Justificación de Principios POO — SmartHome

1. Encapsulamiento

- Todos los atributos de las clases (`_id_hogar` , `_clave` , `_estado_dispositivo` , etc.) se declararon como **privados** (`_`) para proteger el estado interno.
- El acceso y modificación se hace mediante **métodos públicos** (`get_id_usuario()` , `set_dispositivos_activos()`) o **propiedades** (`rol` , `estado_dispositivo`), evitando la manipulación directa.
- Esto asegura que los cambios en el estado de un objeto pasen siempre por una interfaz controlada.

---

2. Abstracción

- Cada clase expone **solo lo necesario** para cumplir su función:
  - `Usuario` abstrae la verificación de clave y el cambio de rol.
  - `DispositivoHogar` abstrae el encendido/apagado y la consulta de si es esencial.
  - `Automatizacion` abstrae la lógica de apagado masivo de dispositivos no esenciales.
- Los detalles internos (estructura de datos, validaciones) quedan ocultos al resto del sistema.

---

3. Composición y Agregación

- **Composición**:
  - `Hogar` contiene `DispositivoHogar` (si el hogar se elimina, los dispositivos dejan de existir).
  - `Automatizacion` contiene `DispositivoHogar` (la automatización no tiene sentido sin ellos).
- **Agregación**:
  - `Usuario` se asocia a `Hogar` y a `DispositivoControl` sin que su ciclo de vida dependa de ellos.
  - `Usuario` puede tener varios `DispositivoHogar` asociados, pero estos pueden existir sin el usuario.

---

4. Modularidad

- Cada clase está en su propio archivo dentro de `modelos/` , lo que facilita el mantenimiento y la escalabilidad.
- Los tests están separados en `tests/` , siguiendo la misma estructura modular.

---

5. Principio de Responsabilidad Única (SRP)

- Cada clase tiene **una única responsabilidad**:
  - `Hogar` : gestionar datos y actividad del hogar.
  - `Usuario` : gestionar credenciales y rol.
  - `DispositivoHogar` : representar y controlar un dispositivo físico.
  - `DispositivoControl` : llevar el conteo y estado de dispositivos.
  - `Automatizacion` : ejecutar reglas automáticas sobre dispositivos.
- Esto reduce el acoplamiento y facilita cambios futuros.

---

6. Reutilización y Escalabilidad

- La estructura permite agregar nuevos tipos de dispositivos o automatizaciones sin modificar las clases existentes.
- Los tests unitarios garantizan que las modificaciones no rompan funcionalidades previas.

Con este texto, el UML y los tests, tu entrega queda **completa y bien fundamentada**:

- Código implementado con TDD.
- Diagrama UML alineado al modelo ER.
- Justificación de principios POO clara y directa.

Diagrama de clases — SmartHome (POO)

Clase Hogar

- **Atributos**
  - `_id_hogar: int`
  - `_registro_actividad: str`
  - `_tiempo_de_conexion: str`
  - `_ubicacion: str`
  - `_tipo_de_vivienda: str`
- **Métodos**
  - `get_id_hogar() -> int`
  - `set_registro_actividad(actividad: str)`
  - `registro_actividad -> str` *(getter)*
  - `calcular_tiempo_formateado() -> str`
- **Relaciones**
  - 1 → N con `Usuario` (*agregación*)
  - 1 → N con `DispositivoHogar` (*composición*)

Clase Usuario

- **Atributos**
  - `_id_usuario: int`
  - `_nombre: str`
  - `_clave: str`
  - `_rol: str`
  - `_tiempo_de_conexion: str`
  - `_edad: int`
  - `_mail: str`
  - `_telefono: str`
  - `_registro_actividad: str`
- **Métodos**
  - `get_id_usuario() -> int`
  - `verificar_clave(clave: str) -> bool`
  - `cambiar_rol(nuevo_rol: str)`
  - `rol -> str` *(getter)*
- **Relaciones**
  - Pertenece a un `Hogar` (*agregación*)
  - 1 → N con `DispositivoControl` (*agregación*)
  - 1 → N con `DispositivoHogar` (*agregación*)

Clase DispositivoHogar

- **Atributos**
  - `_id_dispositivo: str`
  - `_id_usuario_conectado: int`
  - `_ubicacion: str`
  - `_hora_de_conexion: str`
  - `_nombre_dispositivo: str`
  - `_tipo_dispositivo: str`
  - `_marca_dispositivo: str`
  - `_estado_dispositivo: str`
  - `_consumo_energetico: float`
  - `_es_esencial: bool`
- **Métodos**
  - `get_id_dispositivo() -> str`
  - `encender()`
  - `apagar()`
  - `es_esencial() -> bool`
  - `estado_dispositivo -> str` *(getter)*
- **Relaciones**
  - Pertenece a un `Hogar` (*composición*)
  - Pertenece a un `Usuario` (*agregación*)
  - Puede ser controlado por `DispositivoControl` (*composición*)
  - Puede ser parte de una `Automatizacion` (*composición*)

Clase DispositivoControl

- **Atributos**
  - `_id_dispositivo_control: int`
  - `_id_usuario_conectado: int`
  - `_hora_de_conexion: str`
  - `_dispositivos_activos: int`
  - `_dispositivos_apagados: int`
  - `_dispositivos_en_ahorro: int`
- **Métodos**
  - `get_id_dispositivo_control() -> int`
  - `set_dispositivos_activos(cantidad: int)`
  - `set_dispositivos_apagados(cantidad: int)`
  - `set_dispositivos_en_ahorro(cantidad: int)`
  - `dispositivos_activos -> int` *(getter)*
  - `dispositivos_apagados -> int` *(getter)*
  - `dispositivos_en_ahorro -> int` *(getter)*

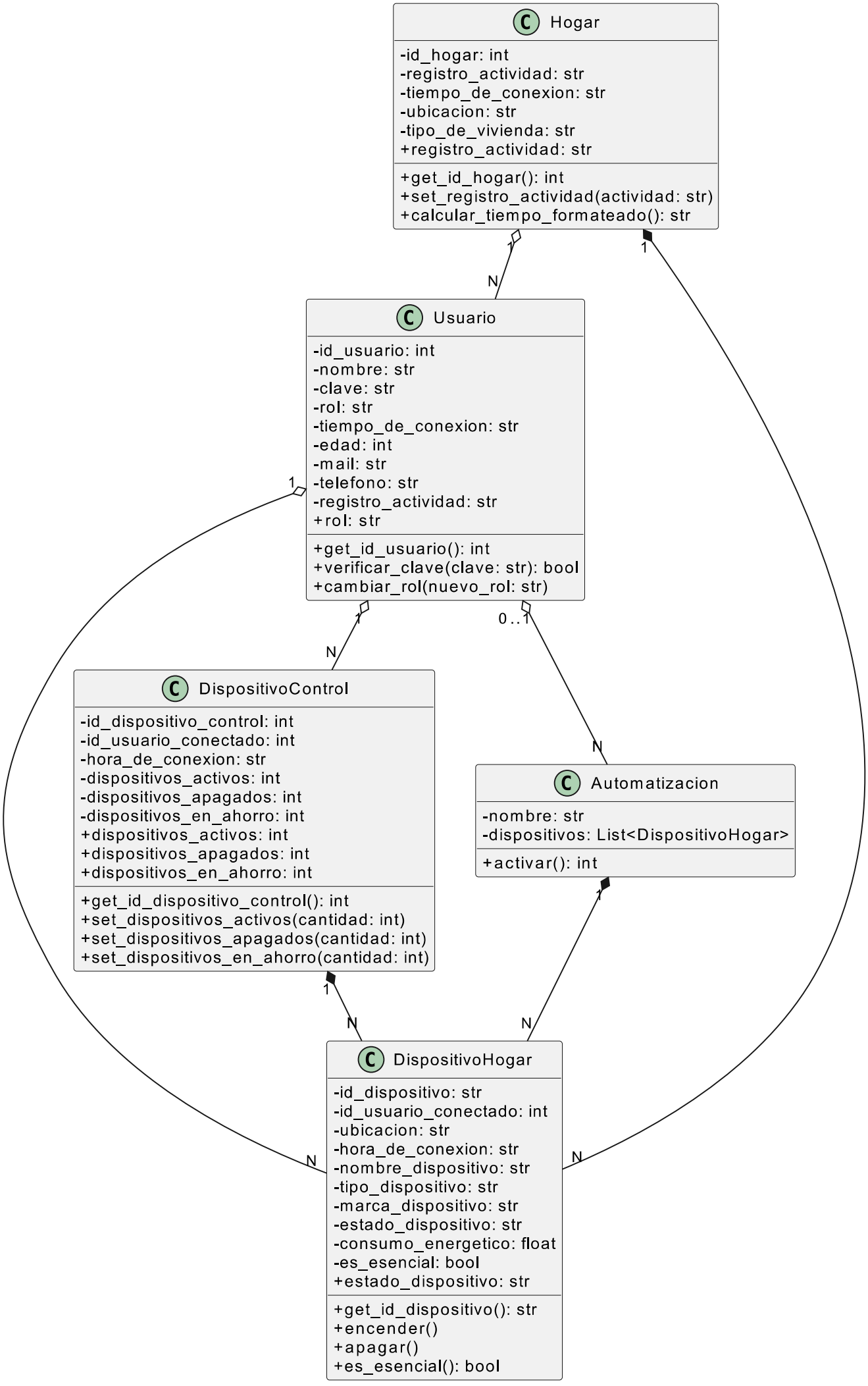
- **Relaciones**
  - Pertenece a un `Usuario` (*agregación*)
  - Controla N `DispositivoHogar` (*composición*)

Clase Automatizacion

- **Atributos**
  - `_nombre: str`
  - `_dispositivos: list[DispositivoHogar]`
- **Métodos**
  - `activar() -> int` *(apaga no esenciales encendidos y devuelve cantidad apagados)*
- **Relaciones**
  - Contiene N `DispositivoHogar` (*composición*)
  - Puede estar asociada a un `Usuario` (*agregación*)

Resumen de relaciones

- **Hogar** 1 — N **Usuario** (*agregación*)
- **Hogar** 1 — N **DispositivoHogar** (*composición*)
- **Usuario** 1 — N **DispositivoControl** (*agregación*)
- **Usuario** 1 — N **DispositivoHogar** (*agregación*)
- **DispositivoControl** 1 — N **DispositivoHogar** (*composición*)
- **Automatizacion** 1 — N **DispositivoHogar** (*composición*)


Hogar
-id_hogar: int
-registro_actividad: str
-tiempo_de_conexion: str
-ubicacion: str
-tipo_de_vivienda: str
+registro_actividad: str
+get_id_hogar(): int
+set_registro_actividad(actividad: str)
+calcular_tiempo_formateado(): str
Usuario
-id_usuario: int
-nombre: str
-clave: str
-rol: str
-tiempo_de_conexion: str
-edad: int
-mail: str
-telefono: str
-registro_actividad: str
+rol: str
+get_id_usuario(): int
+verificar_clave(clave: str): bool
+cambiar_rol(nuevo_rol: str)
DispositivoControl
-id_dispositivo_control: int
-id_usuario_conectado: int
-hora_de_conexion: str
-dispositivos_activos: int
-dispositivos_apagados: int
-dispositivos_en_ahorro: int
+dispositivos_activos: int
+dispositivos_apagados: int
+dispositivos_en_ahorro: int
+get_id_dispositivo_control(): int
+set_dispositivos_activos(cantidad: int)
+set_dispositivos_apagados(cantidad: int)
+set_dispositivos_en_ahorro(cantidad: int)
Automatizacion
-nombre: str
-dispositivos: List<DispositivoHogar>
+activar(): int
DispositivoHogar
-id_dispositivo: str
-id_usuario_conectado: int
-ubicacion: str
-hora_de_conexion: str
-nombre_dispositivo: str
-tipo_dispositivo: str
-marca_dispositivo: str
-estado_dispositivo: str
-consumo_energetico: float
-es_esencial: bool
+estado_dispositivo: str
+get_id_dispositivo(): str
+encender()
+apagar()
+es_esencial(): bool
1
N
0..1