
	Computación	Docente: Diego Quisi Peralta
	Programación Aplicada	Período Lectivo: Septiembre 2020 – Febrero 2021

		<b>FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES</b>	
<b>CARRERA:</b> COMPUTACIÓN/INGENIERÍA DE SISTEMAS		<b>ASIGNATURA:</b> PROGRAMACIÓN APLICADA	
<b>NRO. PROYECTO:</b>	1.1	<b>TÍTULO PROYECTO:</b> Prueba Practica 2 Desarrollo e implementación de un sistema de simulación de acceso y atención bancaria	
<b>OBJETIVO:</b> Reforzar los conocimientos adquiridos en clase sobre la programación en Hilos en un contexto real.			
<b>INSTRUCCIONES:</b>		1. Revisar el contenido teórico y práctico del tema	
		2. Profundizar los conocimientos revisando los libros guías, los enlaces contenidos en los objetos de aprendizaje Java y la documentación disponible en fuentes académicas en línea.	
		3. Deberá desarrollar un sistema informático para la simulación y una interfaz gráfica.	
		4. Deberá generar un informe de la práctica en formato PDF y en conjunto con el código se debe subir al GitHub personal y AVAC.	
		5. <b>Fecha de entrega:</b> El sistema debe ser subido al git hasta <b>17 de enero del 2021 – 23:55.</b>	
<b>ACTIVIDADES POR DESARROLLAR</b>			

## 1. Enunciado:

Realizar un sistema de simulación de acceso y atención a través de colas de un banco.

**Problema:** Un banco necesita controlar el acceso a cuentas bancarias y para ello desea hacer un programa de prueba en Java que permita lanzar procesos que ingresen y retiren dinero a la vez y comprobar así si el resultado final es el esperado.

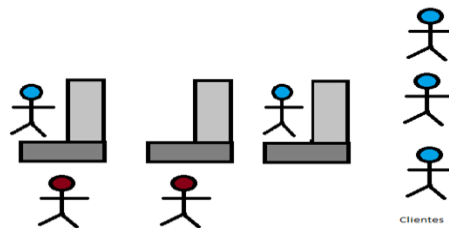
Se parte de una cuenta con 100 euros y se pueden tener procesos que ingresen 100 euros, 50 o 20. También se pueden tener procesos que retiran 100, 50 o 20 euros. Se desean tener los siguientes procesos:

- 40 procesos que ingresan 100
- 20 procesos que ingresan 50
- 60 que ingresen 20.

De la misma manera se desean lo siguientes procesos que retiran cantidades.

- 40 procesos que retiran 100
- 20 procesos que retiran 50
- 60 que retiran 20.


Ademas en el banco, existen 3 cajeros que pueden atender y hay un cola inicial de 10 clientes para ser atendidos, el proceso de atención es de 20 – 15 segundos y los clientes llegan constantemente cada 30 - 50 segundos. Ningún cajero puede atender simultáneamente, adicionalmente el tiempo de moverme de la cola al estante del cajero es de 2 - 5 segundos, esto deberán ser generados aleatoriamente entre los 100 clientes que disponen una cuenta, estos pueden volver a ingresar el numero de veces que sea necesario.



**Se desea comprobar que tras la ejecución la cuenta tiene exactamente 100 euros, que era la cantidad de la que se disponía al principio. Realizar el programa Java que demuestra dicho hecho.**

### Calificación:

- Diagrama de Clase 10%
- MVC: 10%
- Técnicas de Programación aplicadas (Java 8, Reflexión y Programación Genérica): 10%

	<b>Computación</b>	<b>Docente: Diego Quisi Peralta</b>
	Programación Aplicada	<b>Período Lectivo:</b> Septiembre 2020 – Febrero 2021

- Hilos 30%
- Sincronización 10%
- Interfaz Gráfica de simulación 20%
- Informe: 10%

## 2. Informe de Actividades:

- Planteamiento y descripción del problema.
- Diagramas de Clases.
- Patrón de diseño aplicado
- Descripción de la solución y pasos seguidos.
  - Comprobación de las cuentas bancarias e interfaz gráfica.
- Conclusiones y recomendaciones.
- Resultados.

## RESULTADO(S) OBTENIDO(S):

- Interpreta de forma correcta los algoritmos de programación y su aplicabilidad.
- Identifica correctamente qué herramientas de programación se pueden aplicar.

## CONCLUSIONES:

- Los estudiantes identifican las principales estructuras para la creación de sistemas informáticos.
- Los estudiantes implementan soluciones gráficas en sistemas.
- Los estudiantes están en la capacidad de implementar hilos.

## RECOMENDACIONES:

- Revisar la información proporcionada por el docente previo a la práctica.
- Haber asistido a las sesiones de clase.
- **Consultar con el docente las dudas que puedan surgir al momento de realizar la prueba.**

## BIBLIOGRAFIA:

[1]: <https://www.ups.edu.ec/evento?calendarBookingId=98892>

**Docente / Técnico Docente:** Ing. Diego Quisi Peralta Msc.

**Firma:** \_\_\_\_\_



## FORMATO DE INFORME DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA ESTUDIANTES

CARRERA:

ASIGNATURA:

NRO. PRÁCTICA:

TÍTULO PRÁCTICA:

OBJETIVO ALCANZADO:

### ACTIVIDADES DESARROLLADAS

1. Revisar el contenido teórico y práctico del tema

Que es un hilo(Thread)

Un hilo es un proceso ligero o subproceso es una secuencia de tareas encadenadas muy pequeña que puede ser ejecutada por un sistema operática.

Permite que una aplicación realice varias tareas a la vez.

```
private Thread tread;
```

```
tread = new Thread(this);  
tread.start();
```

Que es Runnable

Proporciona un método alternativo a la utilización de la clase Thread, para los casos en los que no es posible hacer que la clase definida extienda la clase Thread. Esto ocurre cuando dicha clase desea ejecutar un hilo.

```
public class ControladorCajero implements Runnable {
```

Patrón que se utilizo:

Patrón Singleton es un patrón que nos permite restringir la creación de objetos pertenecientes a una clase o el valor de un tipo a un único objeto.

```
//Inicio Singleton
private static ControladorCola instance = new ControladorCola();

public static ControladorCola getInstance() {
    return instance;
}

//Fin singleton.
```

Método Run()

Este método Run constituye el cuerpo de un hilo en ejecución, Este es el único método de interfaz Runnable.

```
@Override
public void run() {
```

Es llamado por el método start().

```
tread.start();
```

3. Deberá desarrollar un sistema informático para la simulación y una interfaz gráfica.

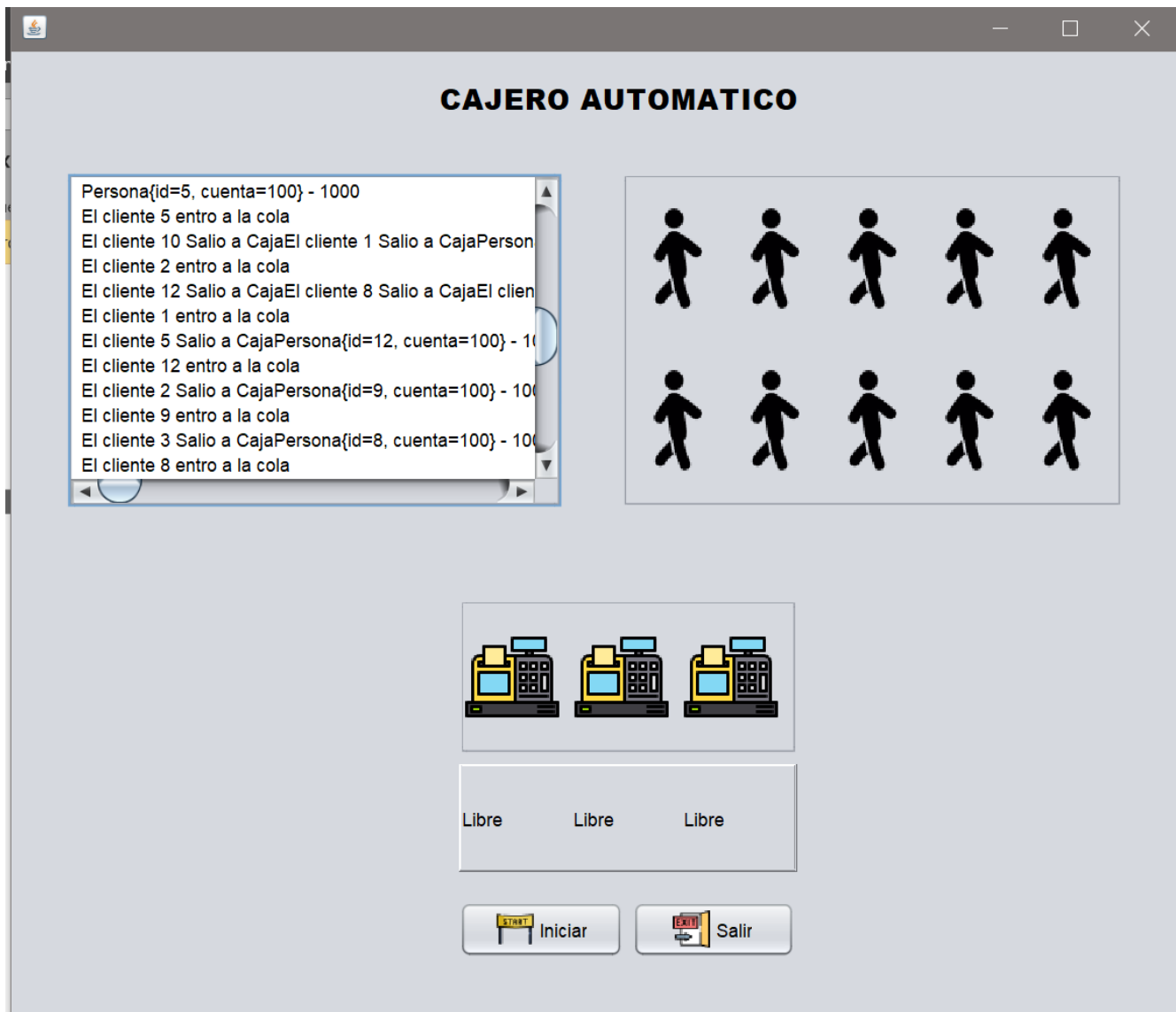
Interfaz creada

Se ha creado la siguiente interfaz donde nos irá mostrando el proceso que realiza en el txtArea y en los JLabel si los cajeros están ocupados o no.

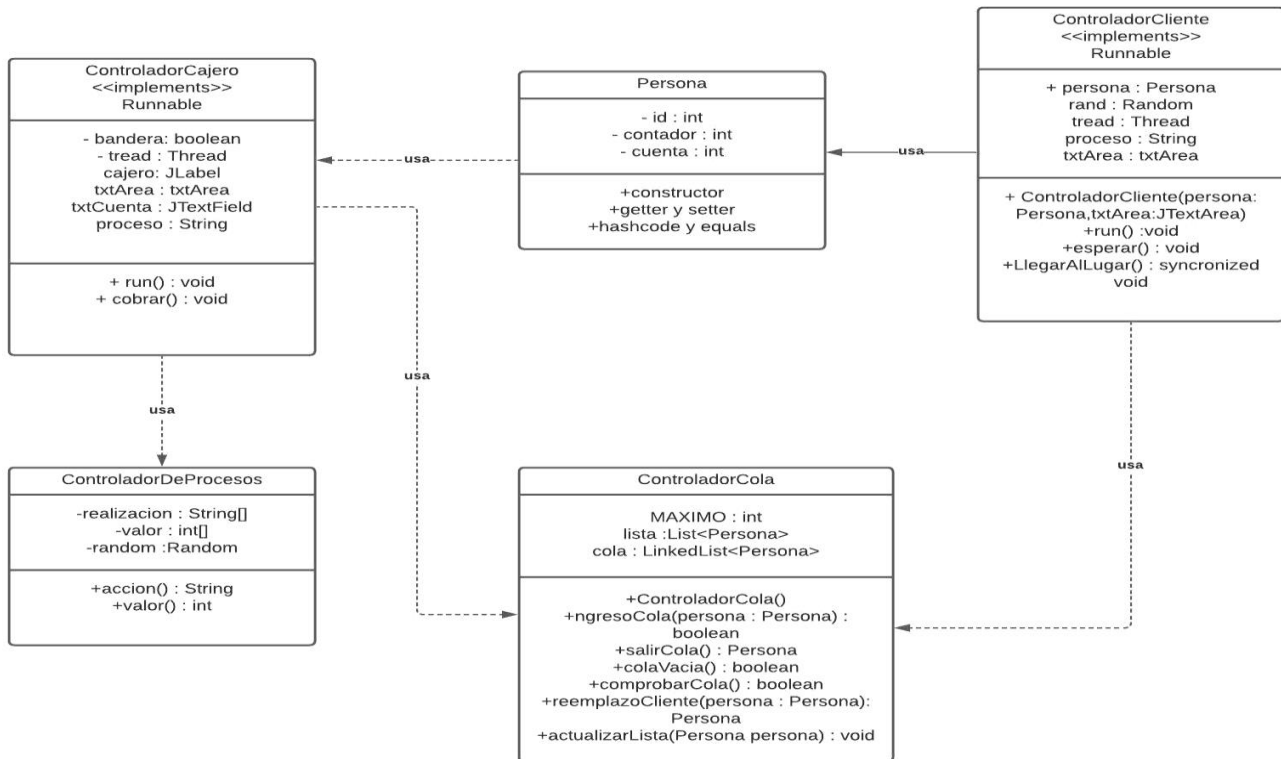


Como vemos nos van dando los resultados y van cambiando los labels dependiendo si están ocupado o no





#### 4. Diagrama UML



### RESULTADO(S) OBTENIDO(S):

- Interpreta de forma correcta los algoritmos de programación y su aplicabilidad.
- Identifica correctamente qué herramientas de programación se pueden aplicar.

### CONCLUSIONES:


Como conclusión se puede decir que trabajar con hilos nos puede ayudar a tener varios procesos de forma simultánea al igual tener un buen entendimiento de estos nos ayudaran a la hora de la creación de programas muy grandes que necesiten de esta implementación

### RECOMENDACIONES:

Recomiendo haber asistido a las clases del ingeniero debido a que con esas bases podemos realizar programas de este tipo, al igual investigar por nuestra propia cuenta como realizar este tipo de proyectos.

**Nombre de estudiante:** Jorge Santiago Cabrera Arias



 <b>UNIVERSIDAD POLITÉCNICA SALESIANA</b> ECUADOR	<b>Computación</b>	<b>Docente: Diego Quisi Peralta</b>
	Programación Aplicada	<b>Período Lectivo:</b> Septiembre 2020 – Febrero 2021

**Firma de estudiante:**



Escaneado con CamScanner