

Trabajo Práctico 1 — Smalltalk

[7507/9502] Algoritmos y Programación III
Curso 2
Primer cuatrimestre de 2022

Alumno:	IGLESIAS Tomás
Número de padrón:	105995
Email:	tiglesias@fi.uba.ar

Índice

1. Introducción	2
2. Supuestos	2
3. Diagramas de clase	2
4. Detalles de implementación	3
4.1. Coste de las Llamadas	3
4.2. Manejo de tarifa en LlamadaNacional	4
5. Excepciones	4
6. Diagramas de secuencia	4

1. Introducción

El presente informe reúne la documentación de la solución del primer trabajo práctico de la materia Algoritmos y Programación III que consiste en desarrollar una aplicación de un sistema de una agencia telefónica en Pharo utilizando los conceptos del paradigma de la orientación a objetos vistos hasta ahora en el curso.

2. Supuestos

Para este modelo se consideró que cuando se registren llamadas de cualquier tipo, nunca recibirán una duración negativa pero sí puede ser cero. Se puede observar en todos los tests que las duraciones son siempre mayores o iguales a cero, siendo el test 13 el responsable de este último caso

Además, se aplica un descuento del diez por ciento a todas las llamadas que se registran con una duración mayor o igual a treinta minutos, independientemente de que la llamada sea nacional o internacional. En los tests 7 y 9 se registran llamadas nacionales e interanacionales respectivamente con duración suficiente para que el descuento sea aplicado y se puede comprobar que es aplicado efectivamente.

Por último, la tarifa internacional es única por cada instancia de TeleAlgo independientemente del país al que se realice la llamada. Dicho supuesto es respaldado por el test 9, donde se aplica una tarifa interanacional por minuto fija y registran varias llamadas internacionales a distintos países.

3. Diagramas de clase

En los diagramas de clase que se muestran a continuación, figuran las entidades responsables de implementar la solución. Las mismas fueron divididas en dos vistas: una general (Figura 1) y una específica para las llamadas (Figura 2). Cabe añadir que algunos mensajes triviales como setters, fueron omitidos de los diagramas optimizar su entendimiento.

En la primera figura se puede observar la clase TeleAlgo cuya mayor responsabilidad es delegar mensajes a otras entidades. Es la clase de más alto nivel y está compuesta por una y solo una ColecciónLlamadas. También dispone de los atributos abonoMensual, que es la abstracción de un monto fijo que debe ser pagado en la vida real, y tarifaInternacionalPorMinuto corresponde al coste por minuto que se cobra a una llamada internacional

ColecciónLlamadas busca facilitar el manejo de las llamadas y permitir a TeleAlgo delegar responsabilidades. Esta hace uso de la entidad Llamada en su atributo llamadas, el cual contiene 0 o varias llamadas registradas de una instancia de TeleAlgo

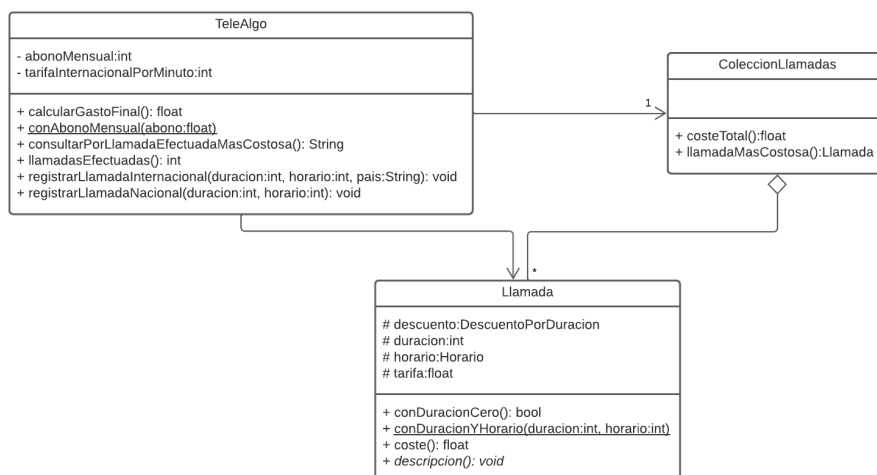


Figura 1: Diagrama general del modelo.

En cuanto a las llamadas propiamente dichas, se tomó la decisión de implementar una clase madre Llamada con los atributos y métodos que se pueden observar en la Figura 2. La clase en cuestión entiende el mensaje `coste` el cual se encargará de devolver el monto correspondiente haciendo uso de sus atributos `tarifa`, `duracion` y `descuento`. Cabe agregar que Llamada dispone del método abstracto `descripcion` el cual deberá ser implementado por todas sus hijas.

LlamadaNacional hereda de Llamada. Cuenta con un método para asignar una tarifa según el horario de la llamada (ver Item 5 Detalles de Implementación para más detalles) que es utilizado en el constructor `conDuracionYHorario()` para que de esa forma las instancias sean inicializadas directamente con su tarifa correspondiente.

LlamadaInternacional es otro tipo de Llamada. Tiene un `pais` que hace alusión al país de destino al cual se hace la llamada y un constructor propio para recibir la tarifa internacional de manera directa.

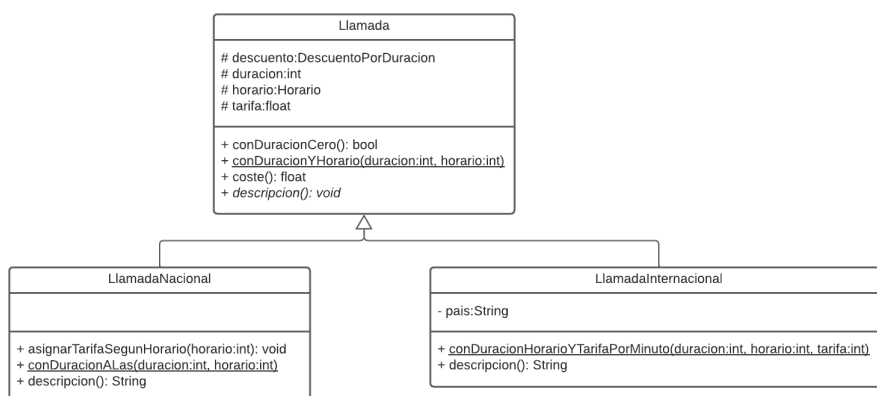


Figura 2: Diagrama de llamadas.

4. Detalles de implementación

4.1. Coste de las Llamadas

Dentro de la clase Llamada se encontrará el método `coste`. Como se mencionó anteriormente, su responsabilidad es responder el coste de la llamada con el resultado del producto entre la tarifa y

la duracion de la llamada menos el descuento de la misma. El descuento sera aplicado a través del mensaje aplicarA el cual recibe el monto al cual se aplicará el porcentaje y devolverá el porcentaje del monto indicado. Por ejemplo, tomando $\text{tarifa} = 1$, $\text{duracion} = 10$ y descuento del 10 porciento, el método `coste` responderá 9. Para un mejor entendimiento, el código en cuestión es el siguiente:

```
coste

costeLlamada := tarifa * duracion.

^ costeLlamada - (descuento aplicarA: costeLlamada).
```

4.2. Manejo de tarifa en LlamadaNacional

Toda Llamada tiene una tarifa de coste por minuto y LlamadaNacional no es la excepción. El comportamiento esperado de LlamadaNacional al recibir el mensaje `coste` es que dependiendo de si la llamada se efectuó en horario hábil o no, responda un valor u otro ya que la tarifa por minuto para llamadas en hora hábil (8 a 20 hs) es de 0,2 y en hora no hábil es de 0,1

Para resolver esto, LlamadaNacional delega en su horario la asignacion de la tarifa, llamando desde `asignarTarifaSegunHorario()` al metodo `asignarTarifa()` de `Horario`, en el cual se encuentra la implementación de lo mencionado. El `Horario` tiene una hora y el algoritmo se encargará de evaluar si está comprendida entre 8 y 20 (plazo de horario hábil) o no y en base a eso asignará la tarifa correspondiente a la instancia de LlamadaNacional.

5. Excepciones

ColeccionLlamadasVacíaError Es lanzada cuando la colección de llamadas en cuestión, es una colección vacía. Su finalidad es dar una explicación concisa de por qué la colección no puede responder mensajes que requieran hacer consultas a sus elementos

DuracionInvalidaError Se lanza cuando la duracion recibida es igual a 0. Creada con el objetivo de validar duraciones de llamadas recibidas

6. Diagramas de secuencia

A continuación se muestran dos diagramas de secuencia para ejemplificar un caso donde se consulta por la llamada mas costosa a una instancia de `TeleAlgo`

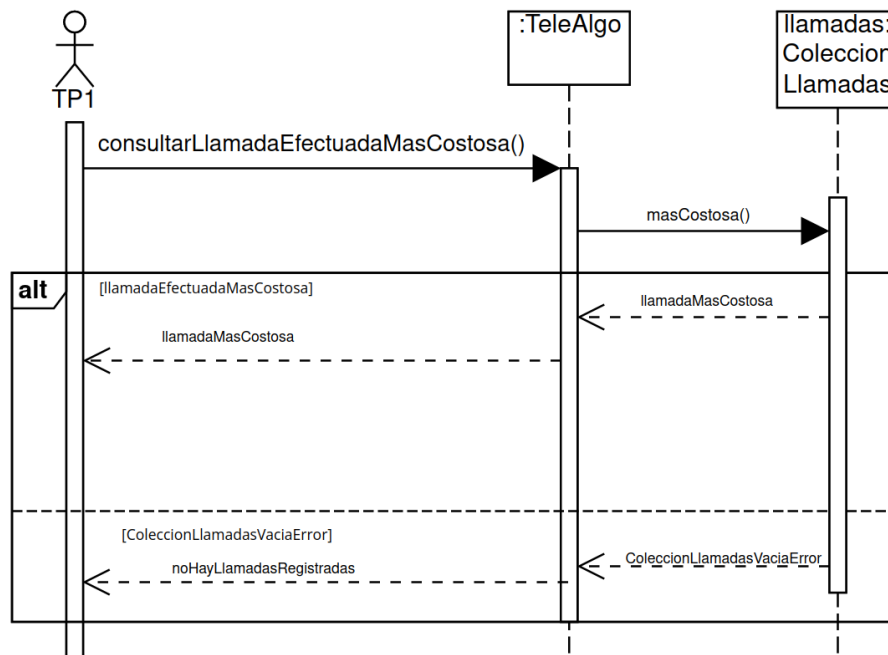


Figura 3: Diagrama de comportamiento de consultarPorLlamadaEfectuadaMasCostosa.

En el diagrama de secuencia de la Figura 3 se observa que se solicita la llamada efectuada mas costosa a una instancia de `TeleAlgo`. Esta delega en `llamadas` (una instancia de `ColeccionLlamadas`) y le solicita la llamada mas costosa. Puede levantar una excepcion de tipo `ColeccionLlamadas-VaciaError` si la colección está vacía o caso contrario contestar la llamada mas costosa. `TeleAlgo` evalúa lo respondido por `llamadas` y si todo salió bien responde la misma llamada al contexto que la solicitó. Caso contrario manejaría el error recibido y responderá que no hay llamadas registradas y por lo tanto no existe la llamada mas costosa

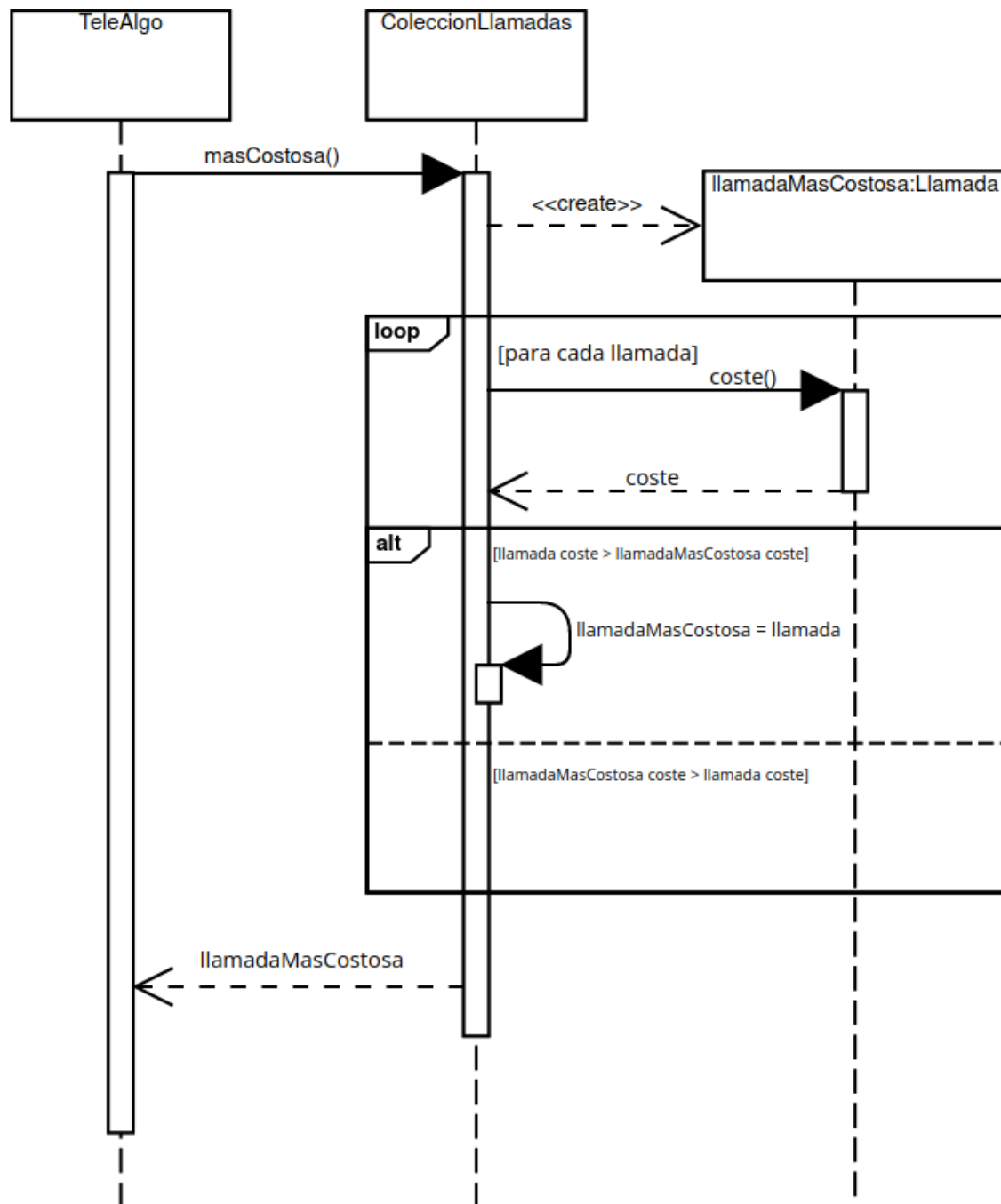


Figura 4: Diagrama de comportamiento interno.

Aquí podemos ver el comportamiento interno dentro de ColeccionLlamadas al recibir el mensaje masCostosa() luego de comprobar que existen llamadas como se indicó en el párrafo anterior.

ColeccionLlamadas almacena en llamadaMasCostosa la primer Llamada dentro de su coleccion de llamadas. Entra en un bucle donde consulta a cada llamada por su coste y a la llamadaMasCostosa para así comparar sus respuestas. Si el valor respondido por llamada es mayor que el respondido por llamadaMasCostosa, ésta última toma el valor de llamada en esa iteración, sino se sigue iterando hasta que no haya mas llamadas en la colección.

Finalmente ColecciónLlamadas responde la llamadaMasCostosa.