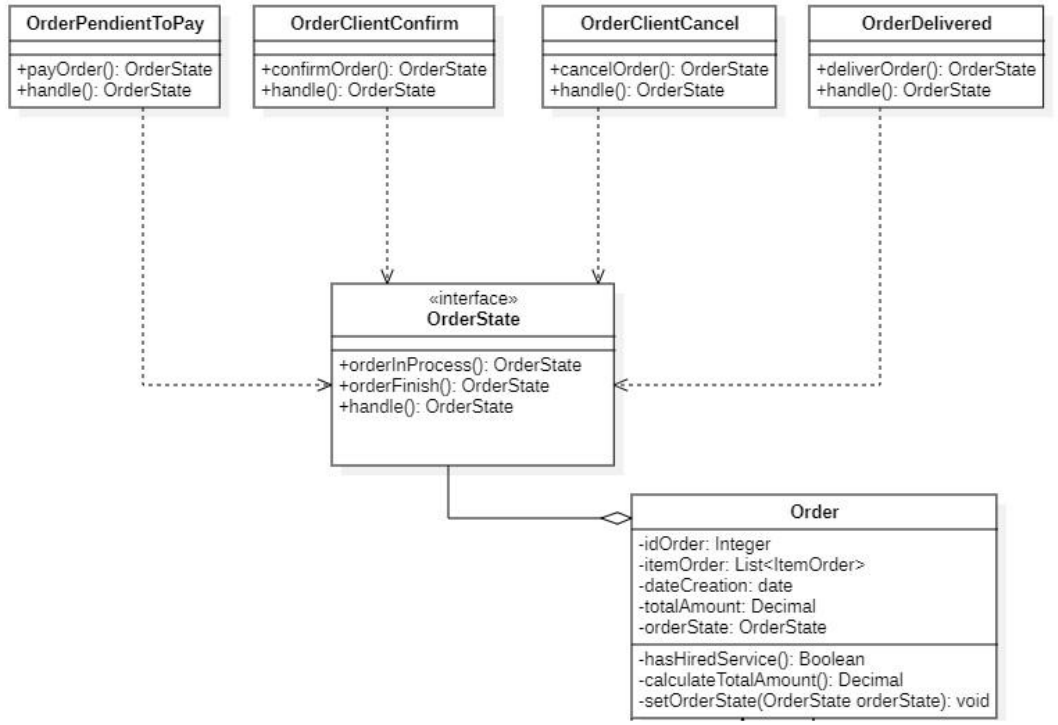
FINNEGANS

Grupo purple
Documentación patrones de diseño usados

Patrón State

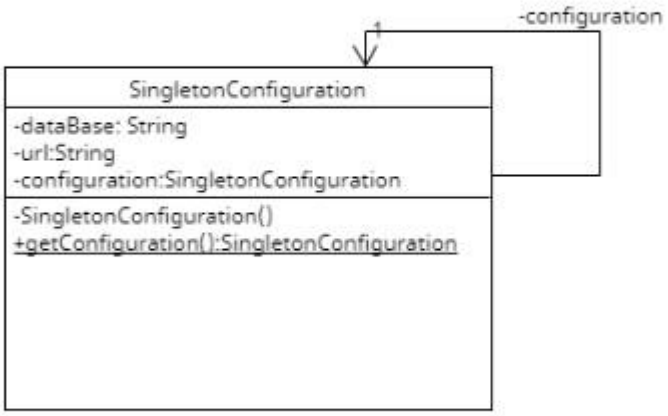


Selección del patrón State para poder cambiar el estado del modelo Order, el estado inicial es de pendientTopay, esperando la confirmación del pago para que la clase Order Setee el estado que viene por parámetro y cambiarlo desde el estado que devuelve cada estado concreto, en el caso de que se confirme el pago se devuelve el estado OrderClientConfirm y en caso contrario se devuelve el estado OrderClientCancel.

Es necesario que el modelo Order tenga instanciado el OrderState y el método setOrderState para poder cambiar los estados en base a las confirmaciones que se vayan realizando en cuanto al pedido, se puede cancelar tanto de OrderPendientToPay como de OrderClientConfirm.

FINNEGANS

Patrón Singleton

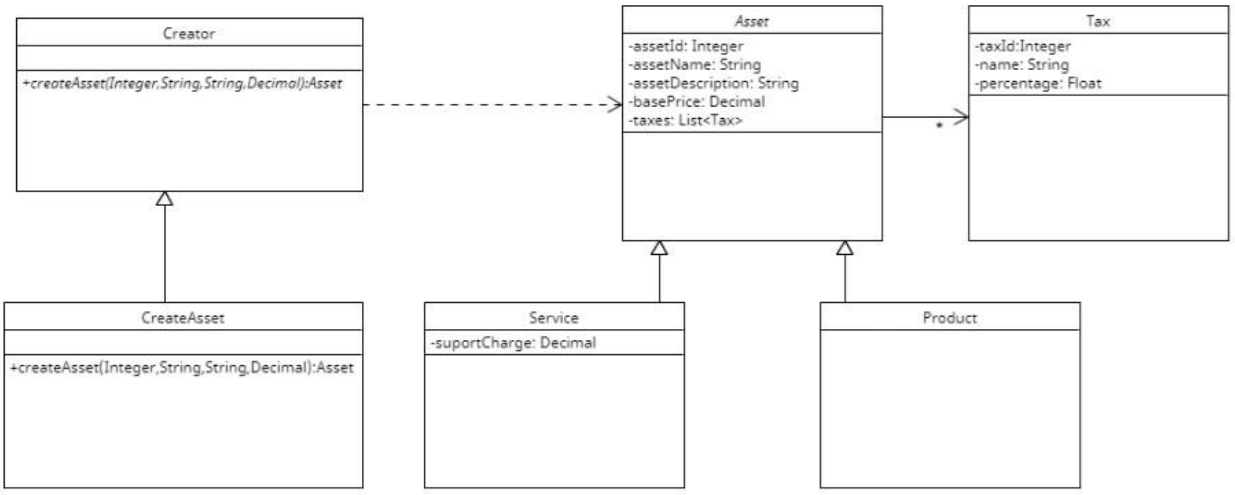


Seleccionamos el patrón singleton en la configuración de la base de datos para que haya una sola conección a esta base que solo se puede instanciar a si misma en caso de no existir, por eso tiene el constructor privado y el método estático getConfiguration que devuelve un SingletonConfiguration desde cualquier clase de la app para acceder a la base de datos

.

FINNEGANS

Patrón Factory



Seleccionamos el patrón Factory con la finalidad de que las clases hijas especifiquen que tipo de Asset va a crearse, puede ser Service o Product en este caso, mediante el método createAsset() que devuelve el tipo de bien que se crea.