

# Seguimiento Final

Angélica Lucía Cruz

## Contexto:

El mundo se ve amenazado por la peor enfermedad que amenaza con propagarse por toda la población humana: Flojera aguda. La Organización Mundial de la Salud (OMS) tras meses de analizar la enfermedad, ha clasificado a la población en tres tipos de casos: personas sanas, infectadas y recuperadas. Por motivos de investigación científica, se nos ha pedido crear un programa que analice el comportamiento de esta enfermedad en los humanos, de modo que si una persona sana tiene contacto con una infectada, haya un 90% de probabilidad de ser infectada, después de 14 días, que serán representados en segundos, estos se recuperarán y no volverán a infectarse ni morirán por la enfermedad. El programa debe de representar de forma clara la cantidad de infectados, personas sanas y recuperadas y su interacción mediante bolitas de colores representativos de cada caso.

## Requerimientos funcionales

RF1	
Descripción	El sistema debe cargar un archivo .TXT con la información de la cantidad de personas sana, infectadas y recuperadas.
Entradas	Archivo TXT
Salidas	
Precondición	Debe existir un archivo .TXT
Postcondición	-

RF2	
Descripción	El sistema debe leer un archivo .TXT con la información de la cantidad de personas sana, infectadas y recuperadas.
Entradas	Archivo TXT.
Salidas	-
Precondición	El programa debe de haber cargado el archivo .TXT correctamente
Postcondición	-

RF3	
Descripción	El programa debe guardar los datos en una lista
Entradas	-
Salidas	Lista
Precondición	Debe de haber cargado un archivo .TXT Debe de haber leído correctamente el archivo .TXT

<b>Postcondición</b>	-
----------------------	---

<b>RF4</b>	
<b>Descripción</b>	El programa debe de crear elipses (personas) a partir de las características del archivo .TXT
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Debe de haber cargado un archivo .TXT Debe de haber leído correctamente el archivo .TXT
<b>Postcondición</b>	-

<b>RF5</b>	
<b>Descripción</b>	Las elipses deben de moverse de manera aleatoria
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Deben de haberse creado las elipses.
<b>Postcondición</b>	-

<b>RF6</b>	
<b>Descripción</b>	El programa debe de diferenciar las elipses por colores determinados según su estado
<b>Entradas</b>	-
<b>Salidas</b>	Debe de haber cargado un archivo .TXT Debe de haber leído correctamente el archivo .TXT
<b>Precondición</b>	-
<b>Postcondición</b>	-

<b>RF7</b>	
<b>Descripción</b>	Las elipses creadas por el programa deben no deben de colisionar entre sí
<b>Entradas</b>	-
<b>Salidas</b>	
<b>Precondición</b>	Deben de haberse creado las elipses.
<b>Postcondición</b>	-

<b>RF8</b>	
<b>Descripción</b>	Al tocarse una elipse de estado saludable con otra en estado de infección, debe de haber un 90% de probabilidad de que la primera se infecte
<b>Entradas</b>	-
<b>Salidas</b>	Cambio de estado en la elipse de estado saludable
<b>Precondición</b>	Las elipses de estado saludable e infectado deben de haberse tocado
<b>Postcondición</b>	-

RF9	
<b>Descripción</b>	El programa debe dar un aviso cuando un 30% de las personas estén infectadas.
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Un 30% de las elipses deben de estar en estado de infección
<b>Postcondición</b>	-

RF10	
<b>Descripción</b>	En el sistema, cuando una elipse que ha sido infectada, tras tocarse con una infectada, deberán pasar 14 segundos para que la primera pase a estado de recuperación.
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Debe ocurrir sólo en las elipses que han sido infectadas por otra elipse infectada, una vez el programa se inicialice
<b>Postcondición</b>	-

RF11	
<b>Descripción</b>	Las elipses que estén en estado de recuperación, no se infectarán nuevamente
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	La elipse debe estar en estado de recuperación
<b>Postcondición</b>	-

RF12	
<b>Descripción</b>	El programa delimitará el lienzo de modo que las elipses colisionan con los bordes del mismo.
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Deben de haberse creado las elipses Las elipses deben estar en movimiento aleatorio
<b>Postcondición</b>	-

RF13	
<b>Descripción</b>	La interfaz del programa debe de mostrar un contador del estado de las elipses
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Deben de haberse creado las elipses
<b>Postcondición</b>	-

<b>RF14</b>	
<b>Descripción</b>	El contador de cada estado debe de estar representado por el mismo color del estado de las elipses
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Deben de haberse creado las elipses
<b>Postcondición</b>	-

<b>RF15</b>	
<b>Descripción</b>	El usuario podrá organizar el contador de la interfaz por ordenamiento parcial según el color
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Deben de haberse creado las elipses Debe de existir el contador
<b>Postcondición</b>	-

<b>RF16</b>	
<b>Descripción</b>	El usuario podrá organizar el contador de la interfaz por ordenamiento Natural según el número de elipses por estado
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Deben de haberse creado las elipses Debe de existir el contador
<b>Postcondición</b>	-

<b>RF17</b>	
<b>Descripción</b>	Cada que se de una nueva infección el programa deberá de lanzar una excepción de contagio que se imprimirá en la consola
<b>Entradas</b>	-
<b>Salidas</b>	-
<b>Precondición</b>	Una elipse infectada debe de haberse tocado con una en estado saludable
<b>Postcondición</b>	-

#### Requerimientos No Funcionales:

- El color de las elipses infectadas debe de ser Rojo
- El color de las elipses sanas debe de ser Verde
- El color de las elipses recuperadas debe de ser Azul.
- El programa debe de ser intuitivo para el usuario

Entidades:

1. Programa

2. Persona

a. Sana

b. Infectada

c. Recuperada