

PROGRAMACIÓN II

Trabajo Práctico 5: Relaciones UML 1 a 1

Alumna: Melisa Martellini

Link repositorio: https://github.com/Melisa-Ines-73/Programaci-n_2/tree/main/5-UML

Ej 1:

Análisis de las relaciones:

- a. Composición: Pasaporte → Foto

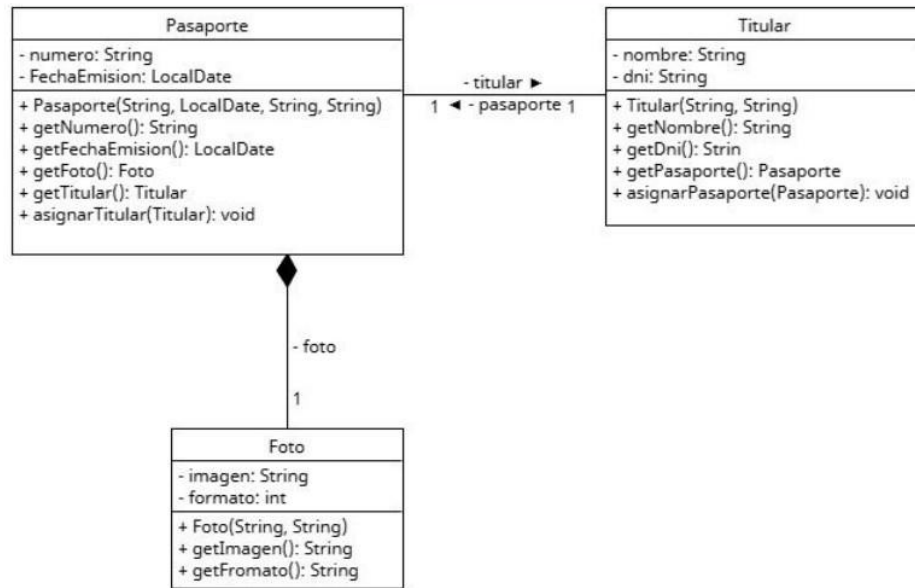
Una Foto es una parte integral e inseparable de un Pasaporte. Si se destruye el pasaporte, la foto deja de tener sentido. El ciclo de vida de la Foto está totalmente ligado al del Pasaporte. Es una relación de todo-parte muy fuerte.

Dirección: Unidireccional. El Pasaporte conoce a su Foto, pero la Foto no necesita conocer al Pasaporte.

- b. Asociación bidireccional: Pasaporte ↔ Titular

Un Pasaporte pertenece a un Titular, y un Titular tiene un Pasaporte. Es una relación entre dos objetos independientes. El titular existe sin pasaporte y viceversa.

Bidireccional: Es útil poder navegar en ambas direcciones: desde el pasaporte saber su titular, y desde el titular preguntar por su pasaporte.



Ej2:

Análisis de las relaciones:

- a. Agregación: Celular → Batería

Una Batería es un componente que puede existir independientemente del Celular. Puede ser fabricada, vendida y reemplazada en diferentes celulares. Es una relación de "tiene-un" donde ambos objetos tienen ciclos de vida independientes.

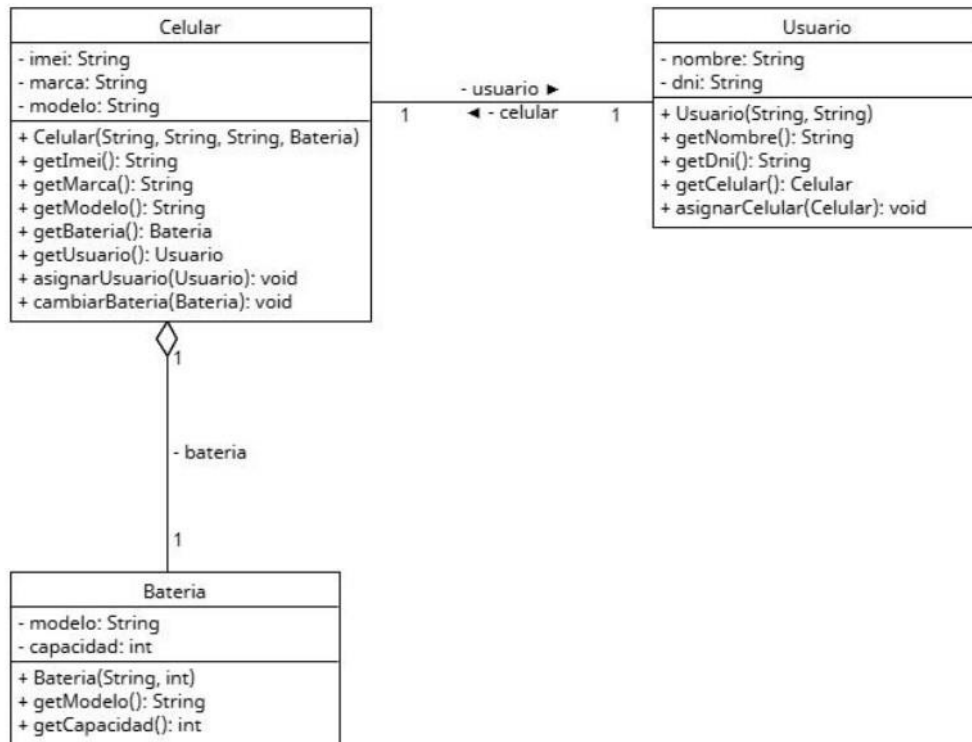
Dirección: Unidireccional. El Celular conoce su Batería, pero la Batería no necesita conocer al Celular.

- b. Asociación bidireccional: Celular ↔ Usuario

Un Usuario puede tener un Celular, y un Celular puede estar asignado a un Usuario. Ambos son entidades independientes que pueden existir por separado.

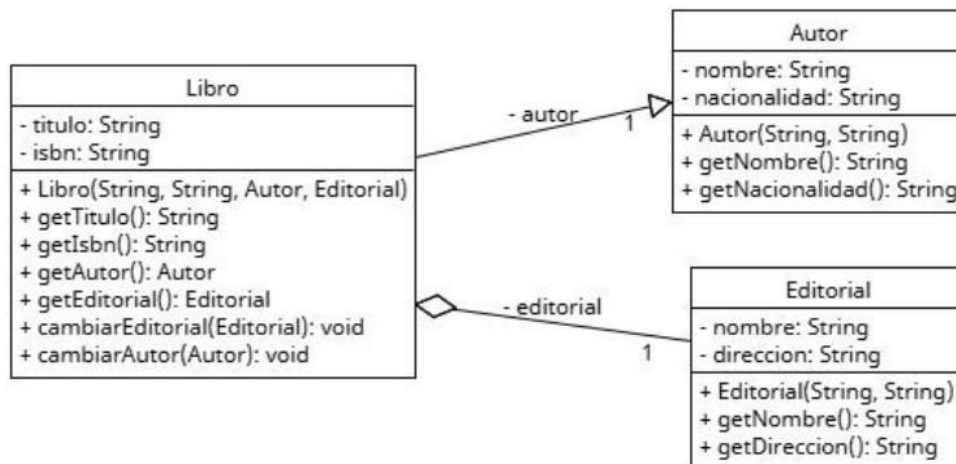
Bidireccional: Permite navegar en ambas direcciones: desde el usuario saber qué celular tiene, y desde el celular saber a qué usuario pertenece.

Diagrama UML. Ejercicio 2: Celular - Bateria - Usuario

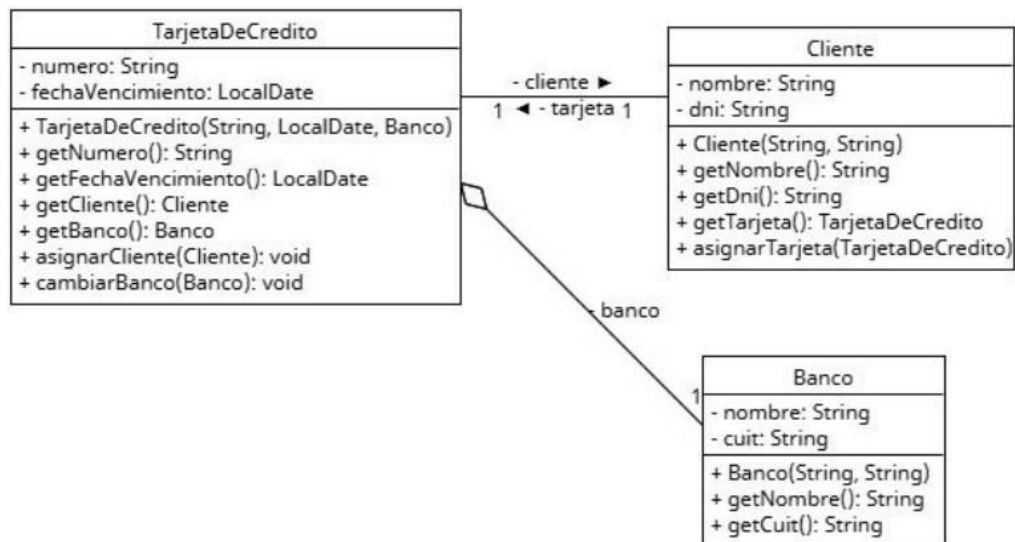


Ej3:

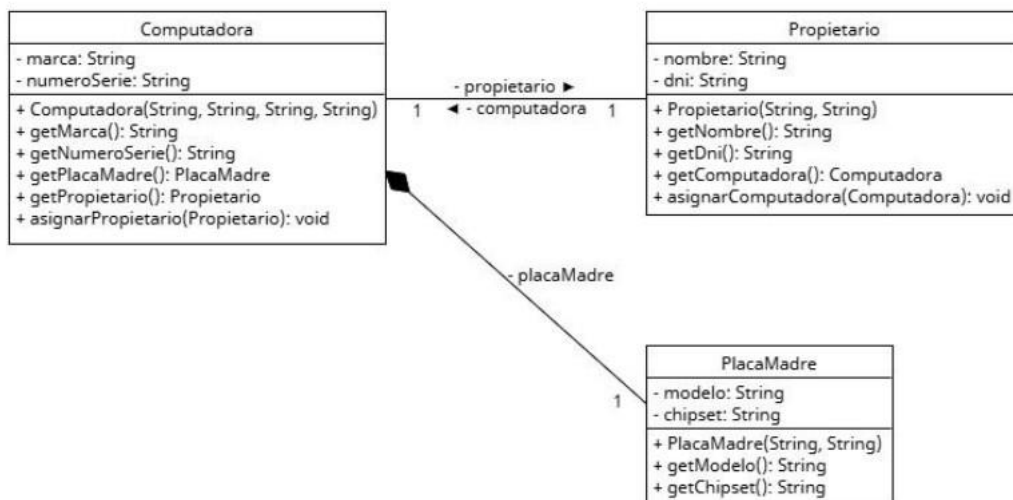
Diagrama UML. Ejercicio 3: Libro - Autor _ Editorial



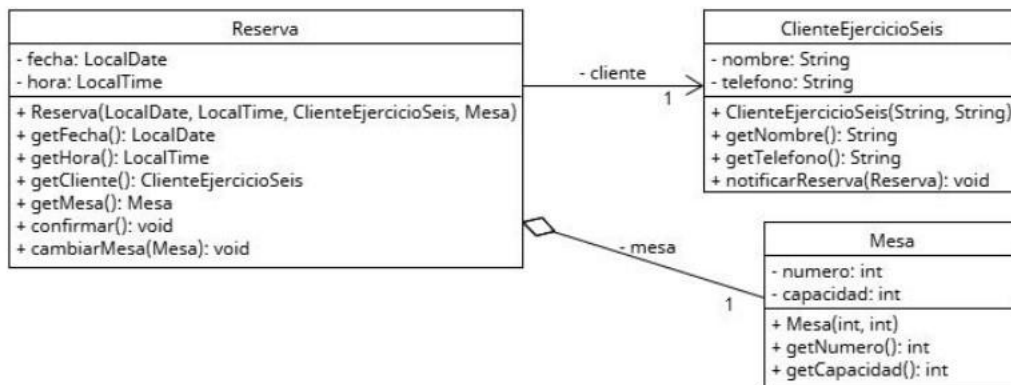
Ej 4



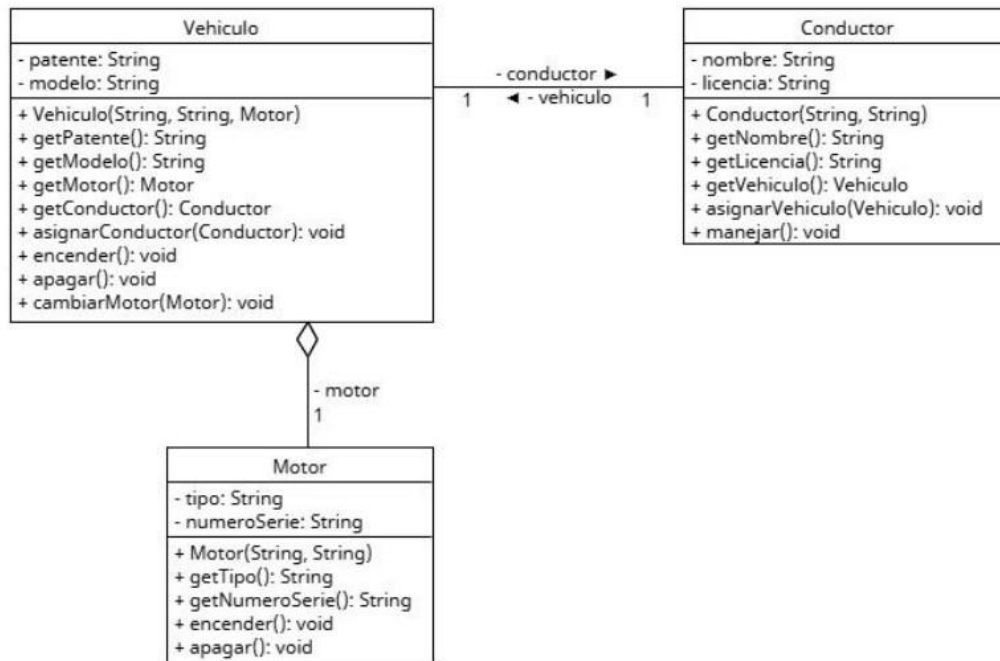
Ej5:



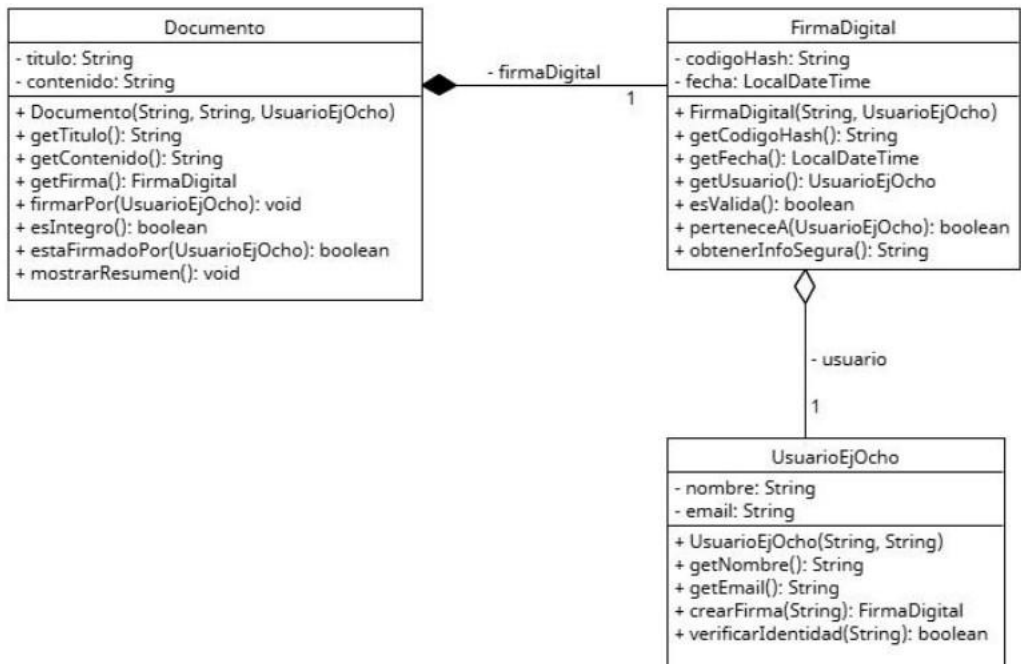
Ej6:



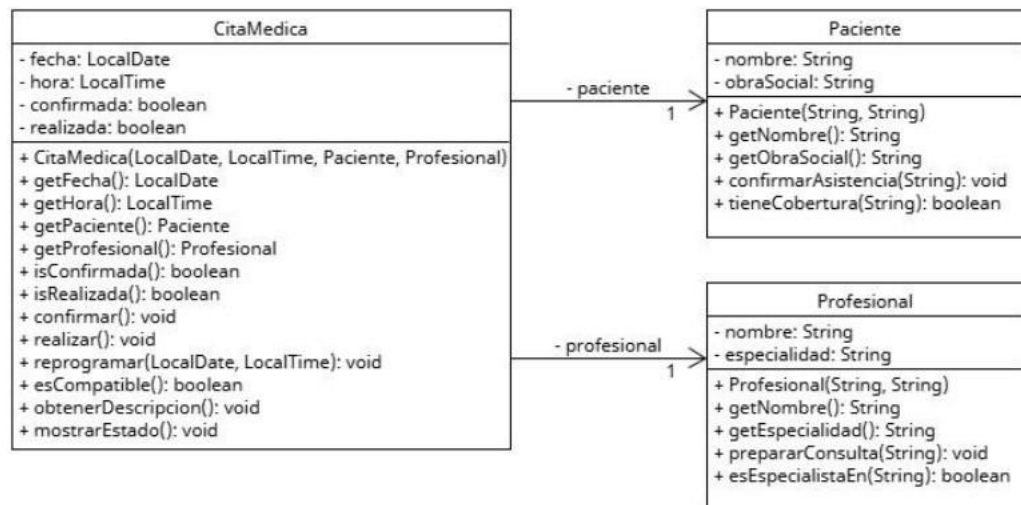
Ej7:



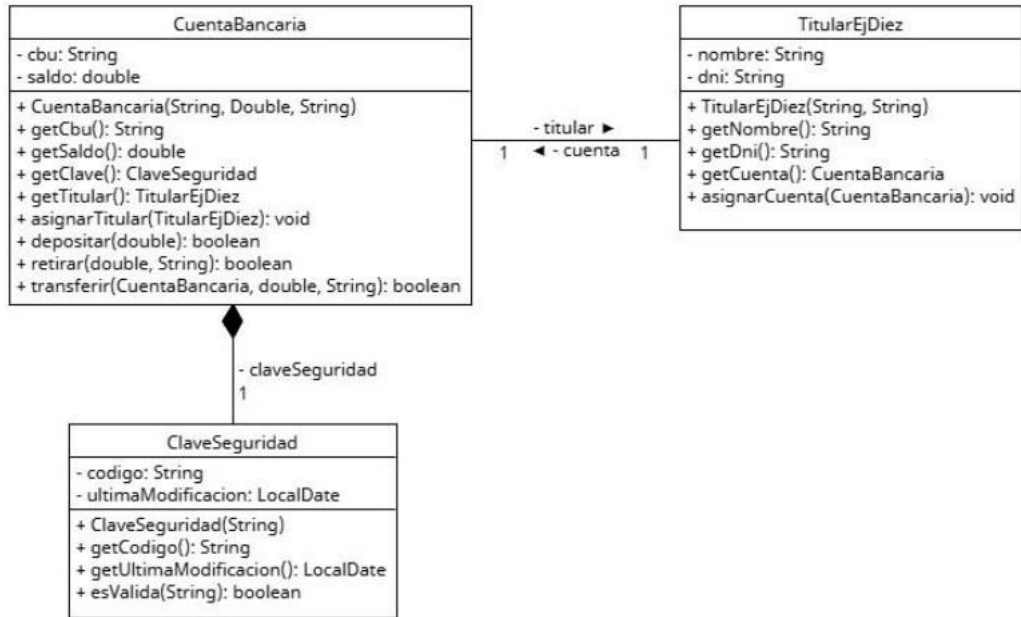
Ej8



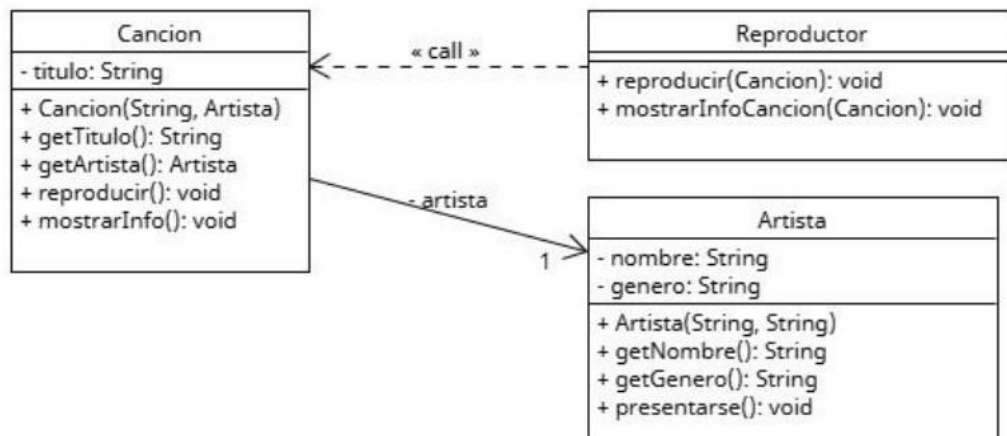
Ej9:



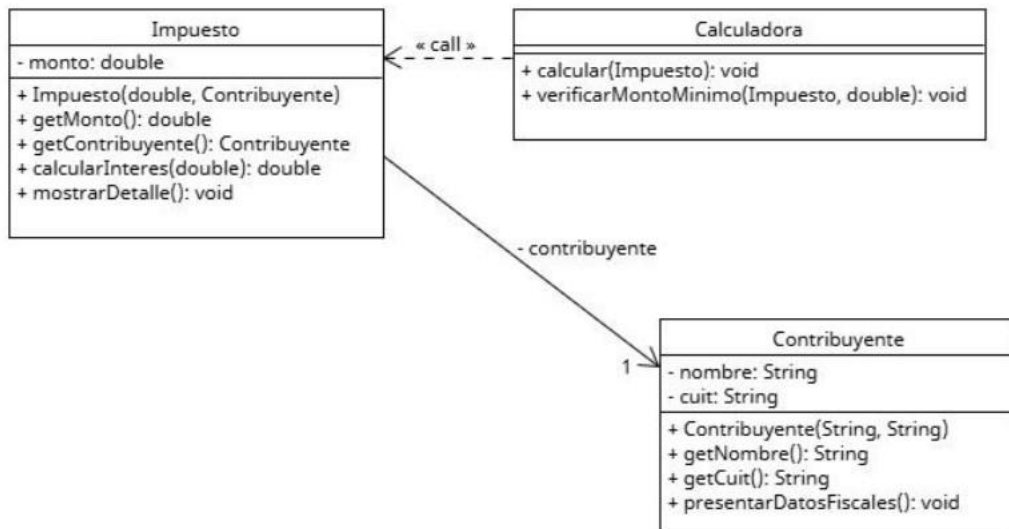
Ej 10



Ej 11



Ej 12



Ej 13:

- a. Asociación unidireccional: CódigoQR → Usuario

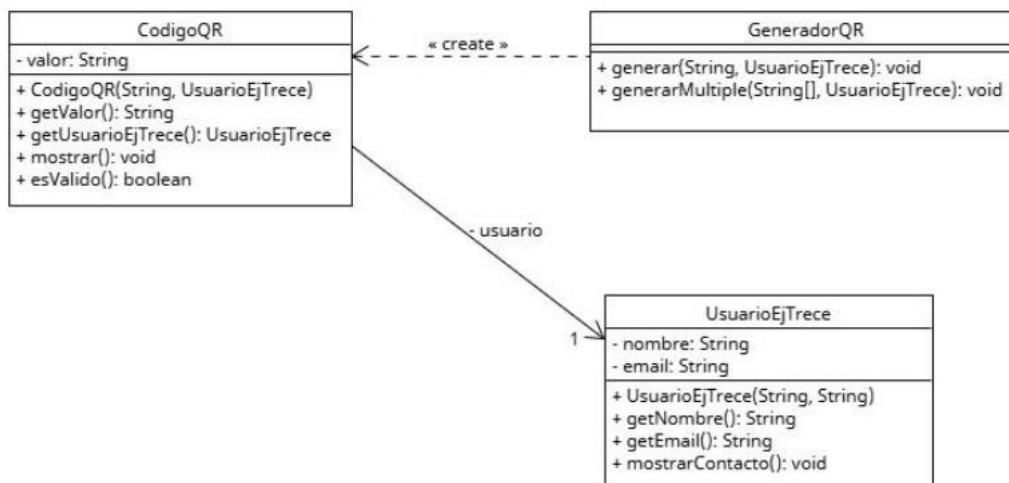
Un CódigoQR está asociado a un Usuario que lo genera.

Dirección: Unidireccional. El CódigoQR conoce al Usuario.

- b. Dependencia de creación: GeneradorQR.generar(String, Usuario)

El GeneradorQR crea un CódigoQR temporalmente dentro de un método.

Dirección: GeneradorQR ----> CódigoQR (línea punteada hacia CódigoQR)



Ej 14:

- a. Asociación unidireccional: Render → Proyecto

Un Render está asociado a un Proyecto que procesa.

Dirección: Unidireccional. El Render conoce al Proyecto.

- b. Dependencia de creación: EditorVideo.exportar(String, Proyecto)

El EditorVideo crea un Render temporalmente dentro del método.

Dirección: EditorVideo ----> Render (línea punteada hacia Render)

