

Tarea # 2

Fecha de Entrega: Vie 15 de Mayo, 2015, 11:55 PM.

Descripción General

El objetivo de esta tarea es experimentar con arquitecturas basadas en eventos para obtener una mejor comprensión de las mismas. El dominio de aplicación de esta tarea es un sistema de seguridad y control para un museo. La tarea comprende de dos partes:

- (1) diseño e implementación y
- (2) descripción y análisis.

La primer parte de la tarea (la de diseño e implementación), requiere actividades de diseño y programación. Se provee un framework, desarrollado anteriormente, y cuyo propietario es una compañía de desarrollo de software. La primera parte de la tarea consiste en modificar la implementación de un sistema base para generar un conjunto de sistemas que soporten los requerimientos descritos en las siguientes secciones de este documento. Se proveen varias clases en que implementan los principales componentes de dicho sistema base. Por ejemplo se provee una gestor de eventos (envío y recepción de eventos de procesos o subprocessos) y una API que permite la interacción con dicho gestor. Esta parte de la tarea se debe realizar en equipo.

La segunda parte de la tarea (la de descripción y análisis), consiste en describir y analizar la arquitectura de los sistemas diseñados e implementados en la primera parte. Para ello, se tendrán que responder un conjunto de preguntas sobre las decisiones de diseño que el equipo realizó. Si bien es posible hablar de la segunda parte como un equipo, la evaluación crítica de esta parte debe hacerse de forma individual. Esta descripción y análisis deberá ser entregada como un reporte escrito.

Contexto de Negocio

Imagina que eres parte de una empresa de software desarrolla sistemas de seguridad y control. Dichos sistemas son normalmente utilizados en diversas instalaciones para el control de la calefacción, ventilación, así como para detectar intrusos, movimiento, humo y fuego. Estos sistemas suelen ser altamente distribuidos, pues utilizan sensores remotos y controladores conectados a monitores. Un requerimiento fundamental para la empresa de software es tener una arquitectura altamente extensible donde los sensores, controladores y monitores puedan ser fácilmente añadidos a los sistemas en tiempo de ejecución. También es importante que los sistemas manejen de manera confiable y oportuna el envío y recepción de eventos entre sensores, monitores y controladores. Las funciones realizadas por los sensores, controladores y monitores se describen a continuación:

- a) Sensor: Permite censar el medio ambiente de forma periódica e informar eventos al gestor de eventos. Ejemplos de estos sensores son sensores de temperatura, humo, movimiento, intrusión y humedad.
- b) Controlador: El controlador tiene uno o más dispositivos conectados y es capaz de controlarlos (encenderlos o apagarlos) con base a la información recibida del gestor de eventos. Así, el controlador actúa como una interfaz entre los dispositivos y el gestor de eventos. Ejemplos de los

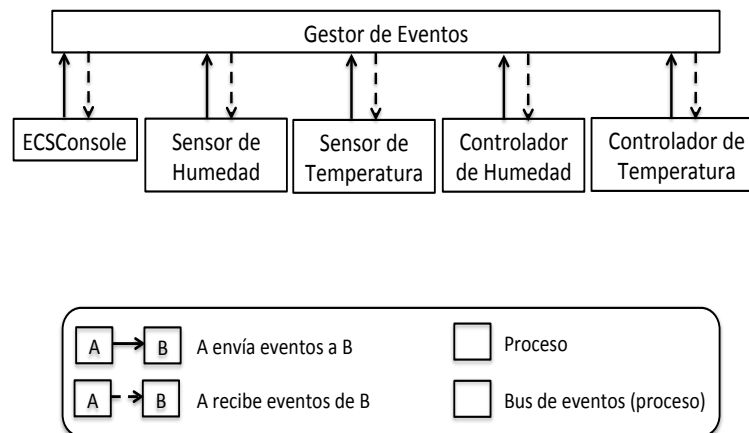
dispositivos que pueden conectarse a un controlador son: calentadores, enfriadores, humidificadores, deshumidificadores, ventiladores, puertas, persianas, aspersores.

- c) **Monitor:** Es una pantalla para uso de los operadores del sistema que les permite observar y responder a las alertas y mensajes generados dentro del sistema. Los monitores también permiten a los operadores configurar aspectos como por ejemplo los niveles de temperatura, humedad o bien activar o desactivar el sistema de seguridad. Los monitores utilizan el gestor de eventos para enviar y recibir datos hacia y desde el resto de los elementos del sistema.

Información del Sistema Base Provisto

a) Descripción

Este es un sistema base para el control de humedad y temperatura en un museo donde se exhiben pinturas al óleo de gran valor. La pinturas deben mantenerse en un ambiente con una temperatura de 70° F - 75° F y con una humedad relativa de 55% al 45%. El sistema base consta de cuatro elementos: un monitor, dos controladores y un sensor de temperatura. La siguiente vista muestra la organización de dicho sistema:



ECSConsole - Este es un proceso implementa una consola de monitoreo y se compone de dos hilos. Uno de los hilos (el principal) sirve como una consola que permite al usuario establecer valores de humedad y temperatura. El otro hilo controla la temperatura y humedad del ambiente mediante el encendido y apagado de la ventilación, calefacción, humidificación y deshumidificación.

Sensor de Temperatura - Este proceso simula el comportamiento de un sensor de temperatura. Normalmente, un sensor debería sólo escribir en el gestor de eventos pero, para simular la acción de un dispositivo real, este proceso lee eventos de temperatura y humedad generados por los controladores de dispositivos y responde a estos eventos mediante la activación o desactivación de

los dispositivos con base en los valores de temperatura. El sensor de temperatura informa eventos de temperatura cada dos segundos.

Sensor de Humedad - Este proceso simula el comportamiento de un sensor de humedad. Normalmente, un sensor debería sólo escribir en el gestor de eventos, pero para simular la acción de un dispositivo real, este lee eventos de humedad generados por el controlador de dispositivos y responde a estos eventos mediante la activación o desactivación de los dispositivos con base en los valores de la humedad. El sensor de humedad informa eventos de humedad cada dos segundos.

Controlador de Humedad - El controlador de humedad es un proceso que interactúa y controla dos dispositivos: un humidificador para añadir humedad al aire, y un deshumidificador para eliminar la humedad del aire. Este elemento sólo se lee información desde el gestor de eventos.

Controlador de Temperatura - El controlador de temperatura es un proceso que interactúa y controla dos dispositivos: un calentador para elevar la temperatura ambiente, y un enfriador para bajar la temperatura ambiente de la habitación. El controlador de temperatura sólo se lee información desde el gestor de eventos.

Es importante resaltar que todos estos procesos se comunican a través de un gestor de eventos. Los eventos están implementados como objetos que incluyen un identificador (ID) y una cadena de texto que se puede utilizar para transmitir un mensaje. Para interpretar los eventos, se utiliza la semántica descrita en las siguientes tablas:

ECSConsole	Responde a eventos con ID 1 y 2
Códigos de acción	
"xxx"	Si ID = 1, entonces la cadena de texto "xxx" contiene el valor, en punto flotante y escala Fahrenheit, de la temperatura del ambiente en grados.
"yyy"	Si ID = 2, entonces la cadena de texto "yyy" contiene el valor, en punto flotante, del porcentaje de humedad relativa.

Sensor de Temperatura	Responde a eventos con ID -5
Códigos de acción	
"H1"	Confirmación de encendido del calentador
"H0"	Confirmación de apagado del calentador
"C1"	Confirmación de encendido del enfriador
"C0"	Confirmación de apagado del enfriador
Notas de aplicación: Considere que este elemento es un simulador. En una aplicación real de este elemento sólo debería escribir la temperatura ambiente en un monitor. Sin embargo, este elemento tiene que simular esta acción. Para ello, el sensor de temperatura escucha los eventos de la confirmación desde el controlador de temperatura y modifica la temperatura mediante la activación o desactivación de el enfriador o el calentador.	

Sensor de Humedad	Responde a eventos con ID -4
Códigos de acción	
"H1"	Confirmación de encendido del humidificador
"H0"	Confirmación de apagado del humidificador
"D1"	Confirmación de encendido del deshumidificador
"D0"	Confirmación de apagado del deshumidificador
Notas de aplicación: Considere que este elemento es un simulador. En una aplicación real de este elemento sólo debería escribir la humedad ambiente en un monitor. Sin embargo, este elemento tiene que simular esta acción. Para ello, el sensor de humedad escucha los eventos de la confirmación desde el controlador de humedad y modifica la humedad mediante la activación o desactivación del humidificador o el deshumidificador.	

Controlador de Humedad	Responde a eventos con ID 4
Códigos de acción	
"H1"	Confirmación de encendido del humidificador
"H0"	Confirmación de apagado del humidificador
"D1"	Confirmación de encendido del deshumidificador
"D0"	Confirmación de apagado del deshumidificador
Notas de aplicación: Este elemento lee los eventos desde el gestor de eventos y sólo responde a los eventos con ID 4. Las cadenas de texto del evento son de la forma XY, donde X indica que el dispositivo: H = humidificador y D = deshumidificador; Y indica 1 = encendido y 0 = apagado.	

Controlador de Temperatura	Responde a eventos ID 5
Códigos de acción	
"H1"	Confirmación de encendido del calentador
"H0"	Confirmación de apagado del calentador
"C1"	Confirmación de encendido del enfriador
"C0"	Confirmación de apagado del enfriador
Notas de aplicación: Este elemento lee los eventos pendientes desde el gestor de eventos y sólo responde a los eventos con ID 5. Las cadenas de texto del evento son de la forma XY, donde X indica que el dispositivo: H = calentador y C = enfriador; Y indica 1 = encendido y 0 = apagado.	

b) Instalación

Los detalles de diseño, notas de aplicación e instrucciones de instalación del gestor de sucesos y el sistema de museos de control ambiental se incluyen en el documento

EMA_DevelopmentFramework.pdf. Esta tarea no requiere modificar el gestor de eventos, por esta razón no se incluye su código fuente. Sin embargo, se provee el código de un sistema base fuente con las clases que permiten crear aplicaciones que utilizan el gestor de eventos. Las instrucciones para instalar y ejecutar el sistema base también se encuentran en el documento EMA_DevelopmentFramework.doc.

Parte I: Diseño e Implementación

Utilizando el código provisto, cada equipo deberá crear tres nuevos sistemas. Cada sistema tiene uno o más requisitos. Cada sistema deberá respetar un modelo de arquitectura orientada a eventos en la mayor medida posible.

Sistema A

Añadir al sistema base las funcionalidades de detección de intrusos y monitoreo. El sistema debería incluir tres alarmas diferentes: ventana rota, puerta rota y detección de movimiento. Asuma que estas alarmas están en un solo controlador. Cree un monitor de seguridad independiente que permita a un operador activar y desactivar el sistema (se puede utilizar el ECSCConsole como modelo). Cuando el sistema está desactivado, los eventos de seguridad no se reportan. Cuando el sistema está activado, intrusiones de seguridad deben ser reportadas a los operadores a través del monitor de seguridad. El tipo de alarma (ventana rota, puerta rota y detección de movimiento) debe ser reportado. Para simplificar las actividades de codificación y prueba simule las alertas de seguridad. Esto es, provea una interfaz de texto simple (use los sensores de temperatura y humedad como modelo) que permita crear un evento de seguridad.

Sistema B

Añada al Sistema A las funcionalidades de detección de incendios y monitoreo. El sistema debe incluir un controlador de alarma de incendio y un controlador de rociadores. Cuando el controlador indique que fuego ha sido detectado, el sistema debe mostrar la alarma en la consola de seguridad. El sistema deberá permitir a un operador el confirmar o cancelar el encendido de los rociadores. Si no hay entrada del usuario en 15 segundos, el rociador se encenderá automáticamente. Una vez encendido, el monitor de detección de incendios debe permitir que el usuario apague el rociador.

Sistema C

Añada al Sistema B la funcionalidad de una consola de servicios de mantenimiento con una función que muestra todos los equipos instalados actualmente en el museo. La consola de mantenimiento debe enumerar los elementos que están conectados al sistema - cada dispositivo debe tener un nombre de dispositivo y una breve descripción. La consola de mantenimiento debe detectar cuando un dispositivo no responde y notificar al usuario a través de la consola de servicios de mantenimiento.

Notas importantes:

- Antes de realizar cualquier actividad de diseño o implementación, asegúrese de comprender la semántica de las arquitecturas basadas en eventos mediante las lecturas de los documentos: ArchBasadasEnEventos_1.pdf, ArchBasadasEnEventos_2.pdf y ArchBasadasEnEventos_3.pdf
- Al entregar sus sistemas, asegúrese de que estén claramente separados. Ponga el código de cada sistema en una carpeta diferente.
- Incluya una descripción muy clara de cómo instalar y ejecutar sus sistemas.
- No asuma que tendremos un ambiente de desarrollo particular instalado.

- No asuma que recompilaremos su código.
- Enviar los tres sistemas vía Moodle (Entregables de Tareas, Proyectos, Exámenes/Archivo con los Sistemas de la Tarea 2) en un archivo .zip con el siguiente formato de nomenclatura: NUM-EQUIPO_NUM-TAREA_FECHA
- Envíe el reporte individual vía Moodle (Entregables de Tareas, Proyectos, Exámenes/Archivo del Reporte Individual de la Tarea 2) en un archivo con el siguiente formato de nomenclatura: NOM-ALUMNO_NUM-TAREA_FECHA.

Parte II: Descripción y Análisis

A pesar de que los sistemas han sido desarrollados en equipo, cada estudiante puede estar de acuerdo o no con las decisiones de diseño e implementación tomadas por el equipo. Considerando lo anterior, contesta las siguientes preguntas. Nota que cada pregunta tiene varias partes. Asegúrate de responder completamente a cada pregunta y sus partes:

1. Para el sistema en general:

- a) ¿Cuáles son los principales “drivers” arquitecturales y cuál es su prioridad?
- b) Desde una perspectiva dinámica, ¿el gestor de eventos es un conector o un componente? Justifica ampliamente tu respuesta.
- c) ¿En qué medida las decisiones de diseño tomadas por el arquitecto en la implementación del sistema base soportan los objetivos de negocio de la empresa de software?

2. Para los sistemas A, B, y C:

- a) Describa la arquitectura del sistema. Existe la libertad de usar cualquier notación, pero por supuesto se espera una buena calidad en la documentación arquitectónica. Asegúrese de incluir las vistas apropiadas.
- b) Discuta qué tan bien las opciones de diseño tomadas por su equipo soportan apoyo los objetivos de negocio de cada uno de los sistemas solicitados en términos de los atributos de calidad correspondientes. Específicamente, discuta
 - ¿cuáles fueron las decisiones de diseño clave?,
 - ¿qué motivó la toma de dichas decisiones?
 - ¿cuáles son las ventajas y desventajas de las mismas?
- c) Discuta cómo las decisiones de diseño tomadas para los sistemas A, B, C impactan en la activación y desactivación del sistema?

3. Para los sistemas A, B, y C:

- a) Describa ¿cuáles decisiones de diseño fueron relativamente fáciles de implementar y cuáles mas difíciles? y ¿qué impacto tuvo el uso de una arquitectura basada en eventos al realizar estas modificaciones?
- b) Describa cómo se promovió o inhibió la reutilización a consecuencia de las decisiones de diseño implementadas y el uso de una arquitectura basada en eventos.

4. Extensión 1 (no requiere ser implementada, sólo pensada y discutida con suficiente nivel de detalle):

Suponga que hay dos instancias del gestor de eventos que se ejecutan en dos diferentes máquinas. Describa cómo se deberían difundir los eventos entre todos los componentes del sistema. Suponga que no se puede modificar el gestor de eventos y sólo un gestor de eventos se puede ejecutar en una máquina al mismo tiempo.

5. Extensión 2 (no requiere ser implementada solo pensada y discutida con suficiente nivel de detalle):

Suponga que debe modificar el algunos de los sistemas sistema para que en caso de un evento de fuego en el museo, cubiertas de vidrio para las pinturas deben bajar y se deben activar los rociadores. Sin embargo, es muy importante asegurarse de que los rociadores no se activarán antes de que las cubiertas están completamente bajadas.

Criterios y Forma de Evaluación

Los criterios a utilizarse son:

- La calidad y el contenido del reporte individual. Se espera sea claro, conciso y completo.
- La consistencia entre las representaciones de diseño y la implementación.
- El grado en que sus soluciones respetan una arquitectura basada en eventos
- La claridad con que se explican las desviaciones a la arquitectura basada en eventos (así como sus efectos).
- Profesionalismo. Por ejemplo la ausencia de errores de la gramática y ortografía en el reporte, la claridad de la presentación del equipo, la entrega en tiempo y forma de los artefactos solicitados, calidad del código fuente.
- El correcto funcionamiento de los sistemas implementados.
- El uso de buenas prácticas de programación. Por ejemplo incluir comentarios que indiquen cómo ha cambiado el sistema base.

La forma de evaluación es como sigue:

Parte I:

- Implementación del sistema A: 10 puntos.
- Implementación del sistema B: 15 puntos.
- Implementación del sistema C: 25 puntos.

Parte II:

- Pregunta 1: 10 puntos.
- Pregunta 2: 10 puntos.
- Pregunta 3: 10 puntos.
- Pregunta 4: 10 puntos.
- Pregunta 5: 10 puntos.