

# ¡BIENVENIDOS!

## Soy Diego Martín

Me dedico a la arquitectura de software y desarrollo .NET



<https://www.linkedin.com/in/diegosasw>

# REQUISITOS

- **Conocimientos básicos de C#**
- **Entorno de desarrollo**
  - **SDK de .NET**
  - **IDE**

# ÍNDICE

- **Planteamiento de un problema**
- **Solución con herencia**
- **Solución con composición**
- **El patrón de diseño de Estrategia**
- **Conclusiones**

# PATRÓN DE DISEÑO ESTRATEGIA CON C#

STRATEGY/POLICY PATTERN

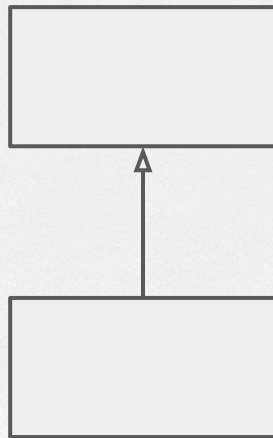


**¿Cómo podría conseguir que algo se pueda comportar de diferente forma?**

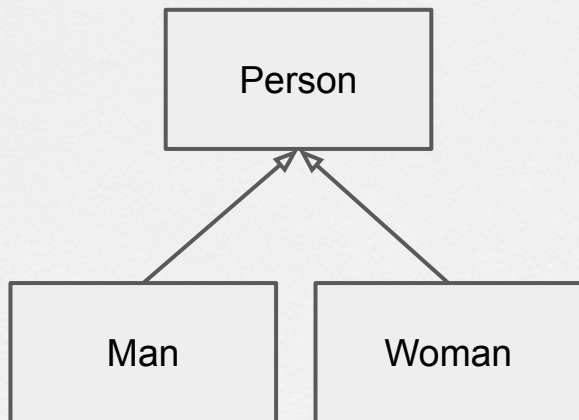


# HERENCIA

- Relación IS-A
- Dependencia fuerte

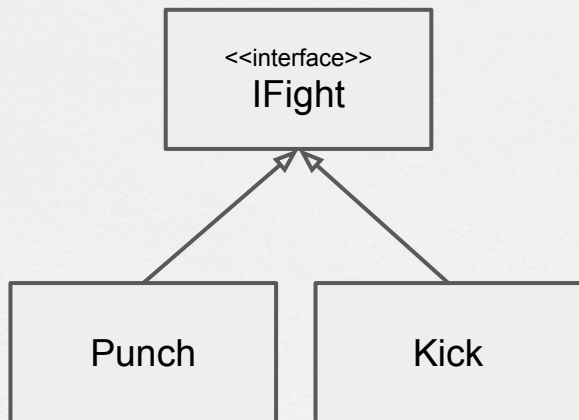


# HERENCIA



- Una *extiende* a otra
- Hereda
  - Propiedades
  - Comportamiento

# HERENCIA



- Una *implementa* otra
- Hereda
  - Contrato de comportamiento



# HERENCIA



- Un guerrero tiene
  - nombre
  - edad
  - aspecto
- Un guerrero puede
  - luchar

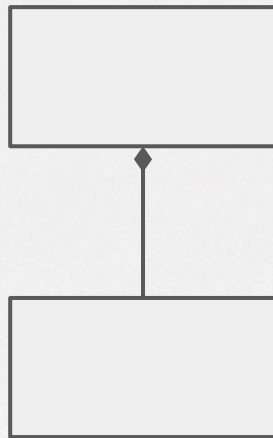
**¡¡CUIDADO!!**

*una tortuga ninja  
es un guerrero,  
... pero ¿es una persona?*

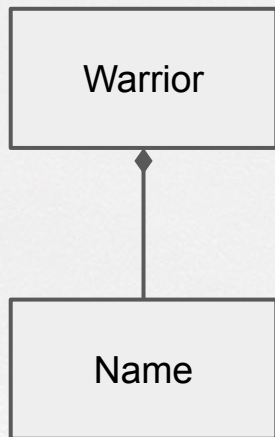


# COMPOSICIÓN

- **Relación HAS-A**
- **Dependencia blanda**

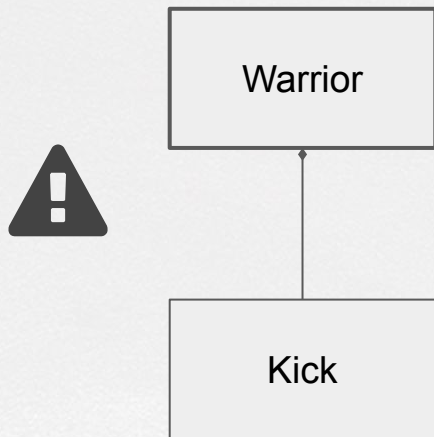


# COMPOSICIÓN



- Una *tiene* a otra
- Se *compone* de o *delega en*
  - Estado
  - Comportamiento concreto

# COMPOSICIÓN



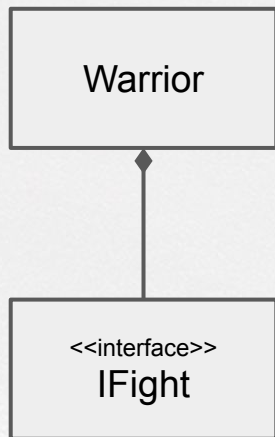
- Un guerrero puede
  - dar patadas

**¡¡CUIDADO!!**

*una boxeador  
es un guerrero,  
... pero ¿puede dar patadas?*



# COMPOSICIÓN



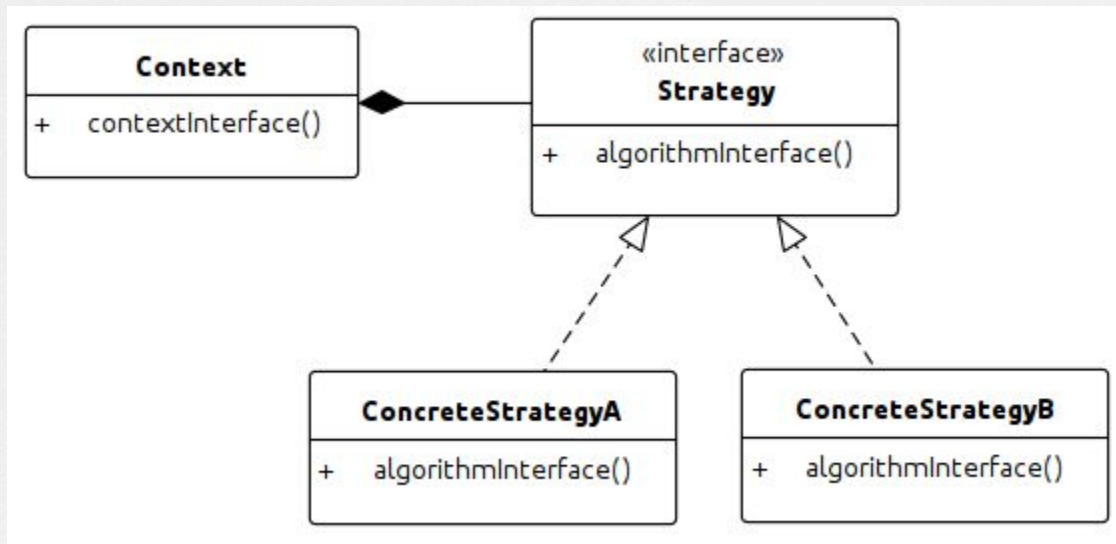
- Una *tiene* a otra
- Se compone de y puede delegar en
  - Contrato de comportamiento

## ENTONCES...

*¿es más flexible delegar en una implementación concreta o en un contrato?*







## PATRÓN DE DISEÑO ESTRATEGIA

Es un patrón de comportamiento que permite cambiar un algoritmo de comportamiento en tiempo de ejecución

# CONCLUSIONES

- Favorece composición frente a herencia
- La composición es una forma de delegación
- Ambas relaciones IS-A (herencia) y HAS-A (composición) son la base de la mayoría de patrones de diseño