PROCESO DE SELECCIÓN ALMUNDO.COM

ATTENDING
call center

DOCUMENTACIÓN GUÍA PARA EJECUTAR
LA APLICACIÓN JAVA

A CALL SYSTEM APP

IVAN LEMUS - 3174395735

REQUERIMIENTOS
PROCESO DE SELECCIÓN

IVAN LEMUS - 3174395735

ATTENDING
call center

https://github.com/idlm07/almundo.callcenter

REQUERIMIENTOS SOLICITADOS

✓ Usar Repositorio GitHub	https://github.com/idlm07/almundo.callcenter
✓ Debe existir una clase Dispatcher encargada de manejar las llamadas, y debe contener el método dispatchCall para que las asigne a los empleados disponibles.	Se creó un modelo de interfaces, entre ella la interface IDispatcher y su clase concreta Dispatcher.
✓ El método dispatchCall puede invocarse por varios hilos al mismo tiempo.	Este método tiene porciones de código con el modificador synchronized, para controlar la concurrencia.
✓ La clase Dispatcher debe tener la capacidad de poder procesar 10 llamadas al mismo tiempo (de modo concurrente).	No se dejó ningún tope de concurrencia. Se realizaron pruebas con 30 llamadas concurrentemente y resolvió sin problemas,
✓ Cada llamada puede durar un tiempo aleatorio entre 5 y 10 segundos.	Se creó un método random el cual genera numeros aleatorios entre 5 y 10, con el fin de que la llamada tarde ese tiempo, mediante un ThreadSleep.
✓ Debe tener un test unitario donde lleguen 10 llamadas.	JUnit Test Class: TestDiezLlamadas (usando la clase ConcurrentTestRunner)

almundo

EXTRAS/PLUS
PROCESO DE SELECCIÓN

IVAN LEMUS - 3174395735

ATTENDING
call center

https://github.com/idlm07/almundo.callcenter

EXTRAS / PLUS

✓ Dar alguna solución sobre qué pasa con una llamada cuando no hay ningún empleado libre.

El algoritmo se encuentra en la clase Dispatcher en el método dispatcherCall, allí se dejó muy juiciosamente los comentarios en el código.

Básicamente, hay configurados a parte de los empleados unos receptores automáticos que pueden ser contestadoras virtuales para que ya sea que atiendan al cliente o simplemente le informen que ya prontamente va a ser atendido por uno de nuestro empleados.

```
public IReceptorLlamada dispatchCall(ILlamada llamada) throws AttendingException {

    //Inicializar variables
    IReceptorLlamada receptorSeleccionado = null;
    int duracionLlamada = 0;

    //Validar receptores de llamadas
    if((empleados   == null || empleados.isEmpty() ) &&
       (automaticos == null || automaticos.isEmpty()))
        throw new AttendingException("No hay ningún recurso disponible para atender la llamada. Intente más tarde. Gracias");


    //Busca si existe un empleado disponible.
    for(IReceptorLlamada empleado : empleados){
        synchronized (empleado) {
            if(empleado.estaDisponible()){

                //Asignar receptor a la llamada
                duracionLlamada = empleado.asignarLlamada(llamada);
                System.out.println("Llamada "+llamada.obtenerId() + " - Asignada a: "+ empleado);
                receptorSeleccionado = empleado;

                break;
            }
        }
    }

    //Si no existe ningun empleado que pueda atender
    // se busca si existe un recurso automático.
    if(receptorSeleccionado == null){
        for(IReceptorLlamada automatico : automaticos){
            synchronized (automatico) {
                if(automatico.estaDisponible()){
                    duracionLlamada = automatico.asignarLlamada(llamada);
                    System.out.println("Llamada "+llamada.obtenerId() + " - Asignada a: "+ automatico);
                    receptorSeleccionado = automatico;
                    break;
                }
            }
        }
    }
```

almundo

IVAN LEMUS - 3174395735

ATTENDING
call center

https://github.com/idlm07/almundo.callcenter

EXTRAS / PLUS

✓ Dar alguna solución sobre qué pasa con una llamada cuando entran más de 10 llamadas concurrentes.

En la clase CallCenter.java, a parte de la estrategia mencionada anteriormente, se estableció un mecanismo de reintentos en caso tal que no haya ningun empleado, ni tampoco una contestadora virtual. En este mecanismo se hacen 3 reintentos de 10 segundos cada uno.

```
public boolean recibirLlamada(ILlamada llamada) throws Exception {

    //Declaración de variables del método.
    IReceptorLlamada    receptor    = null;
    boolean             intentar    = true;
    int                 numIntentos = 3 ;

    //Atender llamada.
    while (intentar){
        try {

            //Delega al Dispatcher que busque un empleado disponible.
            receptor = dispatcher.dispatchCall(llamada);

            //Seguir intentando, pues es una contestadora virtual.
            if("Contestadora Virtual".equals(receptor.obtenerNombre()))
                intentar = true;

            //No seguir intentando, puesto que ya lo encontró.
            else
                intentar = false;

        }
        //No encontró receptor de la llamada.
        catch (Exception e) {

            System.out.println("Llamada " + llamada.obtenerId() + " - EnEspera");
            //Esperar 10 segundos.
            Thread.sleep(10000);

            //Validar numero de intentos
            numIntentos--;
            if(0 == numIntentos){
                intentar = false;
            }

            System.out.println("Llamada " + llamada.obtenerId() + " - Reintento " + numIntentos);

        }
    }

    return receptor != null ? true : false;
}
```

EXTRAS/PLUS
PROCESO DE SELECCIÓN

IVAN LEMUS - 3174395735

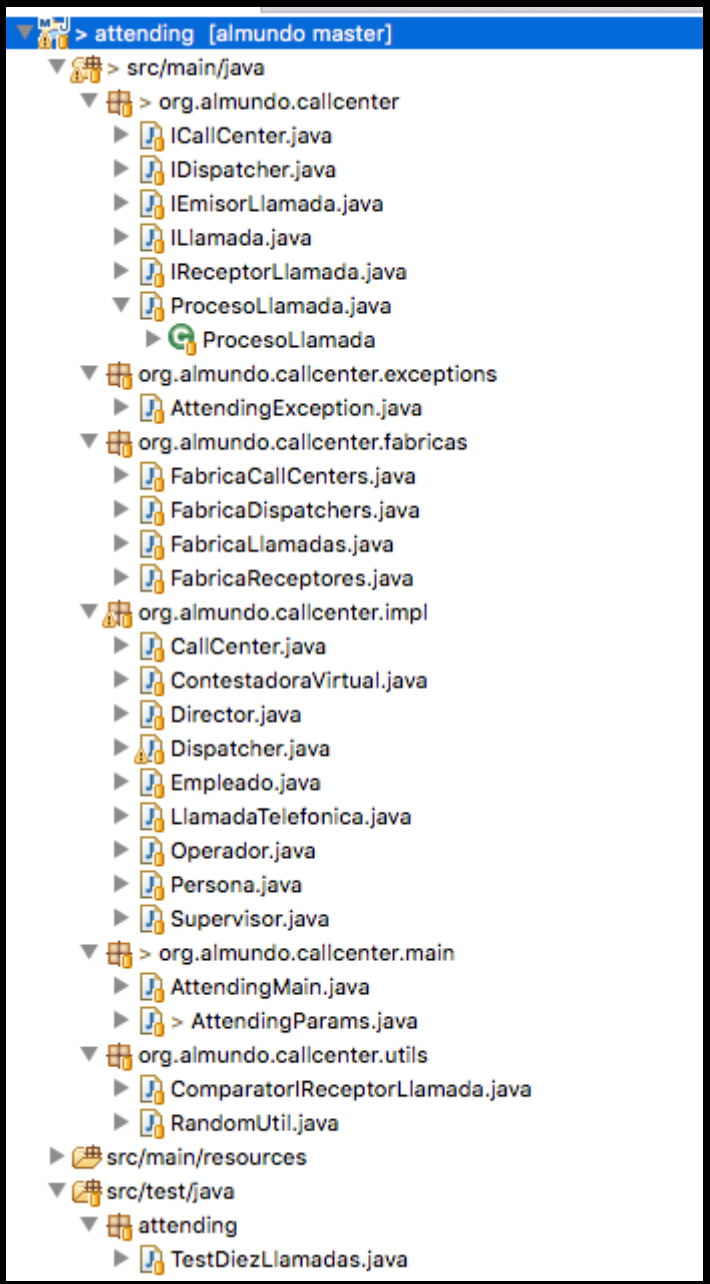
ATTENDING
call center

https://github.com/idlm07/almundo.callcenter

EXTRAS / PLUS

✓ Agregar documentación de código

1.Se realiza este flyer explicativo.
2.Se realizó la debida documentación del código.
3.Se creó logotipo para darle identidad al desarrollo.



ATTENDING
call center

almundo

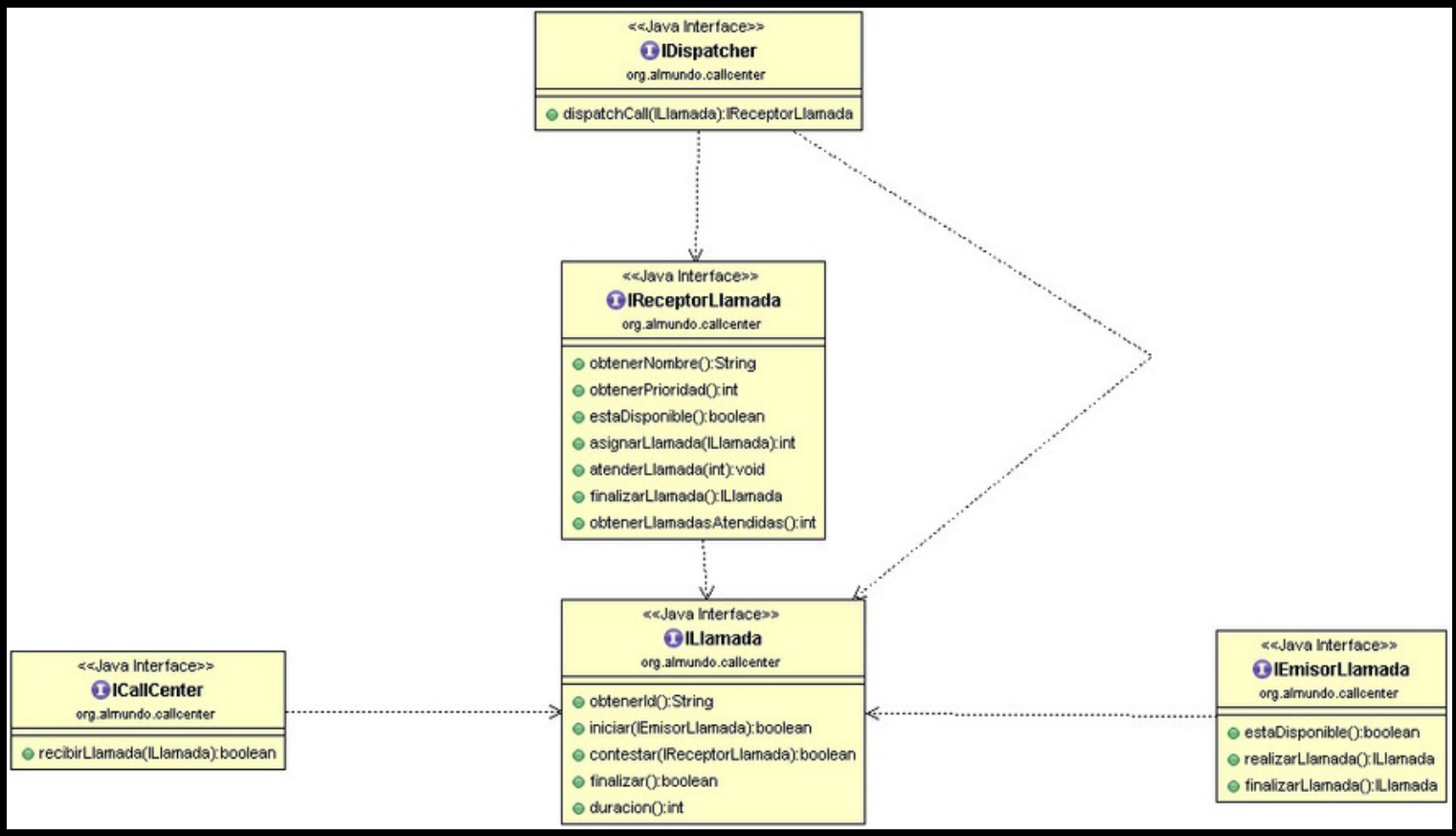
EXTRAS/PLUS
PROCESO DE SELECCIÓN

IVAN LEMUS - 3174395735

ATTENDING
call center

https://github.com/idlm07/almundo.callcenter

DIAGRAMA DE CLASES - INTERFACES



almundo

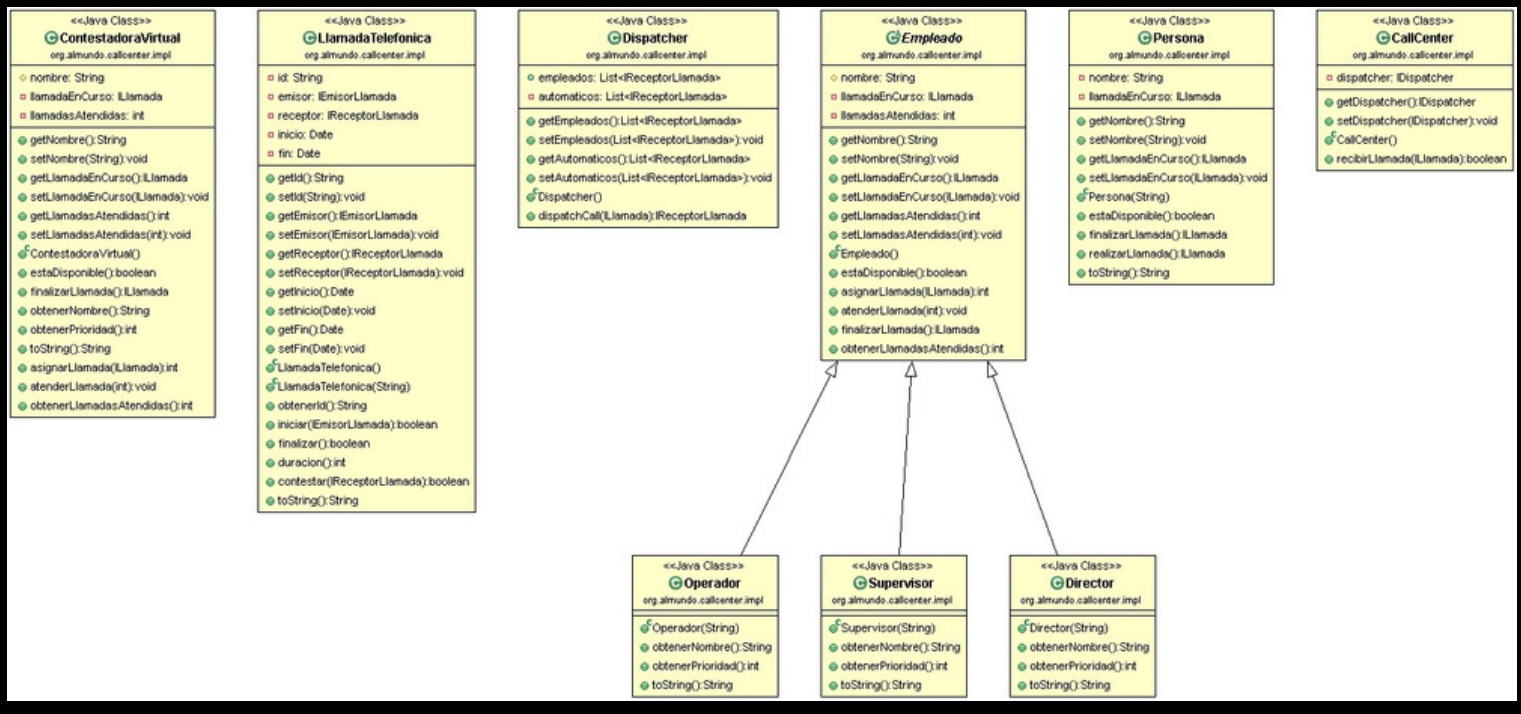
EXTRAS/PLUS
PROCESO DE SELECCIÓN

IVAN LEMUS - 3174395735

ATTENDING
call center

https://github.com/idlm07/almundo.callcenter

DIAGRAMA DE CLASES - IMPLEMENTACIONES



EXTRAS/PLUS
PROCESO DE SELECCIÓN

IVAN LEMUS - 3174395735

Gracias

ATTENDING
call center