

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.

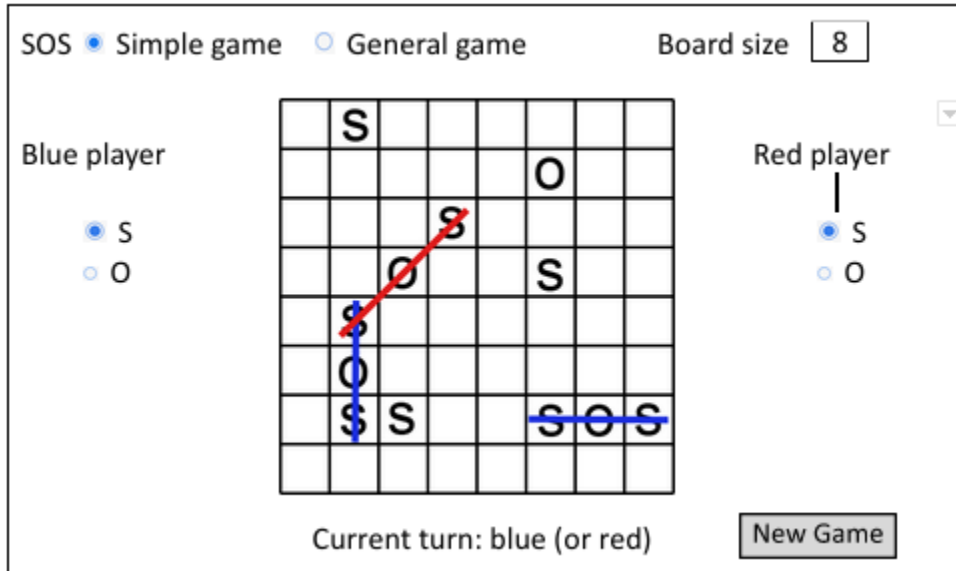


Figura 1. Diseño de GUI de muestra del programa en el Sprint 3

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
JuegoSimple	Producción	246
JuegoGeneral	Producción	159
SosGui	Producción	432
LineaSos	Producción	42
TestComienzaJuego	Prueba	24
TestJuegoGeneral	Prueba	91
TestJuegoSimple	Prueba	69
TestModoJuego	Prueba	26
TestSeleccionaTamano	Prueba	31
Total		1120

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	1.1	JuegoSimple	setTamanoTablero	completo	-
	1.2	JuegoSimple	setTamanoTablero	completo	-
2	2.1	GuiSos	setTipoJuego	completo	-
3	3.1	JuegoSimple	JuegoSimple	completo	-
		JuegoGeneral	JuegoGeneral	completo	-
4	4.1	JuegoSimple	realizarMovimiento	completo	-
	4.2	JuegoSimple	realizarMovimiento	completo	-
5	5.1	JuegoSimple	actualizarEstadoJuego	completo	-
	5.2	JuegoSimple	actualizarEstadoJuego	completo	-
6	6.1	JuegoGeneral	realizarMovimiento	completo	-
	6.2	JuegoGeneral	realizarMovimiento	completo	-
	6.3	JuegoGeneral	realizarMovimiento	completo	-
7	7.1	JuegoGeneral	actualizarEstadoJuego	completo	-
	7.2	JuegoGeneral	actualizarEstadoJuego	completo	-

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	1.1	TestSeleccionaTamano	testSeleccionTamanoValido	Entrada: 10 Salida esperada: 10
	1.2	TestSeleccionaTamano	testSeleccionTamanoInvalido	Entrada: 25 Salida esperada: 8
2	2.1	TestModoJuego	testCambioModoJuegoSimpleAGeneral	Entrada: juego.getTipoJuego().getClass() == JuegoGeneral.class Salida: true
			testCambioModoJuegoGeneralASimple	Entrada: juego.getTipoJuego().getClass() == JuegoSimple.class Salida: true
3	3.1	TestComienzaJuego	testComienzaJuegoSimpleConTamanoSeleccionado,	Entrada: juegoSimple.getCelda(0,0) Salida: JuegoSimple.Celda.S
			testComienzaJuegoGeneralConTamanoSeleccionado	Entrada: juegoGeneral.getCelda(2,2) Salida: JuegoGeneral.Celda.O
4	4.1	TestJuegoSimple	testMovimientoValidoJuegoSimple	Entrada: juego.getTurno() == JuegoSimple.Turno.R OJO Salida: true
	4.2	TestJuegoSimple	testMovimientoInvalidoJuegoSimple	Entrada: juego.getTurno() == JuegoSimple.Turno.R OJO

				Salida: true
5	5.1	TestJuegoSimple	testJuegoSimpleConGanadorAzul	Entrada: juego.getEstadoJuego() Salida: JuegoSimple.EstadoJuego.GANO_AZUL
	5.1	TestJuegoSimple	testJuegoSimpleConGanadorRojo	Entrada: juego.getEstadoJuego() Salida: JuegoSimple.EstadoJuego.GANO_ROJO
	5.2	TestJuegoSimple	testJuegoSimpleConEmpate	Entrada: juego.getEstadoJuego() Salida: JuegoSimple.EstadoJuego.EMPATE
6	6.1	TestJuegoGeneral	testMovimientoQueNoFormeSos	Entrada: JuegoSimple.Celda.S Salida: juego.getCelda(0, 0)
	6.2	TestJuegoGeneral	testMovimientoFormaSos	Entrada: JuegoSimple.Celda.S Salida: juego.getCelda(0, 2)
	6.3	TestJuegoGeneral	testMovimientoNoValido	Entrada: JuegoSimple.Celda.S Salida: juego.getCelda(0, 0)
7	7.1	TestJuegoGeneral	testJuegoGeneralConGanadorAzul	Entrada: juego.getEstadoJuego() == JuegoSimple.EstadoJuego.GANO_AZUL Salida : true
		TestJuegoGeneral	testJuegoGeneralConGanadorRojo	Entrada: juego.getEstadoJuego() == JuegoSimple.EstadoJuego.GANO_ROJO Salida : true
	7.2	TestJuegoGeneral	testJuegoGeneralConEmpate	Entrada: juego.getEstadoJuego() == JuegoSimple.EstadoJuego.EMPATE Salida : true

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	1.1	5	El tablero cambia a tamaño 5	Prueba de la interfaz

				gráfica
	1.2	1	El tablero no cambia de tamaño y muestra mensaje de error	Prueba de la interfaz gráfica
2	2.1	Jugador selecciona el tipo de juego	El juego cambia de modo	Prueba de la interfaz gráfica
3	3.1	Jugador selecciona el tamaño del tablero y tipo de juego.	El tablero cambia al tamaño y modo seleccionado.	Prueba de interfaz grafica.
4	4.1	Jugador azul hace click en celda vacía con 'S' seleccionada	El tablero muestra 'S' en la celda seleccionada y muestra el cambio de turno	Prueba de la interfaz gráfica
	4.2	Jugador rojo selecciona 'O' y hace click en celda ocupada con 'S'	El tablero sigue mostrando 'S' en la celda seleccionada y muestra el cambio de turno	Prueba de la interfaz gráfica
5	5.1	Jugador hace click en casilla y forma SOS en tablero	El tablero muestra una línea sobre el SOS del color del jugador	Prueba de la interfaz gráfica
	5.2	El jugador hace click en última casilla vacía y no se forma SOS	El tablero no se altera	Prueba de la interfaz gráfica
6	6.1	El jugador hace click en una casilla vacía y que no forme SOS	El tablero llena la casilla con la letra seleccionada y muestra el cambio de turno.	Prueba de la interfaz gráfica
	6.2	El jugador hace click en una casilla vacía y que forme SOS	El tablero llena la casilla con la letra seleccionada y no cambia de turno, y se traza el SOS.	Prueba de la interfaz gráfica
	6.3	El jugador hace click en una casilla ocupada	El tablero no cambia de letra ni de turno	Prueba de la interfaz gráfica
7	7.1	Un jugador hace click en la última casilla y el sistema cuenta que el jugador azul tiene mas SOS que el jugador rojo.	El tablero muestra como ganador al jugador azul.	Prueba de la interfaz gráfica
		Un jugador hace click en la última casilla y el sistema cuenta que el jugador rojo tiene mas SOS que el jugador azul.	El tablero muestra como ganador al jugador rojo.	Prueba de la interfaz gráfica
	7.2	Un jugador hace click en la última casilla y el sistema	El tablero muestra empate.	Prueba de la interfaz gráfica

		cuenta que ambos jugadores tienen igual número de SOS		
--	--	---	--	--

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

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)

La clase JuegoGeneral extiende la clase JuegoSimple.

La clase JuegoSimple implementa todos los requisitos para un juego simple.

La clase JuegoGeneral añade los campos numeroSosAzul y numeroSosRojo para contabilizar los puntos de los jugadores ya que en el juego general se requiere que el tablero este lleno y determina al ganador como el que ha formado mas SOS en el tablero o en caso sean iguales se declara empate.

En la clase JuegoGeneral se sobreescribe el método hizoSos para que contabilice los puntos realizados por cada jugador y añada a la lista lineasSos todas las líneas formadas en ese turno.

También se añade el método actualizarPuntos para que actualize los puntos de cada jugador.