

### Reporte Sprint #3

Implementen todas las características que permiten a un jugador humano jugar un juego SOS simple o general contra un oponente humano y **refactoricen su código existente si es necesario**. Las características mínimas incluyen 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 siguiente es un diseño de GUI de muestra.

Se requiere el uso de una jerarquía de clases para hacer frente a los requisitos comunes del juego simple y general. Si tu código para Sprint 2 no ha considerado la jerarquía de clases, es hora de refactorizar su código.

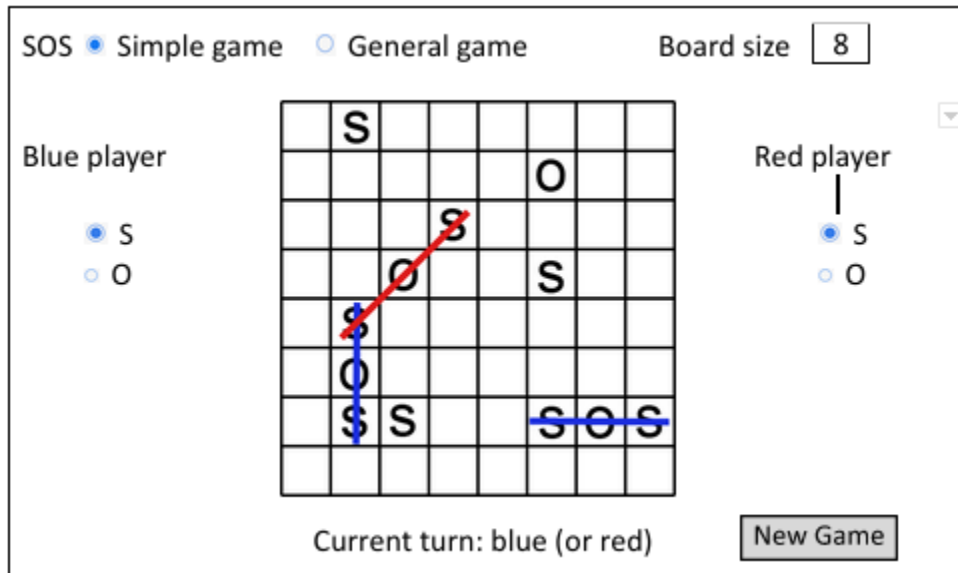


Figura 1. Diseño de GUI de muestra del programa en el Sprint 3

**Entregables: expande y mejora tu entrega para el sprint 2.**

#### 1. Demostración (6 puntos)

Envíen un video de no más de cinco minutos, que demuestre claramente las siguientes características.

- Un juego simple en el que el jugador azul es el ganador.
- Un juego simple empatado con el mismo tamaño de tablero que es 1
- Un juego general en el que el jugador rojo es el ganador y el tamaño del tablero es diferente de 1
- Un juego general empatado con el mismo tamaño de tablero que es 1
- Algunas pruebas unitarias automatizadas para el modo de juego simple
- Algunas pruebas unitarias automatizadas para el modo de juego general

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

#### 2. Resumen del código fuente (2 puntos)

Nombre del archivo de código fuente	¿Código de producción o de prueba?	# líneas de código
GUI	código de producción	173
Tablero	código de producción	85
testTableroVacio	código de prueba	41
testMovimiento	código de prueba	27
testModoDeJuego	código de prueba	19
testGUI	código de prueba	22
Total		

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

### 3. Código de producción vs Historias de usuario/Criterio de aceptación (4 puntos)

Resuman cómo se implementa cada uno de los siguientes criterios de aceptación/historia de usuario en tu código de producción (nombre de clase y nombre de método, etc.)

ID de historia de usuario	Nombre de historia de usuario
1	Escoge el tamaño del tablero
2	Escoge el modo de juego de un tablero escogido
3	Comienza un nuevo juego del tamaño de tablero y del modo de juego elegidos
4	Hacer un movimiento en un juego simple
6	Hacer un movimiento en un juego general

Nombre y ID de la historia usuario	AC ID	Nombre clase(s)	Nombre Método(s)	Estatus (completo o no)	Notas (opcional)
1 Escoge el tamaño del tablero	1.1	Tablero	Tablero	Completo	Se analizará el código para el sprint 4 y en caso sea necesario se refactorizará
	1.2	Tablero	getContenidoDeLasCeldasDelTablero	Completo	Se analizará el código para el sprint 4 y en caso sea necesario se refactorizará
2 Escoge el modo de juego de un tablero escogido	2.1	Tablero	getSelccion	Completo	Se analizará el código para el sprint 4 y en caso sea necesario se refactorizará
	2.2	Tablero	getSelccion	Completo	Se analizará el código para el sprint 4 y en caso sea necesario se refactorizará
3 Comienza un nuevo juego del tamaño de tablero y del	3.1			No iniciado	En el sprint 4 se iniciará con el proceso de la implementación

modo de juego elegidos					
---------------------------	--	--	--	--	--

#### 4. Pruebas vs Historias de usuario/Criterio de aceptación (4 puntos)

Resuman cómo cada uno de los criterios de aceptación/historia de usuario es probado por su código de prueba (nombre de clase y nombre de método) o pruebas realizadas manualmente.

User Story ID	User Story Name
1	Escoge el tamaño del tablero
2	Escoge el modo de juego de un tablero escogido
3	Comienza un nuevo juego del tamaño de tablero y del modo de juego elegidos
4	Hacer un movimiento en un juego simple
6	Hacer un movimiento en un juego general

4.1 Pruebas automatizadas que corresponden directamente a los criterios de aceptación de las historias de usuario anteriores

Nombre y ID de la historia usuario	AC ID	Nombre Clase (s) del código de prueba	Nombre método(s) del código Prueba	Descripción de los casos de prueba (entrada & salida esperada)
1 Escoge el tamaño del tablero	1.1	testTableroVacio		Contiene dos assertEquals. El tamaño esperado para ambos es 4, si el método getNumFilas y getNumColumnas del objeto tablero retorna el mismo valor del tamaño esperado, entonces las pruebas pasarán (estarán en color verde)
	1.2	testTableroVacio	testSizeTableroInvalido	Contiene dos assertEquals. El tamaño esperado para ambos es 1, si el método getNumFilas y getNumColumnas del objeto tablero1 retorna el mismo valor del tamaño esperado, entonces las pruebas pasarán (estarán en color verde)
2 Escoge el modo de juego de un tablero escogido	2.1	testModoDeJuego	testModoDeJuegoSimple	Contiene un assertEquals. El valor esperado es un carácter S, si el método getSeleccion retorna el mismo valor esperado,

				entonces la prueba pasará (estará en color verde)
	2.2	testModoDeJuego	testModoDeJuegoGeneral	Contiene un assertEquals. El valor esperado es un carácter G, si el método getSeleccion retorna el mismo valor esperado, entonces la prueba pasará (estará en color verde)
4 Hacer un movimiento en un juego simple	4.1	testMovimiento	testMovimientoValidoModoSimple	Contiene dos assertEquals. El primero verifica si la casilla está vacía haciendo una comparación del valor esperado y el valor actual y el segundo verifica si el turno le corresponde al jugador A (Azul)

4.2 Pruebas manuales que corresponden directamente a los criterios de aceptación de las historias de usuario anteriores

Nombre y ID de la historia usuario	AC ID	Entrada de caso de prueba	Salida esperada	Notas
1 Escoge el tamaño del tablero	1.1	int 4	int 4	
	1.2	int 1	“Ingrese un tamaño de tablero permitido”	
2 Escoge el modo de juego de un tablero escogido	2.1	char S	char S	
	2.2	char G	char G	
4 Hacer un movimiento en un juego simple	4.1	VACIO char A	VACIO char A	

4.3 Otras pruebas automatizadas o manuales que no corresponden a los criterios de aceptación de las historias de usuario anteriores

Número	Entrada prueba	Resultado esperado	Nombre de clase del código de prueba	Nombre del método del código de prueba
1			testGUI	testTableroVacio

5. Describe cómo la jerarquía de clases en tu diseño trata con los requisitos comunes y diferentes del juego simple y el juego general. (4 puntos)

Por el momento todavía no hemos usado herencia entre las clases que hemos creado (Tablero, GUI), por lo tanto no hay una jerarquía de clases en el diseño presentado en esta primera entrega.