

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA

FACULTAD DE INGENIERÍA

INTRODUCCIÓN A LA PROGRAMACIÓN Y COMPUTACIÓN 1

CATEDRÁTICO:

TUTOR ACADÉMICO:



Fernando Josué López Iboy

CARNÉ: 202300840

SECCIÓN: G

GUATEMALA, 9 DE SEPTIEMBRE DEL 2,024

ÍNDICE

Contenido

ÍNDICE	1
INTRODUCCIÓN.....	2
OBJETIVOS.....	2
1. GENERAL	2
2. ESPECÍFICOS	2
ALCANCES DEL SISTEMA	3
ESPECIFICACIÓN TÉCNICA	3
REQUISITOS DE SOFTWARE.....	3
DESCRIPCIÓN DE LA SOLUCIÓN.....	4
LÓGICA DEL PROGRAMA	5
❖ Clases	5
➤ Librerías.....	5
➤ Variables Globales de la clase	5
➤ Función Main.....	6
➤ Procedimientos, métodos y Funciones utilizadas	6

INTRODUCCIÓN

En el presente informe se documenta la información relacionada con el desarrollo de la aplicación de escritorio para la gestión de experimentos ingresados por los investigadores de IPC Quimik, gestión de estos investigadores, patrones en las muestras y resolución de experimentos. Pudiendo ser esta información editada, agregada y eliminada en la base interna de la aplicación, Para la realización de la aplicación se utilizó el lenguaje de programación JAVA, mediante únicamente código, sin utilizar herramientas de creación de objetos.

OBJETIVOS

1. GENERAL

- 1.1. Familiarizar al estudiante con el lenguaje de programación Java.
- 1.2. Aplicar conocimientos adquiridos en el curso de IPC 1
- 1.3. Elaborar la lógica para presentar una solución a la problemática planteada

2. ESPECÍFICOS

- 2.1. Utilizar el lenguaje de programación Java como herramienta de desarrollo de software.
- 2.2. Aplicación de conceptos de programación orientada a objetos.
- 2.3. Construcción de aplicaciones con interfaz gráfica.
- 2.4. Implementación de sentencias de control, ciclos, arreglos, matrices y librerías de interfaz gráfica.
- 2.5. Aplicación de conceptos de programación para crear herramientas administrativas.
- 2.6. Implementación de soluciones por medio de algoritmos de encriptación.

- 2.7. ● Crear respaldos de información y restablecerla con el uso de serialización

ALCANCES DEL SISTEMA

El objetivo de este manual es exponer la lógica detrás de la aplicación, para que lo entienda alguien relacionado a la materia de la programación

ESPECIFICACIÓN TÉCNICA

REQUISITOS DE SOFTWARE

- La aplicación puede ser ejecutada en cualquier computador, media vez este tenga instalado JAVA, Oracle JDK 8, Netbeans IDE, o en su caso Visual Studio Code

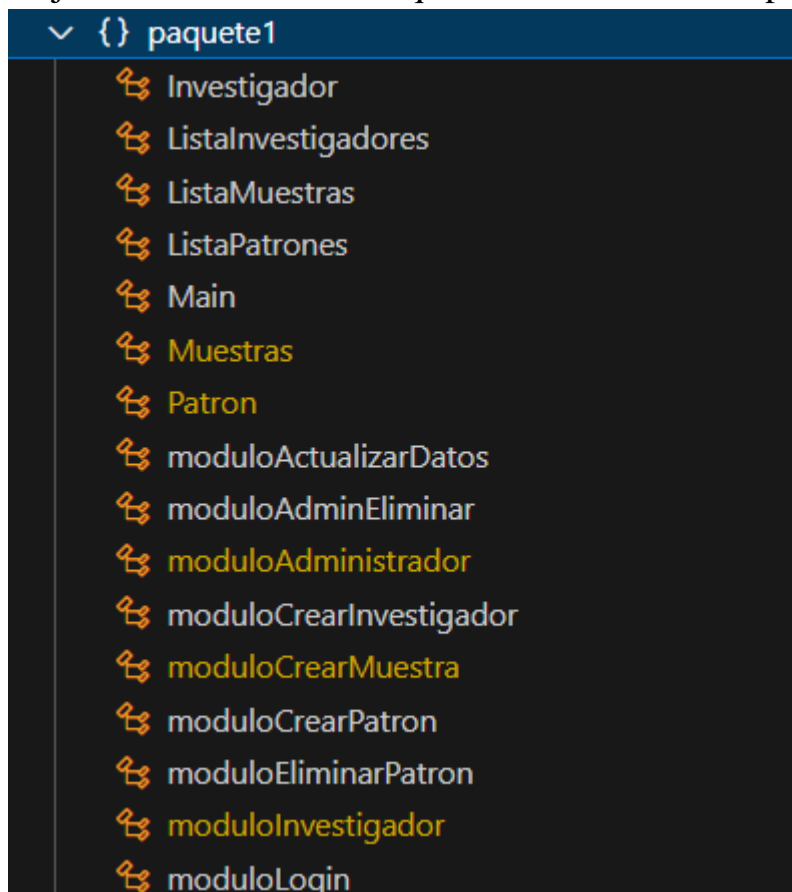
DESCRIPCIÓN DE LA SOLUCIÓN

- Para la solución del enunciado, se crearon clases que giran en torno a Array Lists para realizar los trabajos de modificación, agregación, eliminación y obtención de datos de una manera más sencilla para el desarrollo

LÓGICA DEL PROGRAMA

❖ Clases

Para el desarrollo de la aplicación se crearon múltiples clases, para llevar un mejor control de los datos que se movilizan en la aplicación



➤ Librerías

Para la creación de la aplicación, se utilizaron las librerías, AWT para cambios de colores, IO, para lectura de archivos, Swing, para la creación de las pestañas y ORG para eventos de mouse tal cómo clicar una etiqueta.

➤ Variables Globales de la clase

Para describir variables globales de una manera más sintetizada, se explicará la dinámica de las clases:

El programa consiste en 3 clases principales que giran entorno a Array Lists, Para los investigadores, para las muestras y para los

patrones, Teniendo cómo variables globales, el archivo dónde se almacenan las entradas, la lista dónde se van guardando los objetos, y la cantidad de elementos registrados.

```
public class ListaInvestigadores {
    //-----Variables globales-----
    private static int cantidad=0;
    public static ArrayList <Investigador> listaInvestigadores = new ArrayList<>();
    private static File f = new File(pathname:"C:\\Users\\josue\\Desktop\\Proyecto1-IPC1-Quimik\\investigadores.csv");

    public class ListaMuestras {
        private static int cantidadMuestras=0;
        public static ArrayList <Muestras> listaMuestras = new ArrayList<>();
        private static File f = new File(pathname:"C:\\Users\\josue\\Desktop\\Proyecto1-IPC1-Quimik\\muestras.csv");

        public class ListaPatrones {
            private static int cantidad=0;
            public static ArrayList <Patron> listaPatrones = new ArrayList<>();
            private static File f = new File(pathname:"C:\\Users\\josue\\Desktop\\Proyecto1-IPC1-Quimik\\patrones.csv");
```

➤ Función Main

La función main sólo sirvió para la inicialización de la ventana del login

```
paquete1 > J Main.java > ...
1  package paquete1;
2
3  public class Main {
4
5      Run | Debug
6      public static void main(String [] args){
7          ListaInvestigadores.crearAdmin();
8          moduloLogin ventanaLogin= new moduloLogin();
9          ventanaLogin.setVisible(b:true);
10     }
11 }
12
```

➤ Procedimientos, métodos y Funciones utilizadas

Al haber sido desarrollado completamente con Swing, hay una cantidad inmensa de métodos, pero las principales son las de las clases de ArrayLists, a continuación, se expondrán:

```
18 //-----Método para crear Investigadores-----
19 //(Todos los investigadores se crear con el parámetro "isAdmin" establecido en false)
20 public static void agregarInvestigador(String codigo, String nombre, String genero, String contrasenia){
```

Método constructor, sirve para crear un nuevo elemento que se añadirá a la Array List de investigadores.

```

32 //-----Método para verificar la existencia de un usuario-----
33 public static boolean comprobarIdentidad(String codigo){

```

Método para la verificación de existencia, se envía un código al método, y este devuelve un valor booleano que indica si el elemento existe o no en la lista.

```

43 //-----Método para comprobar la contraseña-----
44 > public static boolean comprobarContraseña(String contra,int pos){...

```

Método de comprobación de contraseña, se envía un código de investigador, y la contraseña para intentar acceder a la información de este, el método comprueba que los datos sean correctos, y en cada caso realizar una acción.

```

51 //-----Método para regenerar el archivo CSV con datos actualizados, eliminaciones, cambios...etc-----
52 public static void regenerarCSV(){

```

Método para regenerar CSV, Se utiliza en las tablas de información al cargar datos, el CSV se vacía y se vuelve a llenar con los datos que contenga la lista, sirve para modificaciones, eliminaciones, e ingreso de nuevos elementos a la lista.

```

85 public static String[] toStringTodosNombres(){

```

Método para que todos los códigos de los investigadores registrados puedan ser ingresados en un JComboBox con facilidad.

```

94 //Método para asignar una muestra al investigador
95 public static void asignarMuestra(String codigoInvestigador, String codigoMuestra){

```

Método de asignación de muestra, sirve para asignar una muestra registrada a un investigador registrado, una vez asignada la muestra no puede ser desasignada.