

QuarenTEAM

*Nombre del grupo: **QuarenTEAM***

*Nombre del proyecto: **Collateral***

*Nombre del documento: **Requerimientos***

Integrantes:

Don Enzo Martin

Quinteros Tomás

Coschica Francisco Nicolás

Hidalgo Juan Nahuel

*Fecha: **05/05/2020***

Historial de Cambios

Versión	Fecha	Cambio	Autor
1.0	10/05/2020	Planificación	Grupo
2.0	07/06/2020	Modificaciones	Grupo
3.0	17/06/2020	Modificaciones	Grupo

Índice:

1.	<u>Requerimientos:</u>	
1.1.	<u>Funcionales.....</u>	<u>2</u>
1.1.1.	<u>Listado.....</u>	<u>2</u>
1.1.2.	<u>Descripción.....</u>	<u>2</u>
1.1.3.	<u>Inputs.....</u>	<u>3</u>
1.1.4.	<u>Outputs.....</u>	<u>3</u>
1.2.	<u>No Funcionales.....</u>	<u>4</u>
1.2.1.	<u>Listado.....</u>	<u>4</u>
1.2.2.	<u>Descripción.....</u>	<u>4</u>
2.	<u>Pruebas de sistema y Matriz de trazabilidad</u>	
3.	<u>Casos de Prueba</u>	
	<u>3.1 Casos de prueba de valores inusuales/extremos.....</u>	<u>8</u>
	<u>3.2 Smoke test.....</u>	<u>8</u>
4.	<u>Diagrama de uso</u>	
5.	<u>Diagrama de arquitectura preliminar</u>	

1- Requerimientos

1.1 Funcionales

1.1.1 Listado:

- 1) Pantalla principal
- 2) Documentos
- 3) Sistema de decisión
- 4) Transiciones
- 5) Hud
- 6) Sonido
- 7) Almacenamiento de decisiones
- 8) NPC
- 9) Escenas

1.1.2 Descripción:

- ☐ 1) Pantalla principal: El usuario al acceder, el juego deberá mostrar una pantalla que le proporcione al usuario distintas opciones.
 - ☐ 1.1) El sistema al iniciar deberá inicializar la escena correspondiente a la pantalla principal
 - ☐ 1.2) El sistema deberá inicializar el canvas correspondiente a la interfaz de usuario de la ventana principal
 - ☐ 1.3) La pantalla principal deberá contar con 2 botones (JUGAR - SALIR)
 - ☐ 1.4) El botón JUGAR deberá estar configurado para al ser oprimido cargar la escena juego en la semana 1
 - ☐ 1.5) El botón SALIR deberá estar configurado para al ser oprimido cerrar el juego
- ☐ 2) Documentos: El sistema deberá mostrar diferentes parámetros para la toma de decisiones como así también el protocolo correspondiente a esa semana.
 - ☐ 2.1) El sistema deberá mostrar un DOCUMENTO con información para tomar la decisión
 - ☐ 2.2) El sistema deberá interpretar en qué semana se encuentra y determinar el protocolo a tener en cuenta
 - ☐ 2.3) La información del documento no deberá estar visible hasta que el usuario interactúe con el documento
- ☐ 3) Sistema de decisión: El sistema deberá darle la posibilidad al usuario de optar entre dos decisiones excluyentes entre sí

<Collateral> Requerimientos

- ☐ 3.1) El sistema deberá darle una opción de decisión afirmativa al usuario
- ☐ 3.2) El sistema deberá darle una opción de decisión negativa al usuario
- ☐ 4) Pantalla de transición: El juego deberá mostrar una transición entre cada semana.
 - ☐ 4.1) El sistema deberá cargar la escena correspondiente a la transición al alcanzar los 3 pacientes atendidos por semana
 - ☐ 4.2) En cada transición se deberán mostrar recortes sobre noticias reales
- ☐ 5) HUD: Debe mostrar el desempeño del jugador acorde a la escena en la que se encuentre, deberá existir un botón para cerrar el juego y además deberá darle la posibilidad al usuario de elegir cómo mostrar su desempeño
 - ☐ 5.1) En la escena de juego el sistema deberá mostrar y actualizar acorde a las decisiones que toma el usuario la barra de desempeño, gráfico de torta y texto
 - ☐ 5.2) El sistema deberá cerrar el juego si se presiona el respectivo botón cuando se está en la vista del documento
 - ☐ 5.3) El sistema en la escena juego deberá contar con botones para alternar entre algoritmos (última decisión y promedio) a la hora de mostrar el desempeño
 - ☐ 5.4) En la escena transiciones y final el sistema deberá mostrar y actualizar acorde a las decisiones que toma el usuario el gráfico de torta
 - ☐ 5.5) En la escena transiciones y final el sistema deberá mostrar un botón para salir del juego.
- ☐ 6) Sonido: Deberá reproducir música.
 - ☐ 6.1) El sistema deberá iniciar la música correspondiente a la escena en la que esté el usuario.
- ☐ 7) Almacenamiento de decisiones: El juego almacenará el resultado de nuestras decisiones.
 - ☐ 7.1) El sistema deberá ser capaz de guardar la decisión tomada para la generación y modificación y control del HUD.
 - ☐ 7.2) El sistema deberá ser capaz de determinar si la decisión tomada por el usuario fue correcta.
- ☐ 8) NPC("non playable character", "personaje no jugable") : El juego deberá contar con un personaje el cual trae el documento al usuario, mostrando su recorrido por el hospital.
- ☐ 9) Escenas: El sistema deberá cargar la escena que corresponda correctamente e ir actualizando el número de semanas.
 - ☐ 9.1) El sistema deberá de tener una escena de menú al inicio.
 - ☐ 9.2) El sistema deberá de tener una escena de juego.
 - ☐ 9.3) El sistema deberá de tener una escena de transición.
 - ☐ 9.4) El sistema deberá de tener una escena final con créditos.

1.1.3 Inputs:

- ❑ Todas las entradas serán administradas por el usuario con el ratón.

1.1.4 Outputs:

- ❑ 1)Pantalla principal: Se darán 3 opciones, la primera que permita comenzar el juego, una segunda que permita seleccionar el nivel gráfico y por último una que le permita salir del mismo.
- ❑ 2)Documentos: Estos pueden ser por ejemplo sexo, edad, historia clínica, etc.
- ❑ 3)Sistema de decisión: La decisión será positiva o negativa.
- ❑ 5)Hud: Muestra los documentos a los que se podrá acceder así como los gráficos para mostrar el desempeño del usuario según la escena que corresponda.
- ❑ 6)Sonido: MP3.

1.2 No Funcionales

1.2.1 Listado:

- ❑ 1)Compatibilidad
- ❑ 2)Legibilidad
- ❑ 3)Tiempo de apertura
- ❑ 4)Mantenibilidad

1.2.2 Descripción:

- ❑ 1)Compatibilidad: El juego deberá ser compatible con las plataformas windows.
- ❑ 2)Legibilidad: El juego se deberá adaptar a la resolución de la pantalla del usuario.
- ❑ 3)Tiempo de apertura: Menos de 5 segundos entre la ejecución del juego a mostrar el menú principal.
- ❑ 4)Mantenibilidad: El sistema no deberá superar los valores establecidos. Los cuales son:
 - ❑ Índice de mantenibilidad: Por encima de 50 para todas las clases.
 - ❑ Complejidad ciclomática: No exceder de 20 a nivel de método.
 - ❑ Profundidad de herencia: no deberá ser mayor a 10 ya que 5 es por defecto ya que el uso de Monobehaviour incluye el máximo recomendado como base.
 - ❑ Acoplamiento de clases: Deberá ser menos a 80 a nivel de clase
 - ❑ Líneas de código: Los métodos no deberán tener más de 20 líneas de código.

2- Pruebas de sistema y Matriz de Trazabilidad

A continuación se detallan todas las pruebas del sistema y al final se muestra la matriz de trazabilidad con los mapeos correspondientes a los requerimientos de sistema.

Nombre:	Jugar PS1
Descripción:	Se inicia el juego
Se ejecuta:	Al oprimir el botón JUGAR
Secuencia:	1) Se aprieta el botón JUGAR
Prerrequisitos	Haber ejecutado el ejecutable y encontrarse en la pantalla de inicio
Resultado esperado	La ventana de juego
Información adicional	N/A

Nombre:	Tomar decisión PS2
Descripción:	Se toman las decisiones del juego
Se ejecuta:	Al oprimir una de las dos opciones disponibles
Secuencia:	1) Se aprieta el botón correspondiente a la decisión que creamos es la correcta
Prerrequisitos	Estar en la ventana del juego
Resultado esperado	Decisión guardada y procesada
Información adicional	N/A

<Collateral> Requerimientos

Nombre:	Salir - 1 PS3
Descripción:	Se sale del juego desde pantalla inicial
Se ejecuta:	Al oprimir el botón SALIR
Secuencia:	1) Se aprieta el botón SALIR
Prerrequisitos	Haber ejecutado el ejecutable y encontrarse en la pantalla de inicio
Resultado esperado	Cerrar el juego
Información adicional	N/A

Nombre:	Salir - 2 PS4
Descripción:	Se sale del juego desde ventana juego
Se ejecuta:	Al oprimir el botón SALIR
Secuencia:	1) Se clickea sobre el documento 2) Se oprime el botón SALIR
Prerrequisitos	Encontrarse en la ventana de juego
Resultado esperado	Cerrar el juego
Información adicional	N/A

Nombre:	Salir - 3 PS5
Descripción:	Se sale del juego desde escena final
Se ejecuta:	Al oprimir el botón SALIR
Secuencia:	1) Se aprieta el botón SALIR
Prerrequisitos	Estar en la ventana de créditos
Resultado esperado	Cerrar el juego
Información adicional	N/A

<Collateral> Requerimientos

Nombre:	Iniciar PS6
Descripción:	Se ejecuta y muestra la pantalla inicial
Se ejecuta:	Al iniciar el archivo .exe
Secuencia:	1) Iniciar el archivo .exe
Prerrequisitos	N/A
Resultado esperado	Ver la pantalla inicial con sus opciones a elegir
Información adicional	N/A

Nombre:	Abrir con diferentes versiones de Windows PS7
Descripción:	Se inicia el juego con el SO Windows 7. Luego se inicia el juego con el SO Windows 10
Se ejecuta:	Al iniciar el archivo .exe
Secuencia:	1)Se inicia el archivo .exe en el SO Windows 7
	2)Se inicia el archivo .exe
Prerrequisitos	Disponer de computadoras con ambos sistemas operativos
Resultado esperado	El juego funcione correctamente en ambos Sistemas Operativos
Información adicional	N/A

<Collateral> Requerimientos

Nombre:	Resolución de pantalla PS8
Descripción:	Se comprueba la correcta adaptación del sistema en pantallas con distintas resoluciones
Se ejecuta:	Al iniciar el archivo .exe
Secuencia:	1)Se inicia el archivo .exe
Prerrequisitos	Distintos monitores
Resultado esperado	La interfaz juego se ajusta a la resolución de cada pantalla
Información adicional	N/A

Nombre:	Tiempo de apertura PS9
Descripción:	Se ejecuta el juego y se mide el tiempo que tarda en llegar a la pantalla de Menú Principal
Se ejecuta:	Al iniciar el archivo .exe
Secuencia:	1) Se inicia el archivo.exe 2) Se inicia el cronómetro 3) Se para el cronometro al llegar al Menu Principal
Prerrequisitos	N/A
Resultado esperado	El juego se abre en menos de 5 segundos
Información adicional	N/A

<Collateral> Requerimientos

Nombre:	Transiciones PS10
Descripción:	El sistema debe cargar la escena que corresponda.
Se ejecuta:	Al tomar 3 decisiones
Secuencia:	1) Se clickea documento
	2) Se toman 3 decisiones
	3) Se espera un corto periodo de tiempo
Prerrequisitos	Estar dentro de la ventana de juego
Resultado esperado	Al tomar 3 decisiones el juego deberá cambiar de escena
Información adicional	N/A

Matriz de trazabilidad

[Link matriz de trazabilidad.](#)

3-Casos de Prueba:

3.1 Casos de prueba de valores inusuales/extremos:

Se realizarán distintas actividades:

- 1)Clickear rápidamente de manera consecutiva cualquier opción válida.
- 2)Seleccionar la misma opción consecutivamente.
- 3)Clicks aleatorios en la pantalla.

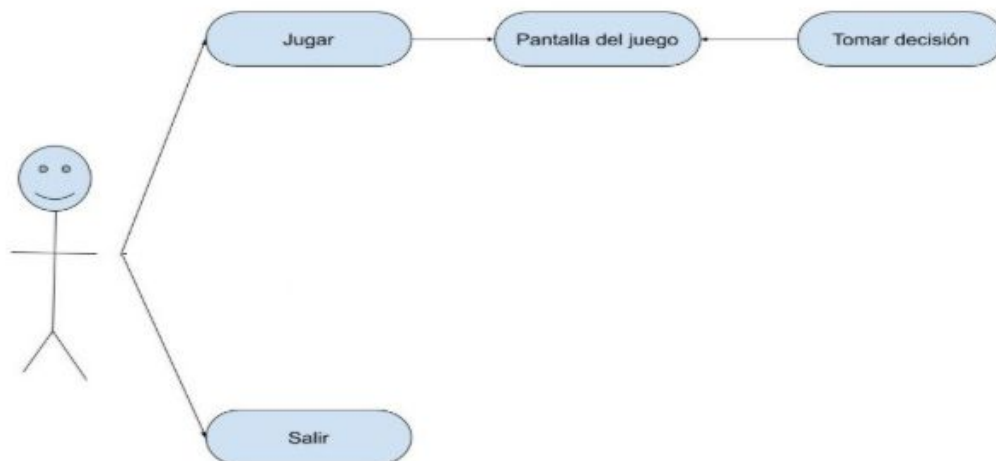
El primer test busca romper el programa mediante algún cruce erróneo de datos, el segundo buscará ver si el juego responde a esto como un error, el tercero intentará ver si el sistema se mantiene inerte ante interacciones aleatorias.

3.2 Smoke Test:

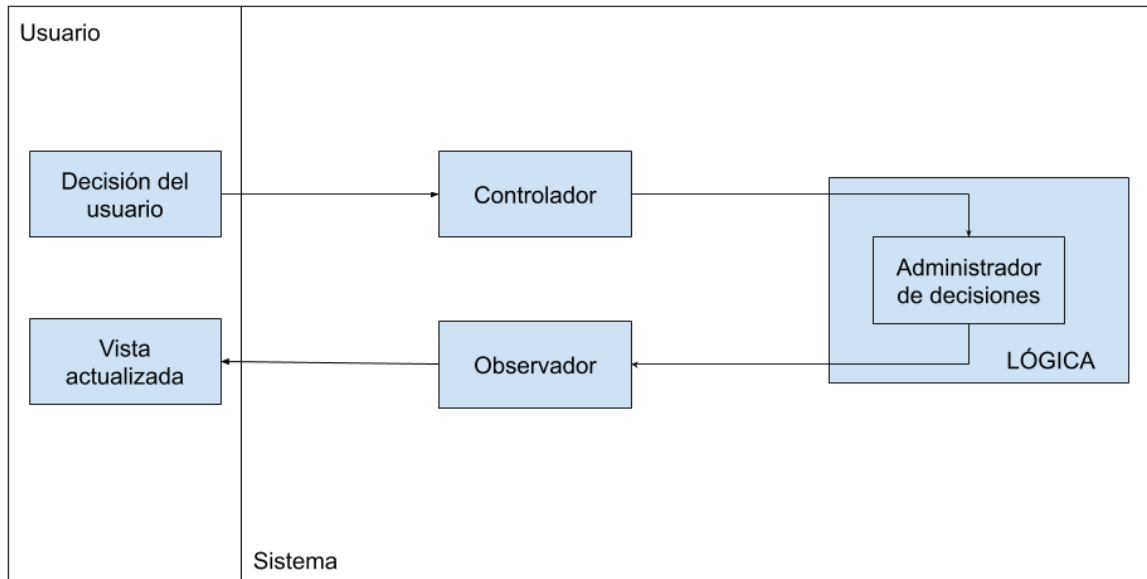
Se ha seleccionado la opción del menú de inicio “Jugar”, “Tomar decisión” y el ejecutable como las pruebas críticas del sistema, sin las cuales el juego no podría ser, por ende sobre estas se testea la funcionalidad de manera continua de nuestro proyecto.

Los casos de uso que actuarían como Smoke serían los test de la matriz de trazabilidad que se corresponden con los números 1, 2, 4, 28, 33, 34, 37 y 42.

4- Diagrama de casos de uso:



5- Diagrama de arquitectura preliminar



Los componentes del diagrama de arquitectura preliminar se corresponden con los siguientes requerimientos.

- Controlador: F1, F2, F3, F4, F5, F6, F8 y F9
- Lógica: F7
- Observador: F5