

The Last Shelter

Acciones y reglas



Versión 1.0

En este documento vamos a especificar las distintas acciones que puede realizar el usuario y los elementos del juego y, además, se van a definir una serie de reglas en las cuales se ve el funcionamiento de dichas acciones con sus precondiciones y resultados.

Miembros:

Andrés Molina López
Arthur Rodríguez Nesterenko
Manuel Ruiz Maldonado
Carlos Sequí Sánchez

Acciones

Acciones del Main Mode

- **Construir**
 - Limitaciones: tener recursos suficientes, que haya espacio válido no ocupado, tener desbloqueada la torre que se quiere construir en el nivel actual.
 - Consecuencias: nueva estructura situada en el espacio (visual), cambio del estado “no ocupado” a “ocupado” del espacio válido elegido, reducción de recursos.
- **Vender**
 - Limitaciones: el atributo vida de la torre debe ser mayor que 0.
 - Consecuencias: desaparece la estructura del espacio, cambia el estado “ocupado” a “no ocupado” del espacio válido en el que estaba, aumentan los recursos en función del precio de venta.
- **Reparar**
 - Limitaciones: la vida de la torre tiene que tener vida mayor que 0 y menor que vida máxima, hay que tener recursos suficientes.
 - Consecuencias: aumento de vida de la torre igual a la diferencia de la vida máxima menos la vida actual, disminución de recursos en función del coste de reparación.
- **Mejorar**
 - Limitaciones: que la torre no esté al nivel máximo, tener recursos suficientes y que esté habilitada la mejora.
 - Consecuencias: cambia la apariencia de la torre, aumenta el nivel de la torre (y con ello los atributos que dependen del nivel), disminuyen los recursos dependiendo del coste de mejora.
- **Poner escudo**
 - Limitaciones: tener recursos suficientes.
 - Consecuencias: reducción de daño sobre la torre, disminución de recursos en precio del escudo.
- **Seleccionar**
 - Limitaciones: objeto disponible para selección.
 - Consecuencias: se muestra un menú contextual alrededor del objeto seleccionado.
- **Disparar (torre)**
 - Limitaciones: objetivo localizado, torre orientada.
 - Consecuencias: disminuir vida del bicho.

- **Localizar objetivo**
 - Limitaciones: al menos hay un bicho en el rango de alcance de la torre.
 - Consecuencias: se selecciona un bicho y se convierte en el objetivo.
- **Orientar torre**
 - Limitaciones: tener un objetivo.
 - Consecuencias: la torre se mueve horizontal y verticalmente para ajustarse a la distancia del objetivo.
- **Colocar torre de energía**
 - Limitaciones: que haya una fuente de energía, que haya 3 espacios adyacentes válidos no ocupados sobre esa fuente de energía, tener recursos suficientes.
 - Consecuencias: aumento de recursos (cada 5 segundos aumentan 5 unidades la energía), cambio de estado de “no ocupado” a “ocupado” en los espacios válidos ocupados, disminución de recursos (lo que se paga por colocar la torre).
- **Mejorar la torre de energía**
 - Limitaciones: que la torre no esté al nivel máximo, tener recursos suficientes y que esté habilitada la mejora.
 - Consecuencias: cambia la apariencia de la torre, aumenta el nivel de la torre (y con ello los atributos que dependen del nivel), disminuyen los recursos dependiendo del coste de mejora.
- **Lanzar ataques especiales**
 - Limitaciones: esperar que se complete el tiempo de recarga, que esté desbloqueado en ese nivel.
 - Consecuencias: tiempo de recarga igual a 0, disminución de vida de bichos dentro del alcance del ataque.
- **Disparar (bicho)**
 - Limitaciones: colisión bicho-torre, bicho orientado.
 - Consecuencias: disminuir vida de la torre objetivo.
- **Avanzar (bichos)**
 - Limitaciones: no tener colisiones.
 - Consecuencias: desplazamiento hacia adelante en su ruta el paso determinado por la velocidad del bicho.
- **Orientar bicho**
 - Limitaciones: tener un objetivo.
 - Consecuencias: el bicho se mueve horizontalmente para golpear a la torre objetivo.

- **Colisión bicho-bicho de la misma ruta**
 - Limitaciones: hay un bicho delante del bicho.
 - Consecuencias: el estado cambia a “espera”.
- **Colisión bicho-bicho de distinta ruta**
 - Limitaciones: se produce la colisión.
 - Consecuencias: se resuelve la colisión.
- **Colisión bicho-torre**
 - Limitaciones: hay una torre en contacto con el radio de alcance del bicho.
 - Consecuencias: el bicho dispara a la torre.
- **Poner vista torre**
 - Limitaciones: torre seleccionada.
 - Consecuencias: cambio de la cámara.
- **Poner vista bicho**
 - Limitaciones: bicho seleccionado.
 - Consecuencias: cambio de la cámara.
- **Poner vista genérica**
 - Limitaciones: no hay.
 - Consecuencias: cambio de la cámara.

Acciones del Mapa

- **Seleccionar nivel**
 - Limitaciones: estar desbloqueado.
 - Consecuencias: mostrar interfaz de selección de dificultad para el nivel elegido.
- **Volver al menú principal**
 - Limitaciones: no hay.
 - Consecuencias: cambia al menú principal.
- **Entrar al research**
 - Limitaciones: no hay.
 - Consecuencias: muestra el menú de research.

Acciones del Research

- **Elegir nivel de mejora**
 - Limitaciones: tener XP suficiente, haber aumentado nivel previo de mejora (si tiene previo).
 - Consecuencias: se seleccionan las mejoras que se quieren realizar, pero no se hacen efectivas.
- **Confirmar nivel de mejora**
 - Limitaciones: pulsar botón "Done".
 - Consecuencias: se cambian los factores de mejora de los distintos parámetros que han sido seleccionados, se vuelve al mapa.
- **Cancelar mejora**
 - Limitaciones: pulsar botón "Cancel".
 - Consecuencias: se vuelve al mapa sin guardar cambios de mejora.

Acciones de Configuración

- **Reanudar**
 - Limitaciones: nivel pausado.
 - Consecuencias: el atributo estado de nivel cambia a running.
- **Salir**
 - Limitaciones: no hay.
 - Consecuencias: finaliza el nivel actual y vuelve al mapa.
- **Reiniciar**
 - Limitaciones: no hay
 - Consecuencias: carga el nivel con su estado inicial.

Reglas

Reglas del jugador

Objetivo: Seleccionar icono estructura

Regla funcional:

- IF HUD.iconoEstructura.Clicked() THEN HUD.iconoEstructura.seleccionado = true

Objetivo: Construir estructura

Regla funcional:

- IF HUD.iconoEstructura.seleccionado AND (torre.precioFabricacion <= level.recursosActuales) AND !level.posicionValidaActual.ocupada AND mouseClicked() THEN construirEstructura() AND level.recursosActuales -= torre.precioFabricacion

Objetivo: Seleccionar estructura

Regla funcional:

- IF level.posicionValidaActual.ocupada AND mouseClicked() THEN HUD.mostrarHUDEstructural()

Objetivo: Desplegar HUD

Regla funcional:

- IF level.posicionValidaActual.ocupada.Clicked() THEN HUD.desplegado = true

Objetivo: Vender estructura

Regla funcional:

- IF HUD.desplegado AND HUD.iconoVenta.Clicked() THEN (level.recursosActuales += estructura.precioVenta) AND estructura.destroyed()

Objetivo: Reparar

Regla funcional:

- IF HUD.desplegado AND HUD.iconoReparar.Clicked() AND (level.recursosActuales >= torre.precioReparacion) THEN (level.recursosActuales -= torre.precioReparacion) AND (torre.health += (torre.MAXHEALTH - torre.health))

Objetivo: Poner escudo

Regla funcional:

- IF HUD.desplegado AND HUD.iconoEscudo.Clicked() AND (torre.precioEscudo < level.recursosActuales) THEN torre.proteger() AND (level.recursosActuales -= torre.precioEscudo)

Objetivo: Mejorar

Regla funcional:

- IF HUD.desplegado AND HUD.iconoMejora.Clicked() AND (torre.precioMejora < level.recursosActuales) THEN torre.mejorar() AND (level.recursosActuales -= estructura.precioMejora)

Objetivo: Seleccionar ataque especial

Regla funcional:

- IF HUD.iconoAtaqueEspecial.tiempoDeCarga == 0 AND HUD.iconoAtaqueEspecial.Clicked() THEN HUD.mostrarHUDAtaqueEspecial()

Objetivo: Disparar ataques especiales

Regla funcional:

- IF HUD.iconoAtaqueEspecial.seleccionado AND level.posicion.Clicked() THEN
dispararAtaqueEspecial() AND
HUD.iconoAtaqueEspecialSeleccionado.tiempoDeCarga == 100

Reglas de la torre

Objetivo: Disparar

Regla funcional:

- IF torre.tieneObjetivo() AND torre.orientada THEN torre.disparar()

Objetivo: Localizar objetivo

Regla funcional:

- IF torre.rango.contieneBicho() THEN torre.objetivo = bicho

Objetivo: Orientarse

Regla funcional:

- IF torre.tieneObjetivo THEN torre.apuntar()

Reglas del bicho

Objetivo: Orientarse

Regla funcional:

- IF bicho.colisionConObjetivo() THEN bicho.apuntar()

Objetivo: Disparar

Regla funcional:

- IF bicho.orientado THEN bicho.disparar()

Objetivo: Avanzar

Regla funcional:

- IF !bicho.colisionConObjetivo AND !bicho.colisionConBicho THEN
bicho.avanzar()

Objetivo: Colisión bicho-bicho en distinta ruta

Regla funcional:

- IF bicho1.colisionConBicho AND bicho1.ruta != bicho2.ruta THEN
bicho1.jumpOver() AND bicho2.avanzar()

Objetivo: Colisión bicho-bicho en misma ruta

Regla funcional:

- IF bicho1.colisionConBicho AND bicho1.ruta == bicho2.ruta THEN bicho2.wait()

Objetivo: Colisión bicho-torre

Regla funcional:

- IF bicho.colisionConTorre THEN bicho.apuntar() AND bicho.disparar()

Reglas del HUD

Objetivo: Poner vista torre

Regla funcional:

- IF torre.seleccionada() AND botonCambioCamara.seleccionado() THEN camara.setCamara(torreSeleccionada)

Objetivo: Poner vista bicho

Regla funcional:

- IF bicho.seleccionado() AND botonCambioCamara.seleccionado() THEN camara.setCamara(bichoSeleccionada)

Objetivo: Poner vista genérica

Regla funcional:

- IF (camara.getCamara() == torre OR camara.getCamara() == bicho) AND botonCambioCamara.seleccionado() THEN camara.setCamara(genérica)

Reglas del mapa

Objetivo: Seleccionar nivel

Regla funcional:

- IF nivel.desbloqueado() AND nivel.seleccionado() THEN nivel.empezar()

Objetivo: Volver al menú

Regla funcional:

- IF botonMenu.disponible AND botonMenu.seleccionado() THEN mapa.disable() AND menu.enable()

Objetivo: Entrar al research

Regla funcional:

- IF botonResearch.disponible() AND botonResearch.seleccionado() THEN mapa.background() AND research.frontground()

Reglas del research

Objetivo: Elegir nivel de mejora

Regla funcional:

- IF mejora.disponible() AND jugador.getXP() >= mejora.coste() AND mejora.seleccionada() THEN jugador.reducirXP(mejora.coste()) AND mejora.enable()

Objetivo: Confirmar nivel de mejora

Regla funcional:

- IF botonDone.seleccionado() THEN research.save() AND research.close() AND mapa.frontground()

Objetivo: Cancelar mejora

Regla funcional:

- IF botonCancel.seleccionado() THEN mejora.disable() AND research.close() AND mapa.frontground()

Reglas de configuración

Objetivo: Reanudar

Regla funcional:

- IF nivel.getEstado() == pausado AND botonReanudar.seleccionado() THEN nivel.setEstado(corriendo)

Objetivo: Salir

Regla funcional:

- IF botonSalir.seleccionado() THEN nivel.close() AND mapa.frontground()

Objetivo: Reiniciar

Regla funcional:

- IF botonReiniciar.seleccionado() THEN nivel.empezar()