

Examen de la Convocatoria Ordinaria de Proyecto de Análisis y Diseño de Software (2020/2021)

NO SE PERMITEN PREGUNTAS DURANTE EL EXAMEN

Limita tus respuestas a lo que dice estrictamente el enunciado. Si necesitas hacer alguna suposición sobre algún detalle que no incluye el enunciado, puedes hacerlo pero siempre de manera fundamentada, razonable y sin contradecir el enunciado. Documenta tales suposiciones en tu respuesta.

Se quiere construir una aplicación para una empresa de reparaciones y reformas del hogar. En esta aplicación, los clientes podrán dar de alta averías, que la aplicación asignará a sus trabajadores para su presupuestado y posterior reparación. Además, esta aplicación deberá interaccionar con un sistema de información externo de una compañía de seguros.

La empresa considera tres tipos de reparaciones para averías: fontanería, electricidad y albañilería. Cualquier reparación se define mediante una descripción, una fecha de parte, una fecha de fin, si son urgentes, y opcionalmente un número de póliza de seguro. Las reparaciones de fontanería han de indicar si requieren un corte general del agua en la calle, y si necesitan el arreglo de la caldera. En este último caso, se debe indicar la marca de la caldera. Las reparaciones de electricidad han de indicar si la avería ha dañado aparatos eléctricos, y una descripción de los daños. Finalmente, las de albañilería deben especificar si es trabajo de interior o exterior, y si requieren trabajo en altura (ej., en un andamio).

La aplicación considerará dos tipos de usuarios: clientes y trabajadores. Ambos vienen descritos por su nombre, teléfono, dirección de correo electrónico, nick y clave. Un cliente puede dar de alta domicilios (con calle, número, puerta, código postal y ciudad) donde podrán solicitarse reparaciones de averías. Por su parte, un trabajador debe indicar su especialidad (albañilería, electricidad, fontanería), y códigos postales en los que trabaja.

Los clientes pueden dar de alta averías indicando el tipo de avería (fontanería, electricidad o albañilería), una descripción de la avería, y en cuál de sus domicilios se ha producido. Al dar de alta una avería, la aplicación asignará un trabajador de la especialidad adecuada para su reparación. En concreto, el algoritmo asignará al trabajador de la especialidad con menor carga de trabajo que trabaje en el código postal donde se produjo la avería. Además, se ha de tener en cuenta que no puede asignarse un trabajador que ya esté asignado a dos reparaciones en fase de ejecución (esto es, con presupuesto aceptado, que hayan comenzado, y que no hayan finalizado). Si no hay trabajadores que cumplan estas condiciones, la reparación de la avería queda en espera, y la aplicación intentará asignarla cuando finalice otra reparación.

Una vez asignada la reparación de una avería a un trabajador, éste contacta telefónicamente con el cliente para concertar una cita con objeto de establecer un presupuesto. La gestión de la cita se realiza de manera externa a la aplicación, y en algunos casos (ej., la sustitución de una lámpara), el presupuesto puede establecerse sin cita presencial. El trabajador se encargará de introducir el presupuesto en la aplicación. Un presupuesto está desglosado en actividades. Cada actividad tiene un nombre y uno o más conceptos que pueden ser de tres tipos: mano de obra (con horas estimadas y precio por hora), material y desplazamiento. Cada concepto debe especificar un nombre, descripción, precio base (sin IVA), e IVA aplicable. Una actividad también puede incluir otras actividades (ej., una reforma del cuarto de baño que incluye cambiar los azulejos de la pared y cambiar la bañera por un plato de ducha, y esta última actividad incluye a su vez cambiar parte del suelo).

El cliente debe acceder a la aplicación para aceptar el presupuesto. Una vez aceptado, la aplicación accede a un sistema externo de la compañía de seguros para comprobar si el cliente tiene una póliza de seguros que cubra la reparación. En caso negativo, la aplicación cobra al cliente el 25% del presupuesto por medio de tarjeta. A continuación, el trabajador asignado procede a la reparación. Cada sesión de trabajo en una reparación ha de subirse a la aplicación indicando la hora de comienzo y fin, así como una descripción del trabajo. Una vez finalizada la reparación, el trabajador ha de indicar en la aplicación el precio final (ya que puede haber variaciones respecto al presupuesto). En ese momento, el cliente debe pagar el resto del precio, a no ser que la reparación sea mediante seguro. En este último caso, la aplicación se encarga de enviar una notificación al sistema de la compañía de seguros.

La aplicación debe incluir funcionalidad para obtener el importe total de las reparaciones realizadas por cada trabajador; la variación del importe final respecto a lo presupuestado por cada reparación; y la facturación imputada a cada cliente, distinguiendo los pagos realizados por el cliente de los cubiertos por la compañía de seguros.

En los apartados 1, 2, 3 y 4 debes utilizar la notación UML para que tus respuestas estén formalizadas y normalizadas como deben estar todos los modelos de análisis y diseño generados en un proyecto software. Expresar ideas correctas en un lenguaje distinto de UML no es aceptable.

Apartado 1. (3 puntos)

Diagrama de casos de uso completo de la aplicación descrita.

Apartado 2. (4 puntos)

Diagrama de clases completo de la aplicación, incluyendo todas las relaciones, atributos y métodos necesarios para implementar la funcionalidad descrita en el enunciado. No incluyas constructores, ni *getters* o *setters* simples (que simplemente devuelvan o cambien un atributo sin más) pero detalla bien el resto de métodos, indicando su nombre, parámetros que recibe y su tipo, y su tipo de retorno.

Apartado 3. (1.5 puntos)

Diagrama de transición de estados que describa el comportamiento de la clase Reparación.

Apartado 4. (1.5 puntos)

Diagrama de secuencia que refleje el método para asignar un trabajador a una reparación.

CONTESTA CADA APARTADO EN HOJAS SEPARADAS