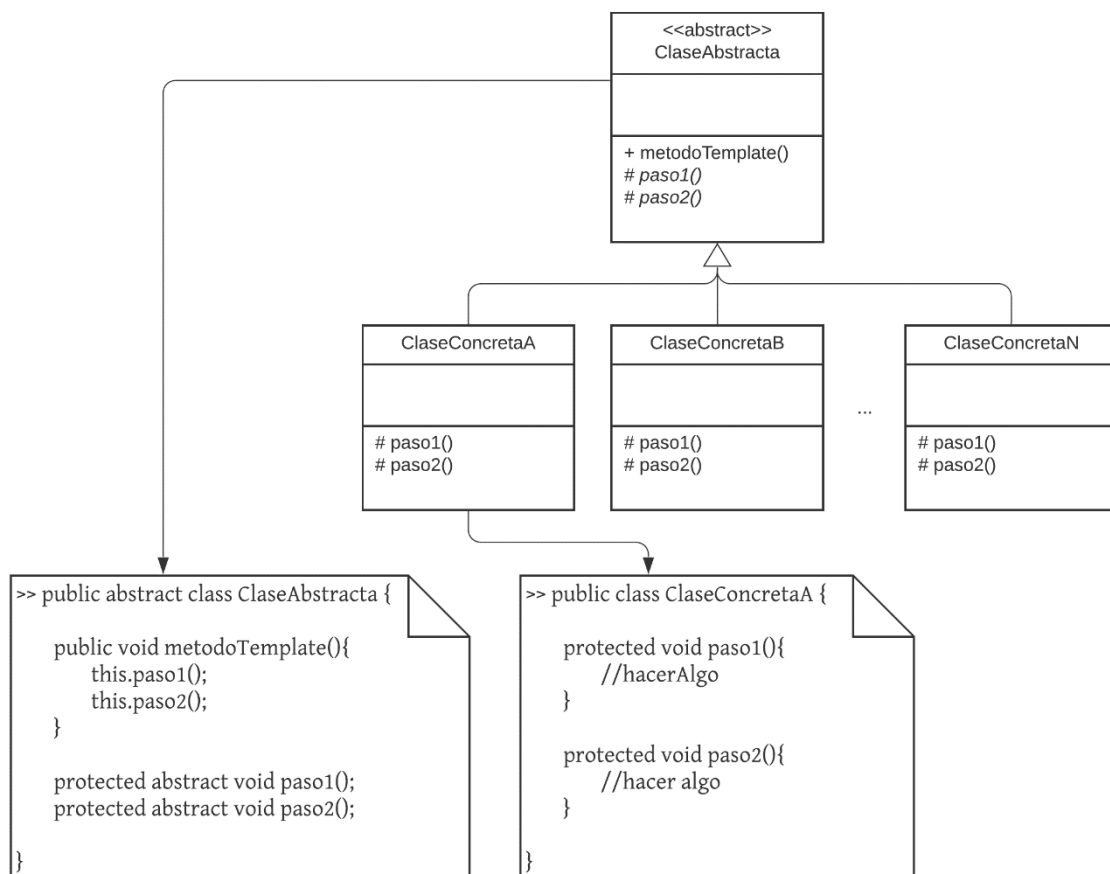


Patrón Template Method

- **Clasificación:** Patrón de comportamiento.
- **¿Qué hace?**
 - ✓ Define el esqueleto de un algoritmo, estableciendo los pasos que sí o sí se deben implementar.
- **Se sugiere su utilización cuando:**
 - ✓ **Varias abstracciones tienen los mismos pasos y orden para realizar una acción, pero cada una de ellas los implementa de forma diferente.**
 - ✓ Se requiere utilizar de forma polimórfica dos o más objetos que pueden ejecutar el mismo algoritmo, respetando sus pasos, pero con implementaciones distintas para cada uno de éstos.
- **Estructura genérica:**



Consideraciones:

- ✓ El método template puede no ser void, así como también pueden no serlo los “pasos”.
- ✓ Pueden existir sentencias intermedias entre los “pasos” que tengan comportamiento en común para todas las clases concretas, en otras palabras, que se haga uso de métodos (concretos, no abstractos) definidos en la clase abstracta.
- ✓ Los “pasos” no necesariamente son 2, sino que pueden ser varios.
- ¿Qué proporciona su uso?
 - ✓ Mayor **mantenibilidad** debido a que el comportamiento de los “pasos” es fácilmente localizable.
 - ✓ Alta **cohesión** en todas las clases involucradas (abstracta y concretas).
 - ✓ **Extensibilidad** para agregar nuevas clases concretas que definan nuevos comportamientos para los pasos.
- Code smells que soluciona/evita de forma directa:
 - ✓ Código duplicado
 - ✓ Clase dios
 - ✓ Herencia rechazada