

CÁTEDRA



Ingeniería de Software II

2019

UNIDAD 3: Mejorar las relaciones de análisis



Realizar el diagrama de clases para el siguiente caso:

1 En este consultorio odontológico trabajan varios profesionales que brindan sus servicios. Cuando un paciente necesita atención, debe solicitar un turno previamente. No se atienden pacientes que no tienen turno.

UNIDAD 3: Mejorar las relaciones de análisis



2

El odontólogo para el que el paciente solicita el turno es el que lo va a atender.

Cada odontólogo tiene una agenda con los días y horarios en los que puede atender, que se crea mensualmente en función de la disponibilidad que el odontólogo informa, con turnos de 30 minutos de duración.

UNIDAD 3: Mejorar las relaciones de análisis



3

Esta agenda genérica representa los días y horarios de atención que tiene disponible ese odontólogo en términos generales y la duración de su consulta.

Esta información se tomará como base para crear la agenda cada mes, considerando para cada mes los días y/u horarios que en ese mes no podrá atender.

UNIDAD 3: Mejorar las relaciones de análisis



4

Cuando el paciente llama por teléfono, se le pregunta el motivo de la consulta y en función de eso se le asigna uno o más turnos.

Por ejemplo, si lo que debe hacerse es un tratamiento de conducto, se le asignan dos turnos de media hora, consecutivos.

UNIDAD 3: Mejorar las relaciones de análisis



Notas:

- Los odontólogos se seleccionan de una lista de nombres de odontólogos, precargada.
- La definición de horarios para el odontólogo elegido muestra para cada día de la semana que atiende los horarios de inicio y de fin y los intervalos de no atención si los hubiera (por ejemplo hora de almuerzo).
- Se deben generar todos los turnos posibles en estado “Disponibles”, para el rango horario definido.
- Se debe permitir modificar el estado de algunos turnos, para bloquearlos.
- Asociar colores a los estados: Disponible (verde); Cancelado (Rojo); Bloqueado (Gris); Reservado (Azul).
- El mes y año para el que se genera la agenda se seleccionan de listas predefinidas.

UNIDAD 3: Mejorar las relaciones de análisis



Diálogo entre el cliente y el equipo de desarrollo

UNIDAD 3: Mejorar las relaciones de análisis



Equipo: Bueno si te parece, comencemos con la agenda de los odontólogos, ¿Qué necesitan para esta funcionalidad?

CLIENTE: Bueno lo que ocurre es que en este consultorio trabajan varios odontólogos y cada uno tiene sus propios horarios de atención y la secretaria se confunde y a veces a dado turnos a gente en horarios que el odontólogo no atendía. Entonces necesitamos que el sistema lleve eso en forma organizada y facilite el trabajo de la secretaria.

Equipo: Ok, ¿las agendas en qué momento se definen?

CLIENTE: La idea es que la se definan a principio del mes, para el mes siguiente y basado en lo que cada odontólogo defina como sus días y horarios de atención. Y sería una agenda para cada odontólogo, para cada mes del año.

Equipo: ¿La duración de cada turno es fija?

CLIENTE: Si la duración de cada turno es de 30 minutos. En todo caso, al momento de asignar turnos si el odontólogo sabe que necesita más tiempo indica a quién registrar el turno que reserve más de un turno continuado, hasta cubrir el tiempo que necesita.

UNIDAD 3: Mejorar las relaciones de análisis



Equipo: Bien, entonces, ¿la idea sería que el sistema busque la definición de horarios para el odontólogo y en función de esa definición genere tanto turnos de 30 minutos como sea posible?

CLIENTE: Exacto! Y los turnos deberían quedar como disponibles (pintados en verde) y permitir que la secretaria bloquee los turnos que ella decida. El turno bloqueado debería estar en gris.

Equipo: ¿Qué otros estados puede tener un turno?

CLIENTE: Además de disponible y bloqueado, el turno puede estar cancelado (en rojo), cuando un paciente decide no asistir a la consulta y reservado (azul), cuando un paciente tomar el turno.

Equipo: ¿Es decir que la Secretaría sólo puede bloquearlos?

CLIENTE: Exactamente.

Equipo: Respecto de la definición de los horarios de cada odontólogo, ¿pueden tener más de una?

CLIENTE: No, vigente en un momento, cada odontólogo puede tener una única definición de horarios.

UNIDAD 3: Mejorar las relaciones de análisis



Y ahora nuestro cliente quiere....

1. Poder consultar historia clínica del paciente.
2. Enviar un aviso al paciente del turno por correo.
3. Imprimir la receta para cada paciente.

UNIDAD 3: Mejorar las relaciones de análisis

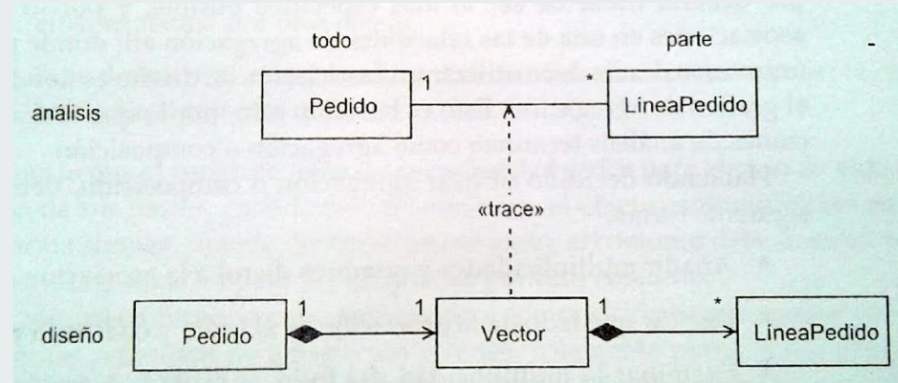
Realice los cambios necesarios en el diagrama de actividad y el diagrama de clases para satisfacer los requerimientos incorporados recientemente.

Pasamos de las clases de análisis a las clases de diseño:

- Verificar que se cumplen las características deseables (totalidad y suficiencia, sencillez, alta cohesión y bajo acoplamiento)
- Utilizar herencia para la reutilización de código.
- Añadir multiplicidad, dirección y nombre a las asociaciones.
- Convertir las asociaciones en agregación o composición cuando sea posible:
 - Decidir qué lado de la asociación es todo y cuál es la parte.
 - Examinar la multiplicidad del todo, si es 0..1, o exactamente 1, puede ser capaz de utilizar composición, o puede optar por añadir un atributo.
 - Asociaciones muchos 1: utilice agregación. Como existen muchos en el todo, no se utiliza composición
 - **Asociación de uno a muchos. Se utilizan colecciones (será dado en clases siguientes)**

UNIDAD 3: Mejorar las relaciones de análisis

Se presenta el caso de una asociación de 1 a muchos en la etapa de análisis.



Se implementa con una clase Vector en diseño (librería estándar de Java.util)

La relación del Pedido (todo) con el Vector es composición ya que es una implementación del todo y no tiene vida fuera de él.

La relación entre Vector y las partes (LineaPedido) puede ser una agregación o una composición

UNIDAD 3: Mejorar las relaciones de análisis



Asociaciones 1 a *.....colecciones!!

En las asociaciones 1 a muchos existe una colección de objetos en el lado parte de la relación. Para implementar esa relación hay que usar las colecciones.

Una clase de colección es una clase cuyas instancias se especializan en gestionar colecciones de otros objetos.

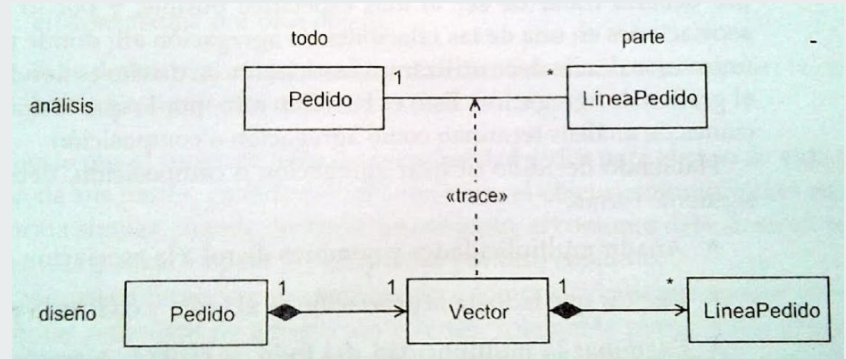
Para su diseño tener en cuenta que todas las clases de colección tienen operaciones para:

- Añadir objetos
- Eliminar objetos
- Recuperar una referencia de un objeto en la colección
- Recorrer la colección

UNIDAD 3: Mejorar las relaciones de análisis

Si identifica que debe modelar con colecciones, hay 4 estrategias de diseño:

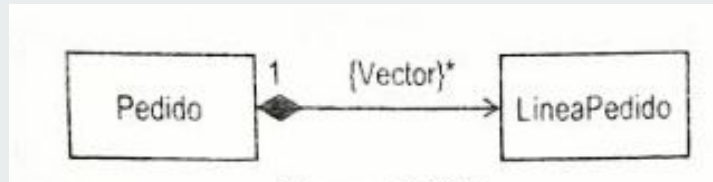
- 1 **Modelar la clase de la colección explícitamente**
- 2 Diga a la herramienta de modelado que genera código que una clase específica se asigne a la colección. Esto se realiza al añadir valores etiquetados a la asociación.
- 3 Añada una propiedad a la relación pero no especifique clase.
- 4 Deje la conversión de las relaciones uno a muchos en clases de colección a los programadores.



UNIDAD 3: Mejorar las relaciones de análisis

Si identifica que debe modelar con colecciones, hay 4 estrategias de diseño:

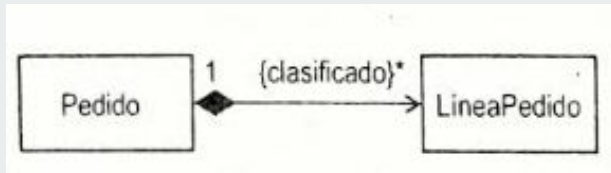
- 1 Modelar la clase de la colección explícitamente
- 2 **Diga a la herramienta de modelado que genera código que una clase específica se asigne a la colección. Esto se realiza al añadir valores etiquetados a la asociación.**
- 3 Añada una propiedad a la relación pero no especifique clase.
- 4 Deje la conversión de las relaciones uno a muchos en clases de colección a los programadores.



UNIDAD 3: Mejorar las relaciones de análisis

Si identifica que debe modelar con colecciones, hay 4 estrategias de diseño:

- 1 Modelar la clase de la colección explícitamente
- 2 Diga a la herramienta de modelado que genera código que una clase específica se asigne a la colección. Esto se realiza al añadir valores etiquetados a la asociación.
- 3 **Añada una propiedad a la relación pero no especifique clase.**
- 4 Deje la conversión de las relaciones uno a muchos en clases de colección a los programadores.



{clasificado}

Elementos en la colección se mantienen en un orden estricto.

{sinorden}

No existe orden de los elementos en la colección.

{único}

Los elementos en la colección son todos únicos; un objeto aparece en la colección como mucho una vez.

{noúnico}

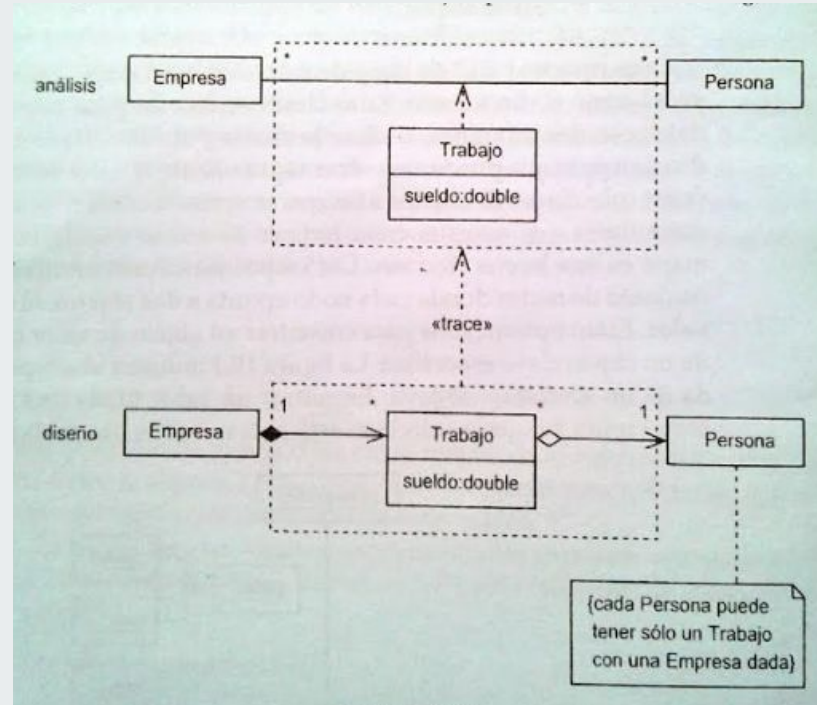
Duplicar elementos está permitido en la colección.

UNIDAD 3: Mejorar las relaciones de análisis

Reemplace las **clases de asociación!!**

Las clases de asociación son artefactos que utilizamos en el análisis que no están directamente soportados por ningún lenguaje de programación orientados a objetos. Por lo tanto hay que eliminarlas del modelo de diseño.

Decida cuál es el todo y cuál las partes y utilice composición y agregación para reemplazarla.



UNIDAD 3: Mejorar las relaciones de análisis



Revise el trabajo propuesto por el equipo para identificar y reemplazar:

1. Asociaciones de 1 a *
2. Clases de asociación

UNIDAD 3: Mejorar las relaciones de análisis



Fuente:

UML 2

- “18 Mejorar relaciones de análisis”