

PRUEBA DE ACEPTACIÓN

Proyecto: Bad Ice Cream

Objetivo de la prueba de aceptación

Validar que la aplicación cumple con los **requisitos funcionales y de diseño definidos**, considerando el comportamiento real implementado del sistema, incluyendo validaciones, mensajes de error, control por teclado, visualización del estado del juego y finalización por tiempo.

Escenario 1: Selección correcta del modo de juego (caso exitoso)

Dado que el usuario se encuentra en el menú principal
Cuando selecciona un modo de juego (Player vs Player o Player vs IA)
Entonces el sistema permite continuar con la configuración del juego
Y habilita las opciones correspondientes según el modo seleccionado

✓ Resultado esperado: el sistema permite avanzar correctamente.

Escenario 2: Selección de perfil IA sin modo de juego (caso fallido)

Dado que el usuario se encuentra en el menú principal
Cuando intenta seleccionar un perfil de IA sin haber elegido previamente un modo de juego
Entonces el sistema muestra una ventana de error
Y el mensaje indica que primero debe seleccionarse un modo de juego

✓ Resultado esperado: el sistema bloquea la acción y notifica al usuario.

Escenario 3: Validación del nombre del jugador humano (caso fallido)

Dado que el usuario está configurando un jugador humano
Cuando deja el campo de nombre vacío
Y intenta iniciar el juego
Entonces el sistema muestra una ventana de error
Y el mensaje indica que el nombre del jugador es obligatorio

✓ Resultado esperado: no se inicia el juego sin nombre válido.

Escenario 4: Configuración inválida de nivel (caso fallido)

Dado que el usuario se encuentra configurando el nivel
Cuando selecciona menos de 2 frutas

O selecciona menos de 1 enemigo
Entonces el sistema impide iniciar el juego
Y muestra una ventana indicando los valores mínimos requeridos

✓ Resultado esperado: el sistema valida correctamente la configuración.

Escenario 5: Inicio correcto del juego con configuración válida (caso exitoso)

Dado que el usuario ha configurado correctamente el tipo de juego (player vs player) y en humano ingreso el nombre
Y ha seleccionado mínimo 2 frutas y 1 enemigo
Y ha definido jugadores válidos
Cuando presiona el botón de iniciar juego
Entonces el sistema inicia el nivel
Y muestra el escenario gráfico
Y se visualiza el HUD con tiempo, puntaje y estado del juego

✓ Resultado esperado: el juego inicia sin errores.

Escenario 6: Movimiento del Jugador 1 mediante teclado (caso exitoso)

Dado que el juego está en ejecución
Cuando el usuario presiona las teclas de dirección

- ↑ Norte
- ↓ Sur
- ← Oeste
- → Este

Entonces el helado del jugador 1 se mueve en la dirección correspondiente

✓ Resultado esperado: el movimiento se ejecuta correctamente.

Escenario 7: Movimiento del Jugador 2 mediante teclado (caso exitoso)

Dado que el juego está en ejecución
Cuando el usuario presiona las teclas

- W (Norte)
- S (Sur)
- A (Oeste)

- D (Este)
Entonces el helado del jugador 2 se mueve correctamente

✓ Resultado esperado: ambos jugadores responden al teclado.

Escenario 8: Creación y destrucción de bloques de hielo (caso exitoso)

Dado que el juego está activo
Cuando el jugador 1 presiona la tecla space
Entonces se crea un bloque de hielo
Cuando presiona la tecla .
Entonces el bloque de hielo es destruido

Y para el jugador 2:

- Q crea hielo
- E destruye hielo

✓ Resultado esperado: las acciones especiales funcionan según las teclas asignadas.

Escenario 9: Interacción con frutas (caso parcialmente exitoso)

Dado que el helado colisiona con una fruta
Entonces el puntaje del jugador aumenta
Pero la fruta permanece visible en el escenario

✓ Resultado esperado: el puntaje se incrementa correctamente, aunque la fruta no desaparece.

Escenario 10: Interacción con enemigos (caso limitado)

Dado que el enemigo está presente en el escenario
Cuando el helado colisiona con el enemigo
Entonces no se produce daño al jugador
Y el enemigo permanece estático

✓ Resultado esperado: el sistema mantiene estabilidad, aunque el comportamiento del enemigo es limitado.

Escenario 11: Visualización del estado del juego (caso exitoso)

Dado que el juego está en ejecución
Entonces el sistema muestra claramente:

- Tiempo restante
- Puntaje
- Información del juego en el HUD

✓ Resultado esperado: la información es visible y comprensible.

Escenario 12: Finalización del juego por tiempo agotado (caso exitoso)

Dado que el tiempo del nivel llega a cero

Entonces el sistema finaliza el juego

Y muestra un mensaje de derrota

✓ Resultado esperado: el juego termina de forma controlada.

Escenario 13: Selección del modo de juego IA vs IA (caso exitoso)

Dado que el usuario se encuentra en el menú principal

Cuando selecciona el modo de juego *Máquina vs Máquina (IA vs IA)*

Entonces el sistema permite continuar con la selección del perfil de IA

Y no solicita nombre ni controles de teclado para jugadores humanos

✓ Resultado esperado: el sistema reconoce correctamente el modo IA vs IA.

Escenario 14: Configuración de jugadores IA (caso exitoso)

Dado que el usuario ha seleccionado el modo IA vs IA

Cuando selecciona los perfiles de IA para ambos jugadores

Entonces el sistema crea dos jugadores de tipo máquina

Y asigna a cada uno su comportamiento de IA correspondiente

✓ Resultado esperado: ambos jugadores son controlados por la IA.

Escenario 15: Inicio del juego en modo IA vs IA (caso exitoso)

Dado que el modo de juego seleccionado es IA vs IA

Y la configuración del nivel es válida

Cuando el usuario inicia el juego

Entonces el sistema inicia el nivel automáticamente

Y los helados comienzan a moverse sin intervención del teclado

✓ Resultado esperado: el juego se ejecuta de forma autónoma.

Escenario 16: Movimiento automático de los helados controlados por IA

Dado que el juego está en ejecución en modo IA vs IA
Cuando el ciclo de actualización del juego se ejecuta
Entonces cada helado controlado por IA calcula su movimiento
Y se desplaza según la estrategia definida en su perfil de IA

✓ Resultado esperado: los helados se mueven usando su lógica de IA.

Escenario 17: Recolección de frutas en modo IA vs IA (caso parcialmente exitoso)

Dado que un helado controlado por IA colisiona con una fruta
Entonces el sistema incrementa el puntaje correspondiente
Pero la fruta permanece visible en el escenario

✓ Resultado esperado: el puntaje se incrementa correctamente.

Escenario 18: Interacción entre IAs (caso limitado)

Dado que ambos jugadores están controlados por IA
Cuando se encuentran en el mismo escenario
Entonces no se produce colisión perjudicial entre ellos
Y continúan ejecutando sus movimientos de forma independiente

✓ Resultado esperado: coexistencia estable de ambas IAs.

Escenario 19: Finalización del juego en modo IA vs IA por tiempo agotado

Dado que el juego está en modo IA vs IA
Cuando el tiempo del nivel se agota
Entonces el sistema finaliza el juego
Y muestra el mensaje de derrota

✓ Resultado esperado: finalización correcta sin intervención del usuario.

Escenario 20: Victoria del juego (caso no implementado)

Dado que se cumplen las condiciones de victoria
Entonces el sistema no muestra mensaje de victoria

✓ Resultado observado: funcionalidad no implementada en esta versión.

Escenario 21: Pausar y terminar el juego (caso parcialmente implementado)

Dado que el usuario presiona el botón de pausa o salir

Entonces el sistema permite detener o finalizar la ejecución

Pero guardar y abrir partida están presentes en la interfaz y no funcionales

✓ Resultado esperado: los botones existen y no generan fallos.

Manejo de excepciones y registro de errores

Dado que ocurre una configuración inválida o una acción no permitida

Entonces el sistema lanza una excepción propia

Y registra el error en el log para los programadores

Y notifica al usuario mediante una ventana emergente

✓ Resultado esperado: errores controlados y registrados.