

## Reporte Sprint #4

Implementa todas las características que permiten a un jugador (humano o computadora) jugar un juego SOS simple o general contra otro jugador (humano o computadora). Las características mínimas incluyen elegir humano o computadora para jugadores rojos y/o azules, elegir el modo de juego (simple o general), elegir el tamaño del tablero, configurar un nuevo juego, hacer un movimiento (en un juego simple o general) y determinar si un juego simple o general ha terminado. El componente informático debe ser capaz de jugar juegos simples y generales completos. Se te anima a considerar estrategias básicas para ganar juegos simples o generales. No se requiere un juego óptimo. El siguiente es un diseño de GUI de muestra. Debes usar una jerarquía de clases para lidiar con los requisitos del oponente de la computadora. Si tu código actual aún no ha considerado la jerarquía de clases, es hora de refactorizar tu código.

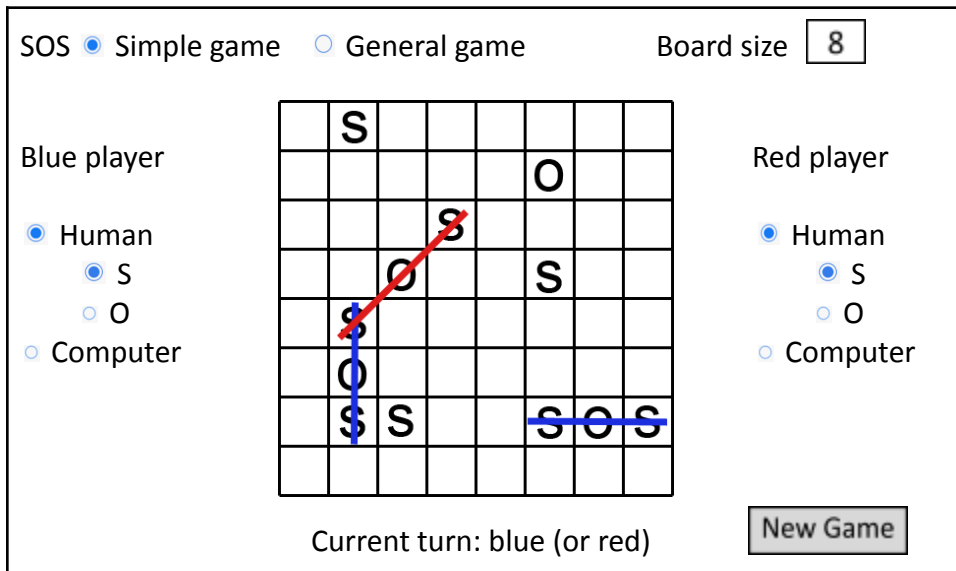


Figura 1. Diseño de GUI de muestra del programa de trabajo para Sprint 3

**Puntos totales: 20**

### 1. Demostración (10 puntos)

Envía un video de no más de cinco minutos, que demuestre claramente que implementaste las funciones del oponente de la computadora y escribiste algunas pruebas de unidad automatizadas.

- 1) Un juego completamente simple donde el jugador azul es un humano, el jugador rojo es la computadora y hay un ganador.
- 2) Un juego general completo donde el jugador azul es la computadora, el jugador rojo es un humano y hay un ganador.
- 3) Un juego simple completo donde ambos lados son jugados por la computadora
- 4) Un juego general completo donde ambos lados son jugados por la computadora.
- 5) Algunas pruebas unitarias automatizadas para el oponente de la computadora.

En el video, debes explicar lo que se está demostrando.

## 2. Historias de usuario para los requisitos del oponente de la computadora (1 punto)

Plantilla de historia de usuario: Como <rol>, quiero <objetivo> [tal que <beneficio>]

ID	Nombre de historia de usuario	Descripción de historia de usuario	Prioridad	Esfuerzo estimado (horas)
7	Iniciar juego con tipo de jugador	Como usuario necesito elegir los tipos de jugador(humano o computadora), el modo y presionar iniciar “Start game” para comenzar hacer el primer movimiento.	3	2
8	Movimiento de la computadora	Como usuario necesito quiero que la computadora sea capaz de jugar automáticamente	3	2

## 3. Criterio de aceptación (AC) para los requisitos del oponente de la computadora (1 punto)

ID y nombre de la historia de usuario	AC ID	Descripción del criterio de aceptación	Estado (completado, por hacer, en progreso)
7 Iniciar juego con tipo de jugador	7.1	AC 7.1 <Iniciar juego> <b>Dado</b> seleccionado el modo de juego y los tipos de jugadores <b>Cuando</b> se el usuario presione el botón para iniciar el juego <b>Entonces</b> el primer jugador podrá hacer su primer movimiento.	completado
	7.2	AC 7.1 <Inicia primero el humano el juego, humano vs computadora> <b>Dado</b> seleccionado el modo de juego y los tipos de jugadores distintos <b>Cuando</b> empiece el juego <b>Entonces</b> el usuario hará su movimiento en el tablero.	completado
	7.3	AC 8.2 <Inicia primero la computador el juego, computadora vs humano> <b>Dado</b> seleccionado el modo de juego y los tipos de jugadores distintos <b>Cuando</b> empiece el juego <b>Entonces</b> la computadora hará un movimiento aleatorio en el tablero;	completado
	7.4	AC 8.3 <Inicia juego entre computadoras> <b>Dado</b> seleccionado el modo de juego y los tipos de jugadores ambos computadoras <b>Cuando</b> empiece el juego <b>Entonces</b> la computadora hará un movimiento aleatorios para los ambos jugadores;	completado
8 Movimiento de la computadora	8.1	AC 8.1 <La computadora hace un movimiento aleatorio> <b>Dado</b> un juego en curso <b>Cuando</b> sea el turno de la computadora <b>Entonces</b> se realizará una jugada en una posición aleatoria del tablero con una letra aleatoria.	completado
	...		

## 4. Resumen de todo el código fuente (1 punto)

Nombre del archivo de código fuente	Código de producción o de prueba	# lineas de ódigo
-------------------------------------	----------------------------------	-------------------

SOSGameBoard	Producción	246
SOSGameConsole	Producción	100
SOSGameGUI	Producción	672
TestSOSGameBoard	Prueba	83
TestSOSGameConsole	Prueba	46
TestSOSGameGUI	Prueba	22
Total		1169

**Debes enviar todo el código fuente para obtener puntos por esta tarea.**

## 5. Código de producción vs Nuevas historias de usuario/Criterios de aceptación (2 puntos)

Resume cómo se implementa cada una de las nuevas historias de usuario/criterios de aceptación en tu código de producción (nombre de clase y nombre de método, etc.)

ID y nombre de la historia de usuario	AC ID	Nombre(s) clases	Nombres(s) métodos	Estado (completo o no)	Notas (opcional)
7 Iniciar juego con tipo de jugador	7.1	GameContent	addContentBottom() startGame.addMouseListener()	completo	Para que se inicie el juego el usuario deberá presionar el botón para iniciar la partida.
	7.2	GameBoard Player()	addMouseListener(Casilla) addContentCasilla(letter) setControl(control)	completo	El usuario dibujara la letra elegida en un casillero vacío
	7.3	GameContent Player()	simulateClick(Casilla, time) setControl(control)	completo	La computadora simulará un click en una casilla vacía después de 0.5 segundos.
	7.4	GameContent Player()	simulateClick(Casilla, time) setControl(control)	completo	
8 Movimiento de la computadora	8.1	SOSGameBoard()	getNumberOfEmptyBoxes() computerPlay()	completo	computerPlay elige una posición aleatoria de entre todas las casillas vacías y pone una S u O (elegida también aleatoriamente) y devuelve un array que contiene la fila, columna y letra de la jugada.

## 6. Pruebas vs Nuevas historias de usuario/Criterio de aceptación (2 puntos)

Resume cómo tu código de prueba (nombre de clase y nombre de método) o las pruebas realizadas manualmente prueban cada una de las nuevas historias de usuario/criterios de aceptación.

6.2 Pruebas manuales que corresponden directamente a algunos criterios de aceptación.

Nombre y ID de la historia usuario	AC ID	Entrada de caso de prueba	Salida esperada	Notas
1	1.1			
	1.2			
	...			
8 Movimiento de la computadora	8.1	El array <b>parameters[]</b> de la función <b>computerPlay()</b>	El efecto de llamar a <b>makePlay()</b> con los valores del array <b>parameters[]</b> debe ser el mismo que el de <b>computerPlay()</b> .	
	...			

### 6.3 Otras pruebas automatizadas o manuales que no correspondan a los criterios de aceptación

Número	Entrada de prueba	Salida esperada	Nombre de clase del código de prueba	Nombre del método del código de prueba
1	La casilla en la posición (0,0) sea diferente de Box.EMPTY después de que la computadora hiciera click en esa casilla	true	SOSGameGUI	simulateClick()

### 7. Presenta el diagrama de clases de tu código de producción y describe cómo la jerarquía de clases en su diseño trata con los requisitos del oponente de la computadora (3 puntos)?

