

***DEPARTAMENTO DE INGENIERÍA E INVESTIGACIONES TECNOLÓGICAS – Análisis de Software***

**ANÁLISIS DE SOFTWARE**

**Documentación Técnica del Producto**

**“Gestor de Libros”**

**Grupo 1**

**Integrantes:**

* Menendez, Leonel
* Raimondo, Pablo
* Ruttimann, Hernan
* Saccella, Claudio
* Sapaya, Nicolas

**Comisión:**

Viernes Noche

**Profesores:**

* Agustín, Gustavo Ariel
* Del Ben, Enzo Antonio
* Landaburu, Roberto Eduardo
* Vinjoy, Marcelo Daniel

**ÍNDICE**

[**Descripción Funcional** 3](#_Toc497991121)

[**Modelo de calidad** 4](#_Toc497991122)

[Niveles de Puntuación 4](#_Toc497991123)

[Criterios de Evaluación de Métricas 5](#_Toc497991124)

[1. Funcionalidad 5](#_Toc497991125)

[2. Eficiencia 6](#_Toc497991126)

[3. Fiabilidad 7](#_Toc497991127)

[4. Mantenibilidad 8](#_Toc497991128)

[5.Usabilidad 9](#_Toc497991129)

[6. Portabilidad 10](#_Toc497991130)

[**Reporte Herramienta de testing** 11](#_Toc497991131)

[**Caja Blanca** 12](#_Toc497991132)

[Mc Cabe 12](#_Toc497991133)

[Condición/Decisión 14](#_Toc497991134)

[**Caja negra** 15](#_Toc497991135)

[**Prueba del Modelo de Calidad** 16](#_Toc497991136)

[1. Funcionalidad 16](#_Toc497991137)

[2. Eficiencia 18](#_Toc497991138)

[3. Fiabilidad 19](#_Toc497991139)

[4. Mantenibilidad 21](#_Toc497991140)

[5.Usabilidad 25](#_Toc497991141)

[6. Portabilidad 27](#_Toc497991142)

[Informe final 30](#_Toc497991143)

[**Prueba de sistemas** 31](#_Toc497991144)

[Prueba de recuperación: 31](#_Toc497991145)

[Prueba de seguridad: 33](#_Toc497991146)

[Prueba de resistencia: 34](#_Toc497991147)

[Prueba de rendimiento: 34](#_Toc497991148)

[Prueba de aceptación: 34](#_Toc497991149)

[**Manual de usuario** 37](#_Toc497991150)

[Descripción 37](#_Toc497991151)

[Objetivo 37](#_Toc497991152)

[Funcionamiento 38](#_Toc497991153)

[a. Inicio 41](#_Toc497991154)

[b. Administración 42](#_Toc497991155)

[Instalación 45](#_Toc497991156)

[**Cubierta del CD** 48](#_Toc497991157)

[**Encuestas:** 51](#_Toc497991158)

[**Resultados:** 53](#_Toc497991159)

# 

# 

# **Descripción Funcional**

El software de gestor de libros, tiene como objetivo la gestión de libros asociados a una biblioteca.

El sistema está dividido en 2 funcionalidades principales:

-Administración de libros:

1. Alta de libros.
2. Baja de libros.
3. Actualización de libros.
4. Alta de editoriales.
5. Baja de editoriales.
6. Actualización de editoriales.
7. Alta de autores.
8. Baja de autores.

9- Actualización de autores.

-Consultas:

1- Consulta.

2- Ordenar datos de libros.

3- Listar datos de libros.

El sistema realiza tanto el almacenamiento de los datos, como la restricción de acceso al software a través de una base de datos administrada de manera local, en el directorio de la aplicación. Dicha restricción de acceso se realiza a través de usuario y contraseña, donde esta última se encuentra encriptada para preservar la seguridad de la información de los usuarios.

# **Modelo de calidad**

Características y subcaracterísticas utilizadas:

1. Funcionabilidad
   1. Seguridad de acceso.
   2. Exactitud.
   3. Adecuación.
2. Eficiencia
   1. Utilización de recursos.
   2. Comportamiento temporal.
3. Fiabilidad
   1. Capacidad de recuperación.
   2. Tolerancia a fallos.
4. Mantenibilidad
   1. Capacidad de ser analizado.
   2. Capacidad para ser modificado.
   3. Estabilidad.
5. Usabilidad
   1. Capacidad de ser entendido.
   2. Capacidad de ser operado.
   3. Capacidad para ser atractivo.
6. Portabilidad
   1. Adaptabilidad.
   2. Instalabilidad.

## Niveles de Puntuación

El algoritmo de calidad consiste en evaluar cada subcaracterística como aceptable, medianamente aceptable, o no aceptable, dependiendo de los criterios que se mencionan posteriormente. Cada característica posee un puntaje y una subcaracterística ponderada, la cual poseerá un mayor puntaje que las demás subcaracterísticas. La sumatoria de los resultados de la evaluación ponderada de las subcaracterísticas, determinarán el puntaje de cada característica. Luego se procederá a sumar los puntajes de las características, y si la sumatoria es igual o mayor a 30, se determinará que el nivel de calidad obtenido es SATISFACTORIO, de lo contrario será NO SATISFACTORIO. Cabe destacar, que para ser considerado Satisfactorio, todas las características deben tener su subcaracterística ponderada como Satisfactorio, si no es así, por más que el puntaje total sea mayor a 30 no será considerado Satisfactorio.

## Criterios de Evaluación de Métricas

### 1. Funcionalidad

* **Seguridad de acceso**

Descripción:

Capacidad del producto software para asegurar la integridad de los datos y la confidencialidad de estos.

**Características a medir:**

* **Encriptación de la contraseña y datos sensibles.**
* **Inicio de sesión.**

Evaluación:

* No aceptable [0]: Ninguna característica cumplida.
* Medianamente aceptable [2]: Una característica cumplida
* Aceptable [4]: Dos características cumplidas.
* **Exactitud**

Descripción:

Es la capacidad del producto software para proporcionar los resultados con el grado necesario de precisión.

Evaluación:

* No aceptable [0]: No se realiza ningún informe.
* Medianamente aceptable [1]: Se realiza 1 informe.
* Aceptable [2]: Se realizan 2 o más informes.
* **Adecuación**

Descripción:

Capacidad del producto software para proporcionar un conjunto apropiado de

funciones para tareas y objetivos de usuario especificados

Evaluación:

* No aceptable [0]: El software no cumple con ninguna funcionalidad requerida por el usuario.
* Medianamente aceptable [1]: El software cumple con una funcionalidad requerida
* Aceptable [2]: El software cumple con al menos dos funcionalidades requeridas.

**Subcaracterística Ponderada:** Seguridad de Acceso.

### 2. Eficiencia

* **Utilización de los Recursos**

Descripción:

Se evaluará la eficiencia del producto software de acuerdo al porcentaje de uso de procesador y de memoria que realice.

Evaluación:

* No aceptable [0]: El producto consume 51% a 100% de uso de procesador y de uso de memoria.
* Medianamente aceptable [2]: El producto consume 26% a 50% de uso de procesador y de memoria.
* Aceptable [4]: El producto consume 25% o menos de uso de procesador y de memoria.
* **Comportamiento temporal**

Descripción:

Se evaluará el tiempo que está el producto software sin informarle al usuario del estado en que se encuentra la solicitud que realizó.

Evaluación:

* No aceptable [0]: El software demora más de 6 segundos en informar el estado de la solicitud.
* Medianamente aceptable [1]: El software demora entre 2 a 5 segundos en informar el estado de la solicitud.
* Aceptable [2]: El software demora menos de 2 segundos en informar el estado de la solicitud.

**Subcaracterística Ponderada:** Utilización de los recursos.

### 3. Fiabilidad

* **Tolerancia a fallos**

Descripción:

Es la capacidad del producto software de mantener la integridad de los datos cuando se producen fallas del sistema.

**Características a medir:**

**• Cuando sucede un error se protegen los datos procesados.**

**• Se realiza un log de actividades que el sistema estaba haciendo.**

Evaluación:

* No aceptable [0]: No se cumple con ninguna ninguna característica.
* Medianamente aceptable [2]: Se cumple con 1 característica.
* Aceptable [4]: Se cumple con las 2 características.
* **Capacidad de Recuperación de Errores**

Descripción:

Es la capacidad del sistema de reanudar sus actividades cuando se producen errores críticos.

**Características a medir:**

**• El sistema reanuda las actividades si se produce una falla crítica.**

**• Reanuda sus actividades y vuelve al estado en que estaba.**

Evaluación:

* No aceptable [0]: No se cumple con ninguna ninguna característica.
* Medianamente aceptable [1]: Se cumple con 1 característica.
* Aceptable [2]: Se cumple con las 2 características.

**Subcaracterística Ponderada:** Tolerancia a fallos.

### 4. Mantenibilidad

* **Capacidad de ser analizado:**

Descripción

Para evaluar la capacidad que tiene el código para ser analizado se tiene en cuenta el promedio de comentarios de todos los métodos del programa.

Evaluación:

* No aceptable [0]: El programa posee un promedio de comentarios menor al 20%.
* Medianamente aceptable [1]: El programa posee un promedio de comentarios entre el 20% y 49%.
* Aceptable [2]: El programa posee un promedio de comentarios mayor al 50%.

* **Capacidad para ser modificado:**

Descripción:

Para evaluar la capacidad que tiene el código para ser cambiado se tomará en cuenta la complejidad ciclomática del método.

Evaluación:

* No aceptable [0]: La complejidad ciclomática es mayor a 20.
* Medianamente aceptable [1]: La complejidad ciclomática es entre 11 y 20.
* Aceptable [2]: La complejidad ciclomática es menor o igual a 10.

* **Estabilidad:**

Descripción:

Para determinar la estabilidad del software se evalúa el promedio de fallas que presenta el producto por prueba.

Evaluación:

* No aceptable [0]: El software presenta un promedio de 4 o más fallas por prueba.
* Medianamente aceptable [2]: El software presenta un promedio de 3 o más fallas por prueba.
* Aceptable [4]: El software presenta un promedio de 2 o menos fallas por prueba.

**Subcaracteristica Ponderada:** Estabilidad.

### 5.Usabilidad

* **Capacidad de ser entendido**

Descripción:

Capacidad que posee el software, para ayudar a los usuarios ante una determinada situación donde se necesite asistencia.

**Características a medir:**

**• Ayuda contextual sobre menús y botones de acción (tooltip).**

**• Manual de ayuda al usuario incorporado en el sistema.**

Evaluación:

* No aceptable [0]: No se cumple con ninguna ninguna característica.
* Medianamente aceptable [1]: Se cumple con 1 característica.
* Aceptable [2]: Se cumple con las 2 características.
* **Capacidad para ser operado**

Descripción

Es la capacidad del producto software de ser utilizado sin asistencia del manual de ayuda ni de un experto.

Evaluación:

* No aceptable [0]: El usuario requiere ayuda de un experto para operar el software
* Medianamente aceptable [1]: El usuario requiere usar el manual de ayuda para poder operar el software.
* Aceptable [2]: El usuario no requiere usar el manual de ayuda para poder operar el software.
* **Capacidad de ser atractivo**

Descripción:

Es la agrupación correcta de funcionalidades dentro del programa en la interfaz gráfica para poder realizar una función específica.

Evaluación:

* No aceptable [0] El usuario utiliza más de 5 pasos promedio para poder realizar una función específica.
* Medianamente aceptable [2]: El usuario necesita entre 4 y 5 pasos promedio para poder realizar una función específica.
* Aceptable [4]: El usuario solamente necesita de 3 o menos pasos promedio para poder realizar una función específica.

**Subcaracterística Ponderada:** Capacidad de ser atractivo.

### 6. Portabilidad

* **Adaptabilidad**

Descripción:

Es la capacidad del producto software de adaptarse a diferentes sistemas operativos sin cambiar su estructura interna.

Evaluación:

* No aceptable [0]: El software solo es compatible con 1 sistema operativo en concreto.
* Medianamente aceptable [2]: El software es compatible con 2 sistemas operativos.
* Aceptable [4]: El software es compatible con 3 o más sistemas operativos.

* **Instalabilidad**

Descripción

El producto software debe poder ser instalado en una cantidad mínima de pasos.

Evaluación:

* No aceptable [0]: El producto se instala en 7 o más pasos.
* Medianamente aceptable [1]: El producto se instala entre 5 y 6 pasos.
* Aceptable [2]: El producto se instala en 4 pasos o menos.

**Subcaracterística Ponderada:** Adaptabilidad.

# **Reporte Herramienta de testing**

Se decidió utilizar el método insertarLibro() de la clase Gestor dado su criticidad.

****

# **Caja Blanca**

## Mc Cabe

public void insertarLibro() {

try {

1

boolean camposCorrectos = recuperarInformacionLibro();

2

3

4

if (camposCorrectos && bd.insertLibro(ISBN, titulo, autor, editorial, edicion, anioPublicacion) == 1) {

informar("El libro se ha insertado con éxito.");

5

actualizarTablaLibros();

}

} catch (SQLException e) {

6

informar("Se ha producido un error al agregar el libro.");

try {

7

bd.close();

8

} catch (SQLException e1) {

9

e1.printStackTrace();

}

bd = new BD();

10

e.printStackTrace();

}

} //FIN

11

Caminos:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Caminos | | | | | | |
| C1 | 1 | 2 | 3 | 11 |  |  |
| C2 | 1 | 2 | 3 | 4 | 11 |  |
| C3 | 1 | 2 | 3 | 4 | 5 | 11 |
| C4 | 1 | 6 | 7 | 8 | 10 | 11 |
| C5 | 1 | 6 | 7 | 9 | 10 | 11 |



La complejidad ciclomática según el algoritmo de McCabe es de 5.

(14 aristas – 11 nodos + 2 = 5)

Por lo tanto, deberán realizarse un mínimo de 5 pruebas para poder probar el programa.

6

2

11/FIN

9

8

7

5

4

3

1

10

## Condición/Decisión

**Código:**

public void insertarLibro() {

try {

boolean camposCorrectos = recuperarInformacionLibro();

if(camposCorrectos && bd.insertLibro(ISBN, titulo, autor, editorial, edicion, anioPublicacion) ==1){

informar("El libro se ha insertado con éxito.");

actualizarTablaLibros();

}

} catch (SQLException e) {

informar("Se ha producido un error al agregar el libro.");

try {

bd.close();

} catch (SQLException e1) {

e1.printStackTrace();

}

bd = new BD();

e.printStackTrace();

}

}

**Resolución:**

D1:C1: try

D2:C2: camposCorrectos

D2:C3: bd.insertLibro(ISBN, titulo, autor, editorial, edicion, anioPublicacion) == 1

D3:C4: try2



# **Caja negra**

**Funcionalidad**: insertarLibro()

**Entradas:**

**ISBN:** longitud de 13 caracteres numéricos.

**Titulo:** caracteres alfanuméricos.

**Edicion:** longitud hasta 10 caracteres numéricos.

**AnioPublicacion:** longitud hasta 10 caracteres numéricos.

|  |  |  |  |
| --- | --- | --- | --- |
| Condición | C. Válidas | C. No válidas | Regla que se aplicó |
| ISBN 13 caracteres | = 13 **(1)** | <> 13 **(6)** | booleana |
| ISBN caracteres numéricos | entero positivo **(2)** | <> entero positivo **(7)** | booleana |
| Titulo caracteres alfanuméricos | Alfanuméricos **(3)** | Caracteres especiales **(8)** | booleana |
| Edicion entre 1 y 10 caracteres | Entre 1 y 10 **(4)** | Blanco **(9.1)**  Más de 10 **(9.2)** | Valido **-** No valido |
| AnioPublicacion entre 1 y 10 caracteres | Entre 1 y 10 **(5)** | Blanco **(10.1)**  Más de 10 **(10.2)** | Valido - No valido |
| AnioPublicacion caracteres numéricos | entero positivo **(11)** | <> entero positivo **(12)** | booleana |

|  |  |  |
| --- | --- | --- |
| Caso | Clases cubiertas | Salida |
| ISBN = “1111111111111“ | 1,2 | OK |
| ISBN = “1” | 2,6 | Error |
| ISBN = “a1111111111111“ | 6,7 | Error |
| Titulo = “Las Palmas 2” | 3 | OK |
| Titulo = “Las Palmas!“ | 8 | Error |
| Edicion = “1222222222“ | 4 | OK |
| Edicion = “ “ | 9.1 | Error |
| Edicion = “12222222223“ | 9.2 | Error |
| AnioPublicacion = “1222222222“ | 5,11 | OK |
| AnioPublicacion = “ “ | 10.1 | Error |
| AnioPublicacion = “12222222223“ | 10.2 | Error |
| AnioPublicacion = “abc123” | 12 | Error |

# **Prueba del Modelo de Calidad**

## 1. Funcionalidad

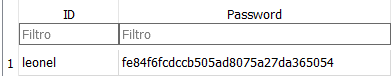
* **Seguridad de acceso**

Descripción:

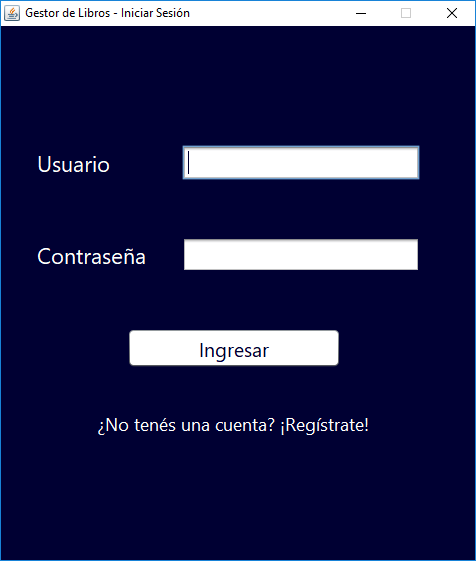
Capacidad del producto software para asegurar la integridad de los datos y la confidencialidad de estos.

**Características a medir:**

* **Encriptación de la contraseña y datos sensibles.**



* **Inicio de sesión.**

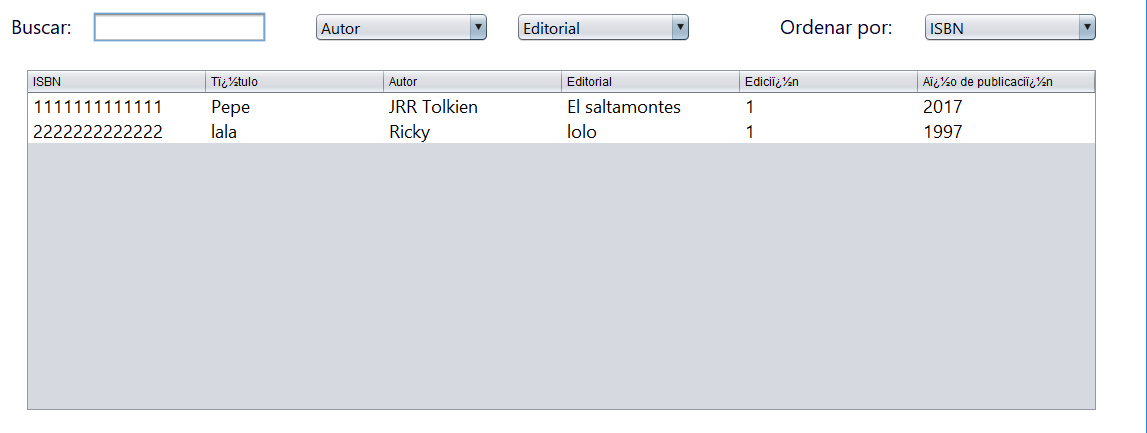


Evaluación: Al contar con ambas características, la **seguridad de acceso** es aceptable y suma 4 puntos.

* **Exactitud**

Descripción:

Es la capacidad del producto software para proporcionar los resultados con el grado necesario de precisión.





Evaluación: Al realizarse 3 informes, la **exactitud** es aceptable y suma 2 puntos.

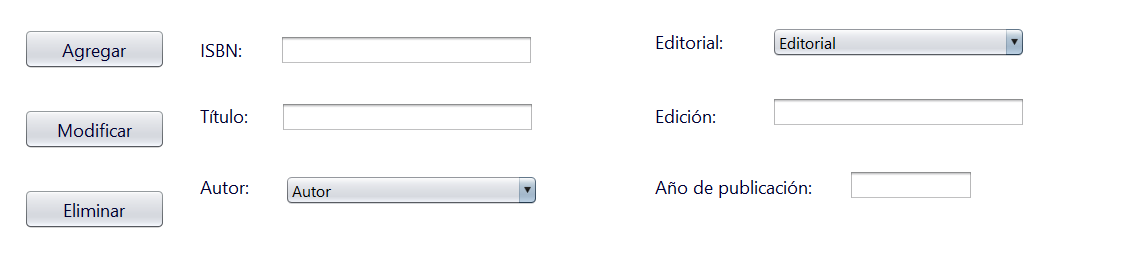
* **Adecuación**

Descripción:

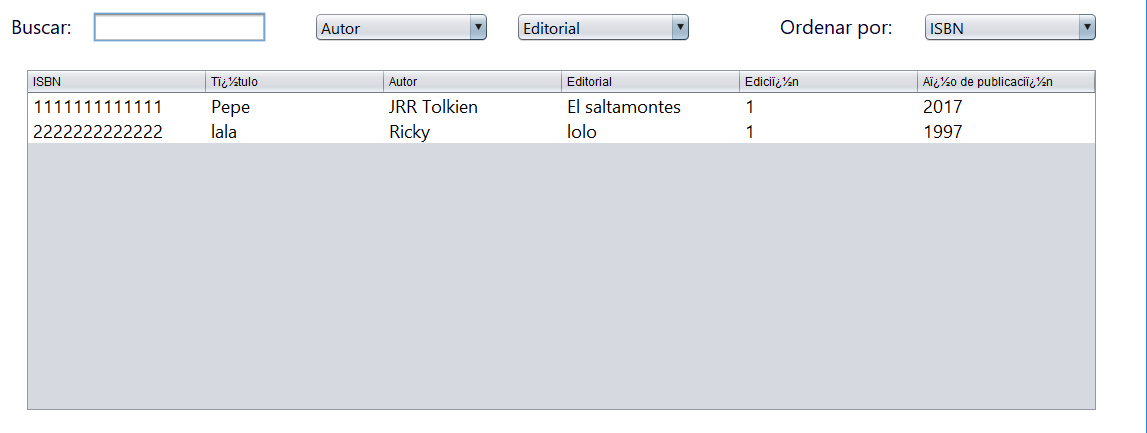
Capacidad del producto software para proporcionar un conjunto apropiado de

funciones para tareas y objetivos de usuario especificados

Insertar nuevo libro



Buscar libros



Evaluación: Al cumplir con al menos 2 funcionalidades requeridas, la **adecuación** es aceptable y suma 2 puntos.

**Subcaracterística Ponderada:** Seguridad de Acceso.

**Subtotal: 8 pts.**

## 2. Eficiencia

* **Utilización de los Recursos**

Descripción:

Se evaluará la eficiencia del producto software de acuerdo al porcentaje de uso de procesador y de memoria que realice.

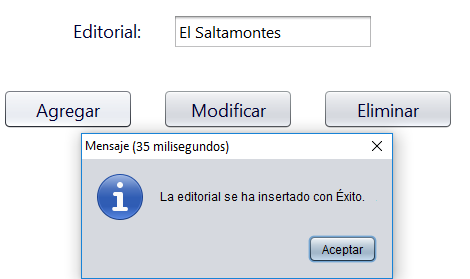
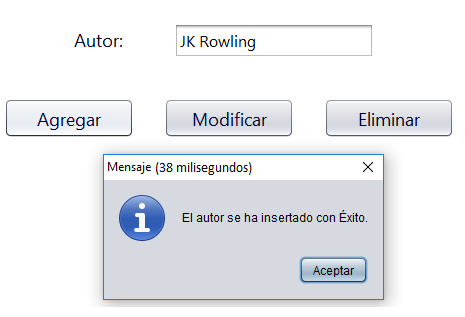


Evaluación: Al consumir menos del 25% de procesador y memoria, la **utilización de los recursos** es aceptable y suma 4 puntos.

* **Comportamiento temporal**

Descripción:

Se evaluará el tiempo que está el producto software sin informarle al usuario del estado en que se encuentra la solicitud que realizó.



Evaluación: Al demorarse menos de 2 segundos en informar el estado de la solicitud, el **comportamiento temporal** es aceptable y suma 2 puntos.

**Subcaracteristica Ponderada:** Utilización de los recursos.

**Subtotal: 6 pts.**

## 3. Fiabilidad

* **Tolerancia a fallos**

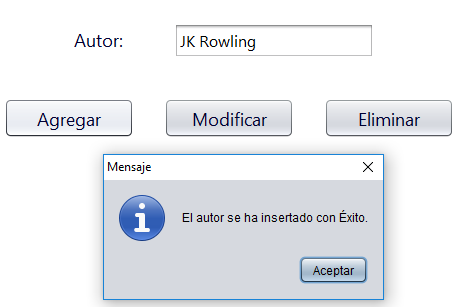
Descripción:

Es la capacidad del producto software de mantener la integridad de los datos cuando se producen fallas del sistema.

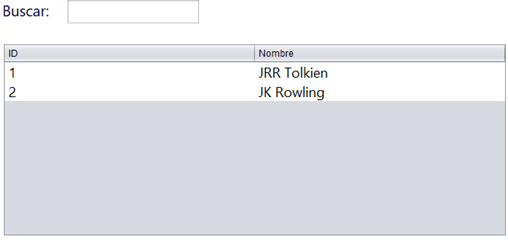
**Características a medir:**

**• Cuando sucede un error se protegen los datos procesados.**

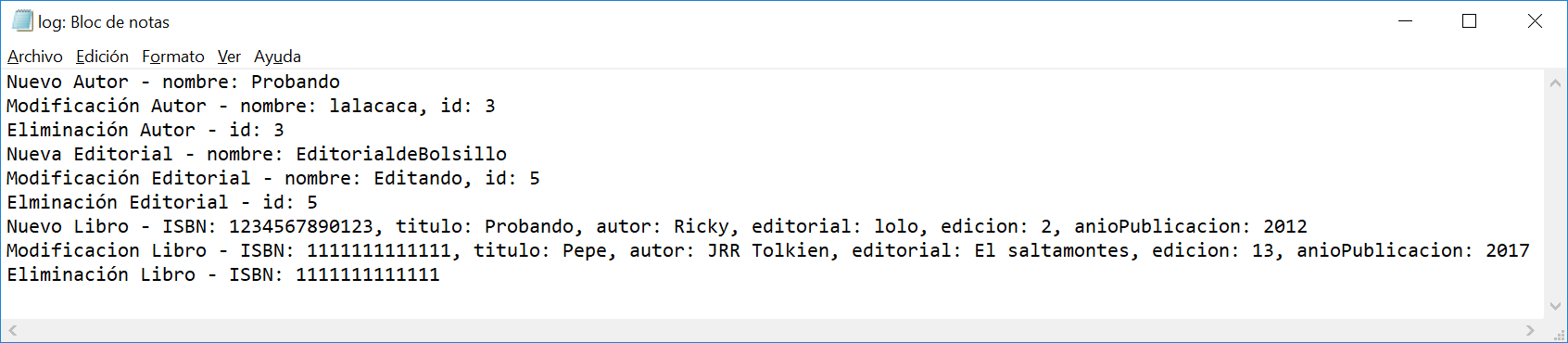
Damos de alta a un autor



Procedemos a realizar el sistema, sin cerrar correctamente el programa. Una vez iniciado el sistema nuevamente, abrimos el programa y notamos que los datos ingresados se encuentran correctamente en la grilla de autores.



**• Se realiza un log de actividades que el sistema estaba haciendo.**

****

Evaluación: Al cumplir con las dos características, la **tolerancia a fallos** es aceptable y suma 4 puntos.

* **Capacidad de Recuperación de Errores**

Descripción:

Es la capacidad del sistema de reanudar sus actividades cuando se producen errores críticos.

Está característica no se ha contemplado.

**Características a medir:**

**• El sistema reanuda las actividades si se produce una falla crítica.**

**• Reanuda sus actividades y vuelve al estado en que estaba.**

Evaluación: Al no cumplir con ninguna característica, la **capacidad de recuperación de errores** es no aceptable y no suma puntos.

**Subcaracteristica Ponderada:** Tolerancia a fallos.

**Subtotal: 4 pts.**

## 4. Mantenibilidad

* **Capacidad de ser analizado:**

Descripción

Para evaluar la capacidad que tiene el código para ser analizado se tiene en cuenta el promedio de comentarios de todos los métodos del programa.

**Clase BD:**

|  |  |  |  |
| --- | --- | --- | --- |
| Método | Cantidad de líneas | Líneas comentadas | % Líneas comentadas |
| BD() | 36 | 22 | 61,11 |
| deleteLibro() | 10 | 2 | 20 |
| getLibro() | 9 | 4 | 44,44 |
| getAll() | 12 | 5 | 41,67 |
| searchLibro() | 23 | 12 | 52,17 |
| insertAutor() | 10 | 2 | 20 |
| updateAutor() | 12 | 3 | 25 |
| deleteAutor() | 10 | 2 | 20 |
| getAutores() | 8 | 4 | 50 |
| searchAutor() | 12 | 6 | 50 |
| insertEditorial() | 10 | 2 | 20 |
| updateEditorial() | 12 | 3 | 25 |
| deleteEditorial() | 10 | 2 | 20 |
| getEditoriales() | 8 | 4 | 50 |
| searchEditorial() | 12 | 5 | 41,67 |
| insertUsuario() | 3 | 3 | 100 |
| deleteUsuario() | 2 | 2 | 100 |
| verificarLogin() | 9 | 3 | 33,33 |
| ordenarResultado() | 12 | 6 | 50 |
| close() | 20 | 20 | 100 |
| agregarTextoAlfinal() | 8 | 0 | 0 |

**Clase Entidad:**

|  |  |  |  |
| --- | --- | --- | --- |
| Método | Cantidad de líneas | Líneas comentadas | % Líneas comentadas |
| Entidad() | 2 | 0 | 0 |
| getId() | 1 | 1 | 100 |
| setId() | 1 | 1 | 100 |
| getNombre() | 1 | 1 | 100 |
| setNombre() | 1 | 1 | 100 |

**Clase Gestor:**

|  |  |  |  |
| --- | --- | --- | --- |
| Método | Cantidad de líneas | Líneas comentadas | % Líneas comentadas |
| Gestor() | 50 | 10 | 20 |
| windowClosing() | 11 | 3 | 27,27 |
| agregarFilaLibro() | 9 | 2 | 22,22 |
| llenarCamposLibro() | 8 | 1 | 12,5 |
| recuperarInformacionLibro() | 11 | 3 | 27,27 |
| actualizarTablaLibros() | 2 | 2 | 100 |
| limpiarTablaLibros() | 4 | 2 | 50 |
| limpiarCamposLibro() | 6 | 0 | 0 |
| buscarLibro() | 22 | 10 | 45,45 |
| traerTodosLosLibros() | 12 | 4 | 33,33 |
| insertarLibro() | 16 | 8 | 50 |
| modificarLibro() | 10 | 5 | 50 |
| eliminarLibro() | 8 | 3 | 37,5 |
| validarCamposLibro() | 18 | 4 | 22,22 |
| agregarFilaAutor() | 5 | 2 | 40 |
| actualizarTablaAutores() | 2 | 2 | 100 |
| limpiarTablaAutores() | 5 | 3 | 60 |
| buscarAutor() | 11 | 5 | 45,45 |
| insertarAutor() | 18 | 8 | 44,44 |
| modificarAutor() | 11 | 6 | 54,55 |
| eliminarAutor() | 8 | 4 | 50 |
| agregarFilaEditorial() | 5 | 2 | 40 |
| actualizarTablaEditoriales() | 2 | 2 | 100 |
| limpiarTablaEditoriales() | 5 | 3 | 60 |
| buscarEditorial() | 11 | 5 | 45,45 |
| insertarEditorial() | 19 | 8 | 42,11 |
| modificarEditorial() | 11 | 6 | 54,55 |
| eliminarEditorial() | 8 | 4 | 50 |
| cargarMenu() | 54 | 9 | 16,67 |
| cargarInicio() | 205 | 25 | 12,2 |
| isCellEditable() | 1 | 0 | 0 |
| cargarAdministracion() | 172 | 21 | 12,21 |
| cargarListeners() | 193 | 58 | 30,05 |
| actionPerformed() | 1 | 1 | 100 |
| keyReleased() | 1 | 1 | 100 |
| mouseClicked() | 4 | 3 | 75 |
| mouseReleased() | 9 | 2 | 22,22 |
| cargarAutoresYEditoriales() | 33 | 15 | 45,45 |
| informar() | 1 | 1 | 100 |

**Clase Launcher:**

|  |  |  |  |
| --- | --- | --- | --- |
| Método | Cantidad de líneas | Líneas comentadas | % Líneas comentadas |
| main() | 20 | 3 | 15 |
| run() | 16 | 2 | 12,5 |

**Clase Libro:**

|  |  |  |  |
| --- | --- | --- | --- |
| Método | Cantidad de líneas | Líneas comentadas | % Líneas comentadas |
| Libro() | 6 | 0 | 0 |
| equals() | 1 | 1 | 100 |
| compareTo() | 1 | 1 | 100 |
| toString() | 1 | 0 | 0 |
| getISBN() | 1 | 1 | 100 |
| setISBN() | 1 | 1 | 100 |
| getTitulo() | 1 | 1 | 100 |
| setTitulo() | 1 | 1 | 100 |
| getAutor() | 1 | 1 | 100 |
| setAutor() | 1 | 1 | 100 |
| getEditorial() | 1 | 1 | 100 |
| setEditorial() | 1 | 1 | 100 |
| getEdicion() | 1 | 1 | 100 |
| setEdicion() | 1 | 1 | 100 |
| getAnioPublicacion() | 1 | 1 | 100 |
| setAnioPublicacion() | 1 | 1 | 100 |

**Clase Login:**

|  |  |  |  |
| --- | --- | --- | --- |
| Método | Cantidad de líneas | Líneas comentadas | % Líneas comentadas |
| Login() | 111 | 15 | 13,51 |
| actionPerformed() | 6 | 0 | 0 |
| mouseEntered() | 2 | 0 | 0 |
| mouseReleased() | 3 | 0 | 0 |
| login() | 12 | 8 | 66,67 |

**Clase Registro:**

|  |  |  |  |
| --- | --- | --- | --- |
| Método | Cantidad de líneas | Líneas comentadas | % Líneas comentadas |
| main() | 21 | 0 | 0 |
| run() | 17 | 0 | 0 |
| Registro() | 130 | 18 | 13,85 |
| actionPerformed() | 6 | 2 | 33,33 |
| mouseEntered() | 2 | 0 | 0 |
| mouseReleased() | 7 | 0 | 0 |
| registrarse() | 27 | 10 | 37,04 |
| registrarEnDB() | 1 | 1 | 100 |

Evaluación: Al poseer un promedio de comentarios de 50,15%, la **capacidad de ser analizado** es aceptable y suma 2 puntos.

* **Capacidad para ser modificado:**

Descripción:

Para evaluar la capacidad que tiene el código para ser cambiado se tomará en cuenta la complejidad ciclomática del método.

|  |  |
| --- | --- |
| searchLibro() |  |
| ordenarResultado() |  |
| registrarse() |  |
| cargarListeners() |  |
| validarCamposLibro() |  |
| buscarLibro() |  |

Para esto se analizó la Complejidad Ciclomática de los 50 métodos existentes en el software. Sólo se muestran los métodos más significativos del sistema.

Evaluación: Al existir métodos con complejidad ciclomática entre 11 y 20, la **capacidad para ser modificado** es medianamente aceptable y suma 1 punto.

* **Estabilidad:**

Descripción:

Para determinar la estabilidad del software se evalúa el promedio de fallas que presenta el producto por prueba.

Se han realizado las pruebas del sistema y no se ha presentado ninguna falla, véase Pruebas del Sistema.

Evaluación: Al no presentar ninguna falla en sus pruebas, la **estabilidad** es aceptable y suma 4 puntos.

**Subcaracteristica Ponderada:** Estabilidad.

**Subtotal: 7 pts.**

## 5.Usabilidad

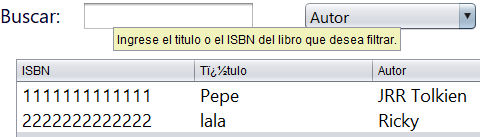
* **Capacidad de ser entendido**

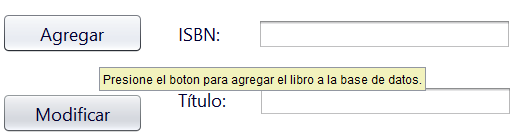
Descripción:

Capacidad que posee el software, para ayudar a los usuarios ante una determinada situación donde se necesite asistencia.

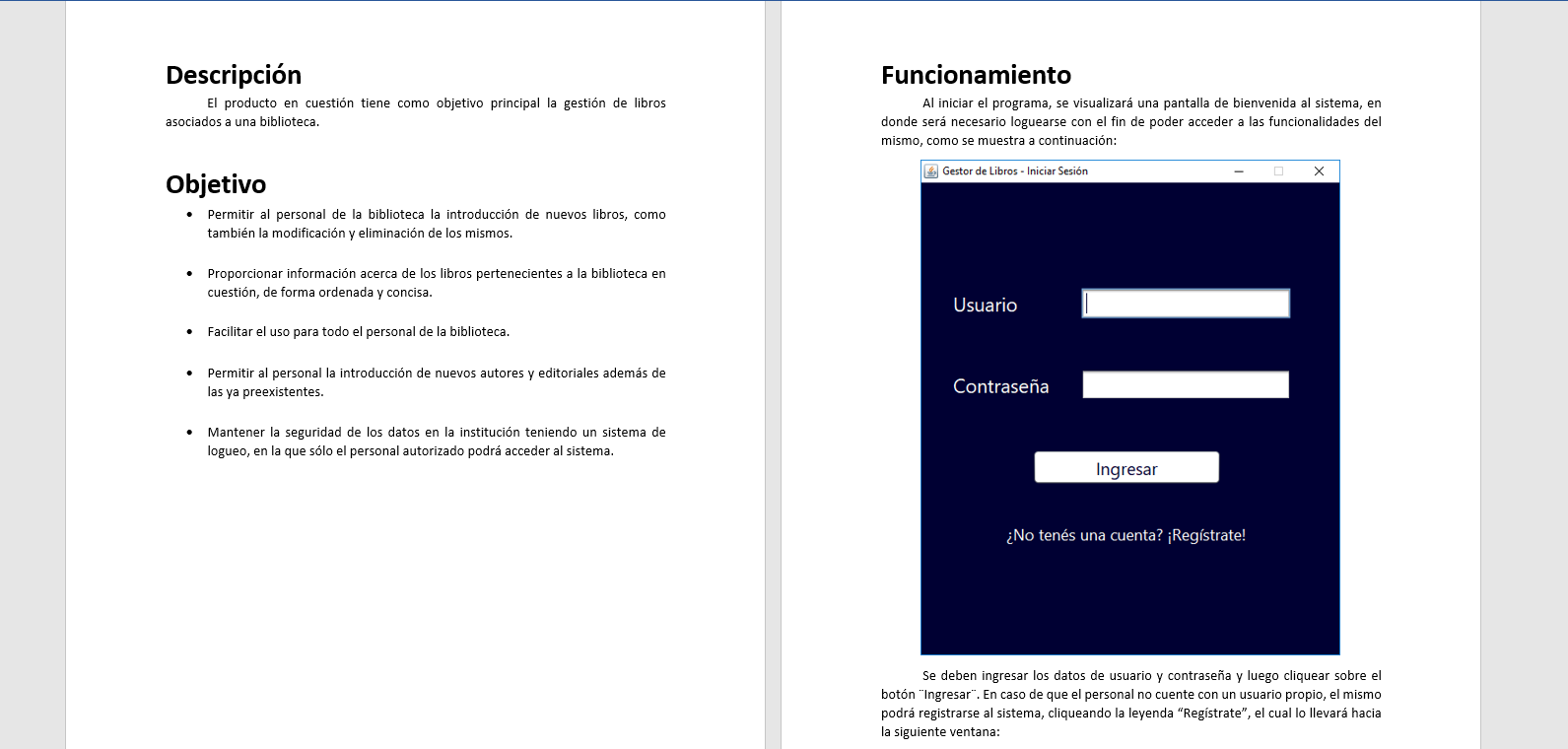
**Características a medir:**

**• Ayuda contextual sobre menús y botones de acción (tooltip).**

****

****

**• Manual de ayuda al usuario incorporado en el sistema.**

****

Evaluación: Al cumplir con las dos características, la **capacidad de ser entendido** es aceptable y suma 2 puntos.

* **Capacidad para ser operado**

Descripción

Es la capacidad del producto software de ser utilizado sin asistencia del manual de ayuda ni de un experto.

Para poder realizar esta prueba se le ha solicitado a un usuario sin conocimientos acerca del ámbito de IT que usara el software, comentándole brevemente que funciones podía realizar y aclarándole que disponía de un manual de usuario en caso de ser necesario. Luego de varios minutos investigando el software, el usuario no tuvo problemas para poder registrar libros, autores y editoriales como tampoco para realizar consultas.

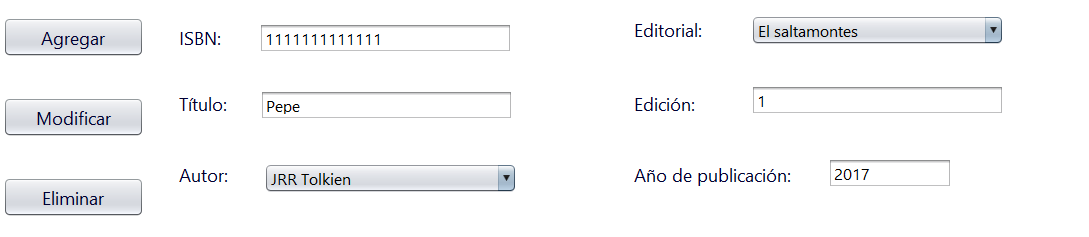
Evaluación: Al no requerir el manual de usuario para poder operar el software, la **capacidad para ser operado** es aceptable y suma 2 puntos

* **Capacidad de ser atractivo**

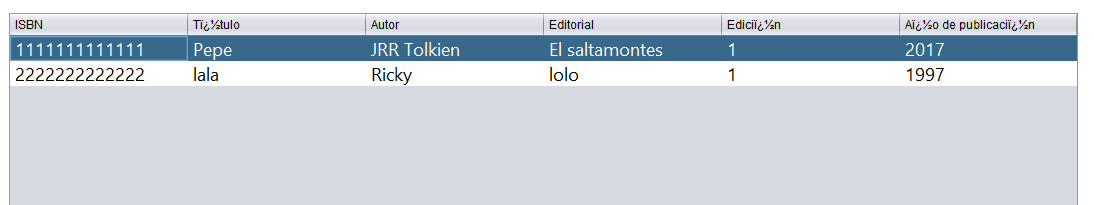
Descripción:

Es la agrupación correcta de funcionalidades dentro del programa en la interfaz gráfica para poder realizar una función específica.

Paso 1: Llenar datos del autor

****

Paso 2: Presionar Agregar y el libro se agregará con éxito



Evaluación: Al necesitar 3 o menos pasos promedio para ejecutar una funcionalidad, la **capacidad de ser atractivo** es aceptable y suma 4 puntos.

**Subcaracteristica Ponderada:** Capacidad de ser atractivo.

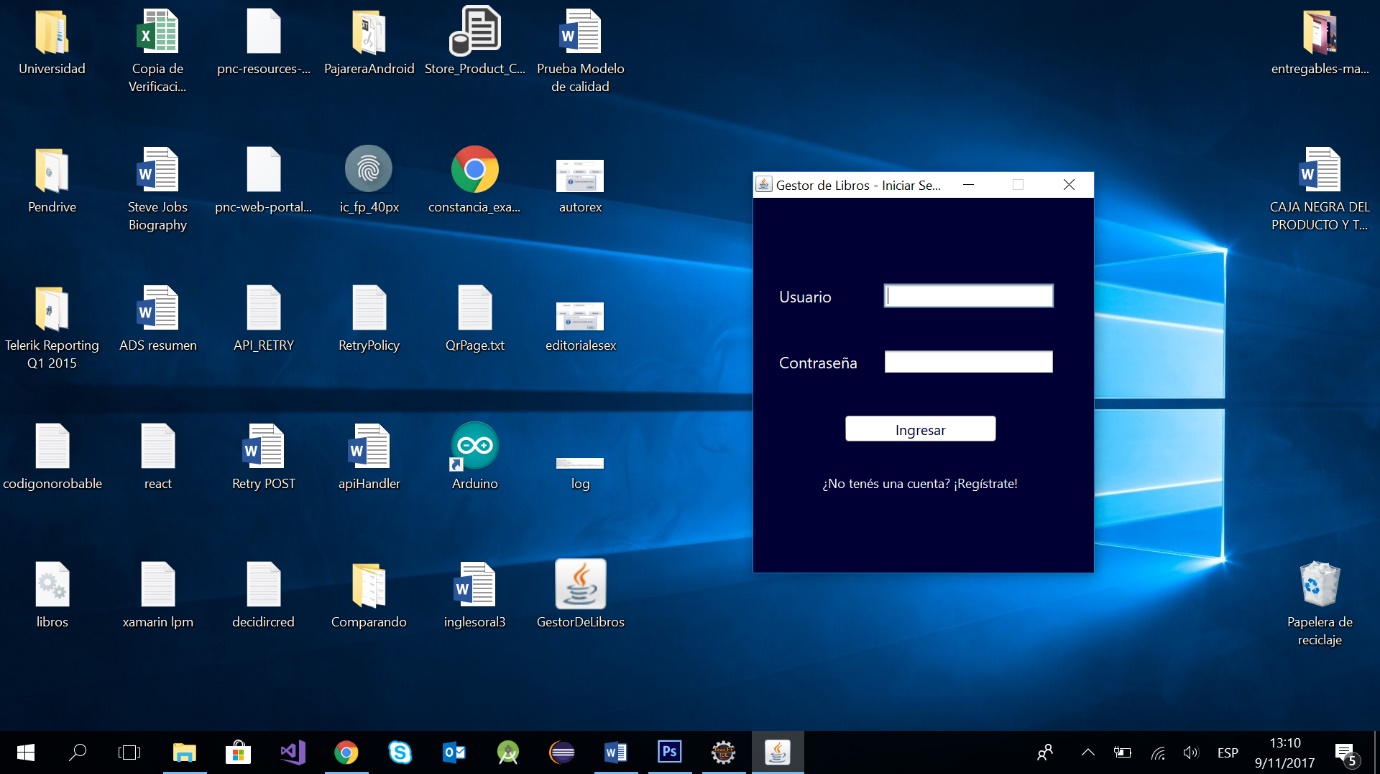
**Subtotal: 8 pts.**

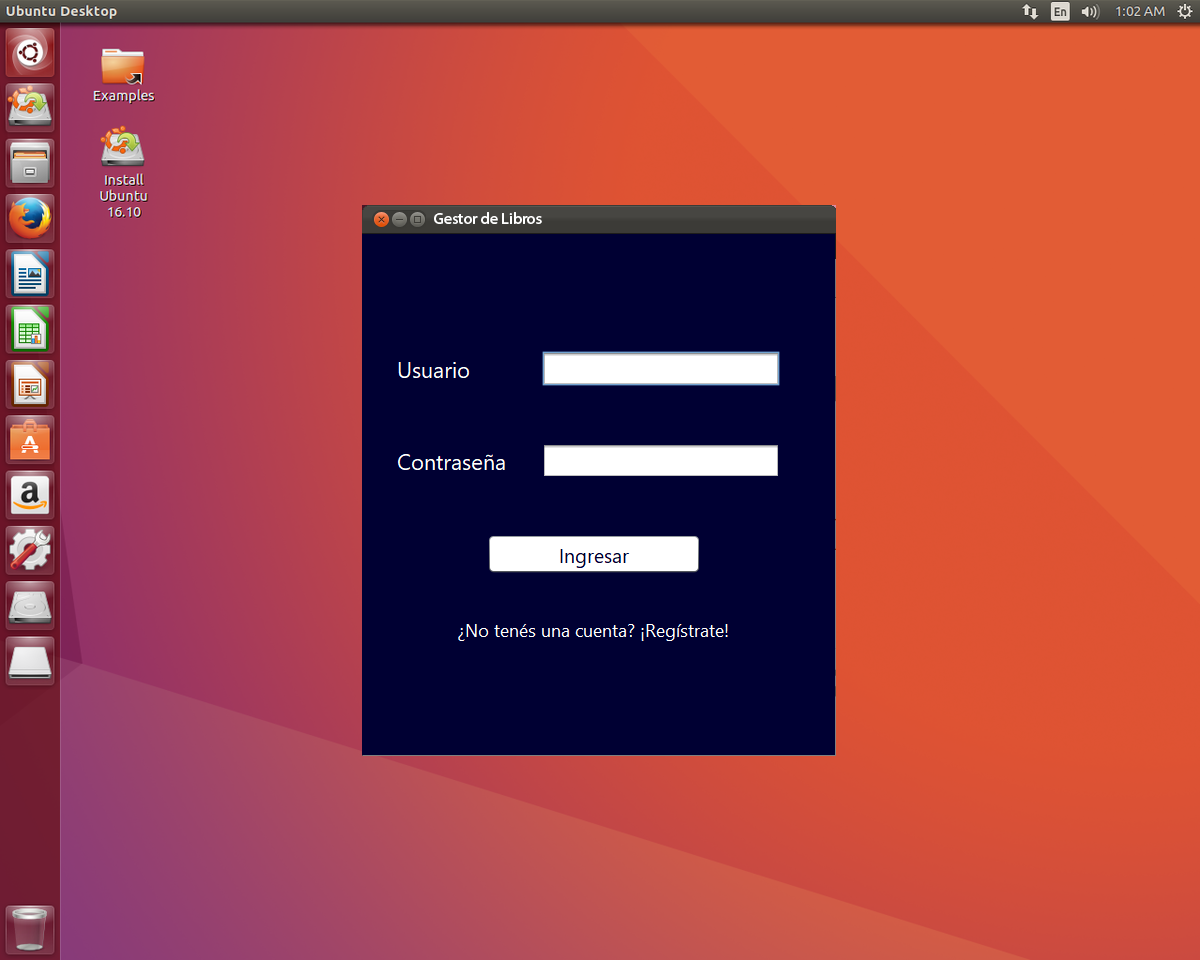
## Portabilidad

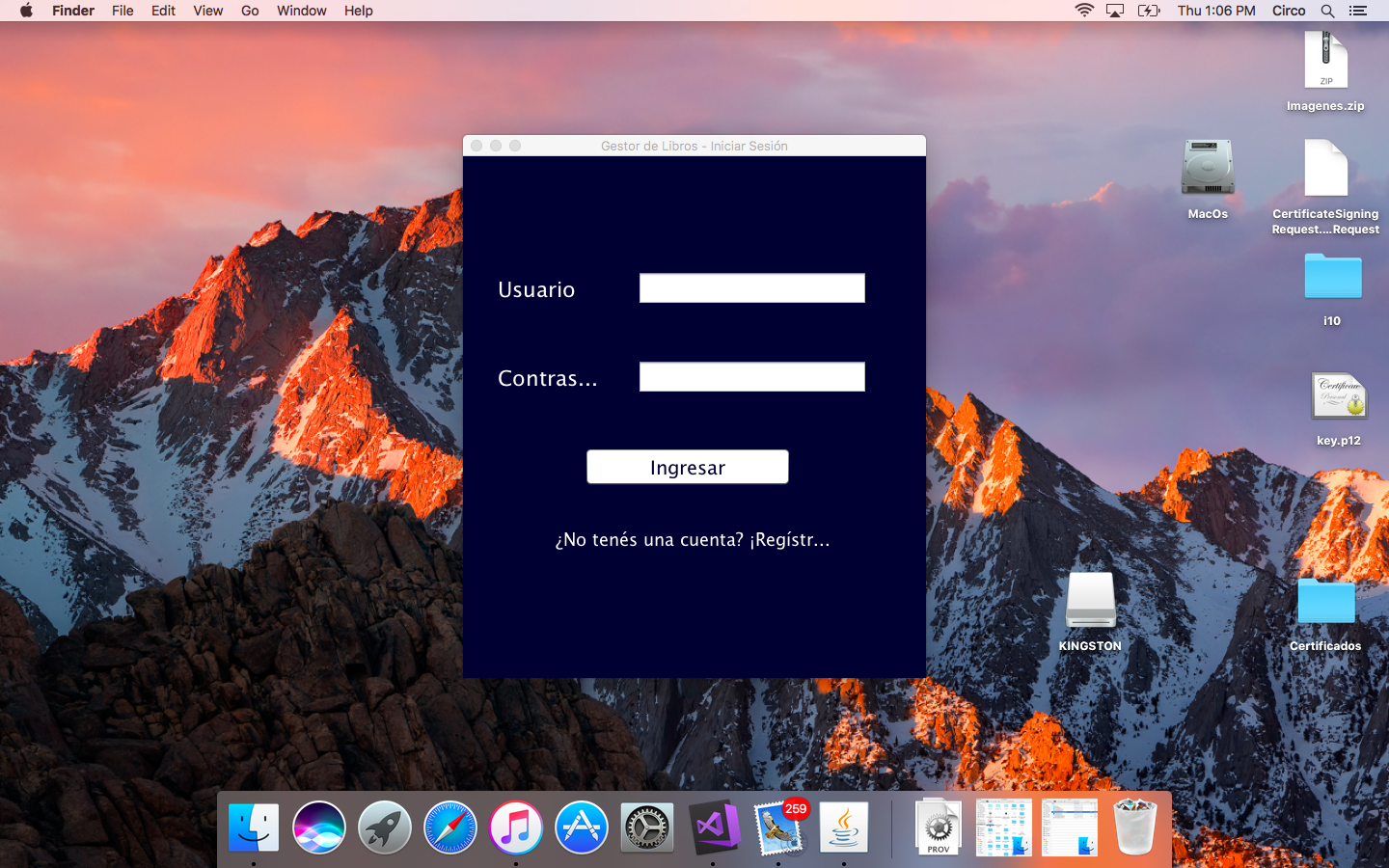
* **Adaptabilidad**

Descripción:

Es la capacidad del producto software de adaptarse a diferentes sistemas operativos sin cambiar su estructura interna.





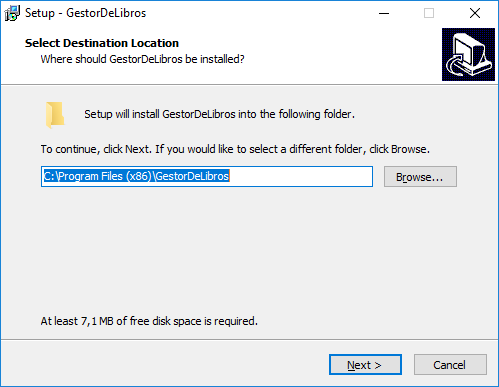
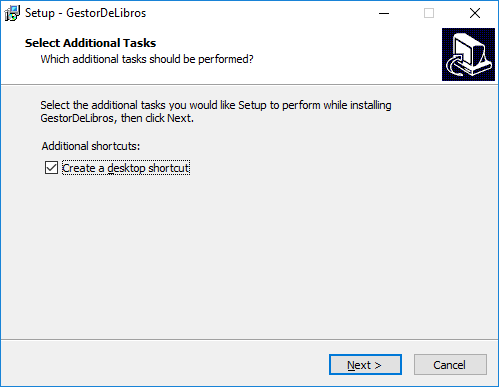


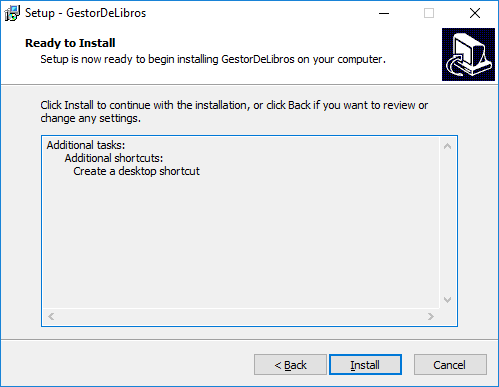
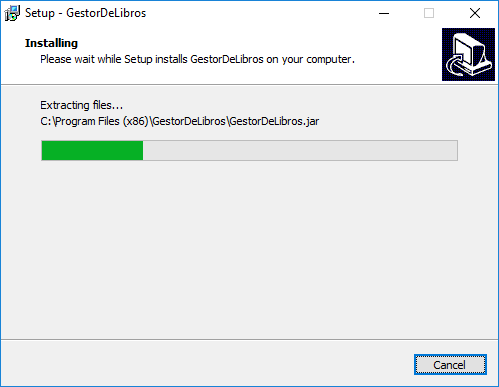
Evaluación: Al ser compatible con 3 o más sistemas operativos, la **adaptabilidad** es aceptable y suma 4 puntos.

* **Instalabilidad**

Descripción

El producto software debe poder ser instalado en una cantidad mínima de pasos.

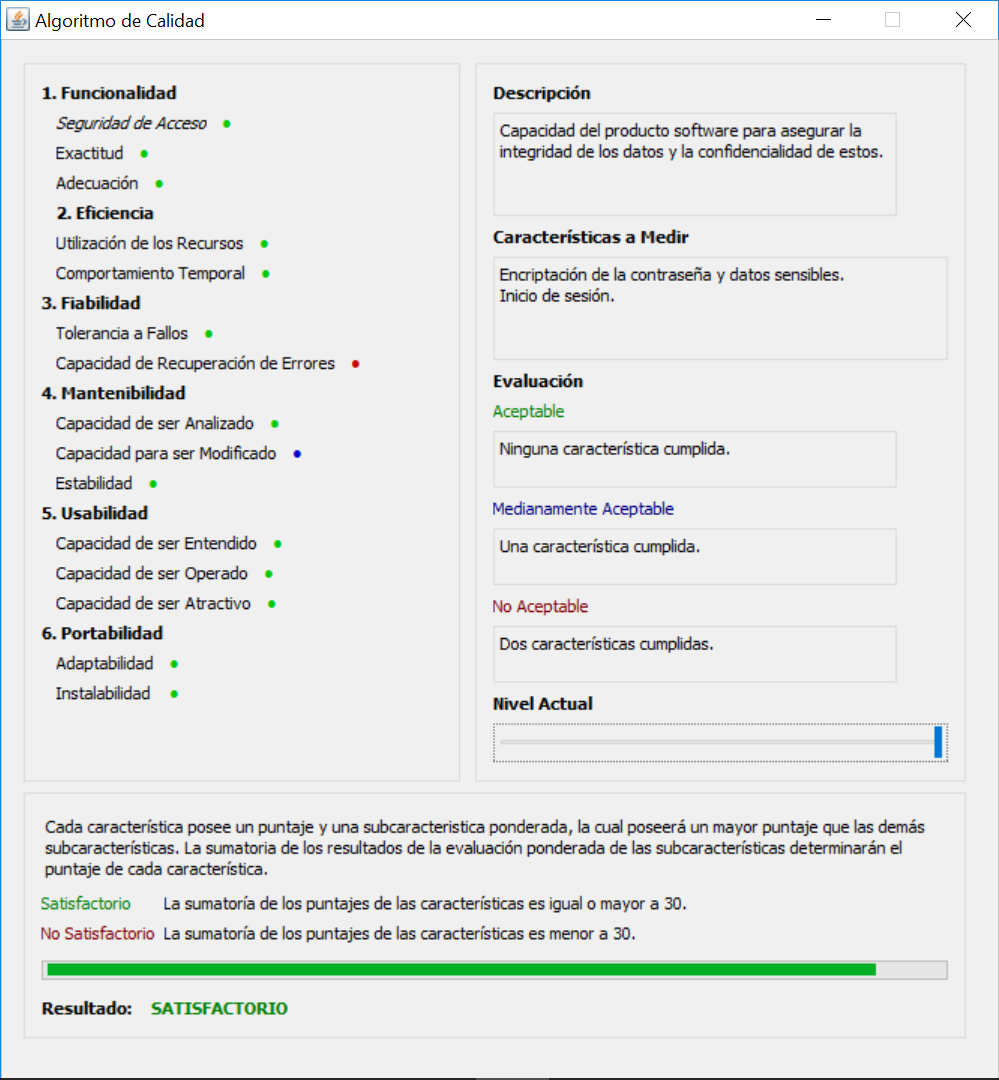
Evaluación: Al instalarse en 4 pasos o menos, la **instalabilidad** es aceptable y suma 2 puntos.

**Subcaracteristica Ponderada:** Adaptabilidad.

**Subtotal: 6 pts.**

Finalmente se ejecuta el algoritmo de calidad con los datos tomados, y obtenemos que nuestro software cumple con el modelo de calidad, sumando un puntaje total de 39 puntos sobre 42 posibles. Y al tener cada característica su subcaracterística ponderada en aceptable, el resultado final del algoritmo es satisfactorio.

## Informe final



# 

# **Prueba de sistemas**

**Hardware utilizado para las pruebas:**

* Procesador: AMD A10 7700K 4C x 3.40 GHz
* Memoria RAM: 8GB.
* HD: 320GB 5400RPM.
* Sistema Operativo: Windows 10 Pro 64 bits.

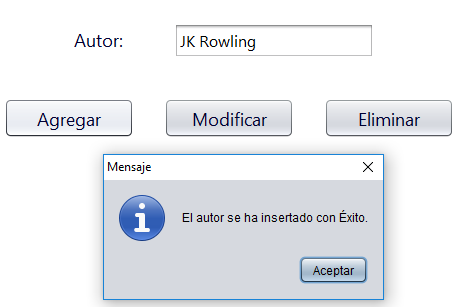
Para esta sección las pruebas de sistemas contempladas en el presente documento son:

1. Prueba de recuperación.
2. Prueba de seguridad.
3. Prueba de resistencia.
4. Prueba de rendimiento.
5. Prueba de aceptación.

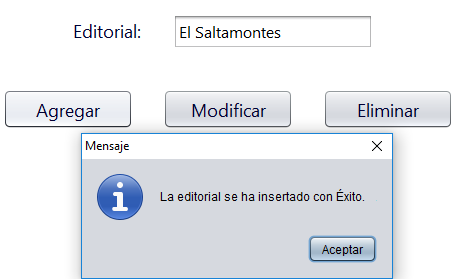
## Prueba de recuperación:

Consiste en forzar el fallo del software y comprobar que la recuperación se lleva a cabo de manera correcta, devolviendo al sistema a un estado coherente.

La prueba consiste en dar de alta un autor y una editorial, reiniciar la computadora en el transcurso y al volver a iniciar el sistema visualizar el autor y editorial correctamente.

****

Damos de alta un autor.

****

Damos de alta una editorial.

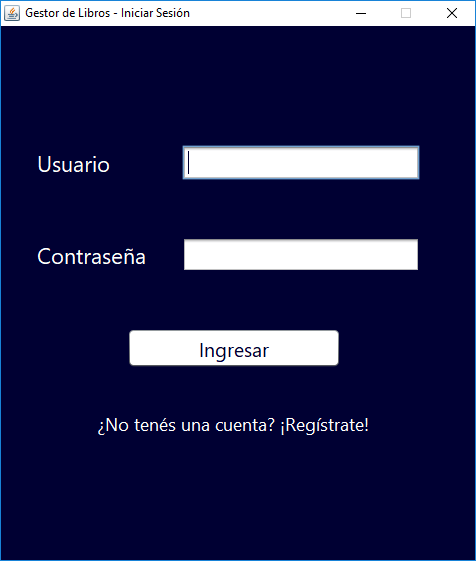
Procedemos a reiniciar el sistema, sin cerrar correctamente el programa. Una vez iniciado el sistema nuevamente, abrimos el programa y notamos que los datos ingresados se encuentran correctamente en su grilla correspondiente.



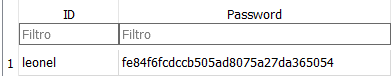
## Prueba de seguridad:

Intentan verificar que los mecanismos de protección incorporados al sistema lo protegerán, de hecho, de penetraciones inadecuadas.

La prueba consiste en que una persona no autorizada (que no esté registrada en el sistema) pueda entrar en el mismo. Para esto se cuenta con una pantalla de Login, que nos pide un usuario y contraseña.



Los distintos usuarios y sus correspondientes contraseñas se guardan en una base de datos en el directorio del sistema, asique para impedir que una persona lea esta base de datos y averigüe los usuarios y contraseñas de los empleados, se procedió a encriptar los mismos.

****

## Prueba de resistencia:

Estas pruebas están diseñadas para que el sistema requiera recursos en cantidad, frecuencia o volumen anormales. La idea es intentar que el sistema se venga abajo por la excesiva tensión a la que se lo somete.

Esta prueba no se realizó ya que el sistema es una aplicación de escritorio monousuario, lo que indica que solo un usuario puede estar accediendo a los datos en un determinado momento. Debido a eso sería imposible realizar una prueba de saturación con peticiones de lectura y escritura a los archivos.

## Prueba de rendimiento:

Es inaceptable que el software proporcione las funciones requeridas fuera de las condiciones de rendimiento exigidas.

La prueba consiste en verificar el uso de CPU y de memoria RAM consumida por el software, asique para esto utilizamos el monitor de recursos ofrecido por Windows.

Como el sistema está basado en Java, el mismo será ejecutado a través del proceso javaw (Java Platform SE binary), el cual genera un ambiente (máquina virtual) que permite la ejecución de la aplicación de manera visual.

****

Como se puede apreciar en las capturas de pantalla anteriores, los valores están dentro de los parámetros normales y no supone ningún “peligro” para el correcto funcionamiento de la computadora

## Prueba de aceptación:

Para esta prueba lo que haremos será realizar un cuestionario de 15 preguntas acerca de las principales características de la calidad del software (Funcionalidad, Fiabilidad, Eficiencia, Portabilidad y Usabilidad) a un grupo de 10 personas.

Para contrastar las respuestas con nuestra ejecución del algoritmo de calidad, evaluamos las 3 respuestas de cada característica, las cuales tienen un puntaje específico detallado más abajo. Si hay 3 respuestas favorables, esa característica sumará el 100% de sus puntos. Si hay 2 respuestas favorables, sumará el 66%. Si hay solo 1 respuesta favorable, el 33%. En el caso de que ninguna respuesta sea favorable, la característica no sumará ningún punto.

A continuación, se mostrarán cuáles son las respuestas favorables según la característica:

Funcionalidad: 1) SI 2) SI 3) SI

Fiabilidad: 1) NO 2) NO 3) NO

Eficiencia: 1) SI 2) SI 3) NO

Usabilidad: 1) SI 2) SI 3) SI

Portabilidad: 1) SI 2) SI 3) NO

Los puntajes de las características son:

* Funcionalidad y Usabilidad: 8 puntos
* Fiabilidad, Eficiencia y Portabilidad: 6 puntos

Por lo tanto, en el caso de las características que tengan 8 puntos, se calcularán de esta forma:

* 1 respuesta favorable: 3 puntos
* 2 respuestas favorables: 5 puntos
* 3 respuestas favorables: 8 puntos

Y en el caso de las características de 6 puntos:

* 1 respuesta favorable: 2 puntos
* 2 respuestas favorables: 4 puntos
* 3 respuestas favorables: 6 puntos

**Funcionalidad**

1. ¿El software cumple con su función?

* SI
* NO

1. ¿Recomendaría el software a sus colegas?

* SI
* NO

1. ¿Le pareció útil el software?

* SI
* NO

**Fiabilidad**

1. ¿Se produjo algún cierre inesperado?

* SI
* NO

1. ¿Realizo alguna acción que no impacto en el sistema?

* SI
* NO

1. ¿En algún momento el software produjo errores?

* SI
* NO

**Eficiencia**

1. ¿El sistema responde rápidamente a sus acciones?

* SI
* NO

1. ¿El sistema devuelve sus consultas rápidamente?

* SI
* NO

1. ¿Siente que su computadora se ralentiza cuando ejecuta el sistema?

* SI
* NO

**Usabilidad**

1. ¿Le pareció fácil utilizar el software?

* SI
* NO

1. ¿Considera que las ayudas del sistema son útiles?

* SI
* NO

1. ‎¿Fue sencillo instalar el sistema?

* SI
* NO

**Portabilidad**

1. ¿El sistema se ejecuta correctamente en distintos sistemas operativos?

* SI
* NO

1. ‎El sistema se instaló correctamente en diferentes sistemas operativos?

* SI
* NO

1. ¿Ocurrió algún problema cuando instalo el sistema?

* SI
* NO

# **Manual de usuario**

## Descripción

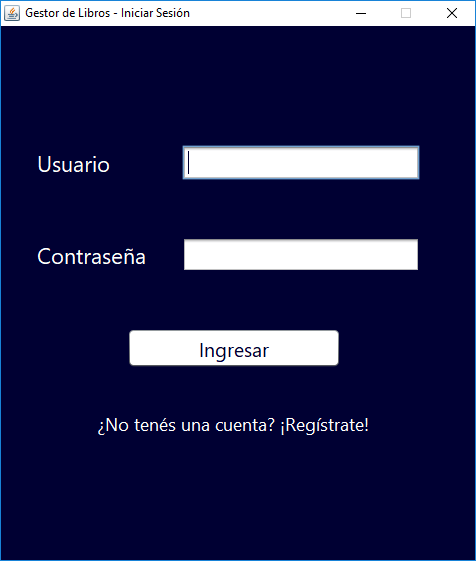
El producto en cuestión tiene como objetivo principal la gestión de libros asociados a una biblioteca.

## Objetivo

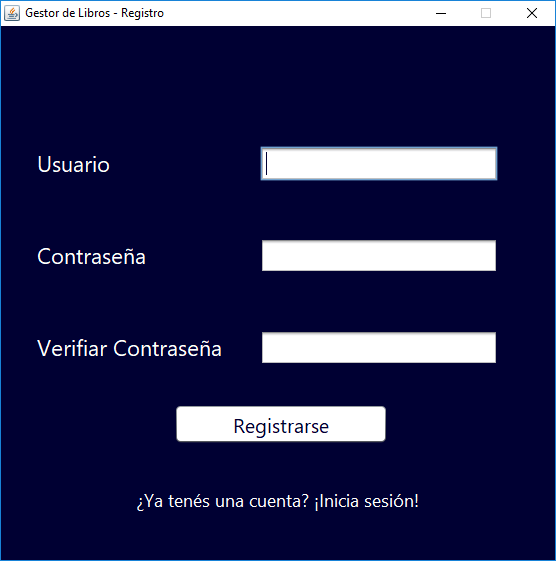
* Permitir al personal de la biblioteca la introducción de nuevos libros, como también la modificación y eliminación de los mismos.
* Proporcionar información acerca de los libros pertenecientes a la biblioteca en cuestión, de forma ordenada y concisa.
* Facilitar el uso para todo el personal de la biblioteca.
* Permitir al personal la introducción de nuevos autores y editoriales además de las ya preexistentes.
* Mantener la seguridad de los datos en la institución teniendo un sistema de logueo, en la que sólo el personal autorizado podrá acceder al sistema.

## Funcionamiento

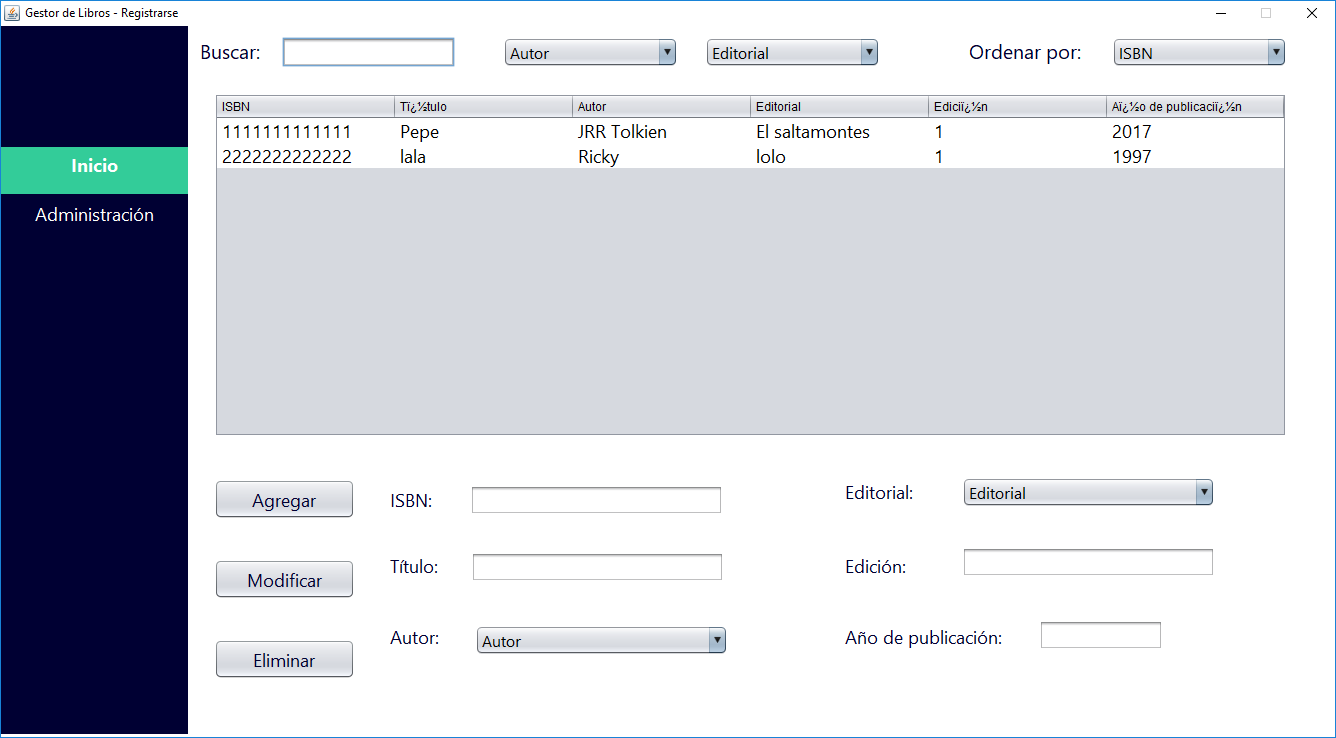
Al iniciar el programa, se visualizará una pantalla de bienvenida al sistema, en donde será necesario loguearse con el fin de poder acceder a las funcionalidades del mismo, como se muestra a continuación:



Se deben ingresar los datos de usuario y contraseña y luego cliquear sobre el botón ¨Ingresar¨. En caso de que el personal no cuente con un usuario propio, el mismo podrá registrarse al sistema, cliqueando la leyenda “Regístrate”, el cual lo llevará hacia la siguiente ventana:



Para poder darse de alta como usuario nuevo en el sistema, serán necesarios dos datos: por un lado, el usuario (el cual no deberá existir en el sistema) y por otro lado la contraseña (que por temas de seguridad debe ser ingresada dos veces). Una vez que se haya llenado el formulario de registración, se deberá cliquear sobre el botón Registrarse y el usuario ya estará dado de alta en el sistema. Una vez registrado (o logueado) correctamente en el sistema se mostrará en pantalla la ventana principal, la cual se ve a continuación:



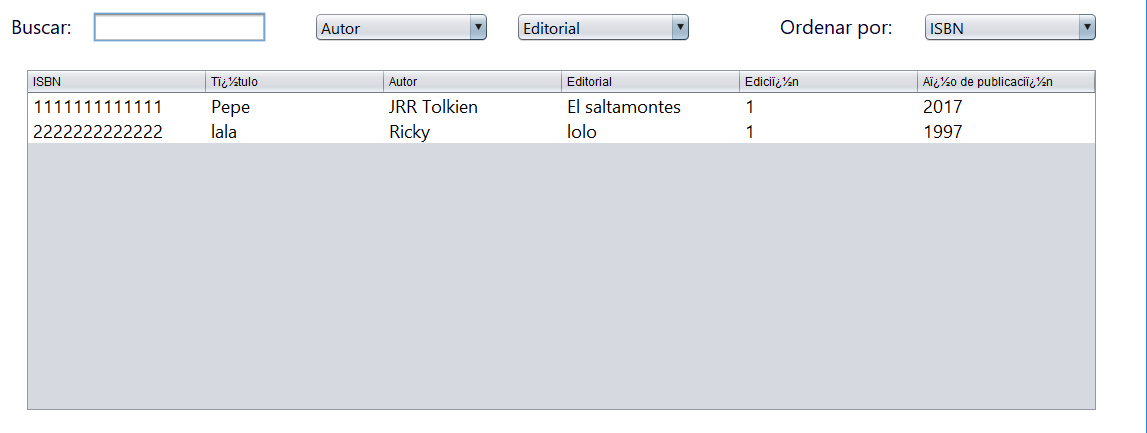
En esta pantalla se visualizarán dos pestañas:

* Inicio
* Administración

### a. Inicio

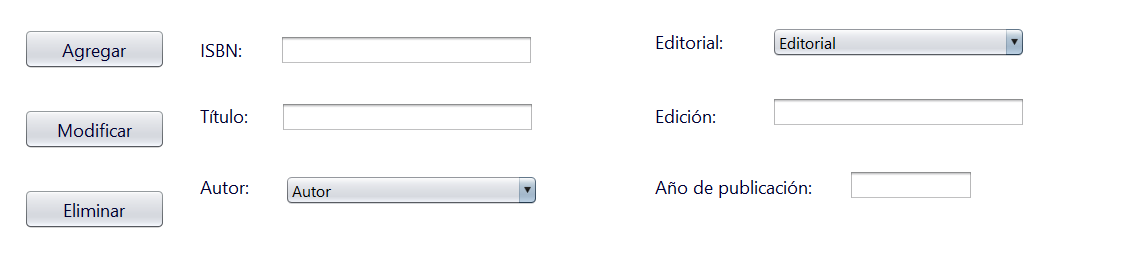
En esta pestaña, se encontrarán dos zonas bien marcadas, nos concentraremos en la primera:

**Consulta de libros**



En esta sección, se podrá realizar la consulta de toda la información de los libros disponibles en la biblioteca. Para ello se deberá introducir, total o parcialmente, el título y/o número del ISBN en el campo “Buscar” (también existe la posibilidad de dejarlo en blanco). Luego, de ser necesario, se pueden aplicar filtros por Autor y/o editorial (seleccionando los mismos de las listas desplegables). Y por último, los resultados se pueden ordenar según el campo deseado, seleccionando el mismo de la lista desplegable a continuación de la leyenda “Ordenar por”.

**Administración de Libros**

****

Esta sección nos permite añadir, modificar o eliminar un libro a nuestro sistema:

* Agregar:

Consiste en el ingreso de un ISBN válido, un título, edición y año de publicación del mismo, y luego seleccionar de la lista desplegable la editorial y autor del mismo. Por último, cliquear en el botón Agregar para registrar el libro en el sistema.

* Modificar:

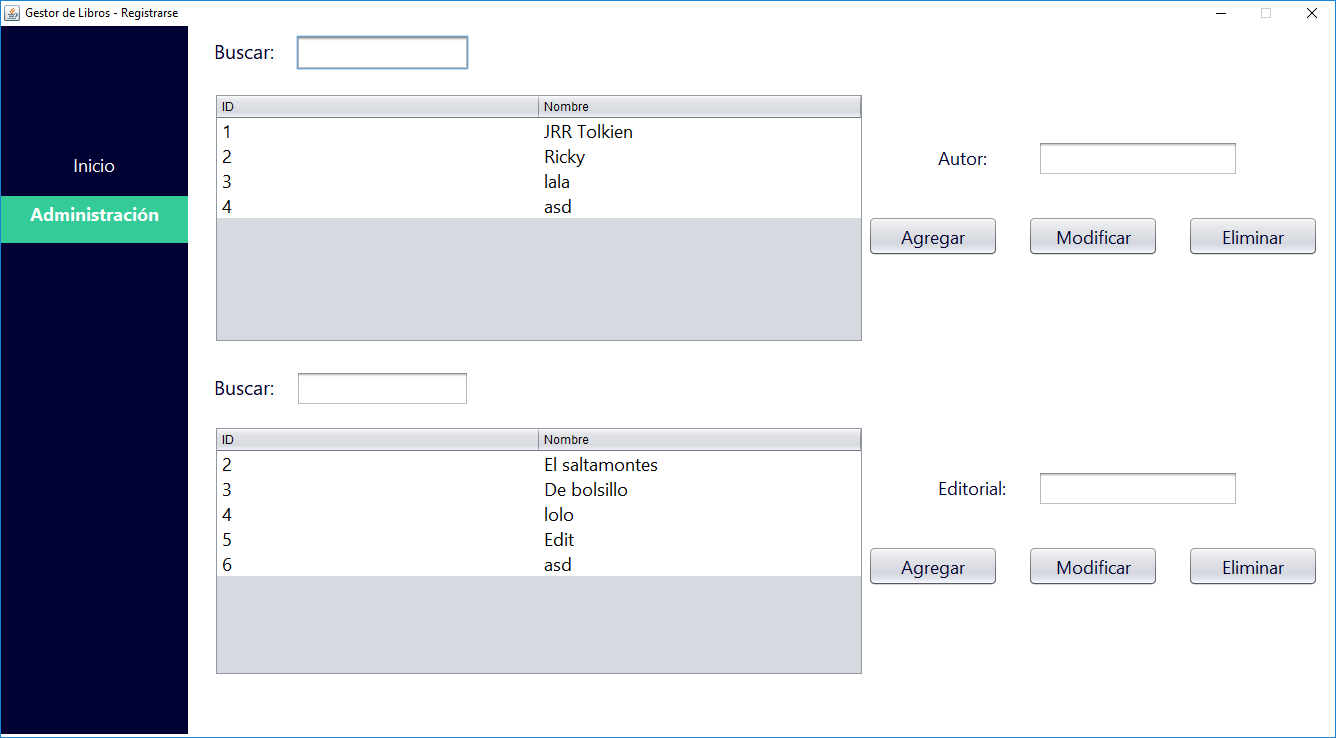
Consiste en seleccionar uno de los libros de la grilla de la sección consulta de libros, y luego colocar los datos correspondientes a la modificación que se desee hacer, una vez llenado el formulario como se indicó anteriormente, se debe cliquear el botón Modificar y el libro será actualizado exitosamente en el sistema.

* Eliminar:

Consiste en seleccionar uno de los libros de la grilla de la sección consulta de libros, y luego cliquear en el botón Eliminar, esto removerá permanentemente el libro del sistema.

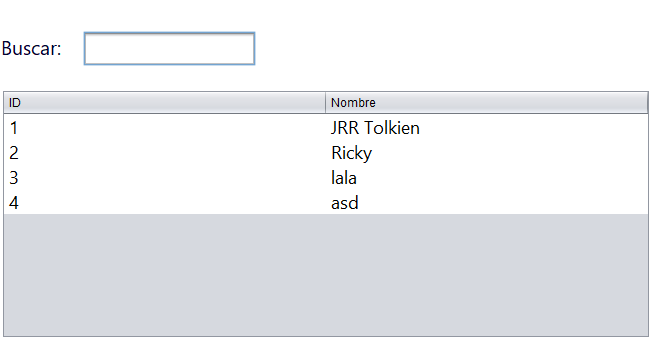
### b. Administración

En esta pestaña, nos encontramos con dos zonas bien definidas, una para la administración de autores y otra para la administración de editoriales:



**Autores**

**Consulta de Autores**



En esta sección, se podrá realizar la consulta de los autores registrados en el sistema de la biblioteca. Para ello se deberá introducir, total o parcialmente, el nombre y/o número del ID en el campo “Buscar” y los resultados se verán inmediatamente reflejados en la grilla.

**Administración de Autores**

****

Esta sección nos permite añadir, modificar o eliminar un autor de nuestro sistema:

* Agregar:

Consiste en el ingreso del nombre del autor deseado y cliquear en el botón Agregar para registrar el autor en el sistema.

* Modificar:

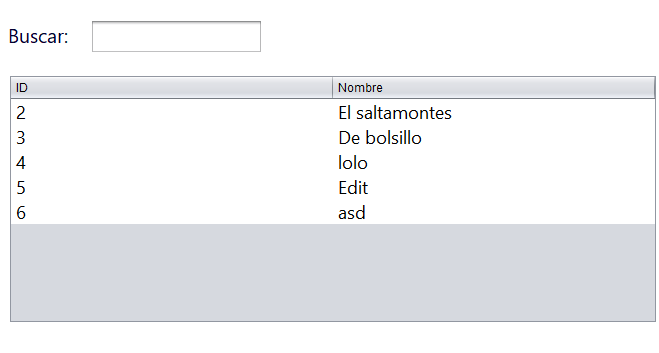
Consiste en seleccionar uno de los autores de la grilla de la sección consulta de autores, y luego colocar el nuevo nombre al autor correspondiente, una vez hecho esto se debe cliquear el botón Modificar y el autor será actualizado exitosamente en el sistema.

* Eliminar:

Consiste en seleccionar uno de los autores de la grilla de la sección consulta de autores, y luego cliquear en el botón Eliminar, esto removerá permanentemente el autor del sistema.

**Editoriales**

**Consulta de Editoriales**



En esta sección, se podrá realizar la consulta de las editoriales registradas en el sistema de la biblioteca. Para ello se deberá introducir, total o parcialmente, el nombre y/o número del ID en el campo “Buscar” y los resultados se verán inmediatamente reflejados en la grilla.

**Administración de Editoriales**



Esta sección nos permite añadir, modificar o eliminar una editorial de nuestro sistema:

* Agregar:

Consiste en el ingreso del nombre de la editorial deseada y cliquear en el botón Agregar para registrar la editorial en el sistema.

* Modificar:

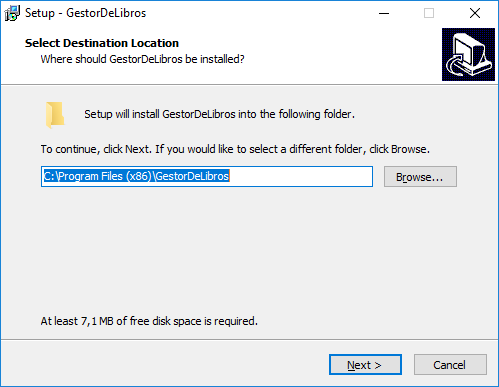
Consiste en seleccionar una de las editoriales de la grilla de la sección consulta de editoriales, y luego colocar el nuevo nombre a la editorial correspondiente, una vez hecho esto se debe cliquear el botón Modificar y la editorial será actualizada exitosamente en el sistema.

* Eliminar:

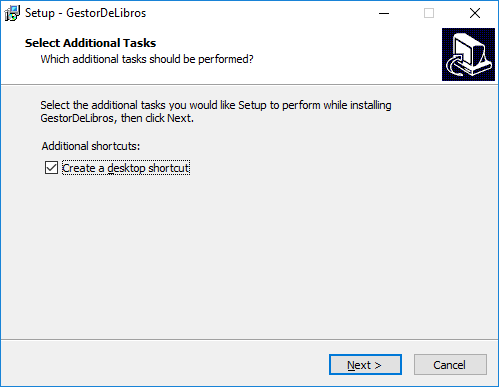
Consiste en seleccionar una de las editoriales de la grilla de la sección consulta de editoriales, y luego cliquear en el botón Eliminar, esto removerá permanentemente a la editorial del sistema.

## Instalación

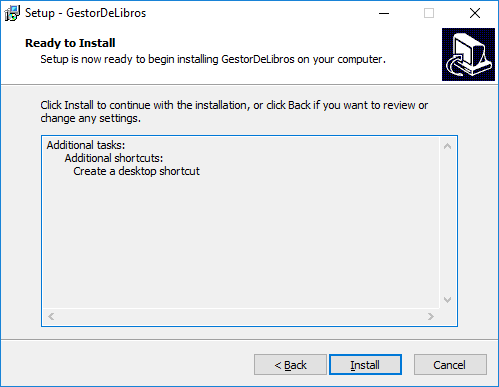
Al hacer doble clic sobre el instalador, se abre una ventana en la que seleccionar la ruta en la cual se desea instalar el software:



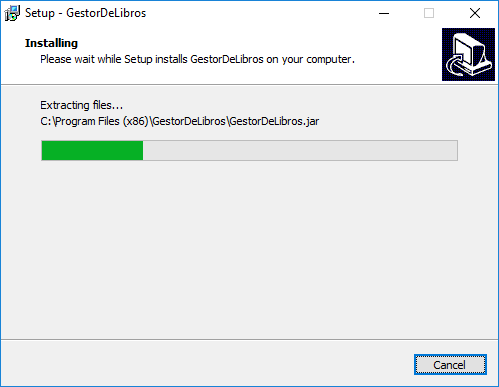
En dicha pantalla se puede cambiar la ruta de instalación por defecto cliqueando en el botón “Browse…”, a su vez indica el espacio requerido en el sistema para poder instalar correctamente el programa. Una vez que se haya seleccionado la ruta de instalación, cliquear el botón “Next”.



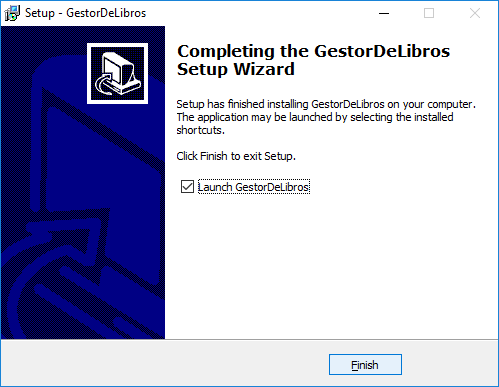
En esta pantalla debe seleccionarse si se desea la creación de un acceso directo en el escritorio. Una vez hecho esto presionar el botón “Next”.



Esta ventana nos indicará la configuración que hemos elegido hasta el momento, en caso de estar de acuerdo presionar el botón “Install” para proceder a instalar el programa en nuestro sistema.



Una vez completada la barra de progreso se abrirá la siguiente ventana:



En esta ventana se debe seleccionar si se desea o no abrir el programa y por último presionar el botón “Finish” para finalizar con la instalación.

# **Cubierta del CD**

Como parte de la norma ISO 9127, además del manual de usuario, se presenta la cubierta de CD (tapa) con la cual es presentado el software de gestor de libros para que los compradores potenciales puedan evaluar la aplicabilidad del paquete y sus requisitos.

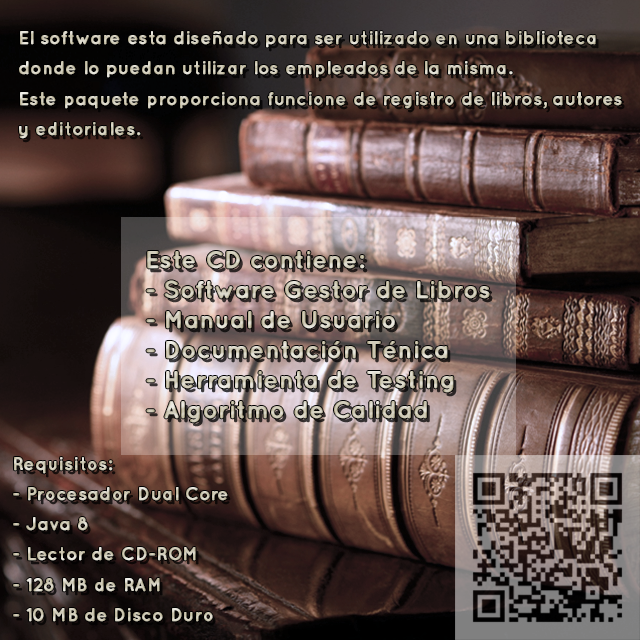
La misma incluye:

* Una breve descripción del software.
* Requisitos de hardware y software.
* Contenido proporcionado.
* Código QR del producto.

**Tapa**



**Contratapa**

****

**Tapa de CD**

# **Encuestas:**

**Funcionalidad**

1. ¿El software cumple con su función?

* SI
* NO

1. ¿Recomendaría el software a sus colegas?

* SI
* NO

1. ¿Le pareció útil el software?

* SI
* NO

**Fiabilidad**

1. ¿Se produjo algún cierre inesperado?

* SI
* NO

1. ¿Realizo alguna acción que no impacto en el sistema?

* SI
* NO

1. ¿En algún momento el software produjo errores?

* SI
* NO

**Eficiencia**

1. ¿El sistema responde rápidamente a sus acciones?

* SI
* NO

1. ¿El sistema devuelve sus consultas rápidamente?

* SI
* NO

1. ¿Siente que su computadora se ralentiza cuando ejecuta el sistema?

* SI
* NO

**Usabilidad**

1. ¿Le pareció fácil utilizar el software?

* SI
* NO

1. ¿Considera que las ayudas del sistema son útiles?

* SI
* NO

1. ‎¿Fue sencillo instalar el sistema?

* SI
* NO

**Portabilidad**

1. ¿El sistema se ejecuta correctamente en distintos sistemas operativos?

* SI
* NO

1. ‎El sistema se instaló correctamente en diferentes sistemas operativos?

* SI
* NO

1. ¿Ocurrió algún problema cuando instalo el sistema?

* SI
* NO

# **Resultados:**

