

## Reporte del Sprint #5

Las principales tareas de esta asignación son:

- (1) Agrega la función de grabar (record) un juego en un archivo de texto. Se requiere la historia de usuario y los criterios de aceptación tanto de grabación como de reproducción
- (2) Realización de un ejercicio de revisión de código.
- (3) Resumir las lecciones aprendidas del Sprint 0 al Sprint 5.

El siguiente es un diseño de GUI de muestra del producto final, donde "Replay" es opcional.

**El trabajo es de caracter individual.**

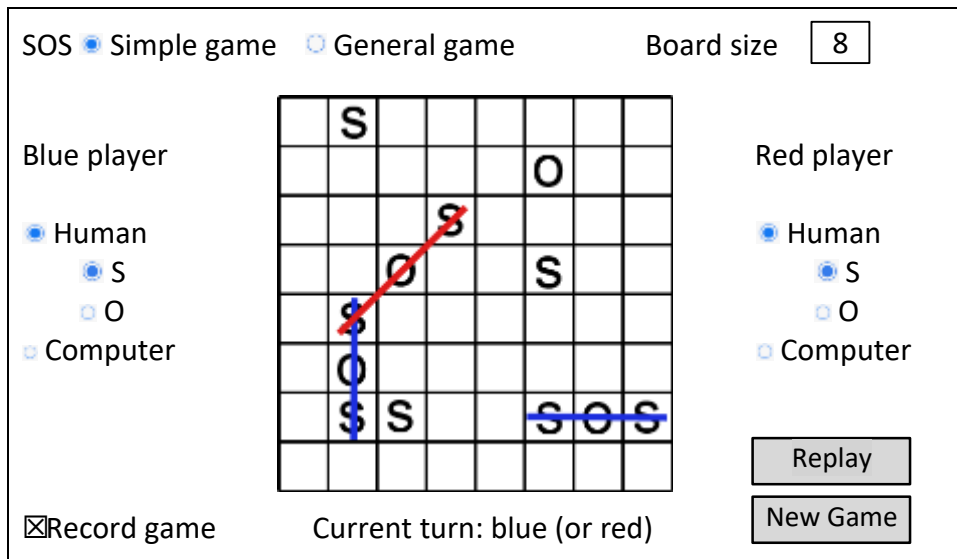


Figura 1. Sample GUI layout of the final product Diseño de GUI del producto final

### Puntos totales

#### 1. Demostración (10 puntos)

Envía un video de no más de 15 minutos, demostrando claramente que has implementado todas las funciones en la siguiente tabla. En el video, debes explicar lo que se está demostrando. **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.**

	Feature
1	Se graba un juego simple completo de dos jugadores humanos.
2	Se graba un juego general completo de dos jugadores humanos
3	Se graba un juego simple completo de jugadores humano-computadora
4	Se graba un juego general completo de jugadores humano-computadora
5	Se graba un juego simple completo de jugadores computadora-computadora
6	Se graba un juego general completo de jugadores computadora-computadora

--	--

Si has implementado la función de "replay" para obtener crédito adicional, debes incluir tu demostración en el video.

## 2. Historias de usuario y criterios de aceptación para los requisitos para los requerimientos Record/Replay (1 punto)

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

Agrega o elimina filas si es necesario

ID	Nombre de historia de usuario	Descripción de historia de usuario	Prioridad	Esfuerzo estimado (horas)
1	Selección del tamaño del tablero	Como jugador requiero poder escoger, en el size board, el número de casillas que tendrá el tablero donde jugaré de tal forma que pueda visualizar el tablero completo.	1	4h
2	Elección del modo de juego	Como jugador necesito elegir el modo de juego para poder iniciar la partida	1	4h
3	Inicia el juego nuevo con el tamaño de tablero elegido	Como jugador requiero poder iniciar un juego nuevo con el tablero elegido tal que pueda probar el juego	1	5h
4	Hacer un movimiento en un juego simple	Como jugador requiero poder hacer un movimiento en el juego modo simple para poder colocar las letras S u O en una casilla vacía.	1	8h
5	Asignación de turno en el juego simple	Como jugador necesito saber cuando es mi turno de colocar S u O en una casilla vacía en el modo de juego simple	2	8h
6	Un juego simple ha terminado	Como jugador necesito saber cuándo he terminado un movimiento simple de tal forma que pueda determinar cuándo ha terminado mi turno.	2	8h
7	Mostrar turno del jugador	Como jugador requiero saber cuándo es mi turno porque así puedo realizar un movimiento.	2	6h
8	Elección de letras S u O	Como jugador requiero elegir entre las letras S u O tal que pueda colocarlas en el tablero.	2	4h
9	Formación de la palabra SOS	Como jugador requiero poder visualizar la palabra SOS rayada con una línea cuando se haya formado en el tablero porque necesito saber la cantidad de palabras SOS que llevo.	3	12h
10	Conteo de palabras SOS	Como jugador requiero ver en el tablero cuantas palabras SOS llevo formadas de esta forma podré saber si puedo ganar el juego.	3	8h
11	Hacer un movimiento en el juego general	Como jugador requiero poder hacer un movimiento en el juego modo general para poder colocar las letras S u O en una casilla vacía.	4	8h

12	Asignación de turnos en un juego general	Como jugador necesito una asignación de turno para poder hacer el siguiente movimiento.	4	24h
13	Un juego general ha terminado	Como jugador necesito que el tablero se llene para poder terminar la partida.	4	8h
14	Asignación de un ganador en un juego general	Como jugador necesito tener el mayor número de SOS para ganar, de tener igual número de SOS se da el empate o de tener menor número de SOS pierdo la partida.	4	8h
15	Visualización del record	Como jugador necesito poder ver el mi partida grabada para poder determinar si es que las fichas que coloqué son correctas	4	3h

ID y nombre de la historia de usuario	AC ID	Descripción del criterio de aceptación	Estado (completado, por hacer, en progreso)
1. Selección del tamaño del tablero	1.1	AC 1.1< Tamaño del tablero permitido> Dado un jugador cuando ingresa un número entero n mayor que 2 en el size board. Entonces el sistema crea el tablero de nxn para el juego.	COMPLETADO
	1.2	AC 1.2 <Tamaño de tablero no permitido> Dado un jugador ingresa en el size board un tamaño no permitido Cuando ingresa dicho orden de casillas Entonces el sistema no crea un juego nuevo y muestra un mensaje donde se especifica que el orden del tablero debe ser mayor o igual a 2.	COMPLETADO
2. Elección del modo de juego	2.1	AC 2.1<Modo de juego simple> Dado un jugador que elige el modo de juego simple, cuando se elige el modo de juego simple, entonces el sistema crea una partida de juego en el modo simple.	COMPLETADO
	2.2	AC 2.2<Modo de juego general> Dado un jugador que elige el modo de juego general, cuando se elige el modo de juego general, entonces el sistema crea una partida de juego en el modo general.	COMPLETADO
3. Inicia el juego con el tamaño de tablero elegido	3.1	AC 3.1 <Inicio de juego de modo simple y tamaño de tablero elegido> Dado un jugador que ingresó un tamaño de cuadrícula válido y seleccionó el modo de juego simple, cuando se elige un tamaño de tablero y modo de juego simple, entonces el sistema muestra el tablero con el tamaño ingresado para que inicie el juego y se registrará bajo las reglas del modo de juego simple	COMPLETADO
	3.2	AC 3.2 <Inicio de juego de modo general y tamaño de tablero elegido> Dado un jugador que ingresó un tamaño de cuadrícula válido y seleccionó el modo de juego general, cuando se elige un tamaño de tablero y modo de juego general, entonces el sistema muestra el tablero con el tamaño ingresado para que inicie el juego y se registrará bajo las reglas del modo de juego general.	COMPLETADO
			COMPLETADO

4. Hacer un movimiento en un juego simple	4.1	AC 4.1 <Hacer un movimiento válido> Dado un símbolo elegido, cuando se elige una celda vacía entonces el símbolo se inserta en dicha celda y si no se ha formado en hilo la palabra SOS entonces se asigna el turno al siguiente jugador.	
	4.2	AC 4.2<Hacer un movimiento inválido> Dado un símbolo elegido, cuando se elige una celda no vacía, entonces no se debería reemplazar el símbolo previo y debe mostrarse un mensaje que diga movimiento no válido.	COMPLETADO
5. Asignación de turno en el juego simple	5.1	AC 5.1<Turno de un jugador> Dado un jugador, cuando coloca en su turno S u O y todavía no se ha formado SOS, entonces se asigna el turno al otro jugador.	COMPLETADO
6.Un juego simple ha terminado	6.1	AC 6.1<Fin de un juego simple por formar SOS> Dado un movimiento válido, cuando este logra formar SOS de forma horizontal, vertical o diagonal, entonces la partida termina y gana el jugador del turno actual.	COMPLETADO
	6.2	AC 6.2<Finde un juego simple por empate> Dado un movimiento válido, cuando este sea el que llene la última celda libre del tablero y no logre formar un SOS, entonces la partida termina y se obtiene un empate.	COMPLETADO
7.Mostrar turno del jugador	7.1	AC 7.1<Turno del jugador en modo simple> Dado un jugador X en modo de juego simple, cuando sea su turno de realizar un movimiento, entonces el sistema debe mostrar un mensaje “turno de X”.	COMPLETADO
	7.2	AC 7.2<Turno del jugador en modo general> Dado un jugador X en modo de juego general, cuando sea su turno de realizar un movimiento, entonces el sistema debe mostrar un mensaje “turno de X”.	COMPLETADO
8. Elección de letras S u O	8.1	AC 8.1<Elegir S u O en modo simple> Dado un jugador en modo de juego simple, cuando es su turno, entonces puede elegir entre S u O para colocarlo en el tablero en una casilla vacía.	COMPLETADO
	8.2	AC 8.1<Elegir S u O en modo general> Dado un jugador en modo de juego general, cuando es su turno, entonces puede elegir entre S u O para colocarlo en el tablero en una casilla vacía.	COMPLETADO
9. Formación de la palabra SOS	9.1	AC 9.1<Formar SOS en modo simple> Dado un jugador “Azul” en el modo de juego simple, cuando realiza un movimiento, entonces el sistema verifica si se formaron hilos SOS, de formarse entonces el sistema mostrará un línea de color Azul superpuesta a cada hilo que se haya formado en el turno actual.	COMPLETADO
	9.2	AC 9.2<Formar SOS en modo general> Dado un jugador “Azul” en el modo de juego general, cuando realiza un movimiento, entonces el sistema verifica si se formaron hilos SOS, de formarse entonces el sistema mostrará un línea de color Azul superpuesta a cada hilo que se haya formado en el turno actual.	COMPLETADO

10. Conteo de palabras SOS	10.1	AC 10.1<Conteo de SOS en modo simple> Dado un jugador "Azul" en modo de juego simple, cuando termine su turno, entonces el sistema mostrará el número de SOS que ha formado hasta su último turno.	COMPLETADO
	10.2	AC 10.1<Conteo de SOS en modo general> Dado un jugador "Azul" en modo de juego general, cuando termine su turno, entonces el sistema mostrará el número de SOS que ha formado hasta su último turno.	COMPLETADO
11. Hacer un movimiento en el juego general	11.1	AC 11.1<Hacer un movimiento válido> Dado un símbolo elegido en el modo de juego general, cuando se elige una celda vacía, entonces el símbolo se inserta en dicha celda	COMPLETADO
	11.2	AC 11.2 <Hacer un movimiento inválido> Dado un símbolo elegido en el modo de juego general, cuando se elige una celda no vacía, entonces no se debería reemplazar el símbolo previo y debe mostrarse un mensaje que diga movimiento no válido.	COMPLETADO
12. Asignación de turnos en un juego general	12.1	AC 12.1 <Turno al otro jugador> Dado un movimiento válido, cuando este no forma una SOS y todavía no se ha llenado el tablero, entonces se asigna el turno al siguiente jugador.	COMPLETADO
	12.2	AC 12.2<Repetición de turno> Dado un movimiento válido, cuando éste forme SOS y todavía no se ha llenado el tablero, entonces el jugador repite otro turno.	COMPLETADO
13. Un juego general ha terminado	13.1	AC 13.1 <Fin de un juego general> Dado un movimiento válido, cuando éste llena la última celda vacía, entonces se termina el juego.	COMPLETADO
14. Asignación de un ganador en un juego general	14.1	AC 14.1 <Asignación de un ganador> Dada una partida terminada, cuando se contabilicen el número de SOS de cada jugador, entonces el que tenga un mayor número gana, si tienen el mismo número empatan.	COMPLETADO
15. Visualización del record	15.1	AC 15.1 <Visualización del record> Dada una partida finalizada Cuando se determinó un empate o un ganador Entonces puedo ver mi partida grabada en un archivo de texto	COMPLETADO

### 3. Revisión de código (4 puntos)

Aplica la revisión del código fuente a una o dos de las clases más importantes (y a otras clases si el tiempo te permite) e informa de los resultados. Además de buscar errores, la revisión debe verificar: (1) si todo el proyecto ha seguido el estándar de codificación de manera consistente, (2) si el proyecto ha seguido los principios de diseño presentados en clase y (3) si hay olores de código que indican la necesidad de refactorización.

Las siguientes listas de verificación proporcionan pautas básicas. Puedes agregar nuevos elementos a cada una de las listas de verificación. Asegúrate de que tus respuestas sean el resultado del ejercicio de revisión del código. Si no hay hallazgos para una entrada, debes proporcionar una explicación. Por ejemplo, si tu respuesta a "¿Se violan las convenciones de nomenclatura?" es no, debes describir una convención de nomenclatura y

presentar un ejemplo. No recibirás puntaje por si tus respuestas son simplemente sí o no sin información adicional.

Clases que han sido revisadas:

- Tablero
- Semi\_IA
- GUI\_SOS
- Menu
- Record

Fecha/hora de duración del ejercicio de revisión del código:

- 06/06/2023 8:00 a.m Tablero
- 06/06/2023 10:00 a.m Semi\_IA
- 06/06/2023 2:00 p.m GUI\_SOS
- 07/06/2023 2:00 p.m Menu
- 07/06/2023 6:00 p.m Record

Checklist	Items Checklist	Conclusiones
Estándares de codificación	Convenciones de nombres	Cada variable y método se explica de buena manera
	Convención de ordenación de argumentos de método	En todos los métodos que tienen como parámetro de entrada filas y columnas, se mantiene en el mismo orden. Y si se requiere un parámetro extra, este se agrega en la parte de derecha.
	Comentarios significativos y válidos.	Cada método tiene su explicación y los test tienen sus respectivos comentarios.
	Estilo consistente de bloques de código	Cada bloque de código tiene un tamaño similar a excepción del método FormacionDelSOS()
	Indentación consistente	Toda la logical implementada se entiende de forma correcta.
Principio de diseño	...	
	Clase o método no bien modularizado	
	Visibilidad adecuada de cada variable, método y clase.	Cada variable se interpreta y visualiza de buena manera. Una la variable que puede resultar confusa es ContenidoDeCeldas; sin embargo, si se observa con detalle se puede entender que se refiere a las fichas que contendrá el tablero.
	Alguna clase con pobre abstracción	La clase tablero, se trató de diseñarla de la forma más abstracta posible, algunas falencias que puede tener es que no se pudo implementar más de una capa de abstracción, solamente tiene dos niveles: la capa que se usa en la GUI y la capa lógica.
	Diseño por contrato ( pre/postcondiciones)	Si se podría haber implementado el diseño por contrato a algunos métodos que ya existían desde sprint anteriores; para este sprint no se utilizó.
	¿Se viola el Principio Abierto-Cerrado?	No, el método tablero permite poder implementar muchos de los métodos que tiene, pero no permite modificarse ningún atributo importante.
Smells código	¿Se viola el Principio de Responsabilidad Única <sup>1</sup> ?	No, tanto la clase tablero, como las GUI's implementadas y la clase de Semi_IA hacen exactamente lo que se requiere, ninguna se traslapa con otras funciones de otra clase.
	Números mágicos	En la implementación de la Victoria del GUI_SOS cuando se juega contra la computadora, el estado del juego se mantiene en PLAYING aun cuando se ha determinado un ganador; sin embargo, el Label que muestra al ganador, lo muestra correctamente. No sé sabe con exactitud por qué pasa eso.

<sup>1</sup> Revisa: [Violation solution for single responsibility principle](#)

	Variable global /clase innecesaria	No se implementaron variables globales, todas las variables se trataron de mantener en un entorno privado.	
	Código duplicado	A excepción del método FormacionDelSOS(), en el resto de código no hay código duplicado.	
	Métodos largos	A excepción del método FormacionDelSOS(); los métodos de todas las clases se mantienen lo más cortos posibles.	
	Larga lista de parámetros	En la clase tablero se pudo haber reducido el número atributos si se hubiera implementado una clase extra llamada jugador, debido a que se tendría que refactorizar completamente todo el código se evitó tomar esta decisión.	
	Expresión demasiado compleja	Todas las expresiones se mantienen simples, ninguna esta fuera de lo común.	
	Switch o if-then-else que necesita ser reemplazado con polimorfismo	Esto está presente en la lógica de la formación del SOS, no se supo cómo implementar una forma más eficiente de resolver el problema a excepción de usar 16 if's.	
	Nombre de método o variable cuya intención no está clara	Todas las variables y métodos estan claros, ninguno tiene una función ambigua	
	¿Algún método similar en otras clases?	No hay métodos similares en otras clases, el único método que está haciendo sobrecargado es RealizarUnMovimiento() en la clase Semi_IA	
	...		
<b>Errores</b>	<b>Fragmento de código con errores</b>	<b>¿Cuál es el error?</b>	<b>¿Por qué es un error?</b>
	Todo el metodo FormacionDelSOS()	Es excesivamente largo e ineficiente	Se podira haber solucionado utlizando algún principio de refactorización o aplicando alguna técnica, por la forma en la que está resuelta resulta ser ineficiente.

#### 4. Resumen de todo el código (1 points)

Nombre del archivo de código fuente	Código de producción o prueba?	# líneas de código
Tablero	testFormacionDelSos	254
Semi_IA	testFormacionDelSos	84
GUI_SOS	testFormacionDelSos	238
Main	-	71
Racord	-	54
Total de líneas de código		701

**No recibirás puntaje por esta tarea a menos que envíes tu código fuente completo.**

5. Resume las lecciones aprendidas de todo el proyecto respondiendo las siguientes preguntas desde la perspectiva de los procesos de desarrollo, codificación, diseño, refactorización y prueba (**4 puntos**):

- ¿Qué ganaste personalmente con el proyecto?  
Aprendí a como poder desarrollar de forma eficiente un proyecto, siento que, si hubiera empezado de forma distinta, el resultado que obtuve no hubiera sido el deseado o al menos como a mí me hubiera gustado.  
Entendí como poder aplicar todo lo visto en clase con este proyecto, si bien, hubieron cosas que no se pudieron implementar porque estaba muy avanzado el proyecto, habían funciones que ya se habían probado con los test pero por no tener algunos conocimientos previos se resolvieron de forma muy lenta, puedo visualizar como poder desarrollar todo desde cero utilizando diseño de contratos, los principios SOLID, los distintos tipos de test que vimos, etc.  
Fue una experiencia muy enriquecedora y divertida de realizar.
- ¿Qué hace bien tu proyecto y qué podría hacer mejor tu proyecto?  
Mi proyecto funciona, cumple con los requisitos mínimos de un juego SOS convencional determina de forma exitosa a un ganador tanto en un juego simple como en un juego general, dibuja de forma correcta el tablero de juego de acuerdo a lo que el usuario desee ingresar como tamaño, las letras S u O se dibujan de forma correcta y se colocan en las casillas con las que el usuario interactúe, lógicamente se crea un tablero virtual en el cual se establecen todas las reglas ya mencionadas.  
Podría mejorar enormemente en la GUI, haciendo un diseño mucho más estilizado y arreglando los distintos smell code que se pueden encontrar, se podrían implementar más modalidades juegos, con respecto al código, se podría implementar otra clase que se llame Jugador y de esta manera poder separar la lógica del puntaje del jugador que esta implementada en la misma del tablero.
- ¿Cómo podrías mejorar tu proceso de desarrollo si desarrollas un juego similar desde cero?  
Como se había mencionado en la primera pregunta, con los conocimientos mucho más completos y consolidados, podría a empezar a definir primeros las particiones de algunas funcionalidades del juego SOS, de esta forma me evito los problemas que tuve en el sprint 2 en el que no sabía exactamente como trabajar los test's de algunos métodos, estaba un poco en el aire algunas ideas, podría poder trabajar con pruebas de especificación, veo que son el tipo de prueba más completo para este proyecto en específico, de esta forma cubriría todos los casos posibles de alguna situación que suscite en el proyecto.

Requisito mínimo para (5): Una página completa a espacio simple, tamaño de fuente no mayor a 12 puntos