

Examen de la Convocatoria Ordinaria

Proyecto de Análisis y Diseño de Software (2011/2012)

Los dueños de la empresa que te encargaron la aplicación de gestión de proyectos han quedado muy satisfechos con tu trabajo, y han decidido encargarte una aplicación para la gestión de errores en los sistemas informáticos que desarrollan para sus clientes. Dicha aplicación será independiente de la que ya les has construido.

El objetivo de la nueva aplicación es permitir a los clientes informar de errores detectados en los sistemas que desarrolla la empresa. De esta manera, cuando un cliente quiere informar de un error, primero ha de identificarse en la aplicación (usuario y clave). Una vez identificado en la aplicación, debe seleccionar el sistema del que quiere informar de un error (p.ej. contabilidad), así como del componente que falla (p.ej. informes de gastos). En caso de no conocer el componente, el cliente puede asociar el error al sistema directamente. Los sistemas y sus componentes vienen descritos por un nombre, una descripción y una serie de palabras clave.

Un error se describe mediante un texto, una serie de palabras clave y una importancia (Crítica, Importante, Normal, Menor o Mejora). Además, el cliente ha de especificar la versión del sistema que falla, la plataforma hardware y el sistema operativo donde se produce el error. La aplicación guardará la fecha en que se informó del error y le asignará un identificador único. Opcionalmente, se podrán añadir adjuntos, que serán bien imágenes con un pantallazo del error, o bien ficheros de código fuente (para usuarios avanzados).

Otros usuarios de la aplicación son los desarrolladores y los jefes de proyecto, que también han de identificarse mediante usuario y clave. Estos últimos tienen asignado el mantenimiento de una serie de sistemas de los que son responsables. Cuando un cliente informa de un error en un sistema, el jefe de proyecto correspondiente puede asignar su resolución a un desarrollador. Si hay desarrolladores sin errores asignados en ese momento, la aplicación comprobará que el error se le asigna obligatoriamente a uno de ellos. Los desarrolladores se encargan de solucionarlo, y podrán actualizar el estado del error según vayan trabajando en él (ver más abajo), así como añadir comentarios al error. La aplicación debe guardar la fecha de la última actualización del error. Los desarrolladores también pueden establecer dependencias que indiquen que un error puede solucionarse sólo cuando se hayan solucionado otros. El jefe puede obtener dos tipos de informe: de actividad de los desarrolladores (errores que tienen asignados), y de errores (errores que hay en los sistemas de los que son responsables, y su estado).

Los errores tienen un estado. Cuando un cliente da de alta un error, éste tiene el estado “*sin confirmar*”. Antes de asignarlo a un desarrollador, el jefe de proyecto revisa los errores sin confirmar. Si a primera vista se consideran errores, pasan al estado “*nuevo*”; en caso contrario, pasan al estado “*cerrado*” (ver más adelante). El jefe de proyecto puede asignar a un desarrollador los errores en estado “*nuevo*”, que pasan al estado “*asignado*”. El desarrollador puede arreglar el error (que pasa a “*resuelto*”), o bien identificar que se trata de un error que ya está siendo procesado (“*duplicado*”), o decidir que no es un error porque no es posible reproducirlo (“*inválido*”). El jefe de proyecto verifica si los errores en estado “*resuelto*” ya no se producen al ejecutar el sistema que antes fallaba. Si la verificación es satisfactoria, pasan al estado “*verificado*”; en caso contrario, pasan al estado “*reabierto*”. Al igual que los errores en estado “*nuevo*”, los errores reabiertos han de ser asignados de nuevo por el jefe de proyecto para ser arreglados de nuevo. Por contabilidad, queremos saber cuántas veces se reabre un error, y el número total de errores que ha arreglado cada desarrollador. Los errores “*verificados*” podrán cerrarse por el jefe de proyecto. Una vez cerrados, se notifica automáticamente por e-mail al cliente que informó del error y al desarrollador que lo arregló (si es que no se cerró directamente).

La aplicación deberá permitir la búsqueda de errores existentes mediante palabras clave. Para esta funcionalidad no es necesario haberse autenticado.

En cualquier momento, cualquier cliente o desarrollador puede añadir un comentario a un error, que puede ser respuesta a otro comentario ya existente. Los comentarios tienen un texto, una fecha y pueden llevar como adjunto imágenes o ficheros de código fuente.

IMPORTANTE: La aplicación descrita está muy simplificada, y no responde a un escenario totalmente realista. Por ejemplo, la gestión de altas y bajas de usuarios es algo que debería incluirse en una aplicación real, pero que queda fuera de lo descrito en el enunciado. Limita tus respuestas a lo que dice estrictamente el enunciado. Es posible que tengas que hacer alguna suposición sobre algún detalle que no incluye el enunciado. Puedes hacerlo, pero siempre de manera bien fundamentada, razonable y sin contradecir el enunciado. Documenta tales suposiciones en tu respuesta.

CONTESTA CADA APARTADO EN HOJAS SEPARADAS

Apartado 1. (2.5 puntos)

- a) Dibuja un diagrama de casos de uso que represente la aplicación descrita.
- b) Describe detalladamente el caso de uso en el que un jefe de proyecto asigna un error a un desarrollador.

Apartado 2. (3.5 puntos)

Dibuja un diagrama de clases que represente la aplicación. No olvides incluir los atributos y métodos necesarios para implementar la funcionalidad que se describe. No es necesario que incluyas constructores, getters o setters.

Apartado 3. (1.5 puntos)

Dibuja el diagrama de transición de estados asociado a la clase error, que describa su comportamiento.

Apartado 4. (1.5 puntos)

Dibuja el diagrama de secuencia que refleje el proceso de asignar un error a un desarrollador.

Apartado 5. (1 punto)

Utilizando la librería *java.swing*, realiza una aplicación que muestre una ventana con un cuadro de texto y un botón. Al pulsar el botón, se debe imprimir por consola el contenido del campo de texto (usando *System.out.println*).