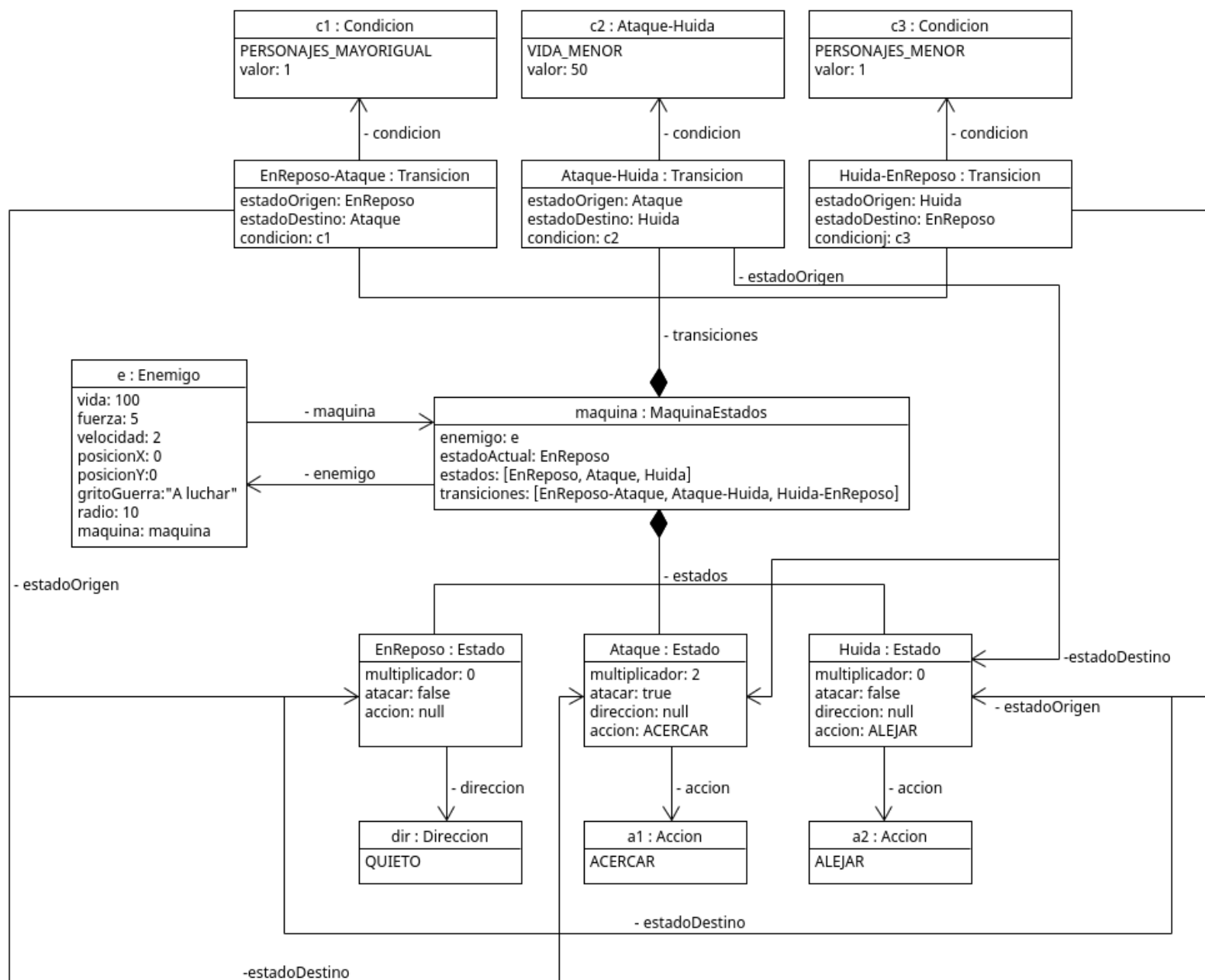


Al igual que en el apartado anterior, usamos el patrón Singleton para asegurar esa instancia única de la clase que representa el juego. Esta clase tiene una composición fuerte con la clase abstracta Vivo, de la cual heredan personajes y enemigos. La clase Juego gestiona los ciclos de los que se hablan en el enunciado, comprueba en cada ciclo si hay personajes vivos para finalizar el juego y, como se pide, es el encargado de proporcionar los vivos que están en el radio de un enemigo.

Todos los vivos, reciben una velocidad y fuerza por parte del juego, además, en cada ciclo, pueden moverse, atacar a otro vivo (para reducir la vida en función de la fuerza del que te ha atacado se usa el reducirVidaAtaque y la vida por ciclo con reducirVidaCiclo), la clase control gestiona estas acciones para personajes, y la máquina de estados para los enemigos en función del estado actual.

En más detalle, un Estado puede tener la variable atacar a true, una dirección no nula o una acción no nula, el valor no nulo en cada ciclo es la funcionalidad a ejecutar, por ello, si en un ciclo el valor de dirección en el estado actual de la máquina de estados es NORTE o el de acción ACERCAR, la máquina invoca el método mover(NORTE) o mover(X, Y) respectivamente correspondiente al enemigo que tiene asociado (un enemigo tiene una y sólo una máquina de estados y viceversa) por composición fuerte, uno no tendría sentido sin el otro ni al revés.

Como la transicionalidad es automática, en cada ciclo, la máquina de estados comprueba que transiciones hay disponibles para el valor de estadoActual e invoca los métodos evaluar con el enemigo que tiene asociado para dichas transiciones, en función de los resultados, la máquina de estados cambia el estadoActual al valor del estadoDestino de la transición.



En el momento en el que se describe en el enunciado la posición inicial del enemigo suponemos que es 0,0, esta debería ir modificándose con el diseño propuesto, por ejemplo, cuando un personaje entre en el radio del enemigo, el ciclo comprobará las transiciones desde 'EnReposo', encontrará la transición 'EnReposo-Ataque' y evaluará su condición asociada, para dicho caso, como c1 comprueba si hay 1 o más personajes y se cumple, la máquina aplicará la transición y se repetirá el proceso con los objetos correspondientes.