Object. Se puede decir en pocas palabras que su funcionalidad es para almacenar y extraer datos de diferentes tipos de fuentes. La solución de este patrón de diseño es tener una aplicación que no está ligada al acceso de datos, y una de las ventajas de este patrón de diseño es cualquier objeto de negocio o aquel objeto que contenga detalles específicos de operación o aplicación no requerirá conocimiento directo del destino final de la información que manipula.

Hemos implementado este patrón de diseño en nuestra clase Usuario el cual le hemos creado una clase por aparte llamada "UsuarioDAO", le estaremos almacenando los datos principales de cada usuario (empleado, administrador y vigilante), teniendo en cuenta que estaremos utilizando las funciones de la conexión a la base de datos y manipulando estos datos sin necesidad de saber de dónde provienen. Extrayendo los datos y almacenando los datos en nuestras tablas de la base de datos. También hemos tomado en cuenta que: los empleados pertenecen al departamento 1, vigilantes pertenecen al departamento 2, Administradores pertenecen al departamento 3.

SCRIPTS DE LA BASE DE DATOS

```
CREATE TABLE DEPARTAMENTO(  
    idDepartamento SERIAL PRIMARY KEY,  
    nombre VARCHAR(50) NOT NULL,  
    ubicacion VARCHAR(100) NOT NULL  
);  
  
CREATE TABLE USUARIO(  
    idUsuario VARCHAR(8) PRIMARY KEY,  
    idDepartamento INTEGER NOT NULL,  
    contrasenia VARCHAR(50) NOT NULL,  
    nombre VARCHAR(100) NOT NULL,  
    apellido VARCHAR(50) NOT NULL,  
    dui VARCHAR(10) NOT NULL,  
    fechaNacimiento Date NOT NULL  
);
```

```
CREATE TABLE REGISTRO(
```

```
    idRegistro SERIAL PRIMARY KEY,
```

```
    idUsuario VARCHAR(8) NOT NULL,
```

```
    entrada BOOL NOT NULL,
```

```
    FechaHora DATE NOT NULL,
```

```
    Temperatura FLOAT NOT NULL
```

```
);
```

```
ALTER TABLE USUARIO
```

```
ADD CONSTRAINT fk_idDepartamento
```

```
FOREIGN KEY (idDepartamento) REFERENCES DEPARTAMENTO(idDepartamento);
```

```
ALTER TABLE REGISTRO
```

```
ADD CONSTRAINT fk_idUsuario
```

```
FOREIGN KEY (idUsuario) REFERENCES USUARIO(idUsuario);
```

```
INSERT INTO DEPARTAMENTO(nombre,ubicacion) VALUES('General','San Salvador')
```

```
INSERT INTO DEPARTAMENTO(nombre,ubicacion) VALUES('Vigilancia','San Salvador')
```

```
INSERT INTO DEPARTAMENTO(nombre,ubicacion) VALUES('Administrativo','San Salvador')
```