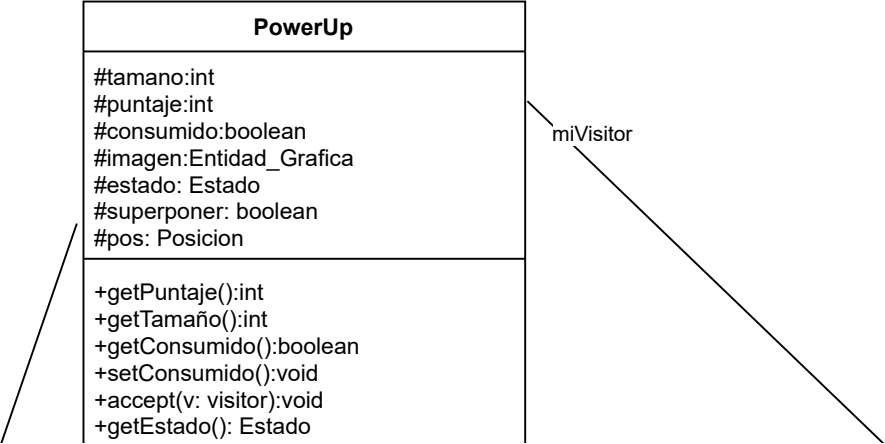
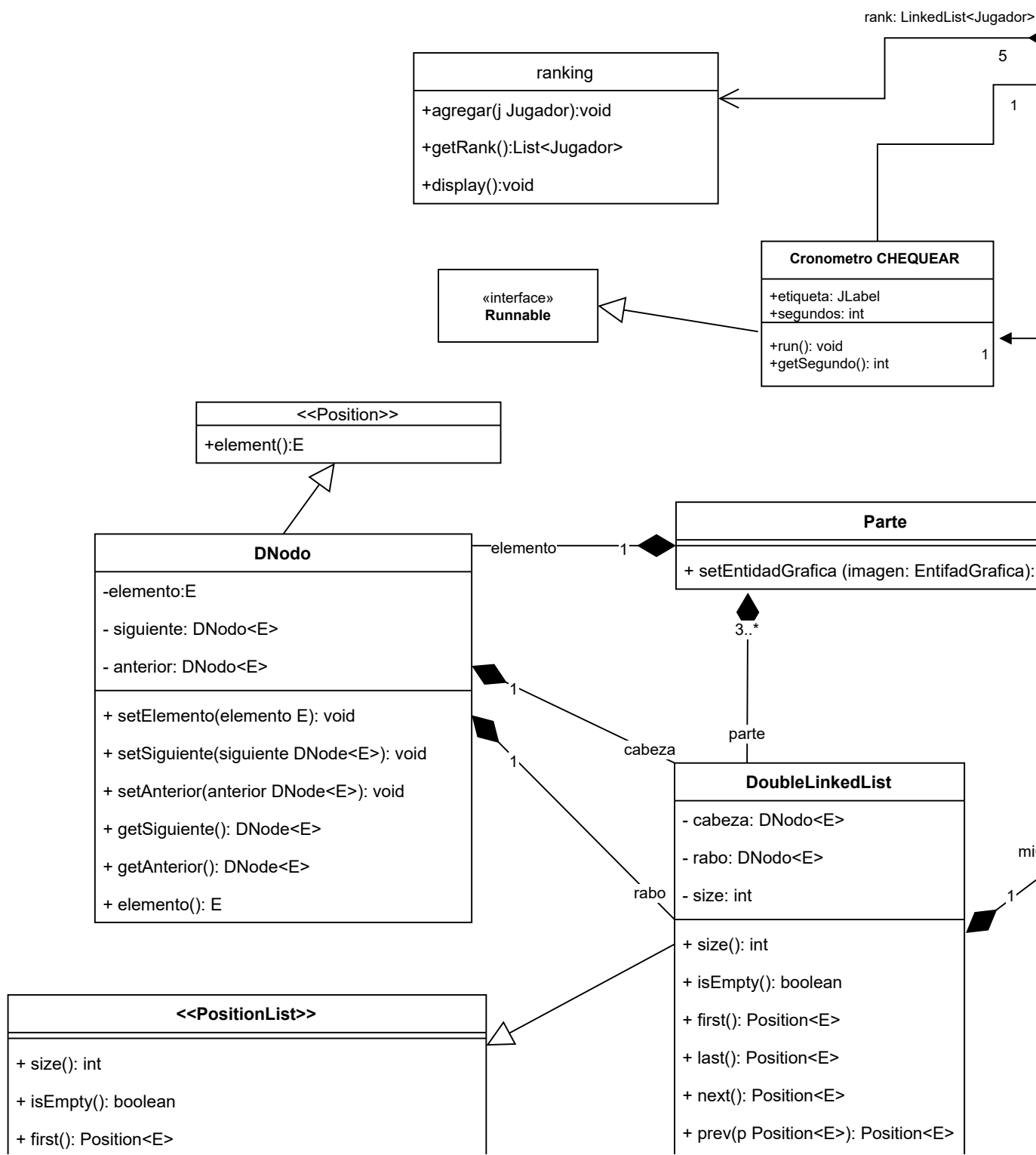
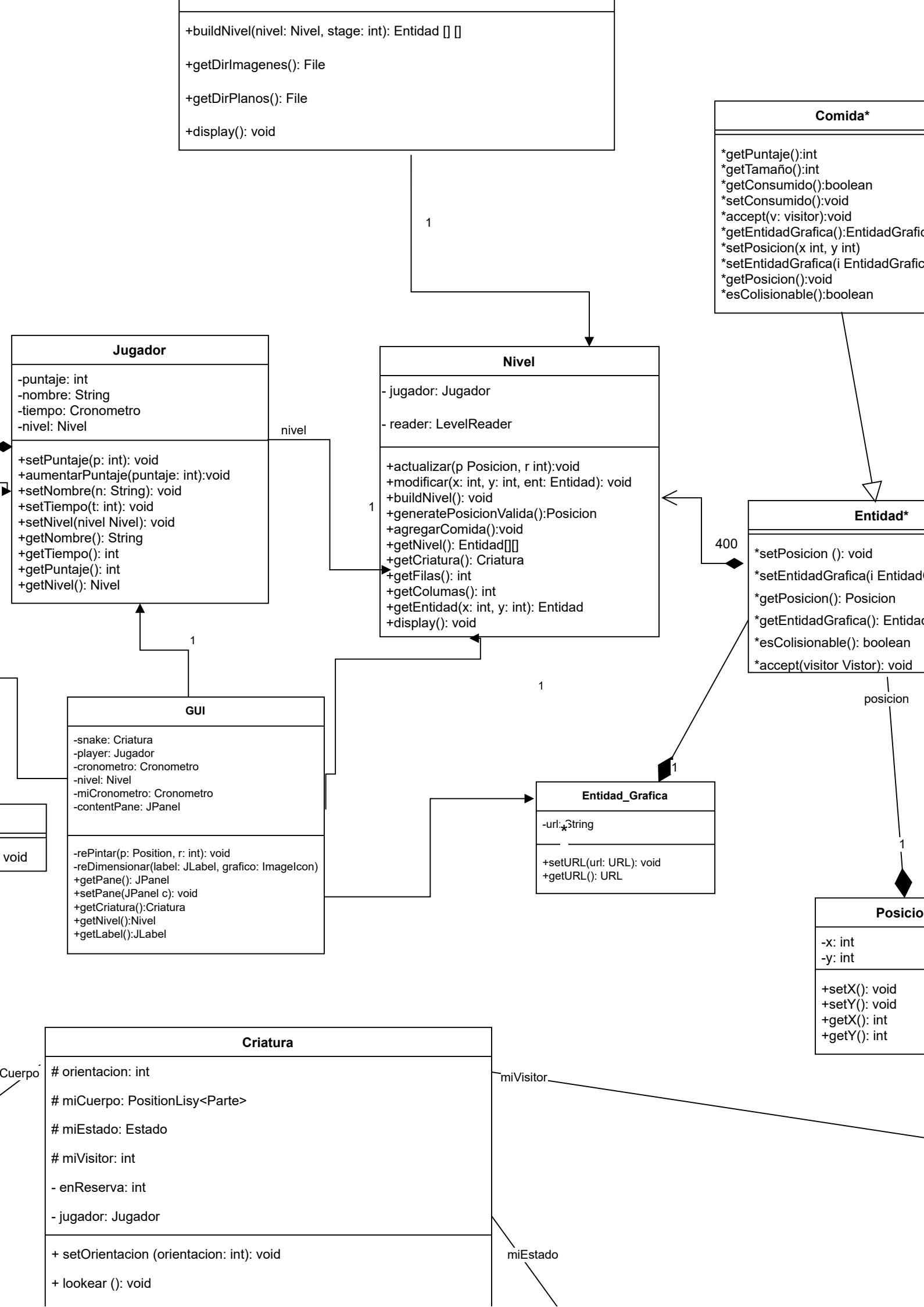
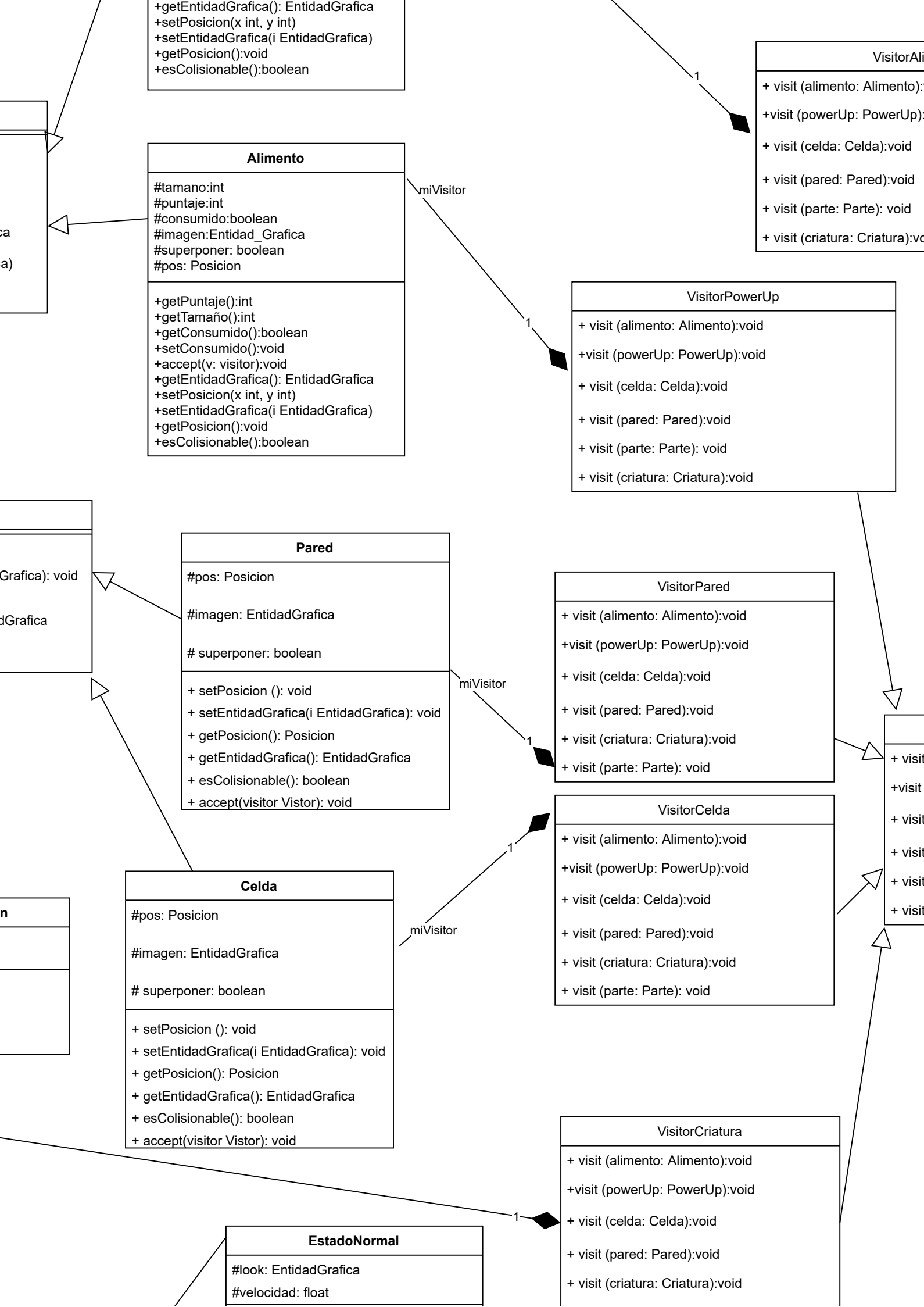


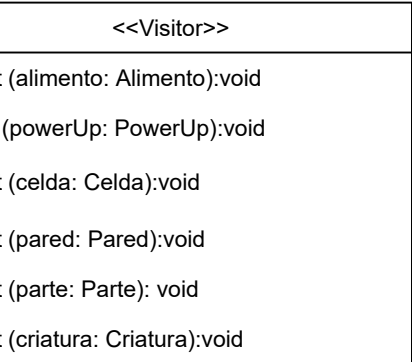
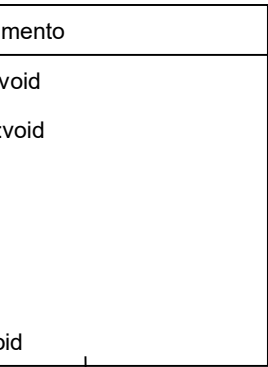
LevelReader
<div>- comida: LinkedList<Entry<Integer,Integer>></div> <div>- muro: LinkedList<Entry<Integer,Integer>></div> <div>- map: char [] []</div> <div>- path: String</div> <div>- DirPlanos: File</div> <div>- DirImagenes: File</div>





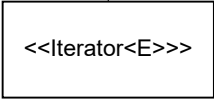
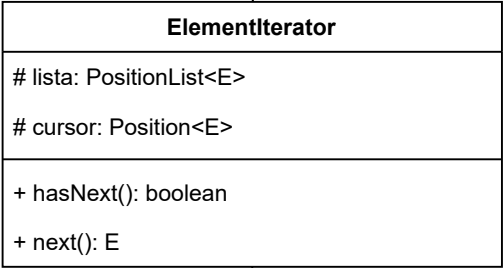






+ last(): Position<E>
+ next(p Position<E>): Position<E>
+ prev(p Position<E>): Position<E>
+ addFirst(element E): void
+ addLast(element E): void
+ addAfter(p Position<E>, element E): void
+ addBefore(p Position<E>, element E): void
+ remove(p Position<E>): E
+ set(p Position<E>, elemento E): E
+ iterator(): Iterator<E>
+ positions(): Iterable<Position<E>>

+ addFirst(element E): void
+ addLast(element E): void
+ addAfter(p Position<E>, element E): void
+ addBefore(p Position<E>, element E): void
+ remove(p Position<E>): E
+ set(p Position<E>, elemento E): E
+ iterator(): Iterator<E>
+ positions(): Iterable<Position<E>>



```
+ setEstado (estado: Estado): void
+ getOrientacion(): int
+ getCuerpo(): Iterator<Parte>
+ getEstado(): Estado
+ getJugador(): Jugador
+ comer (comida: Comida): Criatura
+ morir (): void
+ getReserva(): int
+ getCabeza(): Parte
+ getMovimiento(o int): Posicion
- desplazarEnX (desplazar: int): Posicion
- desplazarEnY (desplazar: int): Posicion
- mover (p: Posicion): void
+ moverArriba (entidad: Entidad): Criatura
+ moverAbajo (entidad: Entidad): Criatura
+ moverIzquierda (entidad: Entidad): Criatura
+ moverDerecha (entidad: Entidad): Criatura
+ accept (visitor: Visitor): void
```

1

