

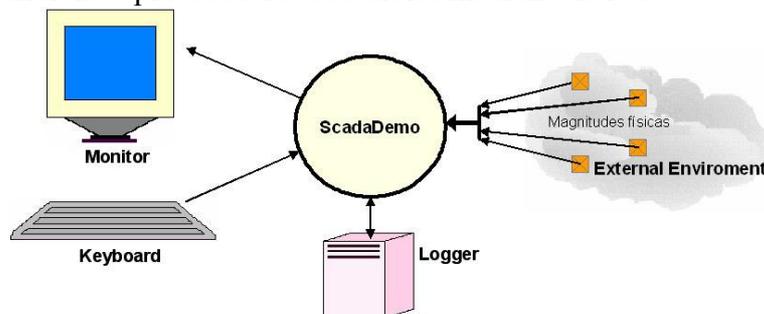
Proyecto: Aplicación ScadaDemo.

Práctica : Diseño de una aplicación concurrente y distribuida.

Autor: J.M.Drake, Laura Barros.

Fecha: 20 Diciembre 2010

La aplicación *ScadaDemo* supervisa un conjunto de hasta 16 magnitudes físicas que son leídas a través de un conjunto de tarjetas de entrada/salida dotadas de conversor A/D e instaladas en diferentes procesadores accesibles mediante la red.



Supervisar una magnitud significa:

- Leer la magnitud a una determinada frecuencia configurable, y almacenar temporalmente los valores leídos en un buffer.
- Calcular el valor promedio de los valores de la magnitud leída, dentro de una ventana temporal establecida para cada magnitud.
- Almacenar en un logger permanente los valores promedios calculados.
- Comprobar si cada valor promedio calculado se encuentra dentro de un rango de alarma definido mediante dos valores de umbral definidos para cada magnitud, y mostrar en la consola un evento de alarma si el resultado de la comprobación es cierta.

Las magnitudes que son supervisadas y las características de la supervisión de cada una de ellas se leen de un fichero de configuración en la fase de arranque de la aplicación. El contenido del fichero de configuración es:

- El nombre de la configuración.
- Una lista con una entrada por cada magnitud que va a ser supervisada. Por cada magnitud se especifica: la línea analógica de la que se lee la magnitud, una etiqueta, el periodo con el que se lee T_{Sampl} , el periodo con el que se registra T_{Reg} , los factores de escala representación/valor físico, los valores que definen el rango de alarma y la prioridad del thread que muestrea la señal.

La interacción entre el operador y la aplicación se realiza a través de un terminal dotado de teclado y monitor. Las opciones de interacción que hay son:

- En el monitor se visualiza un mensaje cada vez que se registra en el logger un valor de alguna magnitud.
- En el monitor se registra un mensaje cada vez que
- El operador pulsa la letra "T" para terminar la aplicación.
- El operador pulsa la letra "C" para recargar la aplicación con un nuevo fichero de configuración

Los requisitos de tiempo real de la aplicación, son:

- Los valores de las magnitudes se adquieren con un jitter máximo de 10 ms.
- El número de muestras de cada magnitud que se promedian antes de ser registrados son siempre el mismo: parte entera de $T_{\text{Reg}}/T_{\text{sampl}}$

- Todos los valores que se registran deben ser visualizados en el monitor, al menos durante un tiempo igual a $T_{Reg}/2$
- Todos los valores que se registran deben ser salvados persistentemente en la base de datos externas.

Ejemplo de caso real: Monitorización de la temperatura en los despachos de CTR.

Se desea medir, registrar y monitorizar la temperatura de 13 despachos de los miembros del Grupo de Computadores y Tiempo Real. La magnitud de cada habitación es leída a través de una tarjeta de entrada/salida que se encuentra ubicada en un equipo. Los equipos se encuentran conectados en red.



En este caso, el fichero de configuración para la aplicación Scada sería:

Fichero de configuración: **ConfigScadaTempCTR.txt**

```
<?xml version=""1.0"" encoding=""UTF-8""?>
<nombreConfiguracion="ConfiguracionBasica">

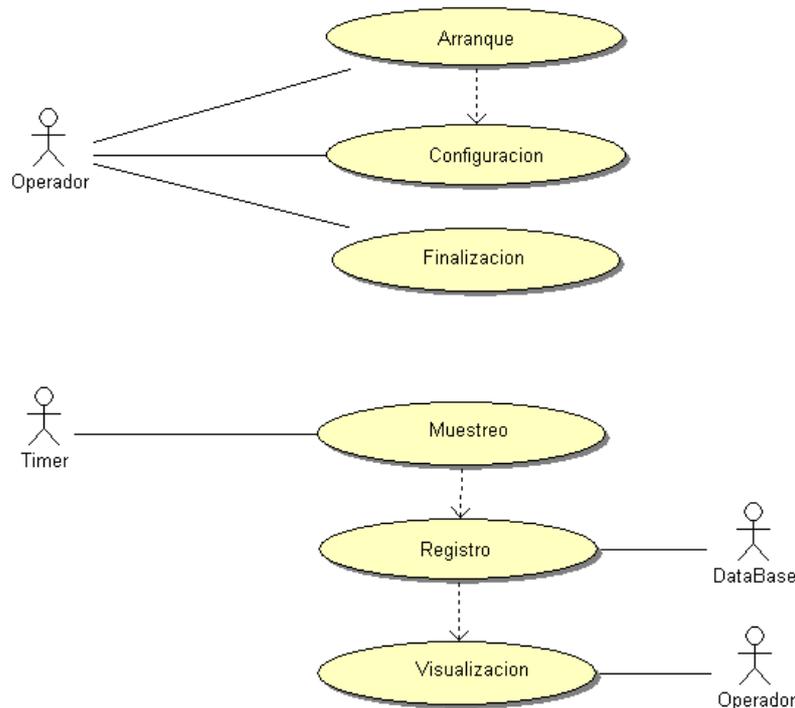
<magnitud label="Temperatura despacho 1 centigrados" prioridad ="5" linea="0"
periododeMuestreo="0.1" periododeRegistro="5.0"
convertFactor ="0.15" convertOffset="30.0 umbralInferior="80"
umbralSuperior="150.0"/>

< magnitud label="Temperatura despacho 2 centigrados" prioridad="5"
line="1" periodoMuestreo="0.1" periodoReg="10.0"
convertFactor="0.10" convertOffset="30.0" umbralInferior="50.0"
umbralSuperior="120.8"/>

< magnitud label="Temperatura despacho 3 centigrados" prioridad="5"
line="2" periodoMuestreo="0.1" periodoReg="10.0"
convertFactor="0.01" convertOffset="1.0" umbralInferior="1.5"
umbralSuperior="5.0"/>

...
</nombreConfiguracion>
```

Casos de uso:



Agente **Operador**: Operador de la aplicación. Interacciona con ella observando los resultados del monitor y pulsando el teclado.

Caso de uso **Arranque**: El operador arranca la aplicación.

Invocación: El operador arranca la aplicación desde línea de comandos.

Actividad:

- Se construye e instancia la aplicación
- Se construye e instancia la GUI que opera como monitor.
- Se invoca el caso de uso Configuración.

Caso de uso **Configuración**: La aplicación se configura de acuerdo con los datos del fichero "ConfigScadaDemo.txt" y se inicia su ejecución.

Invocación: Es invocado por el caso de uso Arranque, o directamente por el operador pulsando la tecla "C" del teclado.

Precondición: El fichero "ConfigScadaDemo.txt" debe estar accesible para ser abierto y leído.

Actividad:

- Si hay muestreo se suspende.
- Si hay accesos abiertos a la base de datos se cierran.
- Se lee y analiza el fichero "ConfigScadaDemo.txt".
- Se inician los procesos de muestreo y registro de magnitudes.

Caso de uso **Finalización**: Se finaliza la ejecución de la aplicación

Invocación: La realiza el Operador pulsando la tecla "T" sobre el teclado.

Actividad: Finaliza el proceso de muestreo y registro y cierra la aplicación:

- Finaliza los procesos de muestreo
- Cierra las conexiones con la bases de datos.
- Finaliza la operación de la consola

Agente **Timer**: Temporizador (reloj) que se programa para que genere un evento cada vez que corresponda la lectura de un nuevo valor de alguna magnitud.

Agente **DataBase**: Base de datos sobre la que se almacenan persistentemente los resultados que se generan en la aplicación. En esta aplicación el registro en la base de datos consiste en imprimir la información en la consola local.

Caso de uso **Muestreo**: Se lee un nuevo valor de una magnitud

Invocación: La invoca el reloj cuando corresponda leer el valor de una magnitud.

Actividad: Se lee un nuevo valor de una magnitud monitorizada.

- Se lee el valor de la magnitud y se acumula.
- Si con la muestra leída se ha finaliza un periodo de registro se invoca el proceso de Registro.

Caso de uso **Registro**: Se registra persistentemente la media de los valores leídos de una magnitud en la la base de datos.

Invocación: Se invoca desde el caso de uso Muestreo cada periodoRegistro.

Actividad: Se registra en la base de datos el valor medio de la magnitud supervisada

- Se invoca Visualización.
- Se lee el tiempo actual.
- Calcula el valor medio
- Se almacena en la base de datos.

Caso de uso **Visualizacion**: Se muestra un valor registrado en el monitor de la consola.

Invocación: Lo invoca el caso de uso Registro

Actividad: Presenta en el monitor el nuevo valor de la magnitud.

Objetivos:

Los objetivos de la práctica son:

- Haciendo uso de UML (no escribir código Java) hay que proponer un conjunto de clases que implementen la funcionalidad de la aplicación.
- La documentación que se genera debe ser la adecuada para que un programador que conozca el lenguaje de programación Java, pueda codificar cualquier clase sin que tenga que tomar decisiones de diseño.

Tareas:

Tarea 1: Realizar el análisis de la aplicación, formulando el diagrama de clases de la aplicación:

- Identificar las clases que constituyen la aplicación, y de cada una de ellas identificar los atributos que constituyen su estado y los métodos que constituyen su interfaz. Establecer entre las clases las relaciones de herencia, agregación, dependencia y visibilidad que existen. Documentar todos los elementos que se incluyan (clases, atributos, métodos y relaciones).
- Describir mediante diagramas de actividad los métodos cuya lógica sea compleja.
- Describir mediante diagramas de estado las clases que tenga un comportamiento complejo.

Tarea 2: Validar el análisis mediante la comprobación de que todos los casos de uso pueden ser implementados utilizando las clases propuestas. Esto se puede llevar a cabo mediante diagramas de secuencias donde se muestren las interacciones entre los objetos.

Por ejemplo:

- **Caso de uso Arranque.**
- **Caso de uso Reconfiguración.**
- **Caso de uso Terminación.**
- **Caso de uso Muestrea, Registra y Visualiza.**

Grupos de prácticas:

Individual o parejas.

Calendario:

Diciembre 2010:

Lunes 20: Planteamiento del trabajo SCADA.

Martes 21: Formación de los grupos. Trabajo en grupo.

Miércoles 22: (Laboratorio) Práctica de ICE (Smart Hunter).

Enero 2011:

Lunes 10: Presentación oral en clase del diseño.

Martes 11: Presentación oral en clase del diseño.

Miércoles 12: Practica ICE II (Cruce Semáforos).

Jueves 13: Práctica ICE II (Cruce Semáforos).

Viernes 14: Repaso de concurrencia/distribuida.

Lunes 17: Presentación de posible solución SCADA (profesor).

Martes 18: Consultas.

Miércoles 19 ó Jueves 20 (pendiente de confirmar): Evaluación distribuido.