



## Sistemas distribuidos Parcial 2

### Descripción del sistema a desarrollar

Las alertas clínicas utilizando sensores y procesamiento de datos se concibieron como una herramienta para detectar de forma temprana el deterioro de la estabilidad hemodinámica del paciente. Ha sido contratado para desarrollar de forma distribuida un sistema que lea 5 indicadores clínicos por medio de varios sensores ubicados en cada habitación de un centro médico, posteriormente, los indicadores deberán ser enviados mediante un cliente a un servidor de alertas el cual analizará cada indicador e identificará si se encuentra en un rango normal. Por otra parte, un conjunto de clientes de notificación se pueden registrar en el servidor de alertas con el propósito de recibir notificaciones automáticas cuando un indicador supere el rango normal.

El cliente debe enviar al servidor de alertas el No de habitación del paciente y la última lectura generada de la frecuencia cardiaca, presión arterial, frecuencia respiratoria, temperatura y saturación de oxígeno. El servidor de alertas cada vez que sufre una excepción registra un mensaje de excepción en un servidor que almacena la información en un log. En la figura 1 se muestra el diagrama de contexto del sistema distribuido.

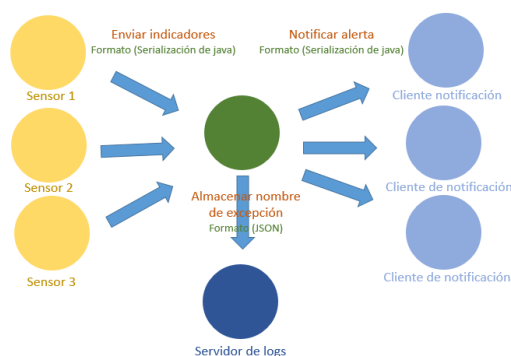


Figura 1. Diagrama de contexto del sistema distribuido

La comunicación entre los sensores y el servidor de alertas es mediante java RMI. La comunicación entre el servidor de alertas y los clientes de notificación es mediante java RMI. La comunicación entre el servidor de alertas y el servidor de logs es mediante sockets y JSON. El servidor de logs va a ser un servidor concurrente construido con sockets e hilos, como el explicado en clase.

### Rangos normales de indicadores clínicos

Para un adulto mayor a 16 años los indicadores clínicos normales deben estar entre los siguientes rangos:

- Frecuencia cardiaca: Latidos por minuto entre 60 y 80.
- Tensión arterial: Presión sistólica entre 110 y 140, y presión diastólica entre 70 y 90.
- Frecuencia respiratoria: ventilaciones por minuto entre 12 y 20.
- Temperatura: entre 36.2 grados centígrados y 37.2 grados centígrados.
- Saturación de oxígeno: entre 95% y 100%.

### Comportamiento al lanzar sensor, servidor y cliente de notificaciones

- Al lanzar el sensor, el programa solicita el no de habitación y posteriormente solicita el ingreso de los indicadores clínicos con el propósito de simular la lectura.
- Los clientes de notificaciones pueden agregarse al sistema distribuido.
- Al lanzar un cliente de notificaciones, internamente enviará al servidor su referencia remota con el propósito de quedar en estado de escucha.



### Comportamiento al leer los indicadores clínicos

- Luego de que un sensor lee los indicadores clínicos, estos son enviados a un servidor, y posteriormente, se pueden volver a ingresar.
- El servidor al leer los indicadores clínicos, determina si están en el rango normal. En caso de que dos o más indicadores no se encuentran en un rango normal se deberá enviar una notificación a varios clientes de notificaciones ubicados en diversas partes de la IPS.
- El cliente al recibir la notificación debe mostrar en una GUI la notificación como se observa en la figura 2.
- Si un indicador presenta una lectura con valor negativo o cero, se debe informar al servidor de logs que el sensor que lee el indicador está presentando una lectura errónea y la fecha de la lectura.



Figura 2. Interface gráfica con la cual interactúa el administrador

### Preguntas a responder

- (v 1.0) Ejecute el sistema, compruebe que las 3 funcionalidades han sido desarrolladas correctamente y explique brevemente cada una de las clases e interfaces que constituyen el sistema distribuido.
- (v 1.0) Explique los 14 pasos que permiten el funcionamiento del sistema distribuido al realizar los Callback. Durante la explicación debe señalar las líneas de código donde se encuentra implementado el paso. Debe basarse en la explicación de los 14 pasos que constituyen el chat grupal.
- (v 0.5) Justifique en que líneas de código se pueden ver reflejadas a) operaciones bloqueantes y no bloqueantes desde la perspectiva de la interoperabilidad del lado del cliente y servidor y b) operaciones idempotentes y no idempotentes.
- (v 0.5) Explique mediante un diagrama los mecanismos que implementaría para controlar la pérdida de peticiones asociados al servicio almacenar mensaje de error, la cual toma como entrada un mensaje de excepción y la fecha en que se generó y produce como resultado su almacenamiento.
- (v 1.0) Realice el diagrama de nodos del sistema implementado y la arquitectura solo de java RMI, considerando que cada cliente de notificaciones antes de comenzar a recibir notificaciones debe registrarse e iniciar sesión en un servidor de usuarios.
- (v 1.0) Explique mediante un gráfico una forma de controlar la operación huérfana determinar si existe alerta médica, la cual toma como entrada los indicadores clínicos y genera como resultado una alerta en el servidor de alertas.



## Mostrar notificación mediante una GUI

Para mostrar la notificación mediante una GUI, debe tener en cuenta el siguiente código donde se muestra la implementación de la interface `notificarClienteNotificacionesCallbackInt`.

```
Public class notificarClienteNotificacionesCallbackImpl extends UnicastRemoteObject implements notificarClienteNotificacionesCallbackInt
{
    Private GUICliente objGUI;

    Public notificarClienteNotificacionesCallbackImpl (GUICliente objGUI) throws RemoteException
    {
        super();
        this.objGUI=objGUI;
    }

    Public void notificar (IndicadoresDTO objIndicadores)
    {
        this.ObjGUI.mostrarNotificacion(objIndicadores);
    }
}
```

## Sustentación

Para la sustentación del segundo parcial debe seguir las siguientes instrucciones:

- Los dos estudiantes deben participar en la respuesta de cada punto
- Debe registrarse en un horario asignado para la sustentación
- Debe sustentar el código fuente de la aplicación, el funcionamiento de la aplicación y la respuesta a las preguntas en un tiempo máximo de 20 minutos.
- En la presentación puede mostrar los diagramas creados, partes de código que realizan funciones por ejemplo registrar el objeto remoto en el NS, y responder a preguntas teóricas.
- Al momento de responder a cada punto del parcial sea ordenado y si es posible coloque un slide que indique cual pregunta se va a responder.
- Al iniciar la presentación agregue una diapositiva inicial que contenga el título del trabajo a desarrollar, el autor, docente, asignatura y programa.