

REL-Imp

Manual de Usuario

Mauricio Holguín Londoño

Universidad Tecnológica de Pereira

Andrés Escobar Mejía

Universidad Tecnológica de Pereira

Carlos Alberto Ramírez Vanegas

Universidad Tecnológica de Pereira

8 de mayo de 2020

Índice

1. INTRODUCCIÓN	9
2. INSTALACIÓN	11
2.1. REQUERIMIENTOS	11
2.2. PROCEDIMIENTO PARA LA EJECUCIÓN	11
3. PRESENTACIÓN DE LA INTERFAZ	13
3.1. PANEL PRINCIPAL	13
3.1.1. Panel de objetos	14
3.1.2. Panel de acciones	17
3.1.3. Panel de sistemas	20
3.1.4. Área de trabajo	21
4. USO DE LA INTERFAZ Y EJEMPLOS	24
4.1. EJEMPLO 1 - ESTRUCTURA SERIE	24
4.2. EJEMPLO 2 - ESTRUCTURA REDUNDANTE	33
4.3. EJEMPLO 3 - REDUNDANCIA DE BAJO NIVEL	41
4.4. EJEMPLO 4 - SUBSISTEMAS	49
5. METODOLOGÍA IMPLEMENTADA	56
5.1. MEDIDA DE LA CRITICIDAD	57

5.2. La medida de Birnbaum	59
5.3. Medida del potencial de mejora	60
5.4. Valor de logro de riesgo	60
5.5. Valor de reducción de riesgo	61
5.6. Medida de importancia de la criticidad	61
5.7. DIAGRAMAS DE BLOQUES	62
5.7.1. Confiabilidad en serie	63
5.7.2. Confiabilidad en paralelo	63
5.7.3. Redundancia activa parcial (k de $n:G$)	64
5.7.4. Redundancia positiva en frío (Stand By)	65
5.7.5. Sistemas irreducibles	66
5.8. ALGORITMO DE BÚSQUEDA	66
5.9. MODELOS UML	68
6. EVALUACIÓN DE LA CALIDAD DEL SOFTWARE	73
6.1. INTRODUCCIÓN	73
6.2. ANÁLISIS CALIDAD DE SOFTWARE REL-Imp	80
6.2.1. Robustez	80
6.2.2. Extendibilidad	81
6.2.3. Desempeño	81
6.2.4. Usabilidad	82

6.2.5. Integridad	82
6.2.6. Portabilidad	83
6.2.7. Compatibilidad	83
6.2.8. Mantenibilidad	83
6.2.9. Documentación	83
6.2.10. Grado de invención o de innovación	84

Índice de figuras

1.	Ejecutable REL-Imp	12
2.	Interfaz principal	13
3.	Panel de objetos	14
4.	Bloque	14
5.	Paralelo	15
6.	Stand By	16
7.	k d n	16
8.	Inicio	16
9.	Fin	17
10.	Panel de acciones	17
11.	Ventana de subsistemas disponibles	18
12.	Subsistema dentro del área de trabajo	19
13.	Ventana de información	19
14.	Panel de sistemas	20
15.	Ventana para renombrar sistema	21
16.	Área de trabajo	21
17.	Botones de opciones de trabajo	21
18.	Análisis del sistema	23

19.	Elemento sin posicionar	24
20.	Elementos debidamente posicionados	25
21.	Bloques dentro del área de trabajo	26
22.	Conector por fuera de un terminal	26
23.	Conector sobre un terminal	26
24.	Inicio de conexión	27
25.	Conexión de dos elementos	27
26.	Elementos en serie	28
27.	Confiabilidad del elemento <i>Bloque_1</i> a lo largo del tiempo	28
28.	Parámetros por defecto de un bloque	29
29.	Distribuciones para los bloques	30
30.	Nuevos parámetros para Bloque_2	31
31.	Medida de Birnbaum para un sistema serie	31
32.	Medida del potencial de mejora para un sistema serie	32
33.	Medida del valor de riesgo para un sistema serie	33
34.	Bloques en paralelo	34
35.	Elemento Paralelo	35
36.	Estructura del elemento Paralelo	35
37.	Estructura paralelo de 3 bloques	36
38.	Estructura paralelo con una serie en sus filas	37

39.	Sistema en paralelo	37
40.	Confiabilidad del sistema paralelo	38
41.	Confiabilidad del sistema	39
42.	Confiabilidad del sistema	40
43.	Confiabilidad del sistema	41
44.	Elementos Stand By y k d n	41
45.	Propiedades Stand By	42
46.	Propiedades k d n	42
47.	Sistema para el ejemplo 3	43
48.	Confiabilidad del sistema redundante en serie	43
49.	Medida de Birnbaum para el ejemplo 3	44
50.	Medida de potencial de mejora para el ejemplo 3	44
51.	Medida de valor de logro de riesgo para el ejemplo 3	45
52.	Medida de valor de reducción de riesgo para el ejemplo 3	45
53.	Medida de importancia de la criticidad para el ejemplo 3	46
54.	Importancia al aumentar número de bloques en StandBy	47
55.	Importancia al aumentar el número de elementos (n) en K d N	48
56.	Importancia al aumentar el número de el número de elementos buenos (K)	49
57.	Asignación de nombre para el sistema 1	49
58.	Sistemas con nuevos nombres	50

59.	Advertencia al tratar de exportar un subsistema	50
60.	Lista de subsistemas disponibles	51
61.	Advertencia al no existir subsistemas	51
62.	Subsistema dentro del área de trabajo	52
63.	Sistema compuesto de subsistemas	52
64.	Confiabilidad total del sistema para el ejemplo 4 (con $t=30000$)	53
65.	Confiabilidad total del sistema para el ejemplo 4 (con $t=10000$)	54
66.	Medida de la importancia por parte de cada uno de los subsistemas (con $t=30000$)	55
67.	Medida de la importancia por parte de cada uno de los subsistemas(con $t=10000$)	56
68.	Sistema en serie	63
69.	Sistema en paralelo	64
70.	Diagrama de componentes	70
71.	Diagrama de paquetes	71
72.	Diagrama de actividades	72
73.	Diagrama de secuencias	73
74.	Certificación del impacto del producto REL-Imp	85

1. INTRODUCCIÓN

La calidad de vida de las personas en todo el mundo depende del funcionamiento seguro y confiable de los sistemas físicos que los rodean, tales como puertos, redes de transporte por carretera, sistemas de comunicación y sistemas de suministro de agua, gas y electricidad, por tanto la resistencia de los sistemas a los fallos durante su funcionamiento es de gran importancia práctica.

Una de las estrategias que podrían considerarse para mejorar la confiabilidad y la resistencia de un sistema es el intercambio de componentes cuando un componente falla, sustituyéndolo así por otro componente del sistema que sigue funcionando. Se han definido varias medidas de importancia (IM) para evaluar y clasificar las contribuciones de los componentes a la probabilidad de fallo de un sistema.

Cuando se analiza la confiabilidad y disponibilidad de un sistema, la medición de la importancia de los componentes suele tener un valor significativo a la hora de priorizar los esfuerzos de mejora, realizar un análisis de compensación en el diseño del sistema o sugerir la forma más eficiente de operar y mantener un sistema. El hecho de centrarse en las áreas más problemáticas del sistema da como resultado las ganancias más significativas.

Para facilitar tales propósitos, se presenta REL-Imp, plataforma portable que permite la identificación de los componentes críticos dentro de un sistema representado por diagramas de bloques (método gráfico por preferencia para la representación de sistemas físicos dentro del estudio de confiabilidad), esto mediante diferentes medidas de importancia. REL-Imp facilita entonces el análisis de sistemas simples y complejos mediante la visualización de las diferentes métricas para cada elemento de un sistema, brindando así la

posibilidad de determinar que componentes son más críticos desde un punto de vista del impacto en el sistema de confiabilidad, proporcionando así una perspectiva cuantitativa sobre los contribuyentes dominantes al riesgo y la sensibilidad del riesgo a los cambios en los valores de los insumos.

Este manual no tiene como finalidad ser un documento exhaustivo acerca de las operaciones realizadas internamente por la interfaz, como tampoco pretende ser un documento guía para el estudio de la confiabilidad de sistemas. En su lugar se describe la metodología utilizada en su funcionamiento, además de los modelos comportamentales que rigen el diseño del presente software.

2. INSTALACIÓN

2.1. REQUERIMIENTOS

Antes de iniciar con la instalación de **REL-Imp**, compruebe los siguientes requisitos:

- Sistema Operativo Windows 7/8/10 (64 bits).
- Intel Core I3 @ 2.3GHz (o equivalente).
- 4 GB de memoria RAM.
- 500MB de espacio libre en el disco de almacenamiento.
- Visor de archivos PDF.

2.2. PROCEDIMIENTO PARA LA EJECUCIÓN

Para ejecutar **REL-Imp**, se debe insertar el CD de distribución en la unidad CD-ROM. Se recomienda iniciar verificando los contenidos de los archivos *leame.txt* y *licencia.txt* disponibles en el CD de distribución.

REL-Imp es un software portable, esto quiere decir que no es intrusivo para el sistema operativo, por tanto no se hace necesaria la instalación de éste dentro del sistema de almacenamiento interno. Basta con copiar los elementos del CD de distribución dentro de un lugar deseado en el equipo anfitrión. Para ejecutar el software basta con hacer doble clic sobre el archivo **REL-Imp.exe** (ubicado dentro de la carpeta **Ejecutable** del CD de distribución), similar al que se observa en la Figura 1.

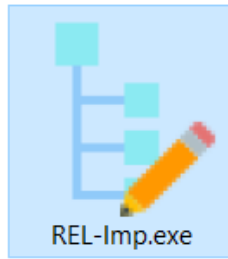


Figura 1: Ejecutable REL-Imp

Para comodidad del usuario, es posible añadir este ejecutable como un acceso directo, o bien sea dentro de la barra de inicio.

3. PRESENTACIÓN DE LA INTERFAZ

Verificados los requerimientos del sistema, es posible acceder a la interfaz por medio del icono generado en la instalación, el cual se observa en la Figura 1. Una vez se ejecuta el software se despliega la interfaz principal del sistema, tal como se observa en la Figura 2.

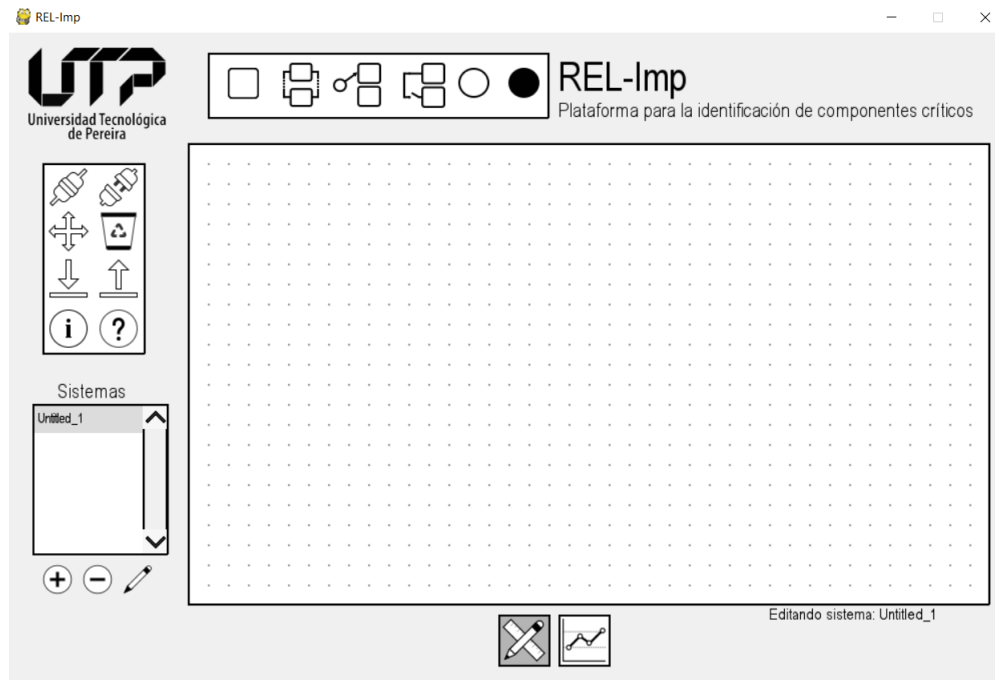


Figura 2: Interfaz principal

3.1. PANEL PRINCIPAL

La interfaz principal se compone de tres partes:

1. Panel de objetos
2. Panel de acciones
3. Área de trabajo

3.1.1. Panel de objetos

Panel que contiene los distintos elementos que permiten la creación de un sistema determinado. La Figura 3 enseña el panel de objetos.

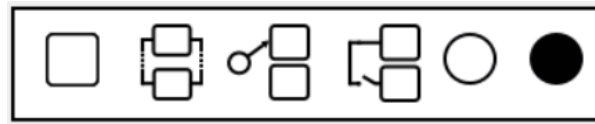


Figura 3: Panel de objetos

Cada uno de estos objetos puede ser posicionado dentro del área de trabajo para dar lugar a sistemas de mayor complejidad. Por defecto cuentan con parámetros iniciales, como lo pueden ser la distribución que asumen a lo largo del tiempo, o la cantidad de elementos en sus filas. La modificación de estos atributos se realiza haciendo clic derecho (adición/reducción de bloques en un elemento con redundancia, distribución que asumen el elemento a lo largo del tiempo, rotación) o doble clic izquierdo (nombre del elemento, parámetros *alpha* y *betha*) sobre el elemento de interés. A continuación, se describen cada uno de estos elementos:

Bloque: Este objeto permite adicionar bloques dentro del área de trabajo para su posterior conexión, bien sea en serie o en paralelo a otros elementos. Haciendo doble clic izquierdo sobre éste se permite modificar su nombre, así como los parámetros *alpha* y *betha*. Mediante el clic derecho sobre éste es posible modificar la distribución que asume, así como su rotación (solo es posible si el elemento se encuentra desconectado). En la Figura 4 se observa dicho objeto.



Figura 4: **Bloque**

Paralelo: Este objeto permite adicionar bloques con redundancia activa dentro del área de trabajo, para su posterior conexión, bien sea en serie o en paralelo a otros elementos. Por defecto, se presentan dos bloques en sus filas. Haciendo doble clic izquierdo sobre la etiqueta del elemento se permite modificar su nombre. Mediante el clic derecho sobre la etiqueta es posible adicionar/reducir la cantidad de sus filas y columnas (esta adición/reducción se refleja visualmente). Cada bloque que se adiciona puede ser configurado de manera independiente, al igual que lo es el elemento **Bloque**. Es posible añadir hasta 5 elementos en sus filas, así como un máximo de 6 elementos en serie por cada una de las filas. Para configuraciones con más elementos, se puede posicionar un bloque adicional o definir subsistemas (ver sección 3.1.2). En la Figura 5 se observa dicho objeto.



Figura 5: **Paralelo**

Stand By: Este objeto permite adicionar bloques que presentan redundancia positiva en frío dentro del área de trabajo, para su posterior conexión, bien sea en serie o en paralelo a otros elementos. Por defecto, se presentan dos bloques en sus filas. Haciendo doble clic izquierdo sobre el conmutador (esfera) se permite modificar su nombre, así como los parámetros *alpha* y *betha*. Mediante el clic derecho sobre el conmutador es posible adicionar/reducir la cantidad de sus filas (esta adición/reducción no se refleja visualmente). Es posible añadir hasta 5 elementos en sus filas. En la Figura 6 se observa dicho objeto.

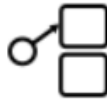


Figura 6: **Stand By**

k d n: Este objeto permite adicionar bloques que presentan redundancia activa parcial dentro del área de trabajo, para su posterior conexión, bien sea en serie o en paralelo a otros elementos. Por defecto, se presentan dos bloques en sus filas. Haciendo doble clic izquierdo sobre la etiqueta del elemento se permite modificar su nombre, así como los parámetros *alpha* y *betha*. Mediante el clic derecho sobre la etiqueta es posible adicionar/reducir la cantidad de unidades mínimas y unidades disponibles (esta adición/reducción no se refleja visualmente). Es posible añadir hasta 5 elementos en sus filas. En la Figura 7 se observa dicho objeto.



Figura 7: **k d n**

Inicio: Este objeto indica el inicio del sistema, se considera obligatorio y solo puede existir uno por sistema. En la Figura 8 se observa dicho objeto.



Figura 8: **Inicio**

Fin: Este objeto delimita el fin del sistema, se considera obligatorio y solo puede existir uno por sistema. En la Figura 9 se observa dicho objeto.



Figura 9: **Fin**

Para finalizar la configuración de cada uno de los objetos seleccionados basta con presionar la tecla *Escape*.

3.1.2. Panel de acciones

Es dentro de esta sección de la interfaz donde se realizan distintas operaciones, tales como la conexión de elementos, la eliminación de elementos dentro del área de trabajo, la importación y exportación de subsistemas, visualización del manual de usuario, entre otros. La Figura 10 enseña el panel de acciones.

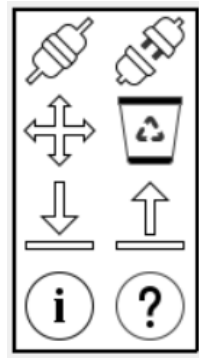








Figura 10: Panel de acciones

A continuación se definen cada una de estas:

- **Conectar**  : Mediante esta opción es posible interconectar los terminales de los distintos objetos que se ubiquen dentro del área de trabajo.

- **Desconectar** : Permite desconectar aquellos elementos que presenten una conexión previamente definida.
- **Mover** : Permite desplazar elementos dentro del área de trabajo. Solo es posible desplazar aquellos elementos que no presenten una conexión.
- **Eliminar** : Permite eliminar elementos del área de trabajo. Si el elemento a eliminar presenta una o más conexiones, cada una de estas también se elimina.
- **Exportar** : Permite exportar el sistema dentro del área de trabajo seleccionada a modo de subsistema. Solo es posible exportar aquellas pestañas que no presenten elementos desconectados.
- **Importar** : Permite importar subsistemas previamente almacenados. Cada vez que se hace clic sobre esta acción, y desde que previamente haya sido exportado por lo menos un subsistema, se despliega una ventana emergente como la que se observa en la Figura 11.

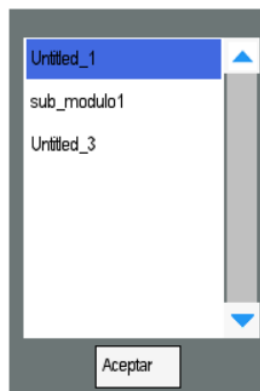


Figura 11: Ventana de subsistemas disponibles

Como resultado de elegir alguno de los subsistemas disponibles, se habilita un elemento rectangular que puede posicionarse dentro del área de trabajo. Este ele-

mento representa todo un sistema previamente almacenado y puede ser accedido haciendo doble clic sobre él. En la Figura 12 se observa un subsistema posicionado dentro del área de trabajo.

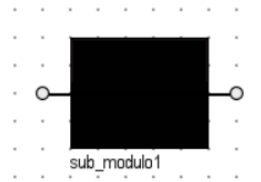


Figura 12: Subsistema dentro del área de trabajo

- **Información** ⓘ: Despliega una ventana emergente que brinda detalles relacionados al presente software, tal como la que se observa en la Figura 13.



Figura 13: Ventana de información

- **Ayuda** ⓘ: Permite acceder al presente manual.

3.1.3. Panel de sistemas

Este panel opera como un contenedor de los nombres de cada uno de los sistemas existentes, y a su vez, dispone de un conjunto de botones que permiten agregar, eliminar y modificar sus elementos. En la Figura 14 se observa el Panel de sistemas.

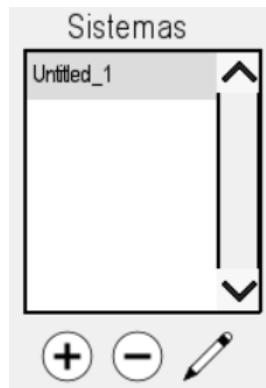





Figura 14: Panel de sistemas

A continuación se describen los botones con los que dispone el Panel de sistemas:

- **Agregar sistema** : Adiciona un nuevo sistema al final de la cola de elementos del Panel de sistemas.
- **Eliminar sistema** : Elimina el sistema que se encuentre seleccionado dentro del Panel de sistemas.
- **Renombrar sistema** : Permite modificar el nombre de la hoja de trabajo actual. Cada vez que se hace clic sobre esta acción se despliega una ventana emergente como la que se observa en la Figura 15.

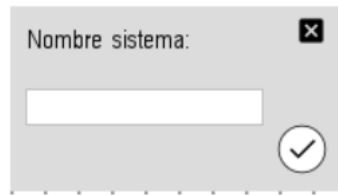


Figura 15: Ventana para renombrar sistema

3.1.4. Área de trabajo

Es dentro de esta sección de la interfaz en la que se realiza el posicionamiento de cada uno de los objetos, dando lugar al diseño y posterior análisis de los sistemas. En la Figura 16 se observa el área de trabajo.

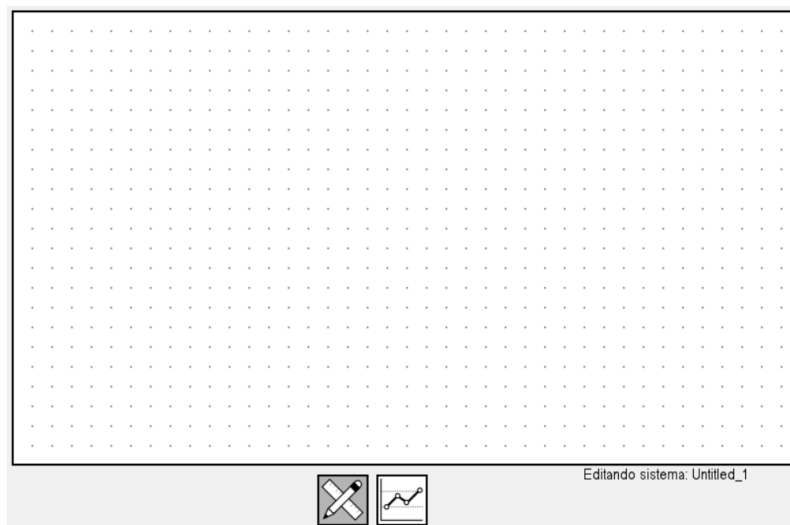




Figura 16: Área de trabajo

Esta sección se divide a su vez en 2 partes, una de diseño y una de análisis del sistema, las cuales pueden ser accedidas por medio de los botones que se observan en la Figura 17.



Figura 17: Botones de opciones de trabajo

A continuación se describe cada una de estas opciones:

- Diseño : Esta es la sección que se presenta por defecto (Figura 16), y dentro de la cual se posicionan cada uno de los elementos que se encargan de formar un sistema. Para avanzar hacia la opción siguiente, el sistema debe encontrarse interconectado en su totalidad. Cada sistema debe poseer un elemento del tipo **Inicio** y uno del tipo **Fin**, indicando el inicio y el fin del sistema, respectivamente.
- Análisis del sistema : En esta sección se realiza un análisis de la confiabilidad total del sistema o alguna medida de importancia de los elementos que lo conforman. Las medidas de importancia disponibles en el sistema son:
 - Birnbaum (ver sección 5.2)
 - Potencial de mejora (ver sección 5.3)
 - Valor de logro de riesgo (ver sección 5.4)
 - Valor de reducción de riesgo (ver sección 5.5)
 - Importancia de la criticidad (ver sección 5.6)

Cabe recalcar que esta sección solo es accesible si todos los elementos, dentro de la hoja seleccionada, se encuentran debidamente conectados. En la Figura 18 se observa esta sección.

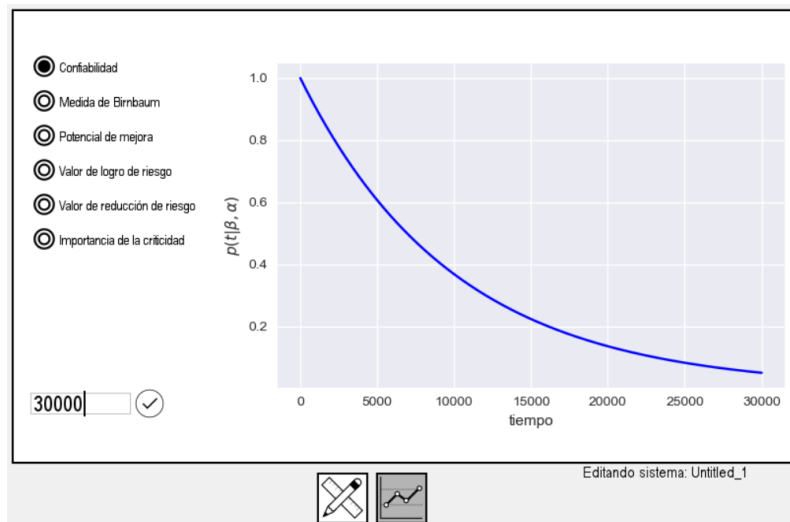



Figura 18: Análisis del sistema

Cabe destacar que es posible tener hasta 14 sistemas independientes dentro del área de trabajo, siendo cada uno de estos accesible por medio de su respectiva pestaña. Inicialmente se cuenta con un solo sistema, en caso de desear adicionar uno nuevo, basta con hacer clic sobre el botón **Agregar sistema** , dando lugar a la creación de una nueva hoja de trabajo.

4. USO DE LA INTERFAZ Y EJEMPLOS

4.1. EJEMPLO 1 - ESTRUCTURA SERIE

Una vez se ejecuta el programa y se presenta la interfaz principal, es posible proceder a adicionar los elementos necesarios para crear un sistema. Para el presente ejemplo se realiza la adición de dos elementos individuales en serie, el cual es un sistema que es confiable si y sólo si todos sus componentes son confiables. Se procede entonces entonces a la adicionan del total de bloques que se desean dentro del área de trabajo. Cabe resaltar que, aunque el orden en que se agreguen los elementos no es de relevancia, se recomienda iniciar con la adición de los objetos **Inicio** y **Fin**, esto debido a que son elementos obligatorios en todo sistema. Para la adición de cada uno de los elementos basta con presionar clic izquierdo sobre el objeto deseado, en este caso se inicia con los elementos **Inicio** y **Fin**, disponibles dentro del panel de objetos. Una vez se realiza dicha acción, el objeto pasa a estar visible en todo momento, siguiendo el desplazamiento del cursor dentro de la interfaz, tal como se observa en la Figura 19.

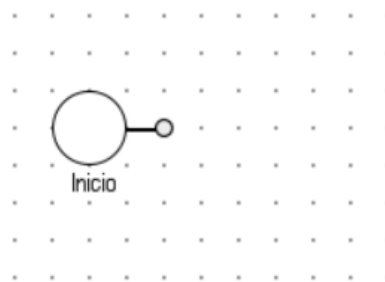


Figura 19: Elemento sin posicionar

Para ubicar el elemento basta con presionar nuevamente el botón izquierdo del mouse, luego de esto es posible seguir adicionando elementos dentro del área de trabajo, solo

hasta que se elija algún nuevo elemento o se presione la tecla **Escape**. El objeto seleccionado sólo puede ser posicionado dentro del área de trabajo, en caso de intentar algún otro movimiento este será obviado. En la Figura 20 es posible observar los objetos **Inicio** y **Fin** debidamente posicionados dentro del área de trabajo y el cursor aún disponible con el elemento **Fin**.

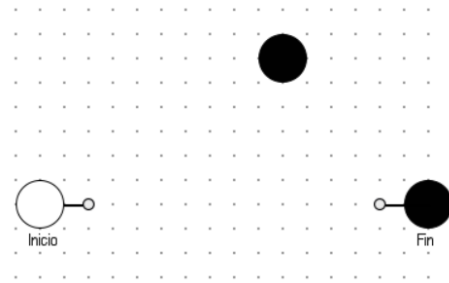


Figura 20: Elementos debidamente posicionados

Una vez se cuenta con los elementos que indican el inicio el final del sistema se procede a realizar la adición de los elementos que lo componen. Para el presente caso solo se adicionan elementos del tipo **Bloque**. A diferencia de los elementos **Inicio** y **Fin** los demás elementos pueden adicionarse repetidamente dentro del espacio de trabajo, y a medida que se agregue nuevos elementos de la misma naturaleza, cada uno de ellos incrementará en pasos de 1 el valor del identificador grupal, esto es, una etiqueta que permite diferenciar cada elemento por medio de una identidad única. Esta etiqueta se encuentra visible debajo de cada elemento. En la Figura 21 se observan dos bloques dentro del área de trabajo, cada uno con su respectiva etiqueta.

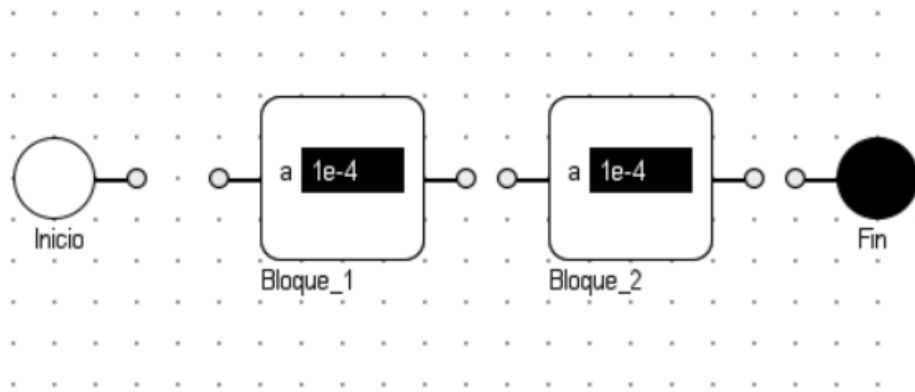



Figura 21: Bloques dentro del área de trabajo

Ya con los elementos debidamente posicionados, se procede a realizar la conexión entre ellos, esto por medio de la acción **conectar** . Al elegir esta acción, se despliega un cursor en forma de cruz, el cual se presenta en color negro si no se encuentra ubicado sobre el terminal de algún elemento, y en rojo si se posiciona sobre un terminal. En las Figuras 22 y 23 se enseña al cursor, tanto fuera de un terminal como sobre un terminal.

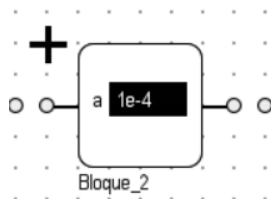


Figura 22: Conector por fuera de un terminal

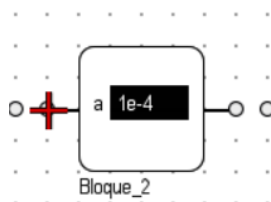


Figura 23: Conector sobre un terminal

Ubicandose entonces sobre un terminal, se hace clic y se dispone a desplazar el cursor

en busca del otro elemento al que se desea conectar. En la Figura 24 se observa el cable que se forma una vez se hace clic sobre un terminal.

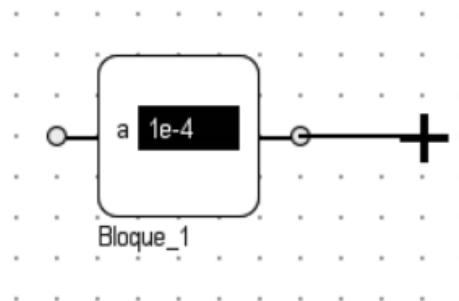


Figura 24: Inicio de conexión

Se hace entonces clic sobre el otro terminal deseado, dando lugar al cierre de la conexión, tornando los terminales conectados a un color negro, tal como se observa en la Figura 25.

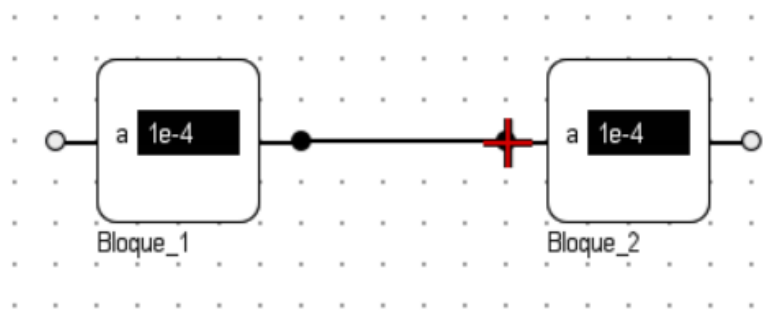


Figura 25: Conexión de dos elementos

Se procede a realizar la conexión de los extremos restantes de los bloques con los elementos **Inicio** y **Fin**, dando lugar al sistema que se observa en la Figura 26.

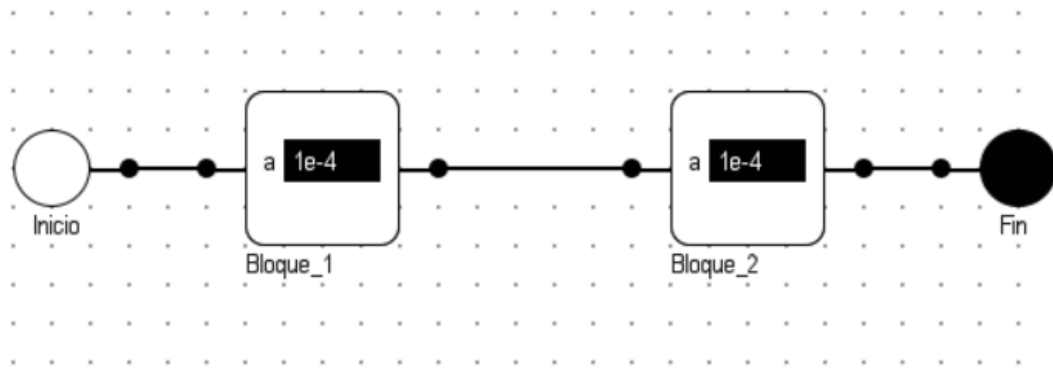



Figura 26: Elementos en serie

Con el sistema ya creado, se procede a realizar un análisis al sistema a partir de la importancia que tiene cada elemento, para ello, se accede a la sección de **Análisis del sistema** , dando lugar a lo que se observa en la Figura 27, donde inicialmente se enseña la confiabilidad total del sistema.

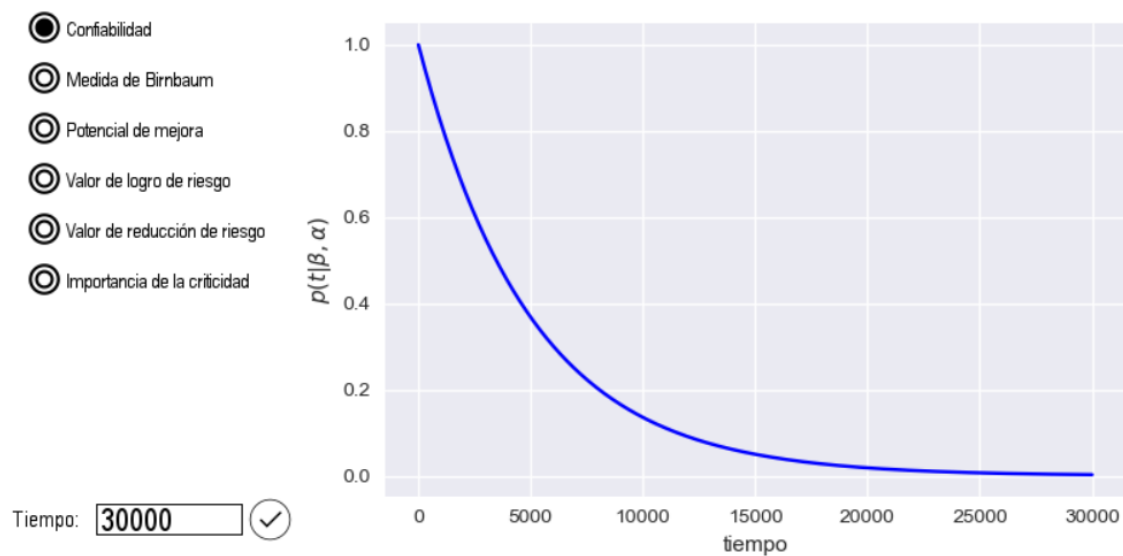


Figura 27: Confiabilidad del elemento *Bloque_1* a lo largo del tiempo

Por defecto se presenta la confiabilidad del sistema, pero el usuario puede elegir en cualquier momento el comportamiento que desea observar mediante el accionamiento

de cada uno de los botones de las métricas presentes. La forma de la curva depende del tipo de distribución y de los parámetros *Alpha* y *Betha* con que este cuente. Por defecto, cada bloque presenta una distribución de tipo exponencial, y un parámetro *alpha* igual a $1e-4$. Es posible elegir entre las distribuciones *Exponencial*, *Rayleigh* y *Weibull*. Para las dos primeras solo es posible modificar el valor de *Alpha*, el parámetro *Betha* asume los valores 1 y 2, respectivamente. En la Figura 28 se observan los parámetros por defecto para un bloque que asume una distribución de tipo exponencial.

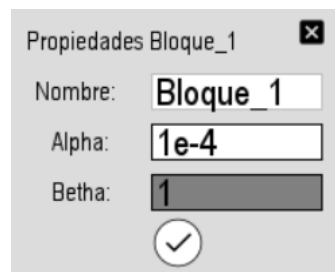


Figura 28: Parámetros por defecto de un bloque

Es responsabilidad del usuario asignar unidades consistentes y uniformes a los parámetros. Por ejemplo, si un parámetro está dado en fallos/mes, la unidad de evaluación será meses, pero si está en fallos/día, la unidad de evaluación será días. Además se debe recordar, que la evaluación no necesariamente es en tiempo, ya que puede ser en distancia, ciclos, etc. Indiferentes de la unidad de evaluación seleccionada, los gráficos siempre muestran la variable "tiempo", en relación a "Tiempo". Para realizar un cambio en la distribución del elemento, basta con hacer clic derecho sobre éste, dando lugar a la ventana que se observa en la Figura 29.

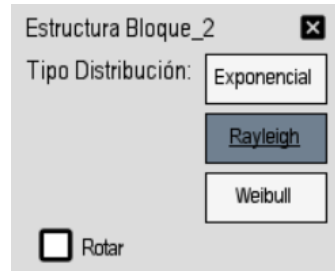


Figura 29: Distribuciones para los bloques

Considerando entonces esta estructura en serie de dos componentes independientes, Bloque 1 y Bloque 2, cada uno con su propio valor confiabilidad p_1 y p_2 , y partiendo del supuesto que $p_1 > p_2$, es decir, el componente 1 es el más confiable de los dos, se procede a realizar un análisis a la importancia que tiene cada uno en el sistema mediante diferentes métricas. Este sistema presenta una confiabilidad $h(p) = p_1 p_2$, lo cual para el caso de la medida de Birnbaum (ver sección ??) da lugar a la obtención de los siguientes parámetros:

1. La medida de Birnbaum para el componente 1 es $I^B(1) = \frac{\delta h(p)}{\delta p_1} = p_2$
2. La medida de Birnbaum para el componente 2 es $I^B(2) = \frac{\delta h(p)}{\delta p_2} = p_1$

Esto significa que $I^B(2) > I^B(1)$, por tanto es posible concluir que cuando se utiliza la medida de Birnbaum, el componente más importante en una estructura de serie es el de menor confiabilidad. Dado a que en el presente software se precisa el ingreso de la tasa de fallos para cada componente, es necesario tener presente que a menor tasa de fallos mayor la confiabilidad del componente, por tanto, al indicar que el bloque 1 es más confiable que el bloque 2, significa que la tasa de fallos del bloque 1 es menor a la del bloque 2. Para permitir esta evaluación se dispone a modificar el parámetro *Alpha* del bloque 2 a un valor de $2e - 4$. Estos cambios se observan en la Figura 30.

Propiedades Bloque_2

Nombre:

Alpha:

Betha:

☒

Figura 30: Nuevos parámetros para **Bloque_2**

Una vez se realiza este cambio se procede a observar el comportamiento que asume la importancia de cada elemento a lo largo de la vida útil del sistema, dando lugar a validar lo previamente mencionado. En la Figura 31 se observa como el Bloque2 tiene mayor importancia durante todo intervalo de operación del sistema, esto debido a que al ser menos confiable para el sistema va a representar un mayor nivel de criticidad.

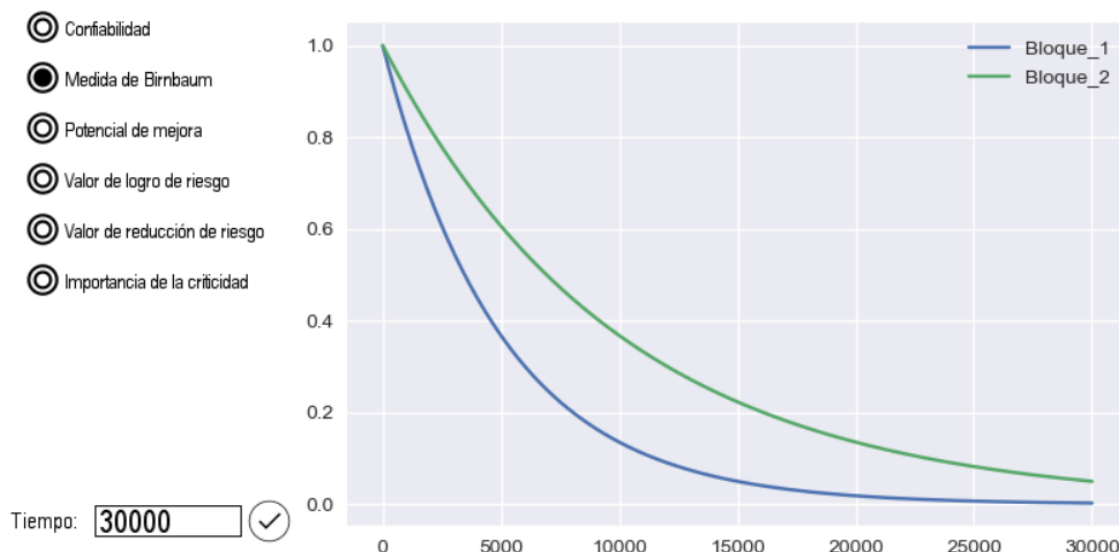


Figura 31: Medida de Birnbaum para un sistema serie

Partiendo de la misma estructura serie, se analiza nuevamente la importancia de los elementos, peso esta vez mediante la medida del potencial de mejora, dando lugar a los siguientes parámetros:

$$1. I^{IP}(1) = 1 * p_2 - p_1 p_2 = p_2(1 - p_1)$$

$$2. I^{IP}(2) = 1 * p_1 - p_1 p_2 = p_1(1 - p_2)$$

Lo anterior significa que $I^{IP}(2) > I^{IP}(1)$, lo que permite concluir que al utilizar la medida del Potencial de Mejora, el componente más importante en una estructura serie es el de menor confiabilidad.

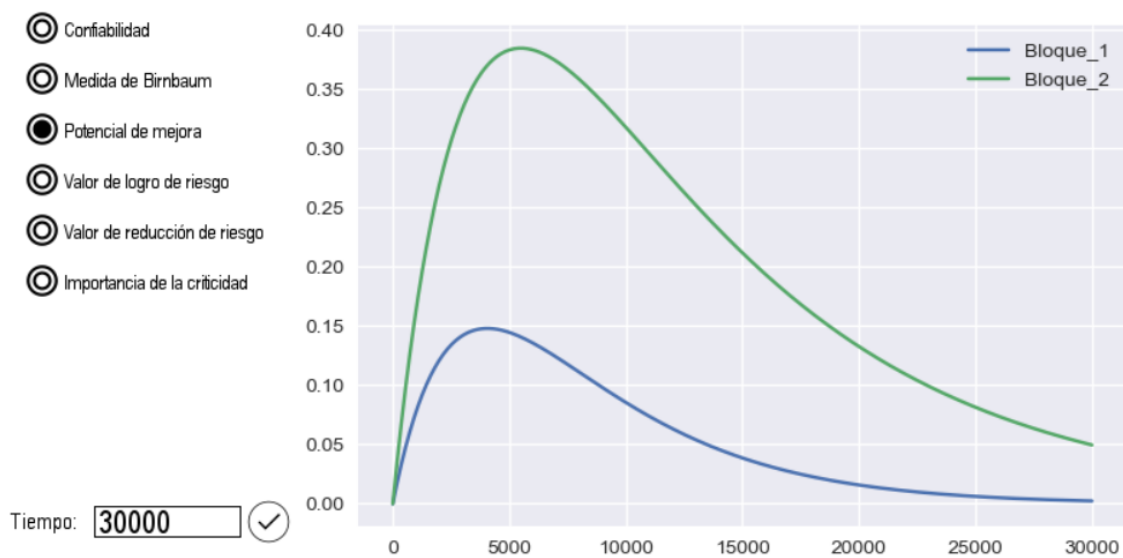


Figura 32: Medida del potencial de mejora para un sistema serie

Se analiza nuevamente el mismo sistema, pero mediante la medida del valor de reducción de riesgo, dando lugar a los siguientes parámetros:

$$1. I^{RRW}(1) = \frac{1-p_1 p_2}{1-p_2}$$

$$2. I^{RRW}(2) = \frac{1-p_1 p_2}{1-p_1}$$

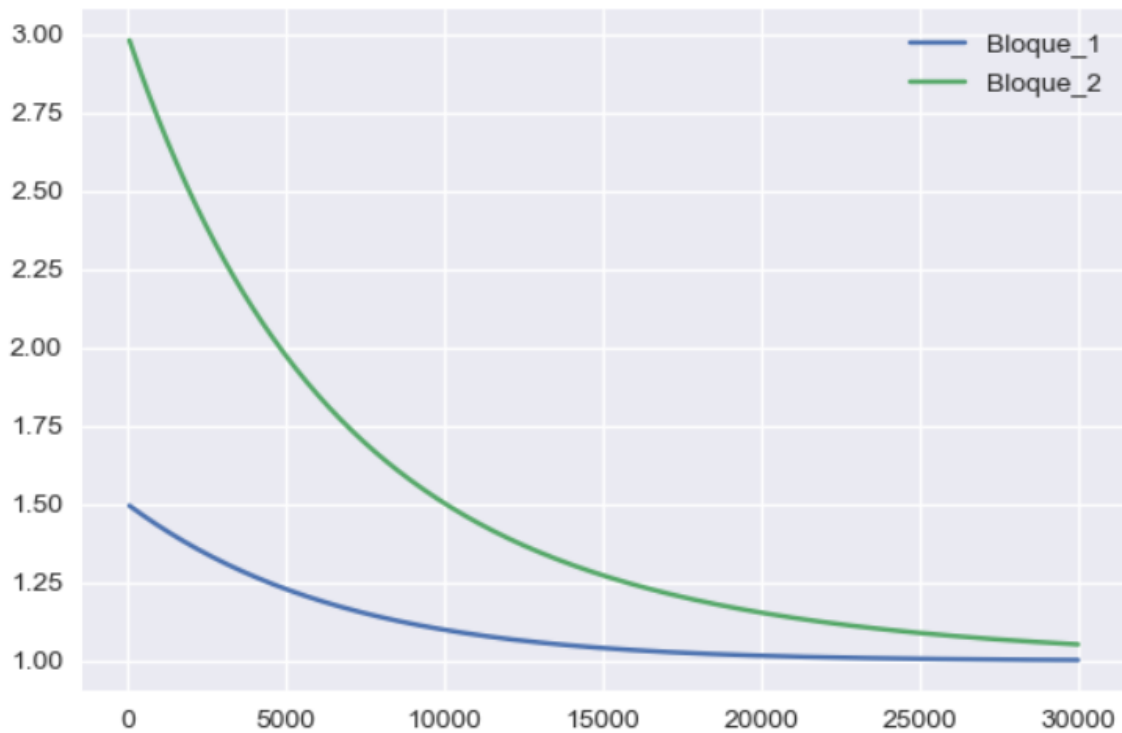



Figura 33: Medida del valor de riesgo para un sistema serie

Por lo tanto, para mejorar la estructura de una serie, es necesario mejorar el componente "más débil", es decir, el componente con la menor confiabilidad.

Nota: El sistema limita al usuario al ingreso de valores numéricos, en caso de intentar adicionar tipos de caracteres ajenos a los datos de naturaleza numérica, éste será omitido.

4.2. EJEMPLO 2 - ESTRUCTURA REDUNDANTE

Se procede a la adición de un nuevo sistema, dando lugar a una nueva hoja de diseño en blanco, esto por medio de accionar clic izquierdo sobre el botón **Agregar sistema** . Se dispone entonces de una hoja en blanco completamente independiente de la hoja en que se creó el ejemplo anterior, permitiendo así dar lugar a un nuevo sistema. Para este ejemplo se expone un caso en el que se desea agregar redundancia al sistema. En

este caso, se dice que la falla del sistema solo ocurre con la falla de todos los elementos. En otras palabras, el sistema es confiable con que solo un elemento sea confiable. Para realizar esta operación, es posible adicionar dos elementos de tipo **Bloque**, y luego conectar los extremos de estos de manera que se genere una estructura de tipo paralelo, tal como se observa en la Figura 34.

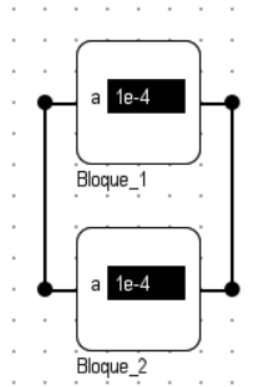


Figura 34: Bloques en paralelo

Esta disposición se torna compleja de realizar a medida que se adicionan más elementos. Para solventar ello, se cuenta con el elemento **Paralelo**, el cual permite adherir elementos de una manera más sencilla. En la Figura 35 se enseña una estructura en paralelo sobre el área de trabajo. Esta se elige y posiciona similar a como se enseñó previamente con el elemento **Bloque**.

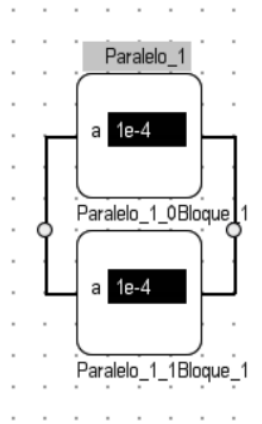


Figura 35: Elemento **Paralelo**

Este elemento cuenta con una etiqueta en su parte superior. Mediante el accionar del clic derecho sobre esta etiqueta es posible modificar la estructura que compone al presente elemento, el cual se presenta a modo de ventana emergente, tal como se observa en la Figura 36.

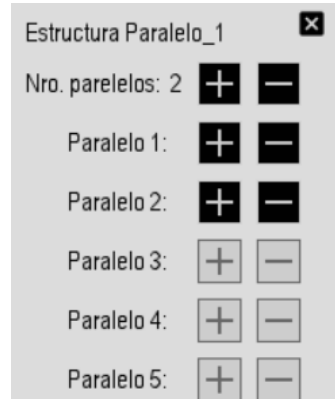


Figura 36: Estructura del elemento **Paralelo**

Por medio de esta ventana se adicionan nuevas filas (paralelos), así como nuevas columnas (series) dentro de estas filas, todo ello por medio de los botones **Agregar/Reducir** . La primer fila de estos botones permite adicionar o eliminar elementos en paralelo. Las filas siguientes permiten adicionar o eliminar elementos en serie a cada uno de

los paralelos disponibles. Esta adición/reducción solo puede realizarse si el elemento se encuentra completamente desconectado. Cada que se adiciona un nuevo paralelo, se habilita dentro del panel de estructura una línea siguiente de botones. Se procede entonces a agregar un nuevo paralelo mediante el accionamiento del botón **Agregar paralelo**, dando lugar a la estructura que se observa en la Figura 37.

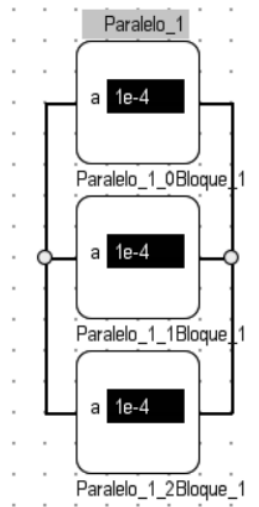


Figura 37: Estructura paralelo de 3 bloques

En caso de desear adicionar un elemento en serie para alguno de los elementos en paralelo, basta con presionar el botón **Agregar serie** en la fila específica del paralelo al que se desea añadir. Haciendo este botón en la línea **Paralelo 2** se da lugar a la estructura que se presenta en la Figura 38. Cada uno de estos bloques puede ser modificado como si se tratase de un bloque individual. Es posible adicionar hasta 5 paralelos, cada uno con un máximo de 6 elementos en serie.

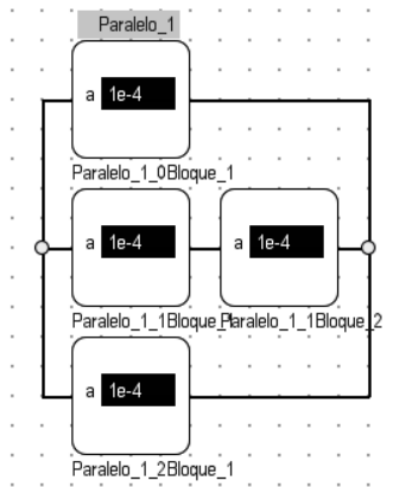


Figura 38: Estructura paralelo con una serie en sus filas

El elemento de tipo **Paralelo** tiene especial significancia cuando se requiere comparar su importancia contra otros elementos, esto debido a que al ser un elemento se evalúa su importancia total, más no la importancia individual de sus propios bloques. Para el presente caso se realiza la validación de un sistema en paralelo de dos componentes independientes, Bloque 1 y Bloque 2 (Figura 39), el cual da lugar a la curva de confiabilidad que se observa en la Figura 40.

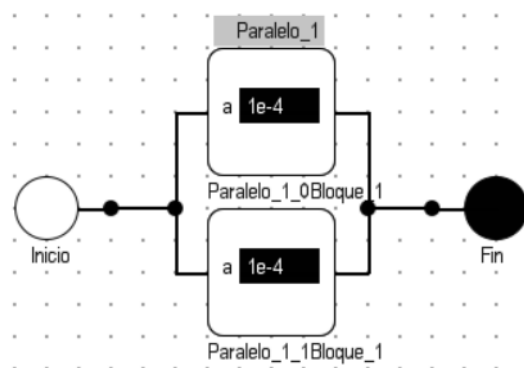


Figura 39: Sistema en paralelo

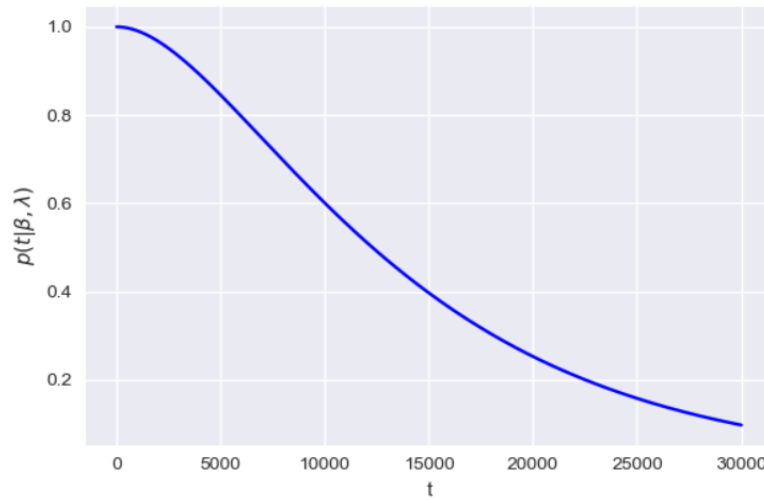


Figura 40: Confiabilidad del sistema paralelo

Considerando estos dos bloques en paralelo con unas confiabilidad del sistema $h(p) = p_1 + p_2 - p_1p_2$, donde p_1 y p_2 son los respectivos valores de confiabilidad para el Bloque 1 y el Bloque 2, y partiendo del supuesto que $p_1 > p_2$, es decir, el componente 1 es el más confiable de los dos, se procede a realizar un análisis a la importancia que tiene cada uno en el sistema. Para la medida de Birnbaum se obtienen los siguientes parámetros:

1. La medida de Birnbaum para el componente 1 es $I^B(1) = \frac{\delta h(p)}{\delta p_1} = 1 - p_2$
2. La medida de Birnbaum para el componente 2 es $I^B(2) = \frac{\delta h(p)}{\delta p_2} = 1 - p_1$

Esto significa que $I^B(1) > I^B(2)$, por tanto es posible concluir que cuando se utiliza la medida de Birnbaum, el componente más importante en una estructura de serie es el de mayor confiabilidad, contrario a la estructura serie, en la que el componente más importante es el de menor confiabilidad. La importancia de ambos componentes se observa en la Figura 41.

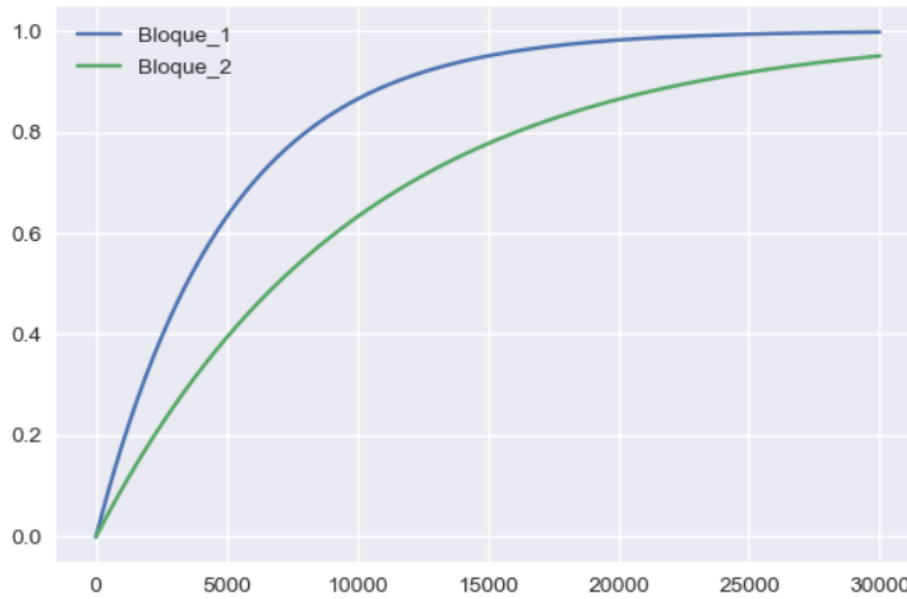


Figura 41: Confiabilidad del sistema

Partiendo de la misma configuración y parámetros de los bloques, se visualiza la importancia de los elementos mediante dos métricas diferentes, como lo son potencial de mejora y valor de logro de riesgo. Para el valor de logro de riesgo se obtienen las siguientes funciones de evaluación:

$$1. I^{IP}(1) = 1 - [p_1 + p_2 - p_1 p_2]$$

$$2. I^{IP}(2) = 1 - [p_1 + p_2 - p_1 p_2]$$

Esto significa que $I^{IP}(1) = I^{IP}(2)$, permitiendo concluir que al utilizar la medida de potencial de mejora, todos los componentes en una estructura en paralelo tienen la misma importancia, lo cual implica el que no ...

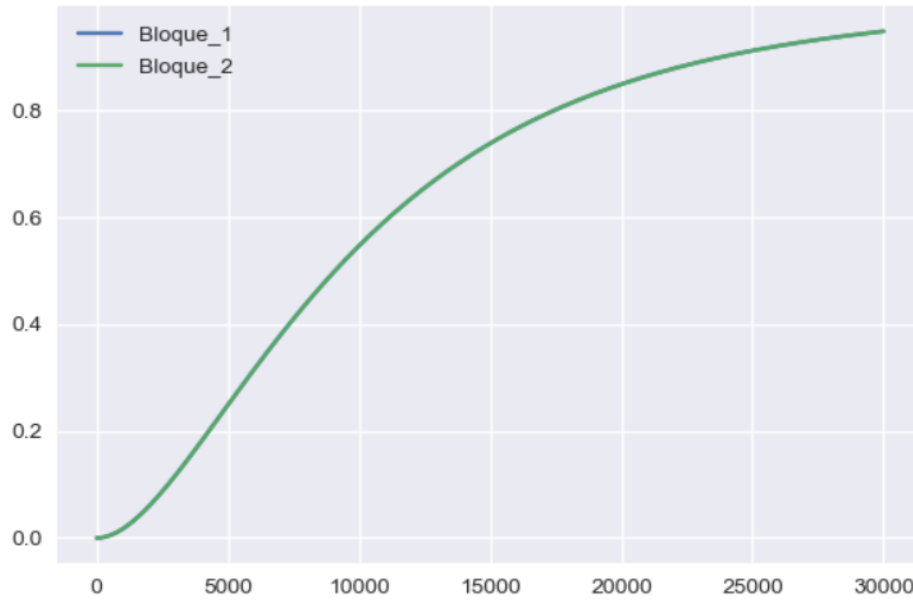


Figura 42: Confiabilidad del sistema

Para el caso de la medida de valor de logro de riesgo, se obtienen los siguientes parámetros:

$$1. I^{RAW}(1) = \frac{1-p_2}{1-(p_1+p_2-p_1p_2)}$$

$$2. I^{RAW}(2) = \frac{1-p_1}{1-(p_1+p_2-p_1p_2)}$$

Esto significa que $I^{RAW}(1) > I^{RAW}(2)$, permitiendo concluir que cuando se utiliza la medida de RAW, el componente más fuerte (es decir, el más fiable) en una estructura paralela es el más importante, tal como se observa en la Figura

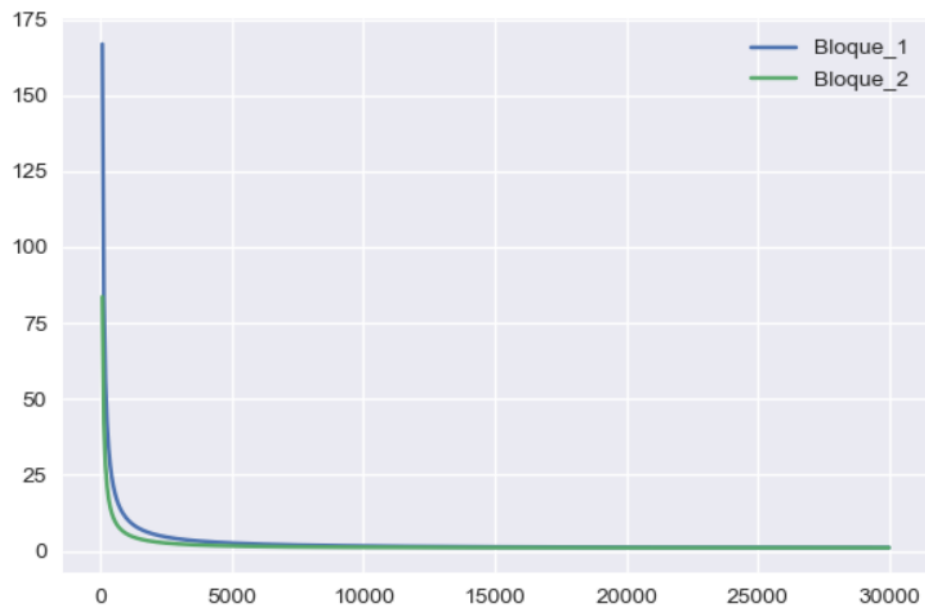


Figura 43: Confiabilidad del sistema

4.3. EJEMPLO 3 - REDUNDANCIA DE BAJO NIVEL

Se procede a la adición de una nueva hoja, la cual va a contener un elemento **Stand By** y un elemento **k d n**, tal como se observa en la Figura 44.

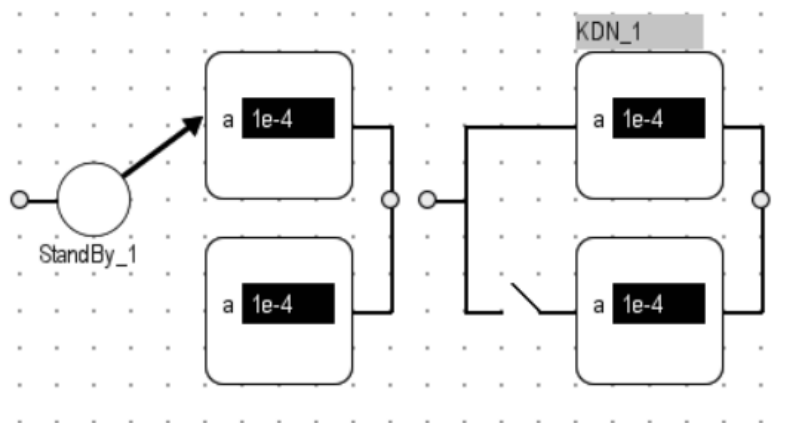


Figura 44: Elementos **Stand By** y **k d n**

Estos elementos pueden configurarse de manera a similar al elemento **Paralelo**, esto es,

mediante el accionamiento del clic derecho sobre estos (conmutador para el **Stand By** y etiqueta para el **k d n**), dando lugar a las ventanas emergentes que se observan en las Figuras 45 y 46.

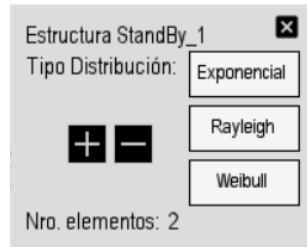


Figura 45: Propiedades **Stand By**

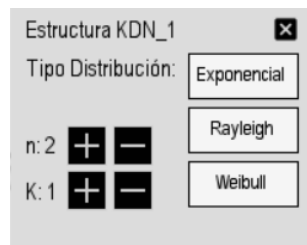


Figura 46: Propiedades **k d n**

Para el **Stand By**, es posible adicionar hasta 5 elementos, así como el tipo de distribución que estos asumen. Para el caso del **k d n**, es posible tener hasta 5 elementos en total (n) y hasta 5 de estos elementos buenos (K), siendo $n \geq K$. Para el presente ejemplo se utiliza un **Stand By** con 2 elementos que asumen una distribución de tipo *Exponencial*, y para el **k d n** se cuenta con un total de 2 elementos ($n = 2$) con 1 de estos buenos ($K = 1$) asumiendo una distribución de tipo *Exponencial*. Para ambos casos se utiliza el valor por defecto de *Alpha* ($1e - 4$). Se dispone entonces a realizar una conexión en serie para estos elementos redundantes, partiendo desde el nodo **Inicio** y pasando por ambos elementos, hasta llegar al nodo **Fin**. Dicha conexión se puede observar en la Figura 47.

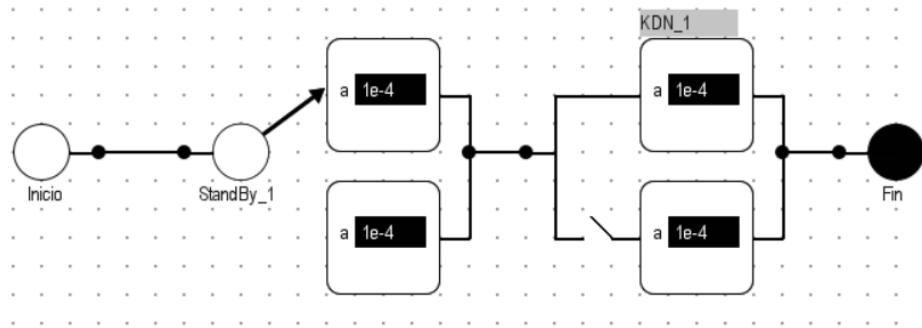


Figura 47: Sistema para el ejemplo 3

Ya con los elementos debidamente conectados, se procede ahora a evaluar el comportamiento de las diferentes métricas que evalúan la importancia de cada uno de los elementos. Para ello, se hace clic sobre la opción **Análisis del sistema** . Dentro de esta sección es posible analizar no solo la confiabilidad que asume el sistema a lo largo del tiempo, sino también la criticidad que presenta cada elemento en un punto dado de su operación. En las Figuras 50 se observa las curva que se presenta por defecto, la curva de confiabilidad.

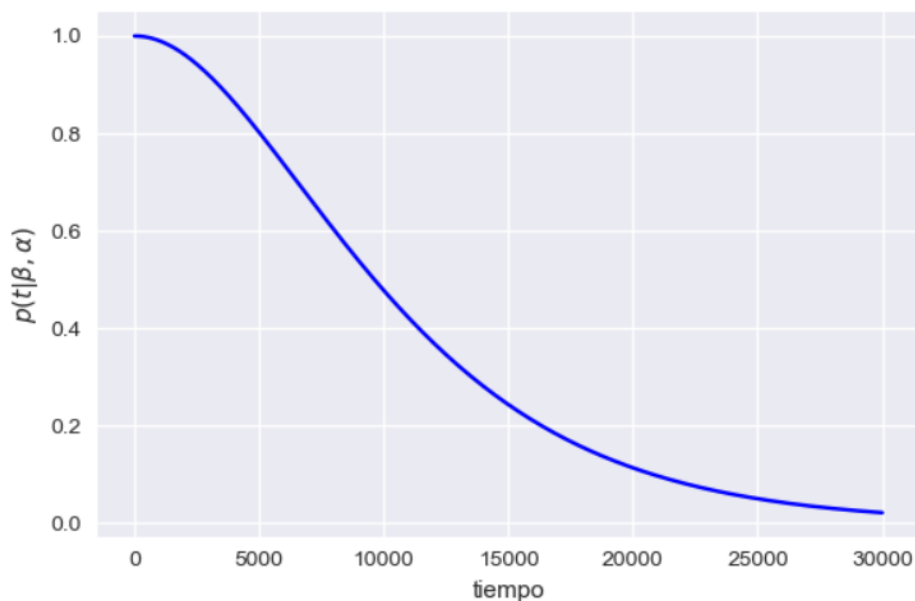


Figura 48: Confiabilidad del sistema redundante en serie

En las Figuras 49-53 se observan las curvas de importancia para las métricas disponibles en la plataforma.

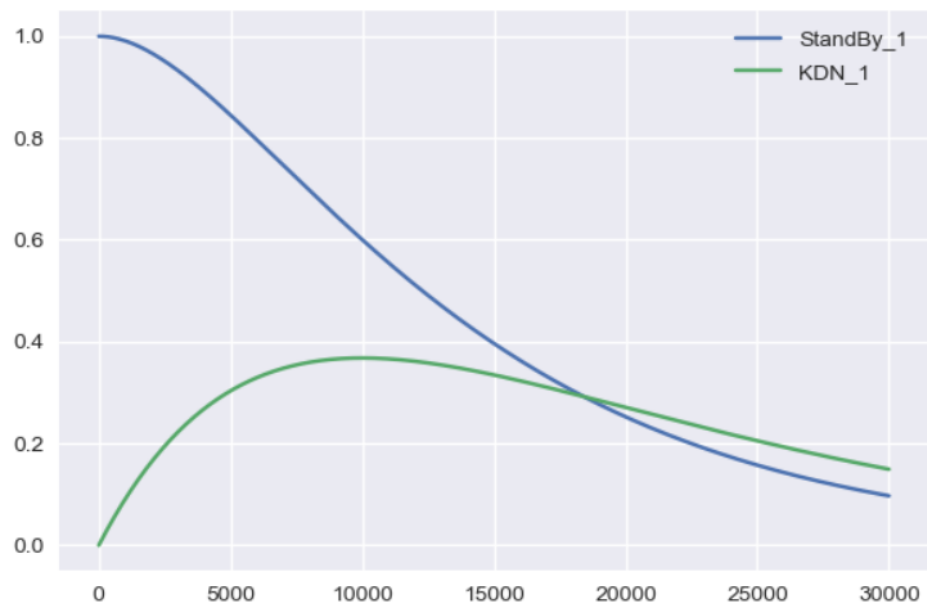


Figura 49: Medida de Birnbaum para el ejemplo 3

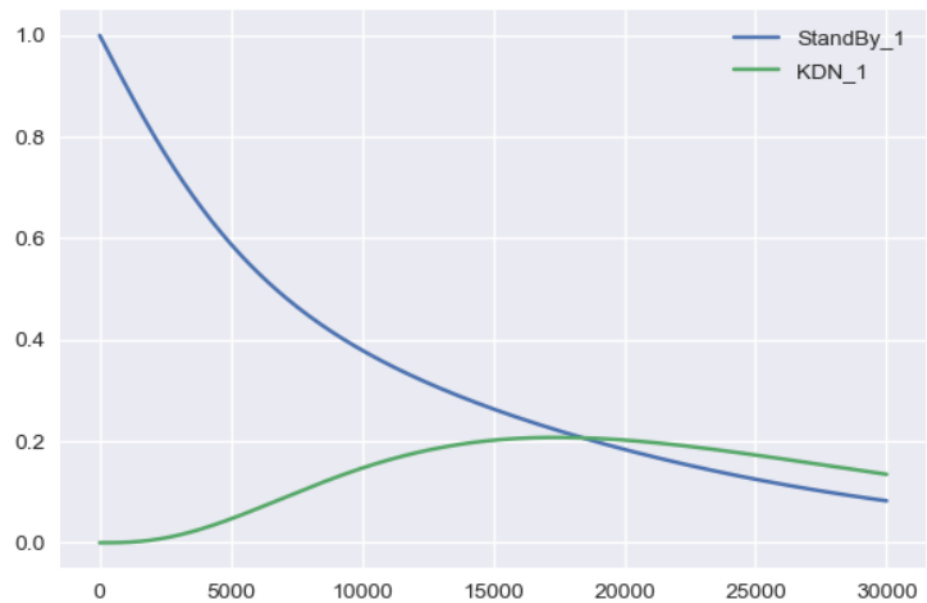


Figura 50: Medida de potencial de mejora para el ejemplo 3

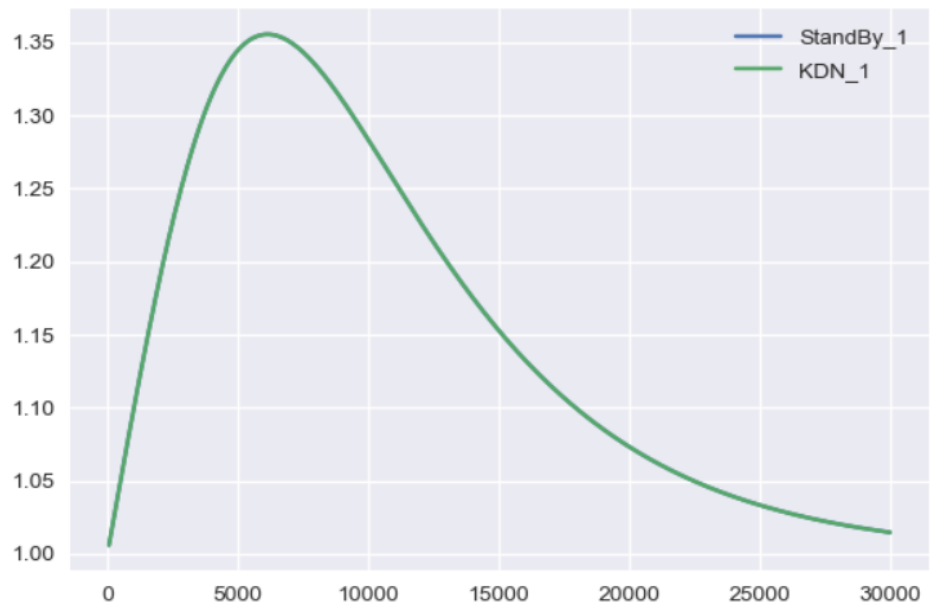


Figura 51: Medida de valor de logro de riesgo para el ejemplo 3

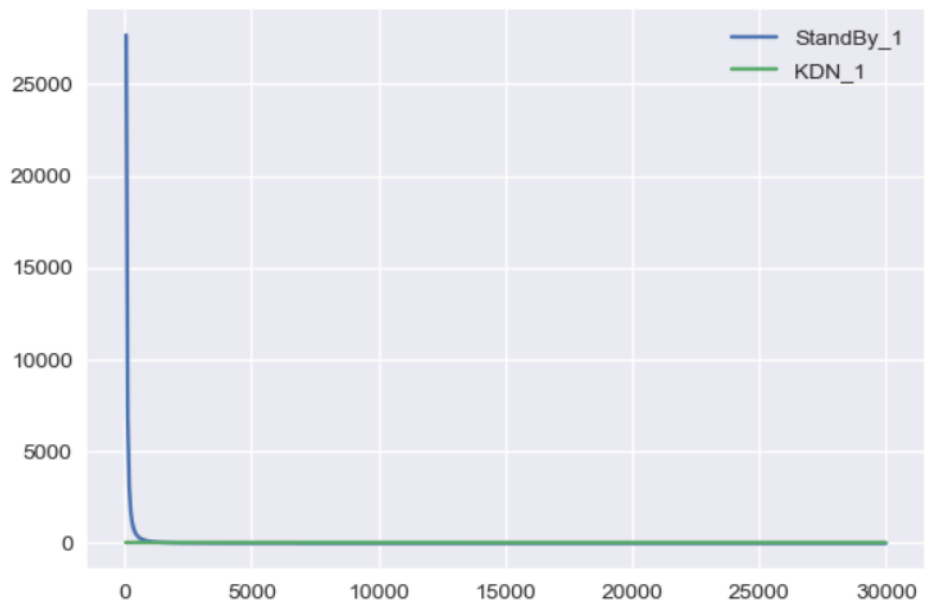


Figura 52: Medida de valor de reducción de riesgo para el ejemplo 3

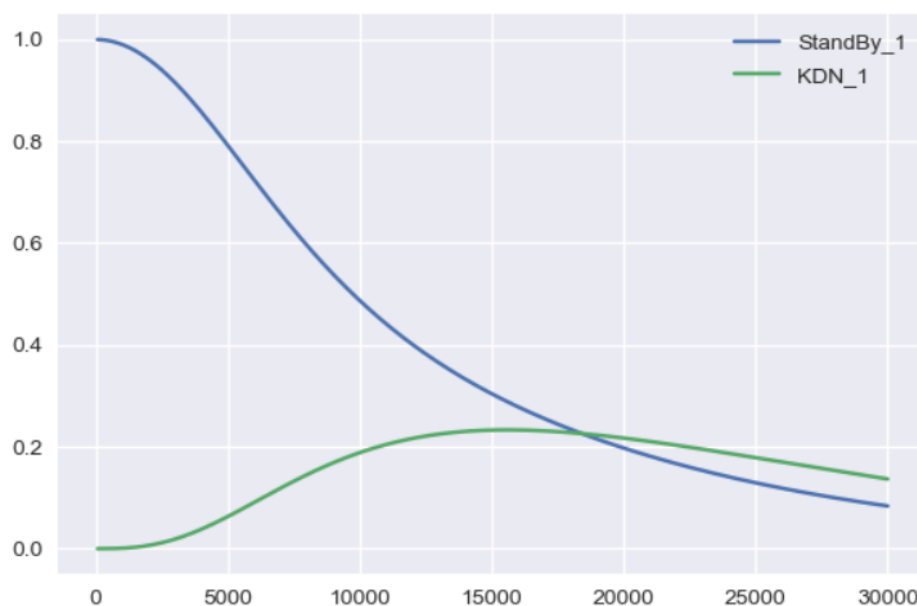


Figura 53: Medida de importancia de la criticidad para el ejemplo 3

En la mayoría de las medidas se llega a un valor de tiempo en el que el bloque **k d n** pasa a tener una mayor importancia que el **StandBy**, en un tiempo aproximado de 18000. Para el caso del valor de logro de riesgo (Figura 51) al encontrarse en serie todos los bloques tienen la misma importancia. Dado a que las medidas en que se traslapan las curvas presentan un comportamiento similar, se procede a observar los cambios que se presentan en ellas por medio de la medida de importancia de la criticidad, esto al realizar cambios en los parámetros de los bloques del sistema. Se inicia entonces por la adición de un bloque extra dentro del elemento **StandBy** por medio del botón Agregar dentro de su ventana de parámetros. La adición de este bloque da lugar a la curva que se observa en la Figura 54.

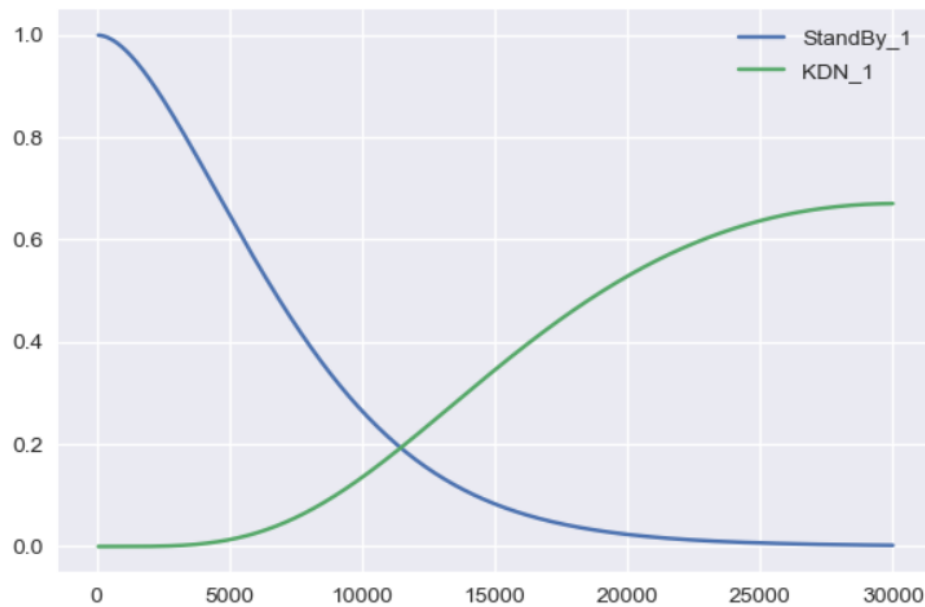


Figura 54: Importancia al aumentar número de bloques en **StandBy**

Esta curva permite observar que un bloque extra en el elemento **StandBy** disminuye el tiempo en que se traslapan las curvas, esto debido a que se está incrementando su confiabilidad, por tanto disminuyendo la criticidad del elemento en caso de presentar una falla. En caso de que se desee disminuir la criticidad que presenta el elemento **k d n** es necesario entonces incrementar la cantidad de bloques, esto mediante el incremento de la variable **n**. Se realiza entonces la adición de dos bloques más, así como la sustracción del bloque previamente agregado dentro del elemento **StandBy**. En la Figura 55 se observa la notoria disminución en la criticidad del elemento **k d n** al aumentar el numero de sus componentes.

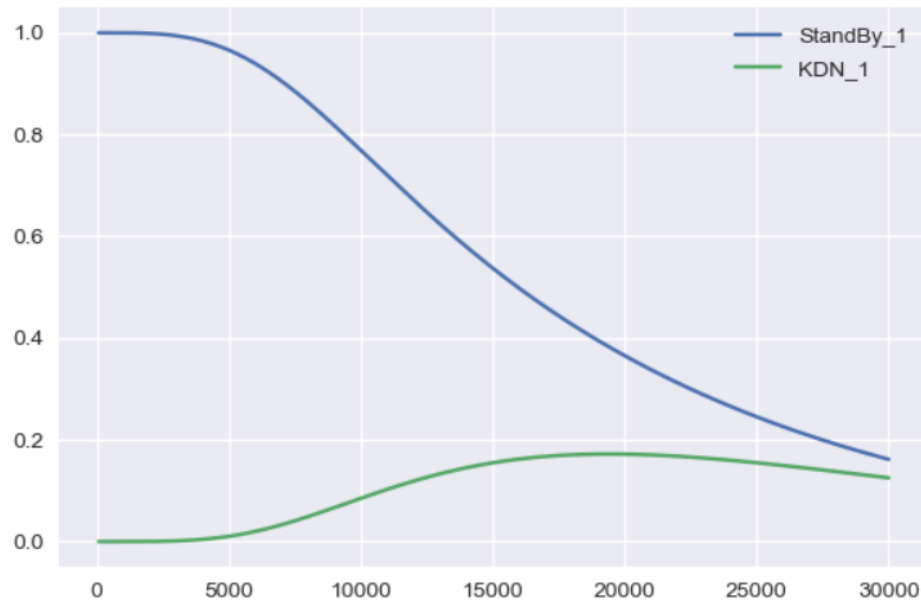


Figura 55: Importancia al aumentar el número de elementos (n) en K d N

En caso de aumentar el número de componentes buenos que requiere el **k d n** (variable K), la importancia del componente debe aumentar, esto debido a que al exigir un mayor número de componentes en operación, también aumenta la probabilidad de falla de alguno de estos, implicando un mayor riesgo para el sistema. En la Figura 56 se observa el efecto de aumentar dicha variable K.

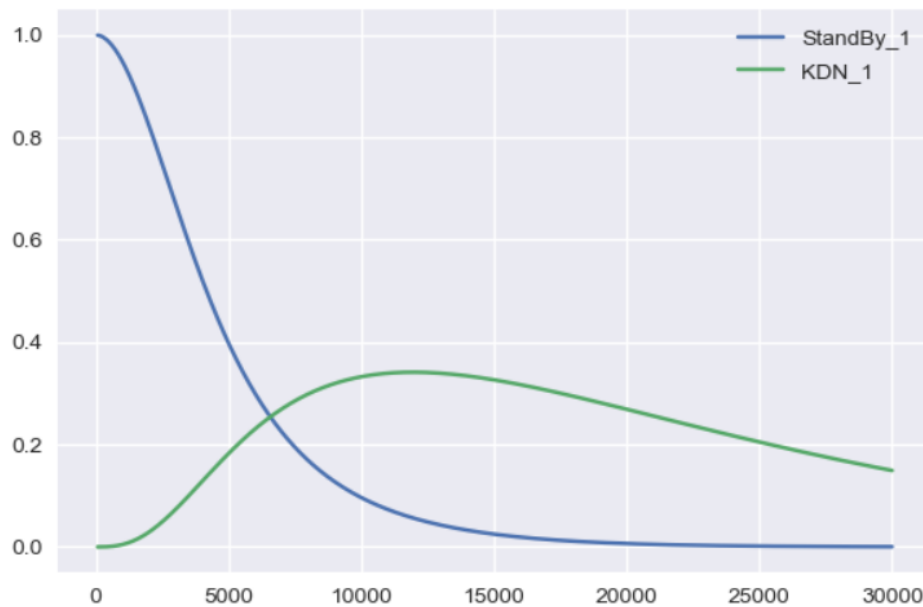



Figura 56: Importancia al aumentar el número de el número de elementos buenos (K)

4.4. EJEMPLO 4 - SUBSISTEMAS

Para este ejemplo, se dispone a la creación un sistema compuesto de subsistemas, siendo los subsistemas cada uno de los ejemplos previamente creados. Por tanto, para coherencia del presente, se realiza una reasignación de nombres a los sistema que contienen a cada uno de los ejemplos. Esto es posible seleccionando el sistema deseado dentro de la lista de sistemas disponibles, y posteriormente la acción **Renombrar** . Para el caso de la sistema 1, el cual cuenta con el nombre por defecto *untitled_1*, se procede a renombrarla mediante la etiqueta *Serie*, tal como se observa en la Figura 57.

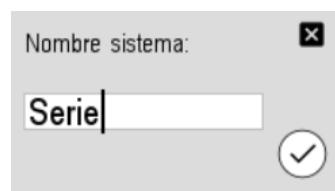



Figura 57: Asignación de nombre para el sistema 1

Para validar el nuevo nombre se hace clic sobre el botón **Okay** . Las pestañas *untitled_2* y *untitled_3* se renombran como *Paralelo* e *Híbrido*. En la Figura 58 se observan los 3 sistemas con sus nuevos nombres.

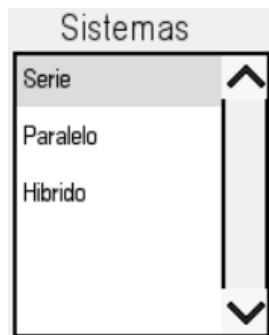



Figura 58: Sistemas con nuevos nombres

Se procede ahora a exportar cada uno de estos sistemas como elementos del tipo **Subsistema**, esto por medio de la acción **Exportar** , dando lugar a la creación de subsistemas. Para realizar esta acción es necesario que todos los elementos dentro de una hoja de diseño se encuentren debidamente conectados, en caso de que esto se incumpla se enseña una ventana de advertencia como la que presente en la Figura 59.

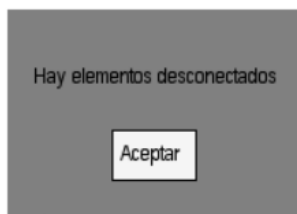



Figura 59: Advertencia al tratar de exportar un subsistema

En caso de que este cuadro de diálogo no se presente, significa que el subsistema ha sido debidamente exportado y se almacena con el nombre que posea su respectiva pestaña. Ya con cada uno de estos debidamente exportados, es posible llamarlos como elementos de un nuevo sistema, esto por medio de la acción **Importar** , lo que da lugar a una

ventana emergente que contiene los nombres de los subsistemas disponibles, tal como se observa en la Figura 60.



Figura 60: Lista de subsistemas disponibles

En caso de no existir subsistema alguno, se presenta la ventana de advertencia que se observa en la Figura 61.

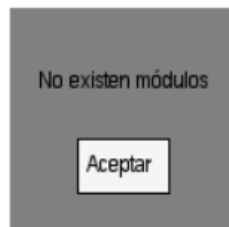


Figura 61: Advertencia al no existir subsistemas

Para elegir el subistema deseado basta con seleccionarlo en la lista desplegable y luego hacer clic sobre el botón **Aceptar**, dando lugar a un bloque negro tal como el que se observa en la Figura 62.

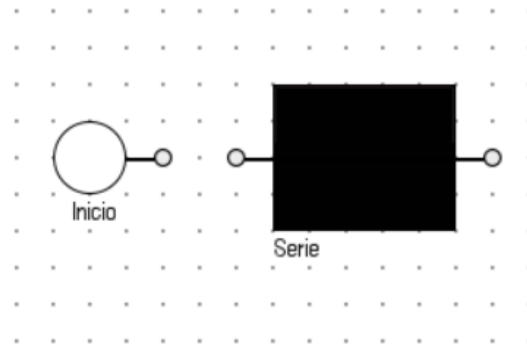


Figura 62: Subsistema dentro del área de trabajo

Es posible acceder al contenido de cada subsistema haciendo doble clic sobre éste, dando lugar a la apertura de la pestaña que contiene los elementos reales del subsistema. Se adicionan entonces los demás subsistemas dentro de la hoja de diseño, dando lugar a su posterior conexión en serie, tal como se observa en la Figura 63.

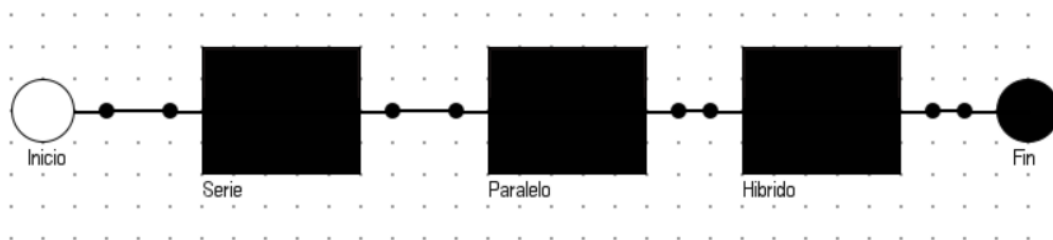



Figura 63: Sistema compuesto de subsistemas

Es entonces posible realizar un análisis sobre los diferentes subsistemas, siendo esto el equivalente a evaluar la confiabilidad total en cada una de las pestañas que los contienen. Por tanto, se procede a observar la confiabilidad y medida de la importancia del sistema por medio de la opción **Análisis del sistema** , dando lugar a la curva de confiabilidad que se observa en las Figura 65.

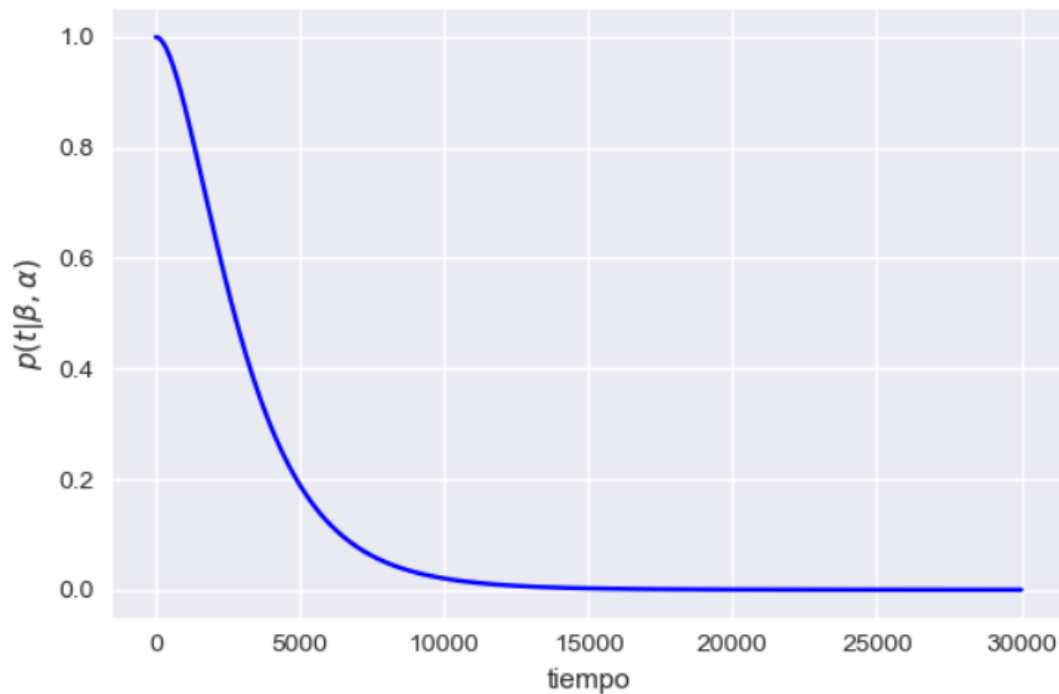


Figura 64: Confiabilidad total del sistema para el ejemplo 4 (con $t=30000$)

Es posible modificar el valor de tiempo para apreciar con mayor claridad el momento en la curva converge a valores cercanos a 0, esto para un valor de tiempo de aproximadamente 10000. Para ello, basta con posicionar el cursor sobre la caja de **Tiempo**, dando lugar a su modificación. Para validar el nuevo valor, es necesario hacer clic sobre el botón **Okay** ☒. En la Figura 65 se observa la confiabilidad del sistema evaluada hasta un tiempo de 10000.

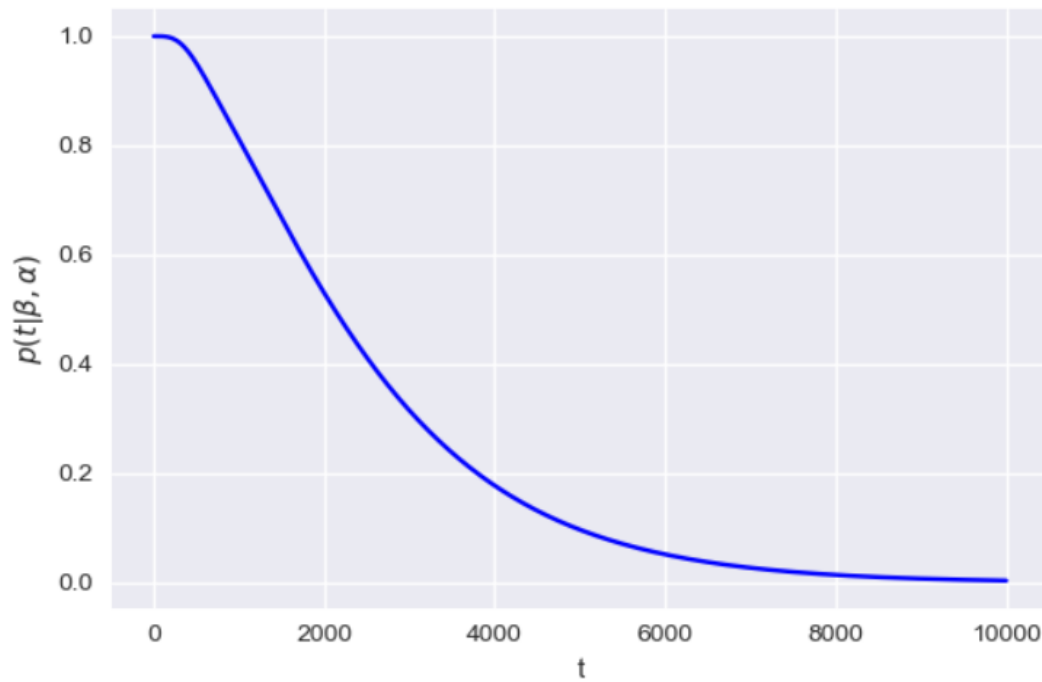


Figura 65: Confiabilidad total del sistema para el ejemplo 4 (con $t=10000$)

Finalmente se procede a observar la importancia que tiene cada elemento sobre el sistema. En la Figura 66 se observa la medida de importancia crítica de cada uno de los elementos.

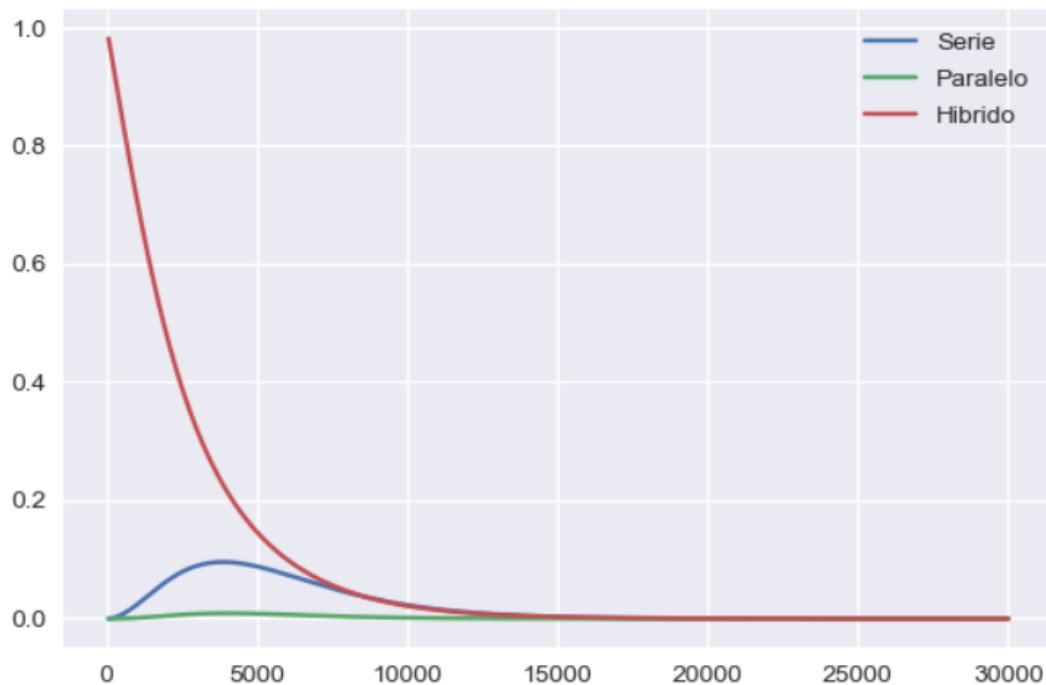


Figura 66: Medida de la importancia por parte de cada uno de los subsistemas (con $t=30000$)

Se observa que es el subsistema paralelo el que menor criticidad presenta, y a su vez es el sistema híbrido (**StandBy** y **k d n**) el que es más crítico en todo instante de tiempo. Se modifica el tiempo de la curva para apreciar con mayor claridad la convergencia de los elementos, tal como se observa en la Figura 67.

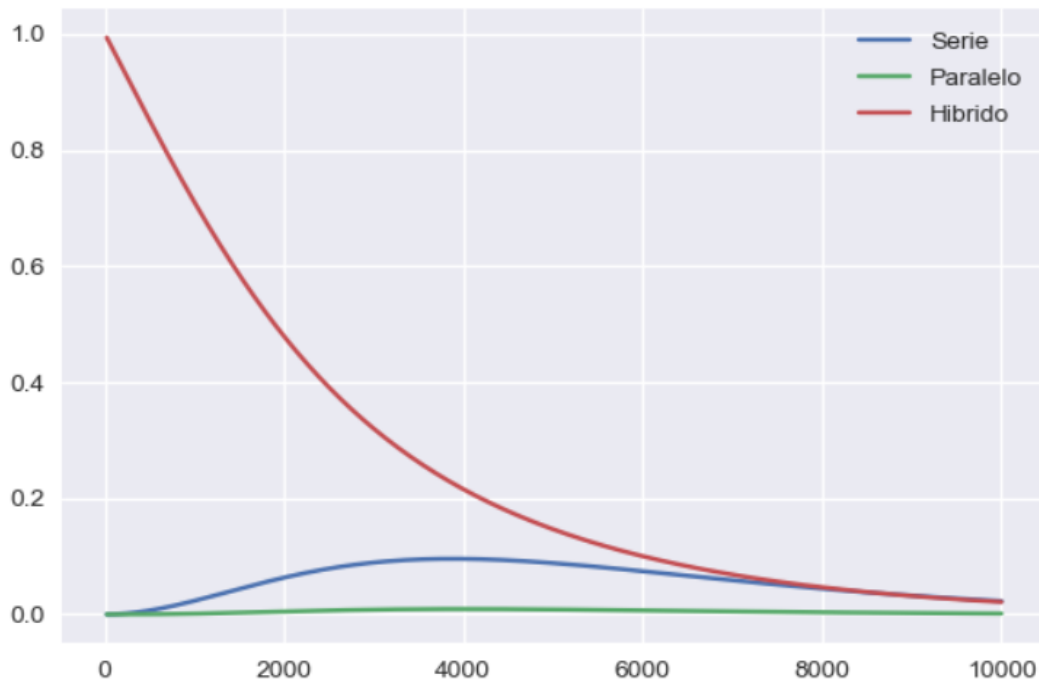


Figura 67: Medida de la importancia por parte de cada uno de los subsistemas (con $t=10000$)

5. METODOLOGÍA IMPLEMENTADA

REL-Imp es un software portable desarrollado en el lenguaje de programación Python, bajo el paradigma de la programación orientada a objetos. Este lenguaje, además de ofrecer módulos para el tratamiento de arreglos y cálculos matemáticos, posee la librería Pygame, la cual permite el desarrollo de software que se basa en módulos de videojuegos en 2D, siendo ésta ideal para la creación de interfaces que requieren de una mayor interacción entre el usuario y el sistema. Dentro de este módulo se encuentran características de especial interés, como lo son los eventos por ratón y teclado, los cuales son piezas esenciales en el desarrollo de entornos gráficos interactivos.

REL-Imp es una herramienta para la identificación de los componentes críticos dentro

de un sistema representado por diagramas de bloques (método gráfico por preferencia para la representación de sistemas físicos dentro del estudio de confiabilidad), esto mediante diferentes medidas de importancia. A continuación se describen brevemente la aspectos presentes dentro del desarrollo de **REL-Imp**.

5.1. MEDIDA DE LA CRITICIDAD

El análisis de criticidad es una metodología que permite establecer la jerarquía o prioridades de procesos, sistemas y equipos, creando una estructura que facilita la toma de decisiones acertadas y efectivas, direccionando el esfuerzo y los recursos en áreas donde sea más importante y/o necesario mejorar la fiabilidad operacional, basado en la realidad actual.

Los valores de confiabilidad, tiempo entre fallos, tiempo de reparación, son de gran importancia en la industria actual. Estos datos son utilizados para mejorar diseños o funciones de maquinas industriales.

Durante el diseño del análisis de la confiabilidad o le evaluación de riesgo de un sistema, los componentes y su configuración puede hacer que unos sean más críticos que otros desde el punto de vista del impacto en el sistema de confiabilidad. Por ejemplo, un conjunto de componentes en serie dentro de un sistema tiene una mayor importancia (para fallar) que si el mismo conjunto de componentes estuviera en paralelo [5]. Las diversas medidas se basan en interpretaciones ligeramente diferentes de la importancia del componente del concepto. Intuitivamente, la importancia de un componente debe depender de dos factores:

- La ubicación del componente en el sistema
- La confiabilidad del componente en cuestión

Quizás, también la incertidumbre en la estimación de la confiabilidad de los componentes.

La importancia del componente puede ser utilizada para:

- Proporcionar una clasificación aproximada de los componentes con respecto a su influencia en la confiabilidad del sistema (o la probabilidad máxima de un evento en un árbol de fallos)
- Ayuda para centrarse en los principales contribuyentes a la falta de fiabilidad del sistema
- Ayudar a relajar las relaciones con los contribuyentes más bajos a la falta de fiabilidad del sistema
- Enfoque en las mejoras con el mayor efecto de confiabilidad
- Indicar la sensibilidad de los parámetros del modelo
- Ayuda para enfocar las revisiones y los estudios de sensibilidad
- Dar prioridad a la búsqueda de fallos en un sistema complejo

Se presentan entonces varios tipos diferentes de medidas de importancia y sus significados.

5.2. La medida de Birnbaum

Birnbaum (1969) propuso la medida de la importancia de la fiabilidad del componente i en la época t , la cual se define por medio de la siguiente ecuación:

$$I^B(i|t) = \frac{\partial h(p(t))}{\partial p_i(t)}$$

Por lo tanto, la medida de *Birnbaum* se obtiene como la derivada parcial de la confiabilidad del sistema $h(p(t))$ con respecto a $p_i(t)$. Este enfoque es bien conocido del análisis de sensibilidad clásico. Si $I^B(i|t)$ es grande, un pequeño cambio en la confiabilidad del componente i resulta en un cambio comparativamente grande en la confiabilidad del sistema en el momento t . Al tomar esta derivada, las confiabilidades de los otros componentes permanecen constantes - sólo se estudia el efecto de variar $p_i(t)$.

En un estudio práctico de confiabilidad de un sistema complejo, una de las tareas que más tiempo consume es encontrar estimaciones adecuadas de los parámetros de entrada (tasas de fallo, tasas de reparación, entre otros). En algunos casos, es posible empezar con estimaciones bastante aproximadas, calcular la medida de importancia de *Birnbaum* para los diversos componentes, o las sensibilidades de los parámetros, y luego dedicar la mayor parte del tiempo a encontrar datos de alta calidad para los componentes más importantes. Los componentes con un valor muy bajo de la medida de *Birnbaum* tendrán un efecto insignificante en la confiabilidad del sistema, y los esfuerzos adicionales para encontrar datos de alta calidad para tales componentes pueden ser considerados una pérdida de tiempo [1].

5.3. Medida del potencial de mejora

El potencial de mejora del componente i en el momento t se define como:

$$I^{IP}(i|t) = h(1_i, p(t)) - h(p(t))$$

$I^{IP}(i|t)$ es, por lo tanto, la diferencia entre la confiabilidad del sistema con un componente perfecto i , y la confiabilidad del sistema con el componente real i . Indica hasta qué punto es posible mejorar la confiabilidad del sistema actual si pudiese sustituirse el componente actual i por un componente perfecto.

En la práctica, no es posible mejorar la confiabilidad $p_i(t)$ del componente i a un 100 % [1].

5.4. Valor de logro de riesgo

El valor del logro del riesgo (RAW) mide el aumento relativo de la falta de confiabilidad del sistema cuando se sabe que el componente i está en un estado de fallo. La falta de confiabilidad nominal del sistema es $1 - h(p(t))$, y cuando se sabe que el componente i ha fallado, la confiabilidad del sistema es $1 - h(0_i, p(t))$. El aumento relativo de la inestabilidad del sistema es por lo tanto:

$$I^{RAW}(i|t) = \frac{1 - h(0_i, p(t))}{1 - h(p(t))} = \frac{Q_0(0_i, t)}{Q_0(t)}$$

Lo cual, para todos los sistemas coherentes, cumple $I^{RAW}(i|t) \geq 1$. Cuando $I^{RAW}(i|t) = 1$, el fallo del componente i (o la presencia del evento básico i) no tiene efecto. El RAW

se considera útil para estimar la importancia del riesgo de los componentes (o características) que se eliminan del sistema [1].

5.5. Valor de reducción de riesgo

El valor de reducción de riesgos (RRW) mide la reducción relativa de la falta de confiabilidad del sistema cuando se sabe que el componente i está funcionando, donde la reducción se mide en relación con la **mejor situación** cuando el componente i está funcionando. La reducción relativa de la falta de confiabilidad del sistema es:

$$I^{RRW}(i|t) = \frac{1 - h(p(t))}{1 - h(1_i, p(t))} = \frac{Q_0(t)}{Q_0(1_i, t)}$$

Lo cual, para todos los sistemas coherentes, cumple $I^{RRW}(i|t) \geq 1$. Cuando $I^{RRW}(i|t) = 1$, mejorar el componente i para que funcione en todo momento no tiene ningún efecto alguno. El RRW se considera útil para delimitar los beneficios de riesgo de las mejoras propuestas [1].

5.6. Medida de importancia de la criticidad

La medida de Importancia de la Criticidad $I^{CR}(i|t)$ del componente i en el tiempo t es la probabilidad de que el componente i sea crítico para el sistema y falle en el tiempo t , cuando se conoce que el sistema falla en el tiempo t .

$$I^{CR}(i|t) = \frac{I^B(i|t) * (1 - h(p(t)))}{1 - h(p(t))}$$

Cabe recordar que cuando el componente i es crítico, los otros componentes están en tal estado que el sistema fallará si y sólo si el componente i falla. Esto también significa que el sistema comenzará a funcionar de nuevo si se repara el componente i .

$I^{CR}(i|t)$ es por lo tanto la probabilidad de que el componente i haya causado una falla en el sistema, cuando se conoce que el sistema está fallando en el momento t . Para que el componente i cause una falla en el sistema, el componente i debe ser crítico y luego fallar. Cuando se repare el componente i , el sistema comenzará a funcionar de nuevo. Es por esto que la medida de importancia crítica puede ser usada para priorizar las acciones de mantenimiento en sistemas complejos [1].

5.7. DIAGRAMAS DE BLOQUES

En confiabilidad, un diagrama de bloques es una representación gráfica de la conexión lógica de los subsistemas y elementos que conforman un sistema. La conexión lógica refleja el comportamiento de los elementos y subsistemas en conjunto desde un punto de vista del aporte a la confiabilidad [3]. Un subsistema puede ser analizado de forma independiente y luego verse como caja negra para el análisis de un nivel mayor. Con los diagramas de bloques, se persigue como objetivo asignar un valor de confiabilidad a cada elemento, subsistema y al sistema partiendo desde el nivel de menor jerarquía hacia el de mayor jerarquía, relacionando componentes o subsistemas en sistemas de mayor nivel, vistos como cajas negras. Por tanto, el sistema de mayor jerarquía es un sistema reducible en subsistemas de menor jerarquía. Se presentan entonces las siguientes topologías.

5.7.1. Confiabilidad en serie

Se asume que los componentes operan de forma independiente uno en relación con los demás. La confiabilidad del sistema R_S con base en la confiabilidad de n componentes en serie, con R_i la confiabilidad del i -ésimo componente, es la probabilidad que la variable aleatoria T tome valores superiores a t tanto para el sistema como para cada componente:

$$R_s = P(t > T) \quad R_i = P(t_i > T)$$

Entonces, la confiabilidad del sistema es la probabilidad que la variable aleatoria T tome valores superiores a cada tiempo t_i .

$$R_s = P(\wedge t_i < T) = \prod_{i=1}^n P(t_i < T) = \prod_{i=1}^n R_i(t)$$

En la Figura 68 se observa la estructura de un sistema en serie.

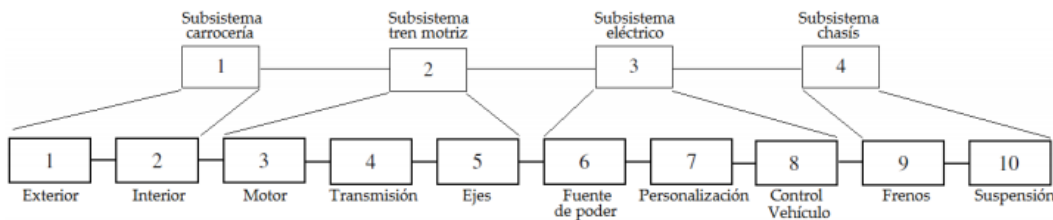


Figura 68: Sistema en serie

5.7.2. Confiabilidad en paralelo

Se define redundancia para un ítem, como la existencia de más de un medio en un instante dado de tiempo para realizar la función requerida. Un sistema está compuesto

por un conjunto de elementos y/o subsistemas relacionados en conexión lógica paralela cuando la falla del sistema sólo ocurre con la falla de todos ellos. En otras palabras, el sistema es confiable con que sólo un elemento sea confiable. El sistema es operacional y desempeña la función requerida si una sola unidad que lo constituye está operacional y cumplen de forma individual su función demandada.

Desde un punto de vista de probabilidad, un sistema en paralelo es inconfiable si todos los elementos son inconfiables, es decir, la inconfiabilidad en un sistema paralelo corresponde a la intersección de eventos independientes. La característica de redundancia implica la independencia de los eventos y se asume la no reparación de las unidades bajo falla hasta que todo el sistema falle. La redundancia incrementa los costos, el peso, la complejidad, el tiempo de diseño, fabricación, ensamble y prueba. Sin embargo todas estas consideraciones juegan a favor de beneficios en confiabilidad, seguridad y cumplimiento de la función requerida. En la Figura 69 se observa la estructura de un sistema en paralelo.

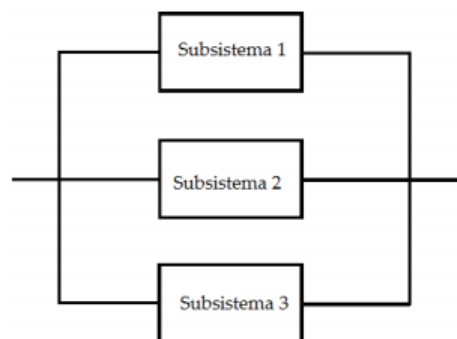


Figura 69: Sistema en paralelo

5.7.3. Redundancia activa parcial (k de $n:G$)

Redundancia activa donde se requiere en todo instante más de un medio para realizar una función requerida. Son ejemplos: sistemas de generación, sistemas de transmisión,

servidores web, sistemas de posicionamiento global, etc. Se usa la nomenclatura k de $n:G$, donde k indica el número mínimo de unidades que deben cumplir su función de las n unidades disponibles y G significa “Bueno” (Good). Por definición, un sistema paralelo en redundancia activa es 1 de $n:G$. Un sistema en serie es n de $n:G$.

Supóngase que el tiempo para falla de n componentes iguales en un sistema k de $n:G$ se distribuye de forma independiente e idénticamente. Sea x el número de componentes que deben estar operativos en el sistema. Entonces x es una variable aleatoria que sigue la distribución binomial y la probabilidad de tener exactamente k componentes operativos es:

$$P(x = k) = C_n^k R_o^k (1 - R_o)^{n-k}$$

Como en un sistema k de $n:G$ se requieren mínimo k unidades, el sistema cumple su función requerida con cualquier número de unidades operativas entre k y n :

$$R_{k/n} = P(x \geq k) = \sum_{i=k}^n C_n^i R_o^i (1 - R_o)^{n-i}$$

5.7.4. Redundancia positiva en frío (Stand By)

Una parte de los medios para realizar una función requerida está destinada a operar, mientras la parte restante está inoperativa (en reposo o en espera) hasta ser requeridos. Cuando la unidad primaria está operativa y la unidad redundante está en reserva, se requiere sensores que detecten la falla de la unidad primaria y un equipo de conmutación que active la unidad redundante con la ocurrencia de la falla (Ej: transferencia red-

planta). En general se habla del equipo de conmutación para incluir tanto a los sensores como al conmutador. Para un modelo exponencial asume el siguiente comportamiento:

$$R_t = \exp^{-\lambda t} \sum_{i=0}^{n-1} \frac{(\lambda t)^i}{i!}$$

5.7.5. Sistemas irreducibles

Algunos sistemas no pueden ser reducidos por métodos convencionales, esto debido a que no presentan conexiones en serie o en paralelo. Para afrontar dichos casos, se presentan diferentes metodologías que permiten dar lugar a la evaluación de un sistema. Entre ellos destaca el método de la descomposición, el cual se basa en el uso del principio de partición del espacio muestral y uso de los teoremas de la probabilidad condicional y probabilidad total. Para el análisis por descomposición se requiere la selección de un elemento que sea responsable de no poderse aplicar una reducción directa.

5.8. ALGORITMO DE BÚSQUEDA

Para el cálculo de la confiabilidad, se visualiza el sistema como un grafo dirigido, el cual se compone de arcos y nodos, siendo cada nodo el equivalente a todos aquellos elementos que se encuentran interconectados [7]. Un grafo, representa un conjunto de nodos unidos en una red. Los algoritmos de búsqueda de caminos son usados para encontrar una ruta de navegación entre un punto de inicio y otro objetivo. La forma más natural para representar grafos en una computadora es la matriz de adyacencia. Sea $V = v_1 \dots v_{|V|}$ el conjunto de vértices del grafo G , y E el conjunto de sus aristas. La matriz de adyacencia

será una matriz de tamaño $|V| \times |V|$, donde la entrada (i, j) será:

$$a_{i,j} = \begin{cases} 1 & \text{si existe una arista de } V_i \text{ a } V_j \\ 0 & \text{en otro caso} \end{cases} \quad (1)$$

Dentro de esta familia de algoritmos se encuentra al algoritmo DFS (Depth-First Search), el cual es una forma sistemática de encontrar todos los vértices alcanzables de un grafo desde un vértice de origen. Su funcionamiento consiste en ir expandiendo todos y cada uno de los nodos que va localizando, de forma recurrente, en un camino concreto. Cuando ya no quedan más nodos que visitar en dicho camino, regresa, de modo que repite el mismo proceso con cada uno de los hermanos del nodo ya procesado. A continuación, se describen los pseudocódigos que dan lugar a la correcta ejecución del algoritmo de búsqueda.

```
1: procedure GRAFO( $v$ )
2:   for vertice  $u$  in  $V[G]$  do
3:     estado[ $u$ ] <- NO VISITADO
4:     PADRE[ $u$ ] <- NULO
5:   end for
6:   tiempo <- 0
7:   for vertice  $u$  do
8:     if estado[ $u$ ] = NO VISITADO then
9:       DFS_VISITAR( $u$ , tiempo)
10:    end if
11:  end for
12: end procedure
```

```

1: procedure DFS_VISITAR(nodou, inttiempo)
2:   estado[u] <- NO VISITADO
3:   timepo <- tiempo+1
4:   d[u] <- tiempo
5:   for V in Vecinos[u] do
6:     if estado[v] == NO VISITADO then
7:       PADRE[u] <- NULO
8:       DFS_Visitar(v, tiempo)
9:     end if
10:  end for
11:  estado[u] <- TERMINADO
12:  timepo <- tiempo+1
13:  f[u] <- tiempo
14: end procedure

```

5.9. MODELOS UML

Para la correcta visualización e interpretación del software, se hace uso de las herramientas del lenguaje de modelado “UML”, el cual es un estándar aprobado por la ISO como un modelo general para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. Estos se dividen en las siguientes tres categorías: clasificación estructural, comportamiento dinámico, y la gestión del modelo. Dentro de cada una se exponen dos modelos distintos, esto con la finalidad de captar y enumerar exhaustivamente los requisitos y el dominio de conocimiento, de forma que todos los implicados

puedan entenderlos y estar de acuerdo con ellos. Los diversos modelos de un sistema de software pueden capturar requisitos sobre su dominio de aplicación, las formas en que los usuarios harán uso de éste, su división en módulos, los patrones comunes usados en un sistema, entre otras [8]. A continuación, se presentan los modelos implementados dentro de cada una de las clasificaciones previamente mencionadas.

Estructurales: Dentro de esta clasificación se describe los elementos del sistema y sus relaciones con otros elementos. Para la presente clasificación se enseñan el diagrama de clases, el diagrama de componentes y el diagrama de paquetes.

- Diagrama de componentes: Este permite visualizar la estructura de alto nivel del sistema y el comportamiento del servicio que estos componentes proporcionan y usan a través de la interfaz. Dentro de este modelo se observa al usuario como principal actor dentro de los módulos de la interfaz, siendo este quien tiene acceso a cada uno de los demás módulos restantes.

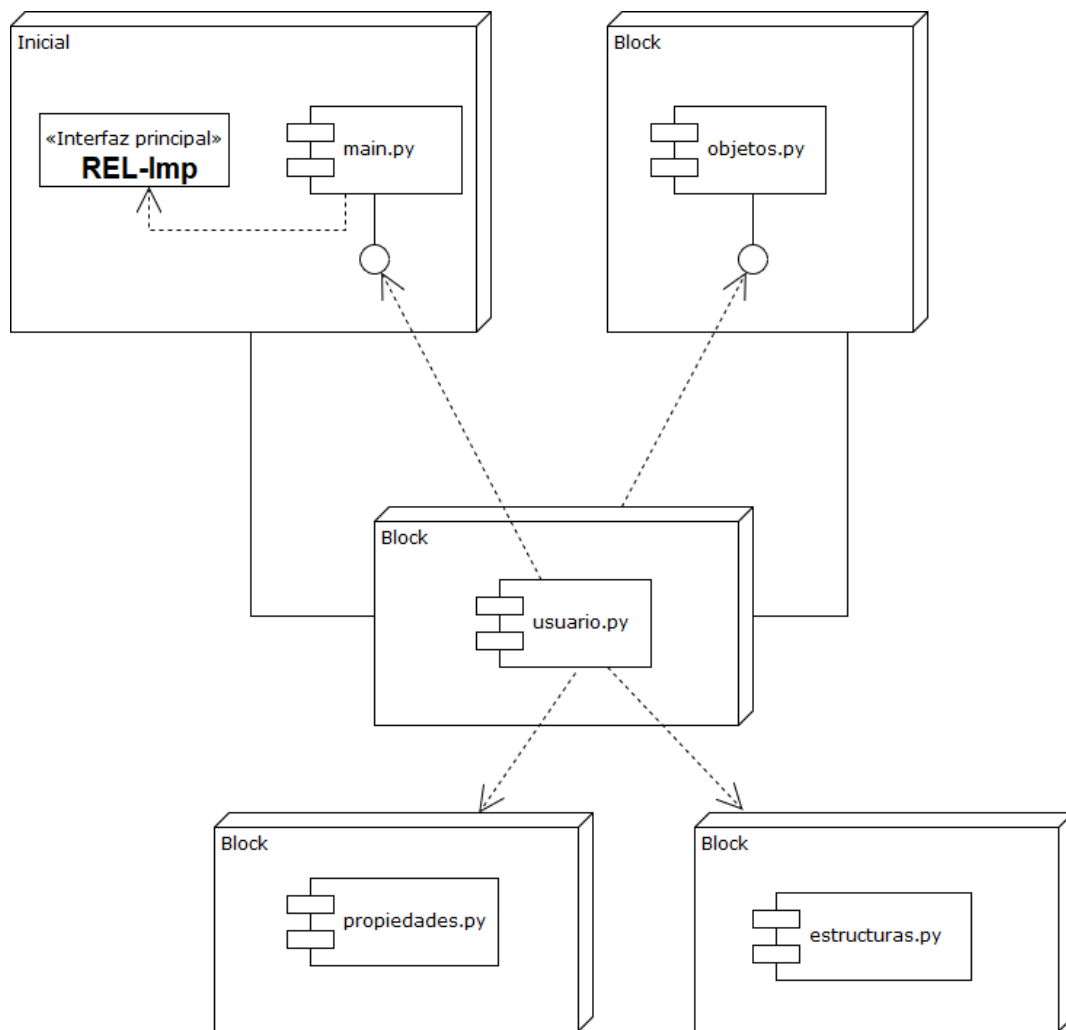


Figura 70: Diagrama de componentes

- Diagrama de paquetes: Permite dividir un modelo para agrupar y encapsular sus elementos en unidades lógicas individuales. Por medio del presente diagrama se visualiza la relación existente entre cada uno de los módulos de la interfaz, incluyendo el módulo encargado de la carga de los eventos y renderización de gráficos.

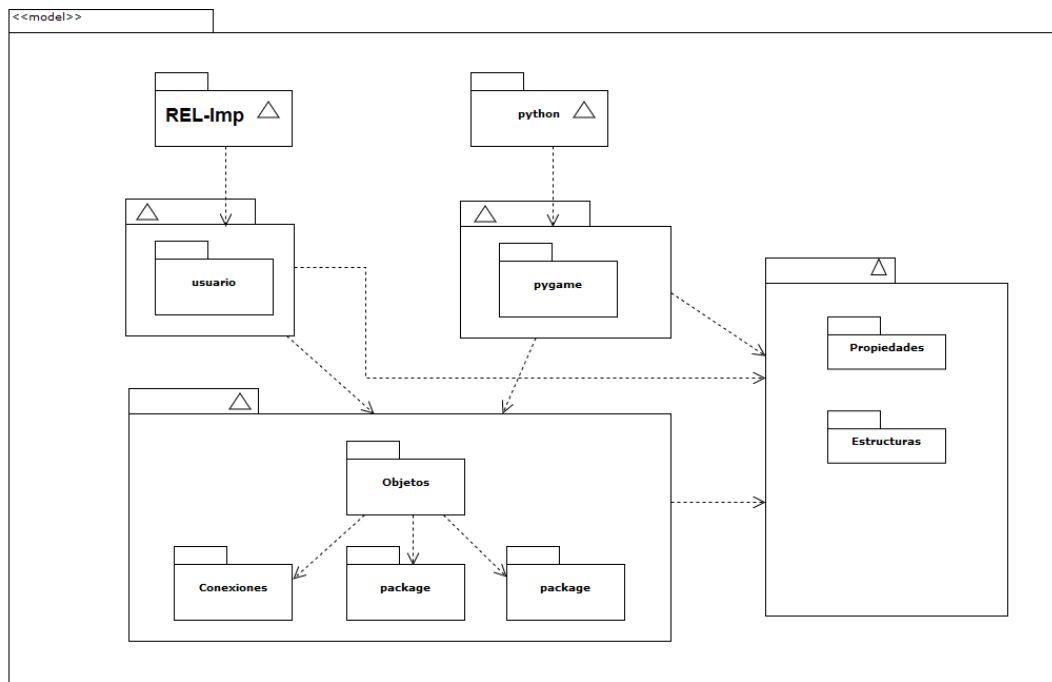


Figura 71: Diagrama de paquetes

Comportamiento: Enseña el comportamiento dinámico de los objetos en el sistema. Se presenta por medio del diagrama de actividades.

- Diagrama de actividades: Permite describir las actividades que realiza cada modulo del programa, así como la relación que tiene con los distintos módulos presentes. Dentro de este se observa la interacción entre las distintas acciones presentes en la interfaz, aclarando cual es el inicio del programa y cual puede ser considerado como el punto final de éste.

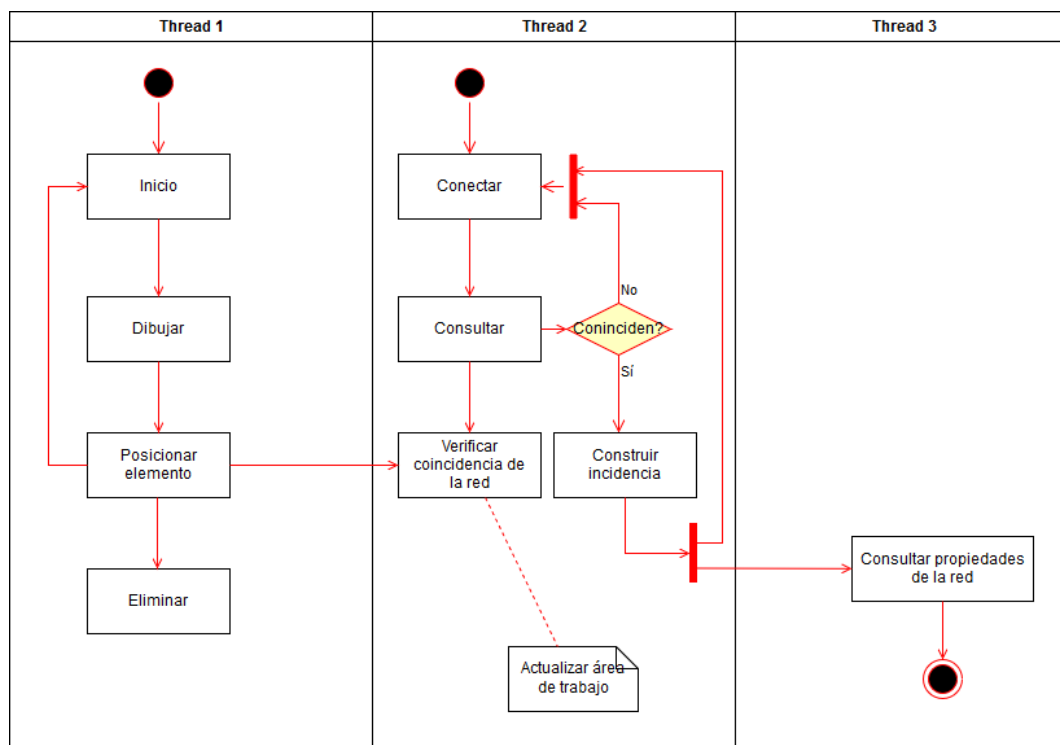


Figura 72: Diagrama de actividades

Gestión del modelo: Describe la organización de los propios modelos en unidades jerárquicos. La implementación presente es a través del diagrama de secuencias.

- Diagrama de secuencias: Permite determinar de manera gráfica el orden en que se ejecuta cada acción dentro del programa. En este se visualiza la jerarquía de las acciones, teniendo cada una de estas que respetar un orden para cumplir con la correcta ejecución del programa.

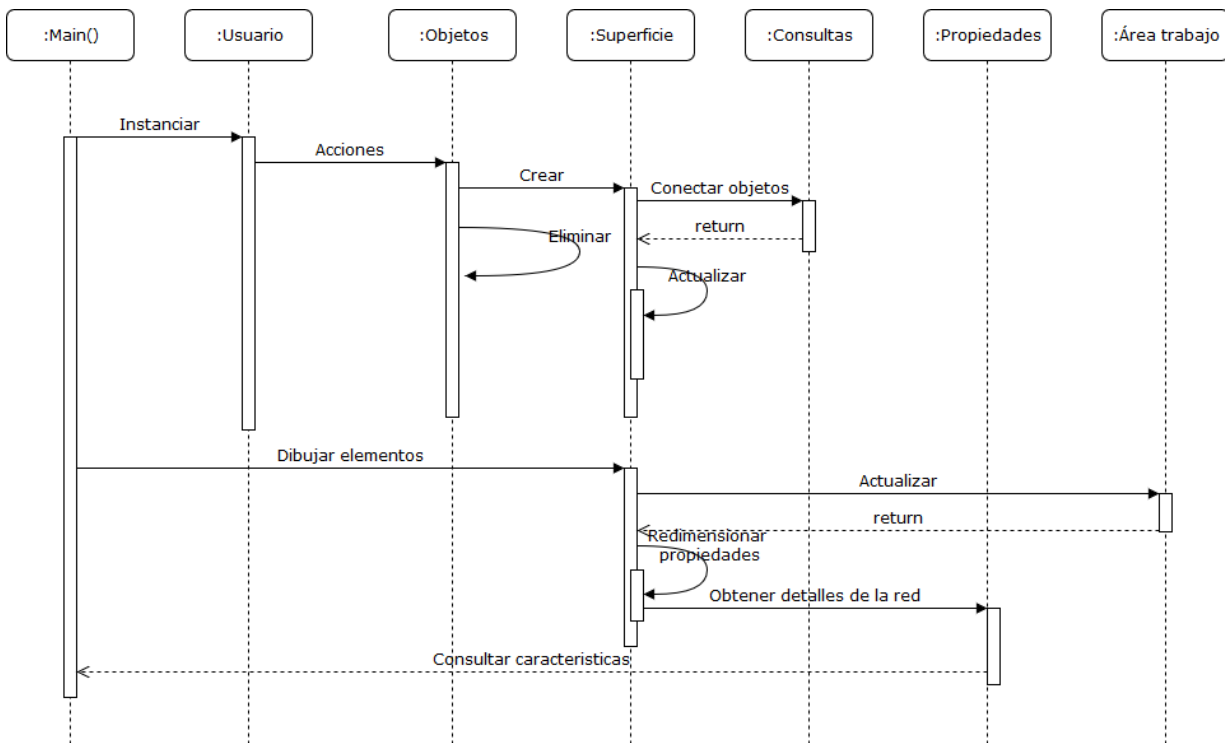


Figura 73: Diagrama de secuencias

6. EVALUACIÓN DE LA CALIDAD DEL SOFTWARE

6.1. INTRODUCCIÓN

Como parte del proceso de evaluación para productos de Software al interior de la Universidad Tecnológica de Pereira, el presente documento presenta los aspectos preponderantes durante el proceso de planeación, diseño, implementación, verificación y validación del presente producto de Software, denominado REL-Imp.

A nivel mundial, existe un conjunto de estándares como la normas ISO/IEC 9126, NTP/ISO 17799, ISO 27001, BS 25999 y la ISO/IEC 19505, las cuales permiten dar

guías y buenas prácticas para la evaluación de la calidad de software, su seguridad y su integridad, siendo todas ellas ampliamente difundidas, aceptadas y consideradas como referentes internacionales.

El estándar ISO/IEC 9126 presenta un modelo para la evaluación de la calidad del software mediante el siguiente conjunto de características: Funcionalidad, Fiabilidad, Usabilidad, Eficiencia, Mantenibilidad, Portabilidad y Calidad en Uso. A su vez, cada una de estas características se define y divide en un conjunto de sub-características, de la siguiente manera:

- Funcionalidad - atributos que se relacionan con la existencia de un conjunto de funciones y sus propiedades específicas.
 - Idoneidad
 - Exactitud
 - Interoperabilidad
 - Seguridad
 - Cumplimiento de normas
- Fiabilidad - atributos relacionados con la capacidad del software de mantener su nivel de prestación bajo condiciones establecidas durante un período establecido.
 - Madurez
 - Recuperabilidad
 - Tolerancia a fallos

- Usabilidad - atributos relacionados con el esfuerzo necesario para su uso, por un conjunto de usuarios.
 - Aprendizaje
 - Comprensión
 - Operatividad
 - Atractividad
- Eficiencia - atributos relacionados con la relación entre el nivel de desempeño y la cantidad de recursos
 - Comportamiento en el tiempo
 - Comportamiento de recursos
- Mantenibilidad - atributos relacionados con la facilidad de extender, modificar o corregir errores en un sistema software.
 - Estabilidad
 - Facilidad de análisis
 - Facilidad de cambio
 - Facilidad de pruebas
- Portabilidad - atributos relacionados con la capacidad de un sistema software para ser transferido desde una plataforma a otra.
 - Capacidad de instalación

- Capacidad de reemplazo
- Adaptabilidad
- Co-Existencia

El estándar de seguridad de la información ISO 17799:2000, es la forma generalizada de contar con un estándar que permite reconocer o validar el marco de referencia de seguridad aplicado por las organizaciones, el cual se basa principalmente en la primera parte del estándar BS 7799 conocida como Código de Prácticas (BS 7799 Part 1: Code of Practice), emitida por el BSI (British Standard Institute). El ISO 17799, es una guía de implementación del sistema de administración de seguridad de la información, y se enfoca en tutelar los siguientes principios de la seguridad informática:

- Confidencialidad - asegurar que únicamente personal autorizado tenga acceso a la información.
- Integridad - garantizar que la información no será alterada, eliminada o destruida por entidades no autorizadas.
- Disponibilidad - asegurar que los usuarios autorizados tendrán acceso a la información cuando la requieran.

Estos principios de protección de información son el núcleo básico, sin embargo se debe tener presente siempre que, dependiendo de la naturaleza y objetivos particulares, estos elementos de protección se pueden modificar o tener énfasis diferentes.

La ISO/IEC 19505 Information technology - Open Distributed Processing - Unified Modeling Language (UML), es un estándar que describe un lenguaje gráfico para visualizar,

especificar, construir y documentar un sistema. UML permite describir un esquemático del sistema, donde se incluyen aspectos conceptuales tales como los procesos, las funciones del sistema y los aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos, etc. Es importante remarcar que UML es un lenguaje para describir el modelo, para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. Los tipos principales de diagramas UML son:

- UML estructural: muestra la estructura estática de los objetos en un sistema.
- UML de comportamiento: muestra el comportamiento dinámico de los objetos en el sistema.
- UML de interacción: muestra una vista sobre los aspectos dinámicos de los sistemas modelados.

En el modelo de evaluación de Software que sigue la Universidad Tecnológica de Pereira se vislumbra un alineamiento con estos modelos internacionales al tener presente los siguientes criterios de evaluación, los cuales se pueden encontrar como características o sub-características de los modelos ISO:

- Robustez - sólido aún en situaciones difíciles.
- Extendibilidad - fácil de que le sean añadidas nuevas características.
- Desempeño - hace lo que tiene que hacer en el tiempo requerido, no desperdicia espacio en RAM ni en disco.

- Usabilidad o amigable al usuario - fácil de usar desde el punto de vista del usuario final.
- Integridad - que la información no se pierda ni se la puedan modificar o cambiar o capturar personas no autorizadas; o que la información almacenada permanezca consistente.
- Portabilidad - que pueda portarse fácilmente de una plataforma a otra.
- Compatibilidad - que sea compatible con anteriores versiones, si las hay.
- Mantenimiento - que sea de fácil mantenimiento.
- Documentación - que esté suficientemente documentado.
- Grado de invención o de innovación. Este aspecto, además se desglosa en las siguientes dos observaciones:
 - ¿Cómo es el desarrollo de la temática con relación a la productividad nacional en el área tratada (estado del arte)?
 - ¿Considera usted que el software tiene impacto y contribuye a las necesidades de la academia u organizaciones públicas o privadas en el ámbito regional, nacional o internacional?

Con el fin de comparar lo descripto por normas y por el modelo de evaluación de Software que sigue la Universidad Tecnológica de Pereira, se presenta la siguiente tabla:

Criterio	Modelo UTP	Modelos ISO
Robustez	Sólido aún en situaciones difíciles	Capacidad de reaccionar apropiadamente ante situaciones excepcionales
Extendibilidad	Fácil de que le sean añadidas nuevas características	Facilidad de adaptación del software a nuevos requisitos o cambios en la especificación.
Desempeño	Hace lo que tiene que hacer en el tiempo requerido, no desperdicia espacio en RAM ni en disco	Relacionado como eficiencia, o atributos relacionados con la relación entre el nivel de desempeño y la cantidad de recursos. Se tiene presente su comportamiento en el tiempo y su comportamiento de recursos.
Usabilidad o amigable al usuario	Fácil de usar desde el punto de vista del usuario final	Atributos relacionados con el esfuerzo necesario para su uso, por un conjunto de usuarios
Integridad	Que la información no se pierda ni se la puedan modificar o cambiar o capturar personas no autorizadas; o que la información almacenada permanezca consistente	Capacidad de proteger sus diferentes componentes contra los procesos o elementos que no tengan derecho de acceso a los mismos.
Portabilidad	Que pueda portarse fácilmente de una plataforma a otra	Capacidad de ser transferido y adaptado desde una plataforma a otra
Compatibilidad	Que sea compatible con anteriores versiones, si las hay	En la característica Portabilidad, las subcaracterísticas de: <ul style="list-style-type: none"> ■ Co-Existencia ■ Capacidad de reemplazo
Mantenimiento	Que sea de fácil mantenimiento	Definido como mantenibilidad, o atributos relacionados con la facilidad de extender, modificar o corregir errores en un sistema software
Documentación	Que esté suficientemente documentado	Descripción mediante modelos UML y documentación general por manuales.

Cuadro 1: Tabla comparativa.

Grado de invención o de innovación	¿Cómo es el desarrollo de la temática con relación a la productividad nacional en el área tratada (estado del arte)? ¿Considera usted que el software tiene impacto y contribuye a las necesidades de la academia u organizaciones públicas o privadas en el ámbito regional, nacional o internacional?	
------------------------------------	---	--

Cuadro 2: Tabla comparativa.

A continuación, se muestra el análisis respectivo de calidad del software con el marco de referencia empleado por la Universidad Tecnológica de Pereira, con el fin de mostrar cómo estos aspectos son tenidos en cuenta dentro del desarrollo del software REL-Imp.

6.2. ANÁLISIS CALIDAD DE SOFTWARE REL-Imp

6.2.1. Robustez

REL-Imp es sólido ante diferentes tipos de acciones de entrada, ya sean eventos de ratón y teclado, o parámetros ingresados por el usuario. Para el ingreso de parámetros, se restringe desde un inicio al uso únicamente de valores numéricos, en caso del usuario tratar de acceder un carácter de naturaleza distinta, el sistema omitirá dicha acción. En el caso de los eventos de ratón y teclado, se permite el accionamiento de ambos individuos

con la finalidad de interactuar con los diferentes objetos y propiedades de la interfaz, como lo es la adición de elementos dentro del área de trabajo por medio del mouse, o el uso de diferentes teclas para el accionamiento de eventos, tal como la eliminación de elementos activos o adición de parámetros en las ventanas emergentes.

6.2.2. Extendibilidad

El modelo que rige el diseño del software se basa en el paradigma de la programación orientado a objetos, respetando éste cada una de sus propiedades intrínsecas, como lo son la herencia, encapsulamiento, polimorfismo, entre otras. Debido a ello se facilita el realizar la adición de nuevos ítems al presente software. Dentro de la sección 5 del manual de usuario del software es posible observar el despliegue de cada uno de los elementos que lo componen, así como de la manera en que se interrelacionan entre sí, esto a través del lenguaje unificado de modelado (UML).

6.2.3. Desempeño

REL-Imp posee pocas exigencias en cuanto a la relación desempeño-costos computacional. Éste puede ejecutar sistemas que posean aproximadamente 20 elementos dentro del área de trabajo sin conllevar a un colapso de la interfaz, esto sin tener presente que en cada pestaña es posible tener subsistemas que cuentan internamente con más elementos. Para cumplir con dichas condiciones, es recomendable que el equipo anfitrión posea los siguientes componentes (o superiores a estos):

- Sistema Operativo Windows 7/8/10 (64 bits).

- Intel Core I3 @ 2.3GHz (o equivalente).
- Memoria RAM de 4GB mínimo.
- 500MB de espacio libre en el disco de almacenamiento.
- Visor de archivos PDF.

6.2.4. Usabilidad

La interfaz gráfica de REL-Imp es de naturaleza intuitiva, cuenta con elementos visuales que facilitan la rápida comprensión de su funcionamiento, tanto para usuarios ya relacionados con la temática del campo, como para personas que no han tenido experiencia con programas similares. En caso de un usuario requerir de una guía exhaustiva, se dispone de un manual de usuario que clarifica cada aspecto necesario para el correcto aprovechamiento del software.

6.2.5. Integridad

El producto final a entregar consta de un archivo binario protegido contra escritura, esto es, un ejecutable que permite observar e interactuar al usuario con cada uno de los elementos y propiedades de la interfaz, todo ello bajo restricciones mínimas (necesarias para garantizar la robustez del sistema), como lo es el ingreso de parámetros y eventos de ratón y teclado. Por cuestiones de evaluación, la presente distribución anexa dentro de sus elementos la carpeta de código fuente, esto con la finalidad de permitir al evaluador corroborar de manera propia cada uno de los aspectos mencionados dentro de la metodología implementada para el desarrollo del software.

6.2.6. Portabilidad

REL-Imp es un programa portable, no requiere de instalación dentro de un equipo anfitrión. Éste cuenta con cada uno de los módulos y paquetes necesarios para su correcta operación, por tanto puede ser migrado entre el número de plataformas deseado sin implicar esto deterioro o fallos en el sistema.

6.2.7. Compatibilidad

REL-Imp al ser un programa portable no requiere del uso de recursos compartidos propios del sistema operativo, implicando esto la no interferencia entre los diferentes hilos presentes en el momento de ejecución del software, garantizando así la coexistencia con los demás procesos activos.

6.2.8. Mantenibilidad

El programa es construido en su totalidad por módulos, los cuales a su vez contienen objetos. Por tanto, en caso de ser requerida una intervención, es posible recurrir a los modelos que describen dichas estructuras y corregir o adicionar de manera satisfactoria la necesidad presentada.

6.2.9. Documentación

Dentro de la sección 5 del manual de usuario de REL-Imp se ofrece a un mayor nivel de detalle cada uno de los aspectos que componen el software, desde la construcción hasta

el funcionamiento. Se cuenta con distintos diagramas UML, como lo son los diagramas de clasificación estructural, el de comportamiento dinámico, y la gestión del modelo.

6.2.10. Grado de invención o de innovación

Para proveer información acerca del nivel de innovación con el que cuenta el software REL-Imp, se proporciona el documento anexo en la figura a continuación, el cual menciona el impacto a nivel académico presentado por el presente software, esto por medio de su uso como herramienta que permite mejorar los métodos de enseñanza - aprendizaje al permitir poner en práctica conceptos y desarrollo de metodologías para diseñar, simular y analizar sistemas simples y complejos utilizando el método de diagramas de bloques. En éste es posible visualizar los resultados de evaluaciones de confiabilidad operativa, tanto del conjunto de elementos que componen un sistema como de sus elementos de manera individual. Además, permite hallar la medida de importancia que tiene cada elemento sobre el comportamiento del sistema, facilitando de esta manera determinar que componentes son más críticos desde un punto de vista del impacto en el sistema de confiabilidad. Además de éste, también se incluyen eventos y publicaciones en las que REL-Imp tuvo participación como elemento de modelamiento.



Universidad
Tecnológica
de Pereira
Pereira, 25 de Enero de 2020

Ingeniero
Jhoniers Gilberto Guerrero Erazo
Vicerrector Académico
Presidente CIARP
Universidad Tecnológica de Pereira

Referencia: certificación de impacto del producto REL-Imp empleado por la comunidad académica.

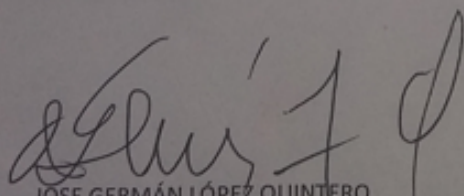
Cordial Saludo

Por medio del presente documento me permito certificar que el producto de software denominado **REL-Imp** es empleado por la comunidad de estudiantes y profesores del programa de Ingeniería Eléctrica y de la maestría en Ingeniería Eléctrica de la Universidad Tecnológica de Pereira.

El impacto del producto **REL-Imp** empleado por la comunidad descrita, se evidencia principalmente en su uso, tanto a nivel de pregrado como a nivel de posgrado, como herramienta que permite describir, representar y analizar sistemas de confiabilidad por medio de métodos gráficos, como lo son los diagramas de bloques (Nivel de Pregrado), así como determinar la configuración y los componentes que pueden hacer que un sistema sea crítico desde el punto de vista del impacto y aporte de cada elemento en la confiabilidad global del sistema (Nivel de Posgrado).

La plataforma de software desarrollada permite construir, simular y analizar sistemas de diferentes complejidades, esto por medio del método de diagramas de bloques. Este software proporciona una clasificación de los componentes con respecto a su influencia en la confiabilidad del sistema, facilitando de esta manera determinar que componentes son más críticos desde un punto de vista del impacto según la configuración, valor individual de confiabilidad y aporte total al sistema. Algunas áreas específicas del conocimiento donde se emplea la herramienta son las relacionadas con: En pregrado en Estadística, probabilidad y mantenimiento eléctrico; en posgrado en Teoría de la confiabilidad y el riesgo; en ambos niveles de enseñanza, se cubre en general las áreas de mantenimiento y automatización.

Atentamente,


JOSE GERMÁN LÓPEZ QUINTERO
Director Programa de Ingeniería Eléctrica
Universidad Tecnológica de Pereira

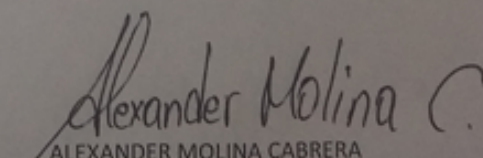

ALEXANDER MOLINA CABRERA
Decano Facultad de Ingenierías EEFCC
Universidad Tecnológica de Pereira

Figura 74: Certificación del impacto del producto REL-Imp

Proyectos de grado, en pregrado:

Título: Metodología para optimización, a través de algoritmos genéticos, de sistemas de manufactura flexible libres de bloqueos modelados mediante redes de Petri.

Universidad Tecnológica de Pereira - Ingeniería Eléctrica, 2017

Personas orientadas: Cristian Camilo Grisales Londoño, David Daniel Villegas Trujillo

Artículos en eventos:

Título: Planificación de movimientos para sistemas de navegación autónoma terrestre utilizando lógica temporal lineal

Universidad Tecnológica de Pereira

Autores: Jorge Luis Martínez Valencia, Mauricio Holguín Londoño, Andrés Escobar Mejía

Evento: II Congreso Internacional Electromecánica y Eléctrica

ISBN: 978-9942-759-09-2

Lugar: Latacunga, Ecuador, julio 12-14 de 2017.

Memorias en eventos:

Título: Metodología para definir el conjunto de condiciones mínimas y necesarias, para la detección de errores en código, bajo filosofía TDD, en marco de una aplicación para detección de objetos por características morfológicas

Autores: Juan Felipe Grajales Gonzalez, Mauricio Holguín Londoño, Germán Andrés Holguín Londoño

Evento: Congreso Internacional de DESARROLLO DE SOFTWARE

ISBN: 978-9942-759-94-8

Lugar: Instituto Superior Tecnológico del Azuay/ Cuenca, Ecuador. 24-26 Octubre del 2018.

Referencias

- [1] RAUSAND, M. Y HOYLAND A. *System Reliability Theory Models, Statistical Methods, and Applications*, Wiley, 2004
- [2] BIROLINI, A. *Reliability Engineering: Theory and Practice* Springer. 1999.
- [3] ANAND, A. Y RAM, M *System Reliability Management: Solutions and Technologies*
- [4] OROZCO ÁLVARO, GUARNIZO CRISTIAN, HOLGUÍN MAURICIO, *Automatismos Industriales* Universidad Tecnológica de Pereira, 2008
- [5] FRENKEL, I Y LISNIANSKI A., *Recent Advances in System Reliability: Signatures, Multi-state Systems and Statistical Inference*
- [6] LISNIANSKI, A. *Multi-state system reliability*, 2003
- [7] BONDY, J. Y MURTY, U. *Graph theory with applications*, 2013
- [8] J. RUMBAUGH, I. JACOBSON, G. BOOCH *El Lenguaje Unificado de Modelado. Manual de referencia* Pearson Educación S.A. Madrid, 2000.