

# Trabajo Práctico 1 — Smalltalk

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

Alumno:	SOBRAL, Francisco
Número de padrón:	108603
Email:	fsobral@fi.uba.ar

## Índice

1. Introducción	2
2. Supuestos	2
3. Detalles de implementación	2
4. Excepciones	2
5. Diagramas de clase	3
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 para una empresa de telefonía en Pharo utilizando los conceptos del paradigma de la orientación a objetos vistos hasta ahora en el curso.

## 2. Supuestos

En mi caso, a través de las pruebas, interprete que TeleAlgo era una empresa de telefonía, la cual se encargaba de almacenar llamadas de los usuarios, para más adelante facturarlas y utilizar su información. En cuanto a suposiciones, yo asumo que una llamada no puede tener una duración negativa, y por ende, en caso de tenerla, el programa lanzaría un error ya que el numero ingresado es invalido. Por otra parte, asumo que el horario habilitado es de 1 a 24 horas y como consecuencia, cualquier llamada realizada fuera de dicho horario también va a ser tomada como invalida y finalizara la ejecución del programa. Para agregar, las tarifas aplicadas a viajes internacionales tampoco pueden tener un valor negativo, en caso de tenerla, las consecuencias serían las expresadas en el ítem anterior. Para concluir, mi última suposición, fue que los viajes Nacionales siempre van a tener como destino 'Argentina'.

## 3. Detalles de implementación

En este programa, yo opte por utilizar 4 clases, de las cuales dos son clases hijas de una. Para empezar, la clase TeleAlgo es la encargada de almacenar las llamadas de los usuarios, y la clase Llamada es la encargada de llevar a cabo dichas llamadas. Decidí utilizar herencia ya que hay dos tipos de llamadas: las nacionales y las internacionales. Como consecuencia de esto, ambas clases tienen cosas en común, así como también tienen sus diferencias. A raíz de esta situación, decidí crear una clase padre con nombre Llamada, en la cual va a crear los atributos que tienen en común (duración de llamada, horario de llamada y el destino de esta). Entonces, creo estos atributos en la clase padre y luego simplemente lo llamo en las hijas (agregando los atributos particulares de las clases hijas). La ventaja de esto fue que no tuve que repetir el código para generar dichos atributos de cada clase, sino que los cree una vez y las hijas lo heredaron. En cuanto a delegación, lo utilizo en la clase TeleAlgo ya que delega toda responsabilidad de las llamadas a la clase Llamada. La función de TeleAlgo es almacenar el objeto generado por la clase Llamada para luego utilizar su información.

## 4. Excepciones

Para perfeccionar mi programa, opte por utilizar tres excepciones, las cuales tienen como consecuencia la detención del programa.

**HoraInvalidaError** En primer lugar, este mismo aparece cuando el usuario ingresa un horario invalido, por ejemplo 25 horas. Los horarios validos son entre la 1 y las 24.

**MinutosInvalidosError** En segundo lugar, tenemos el error provocado por el ingreso de minutos inválidos los cuales serían minutos de valor negativo.

**TarifaInvalidaError** Y por último, este es ocasionado por ingresar una tarifa para llamada internacional de valor negativo.

## 5. Diagramas de clase

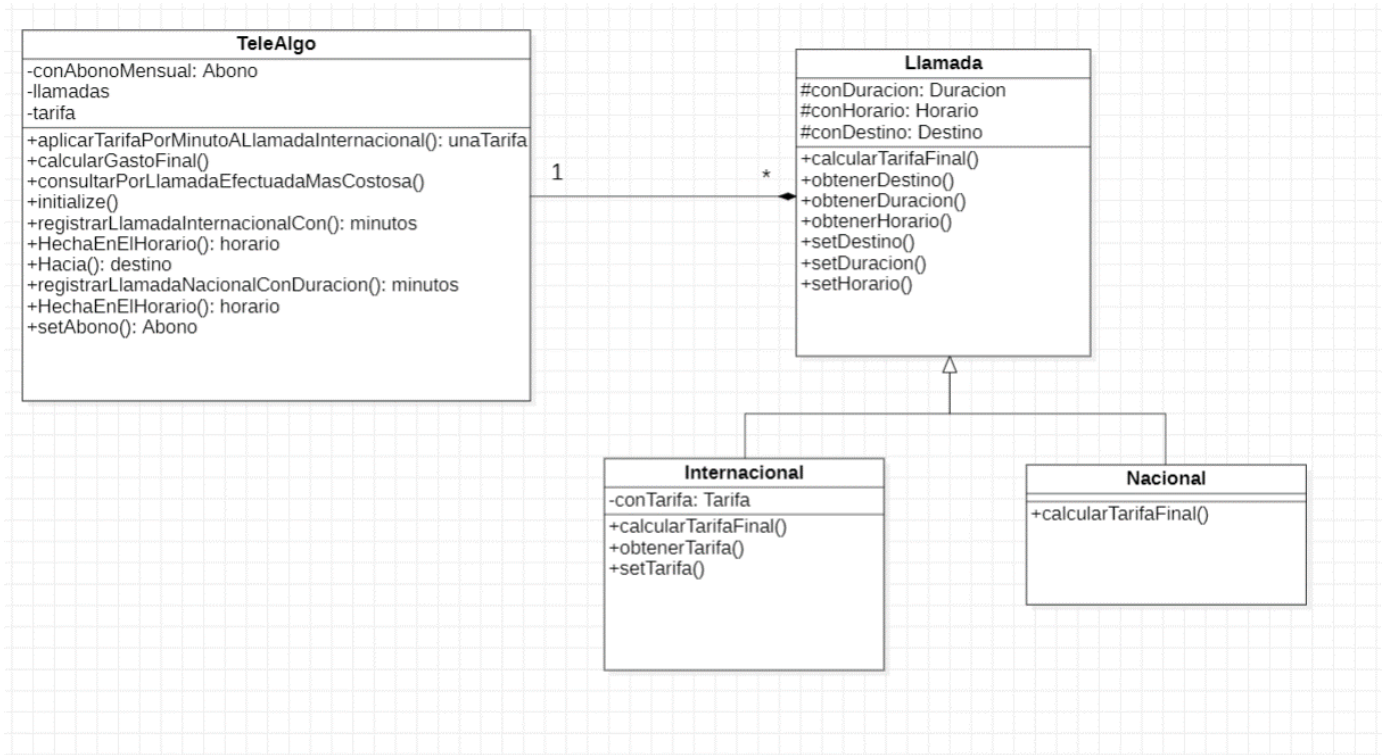


Figura 1: Diagrama de Clases.

## 6. Diagramas de secuencia

Como se puede ver, este diagrama muestra el paso a paso para obtener la llamada más cara, mostrando bien la creación de los objetos necesarios y sus distintos llamados para cumplir el objetivo a resolver.

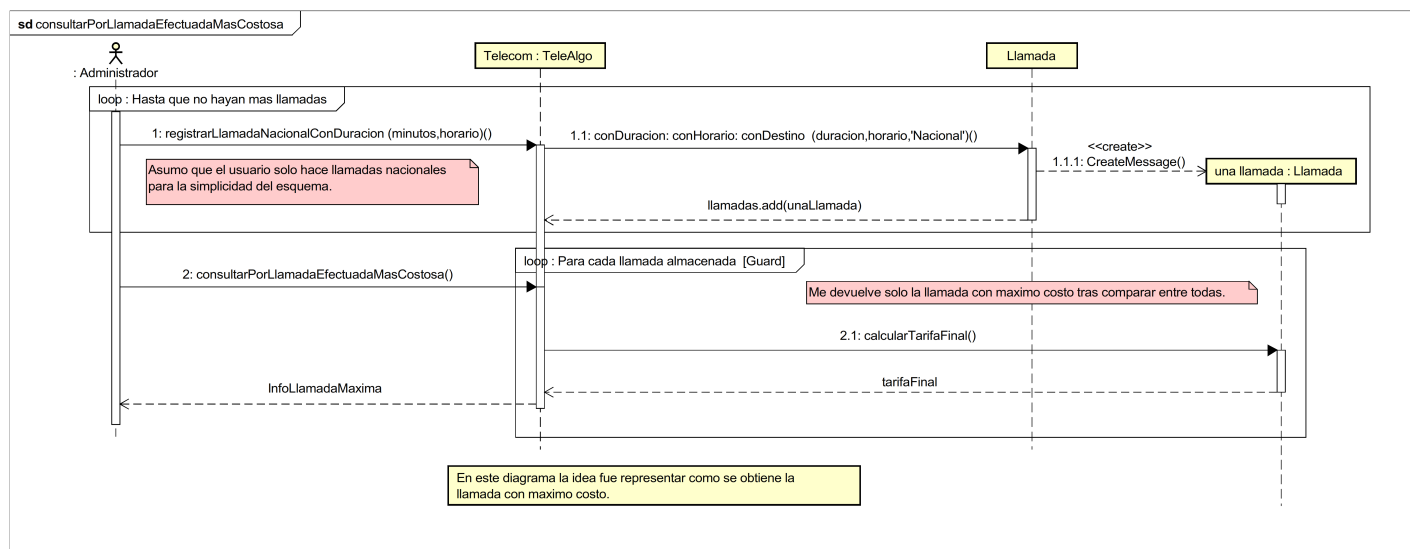


Figura 2: Consulta llamada mas cara.

En el siguiente diagrama, vemos como se obtiene la tarifa final de una llamada, mostrando el paso a paso.

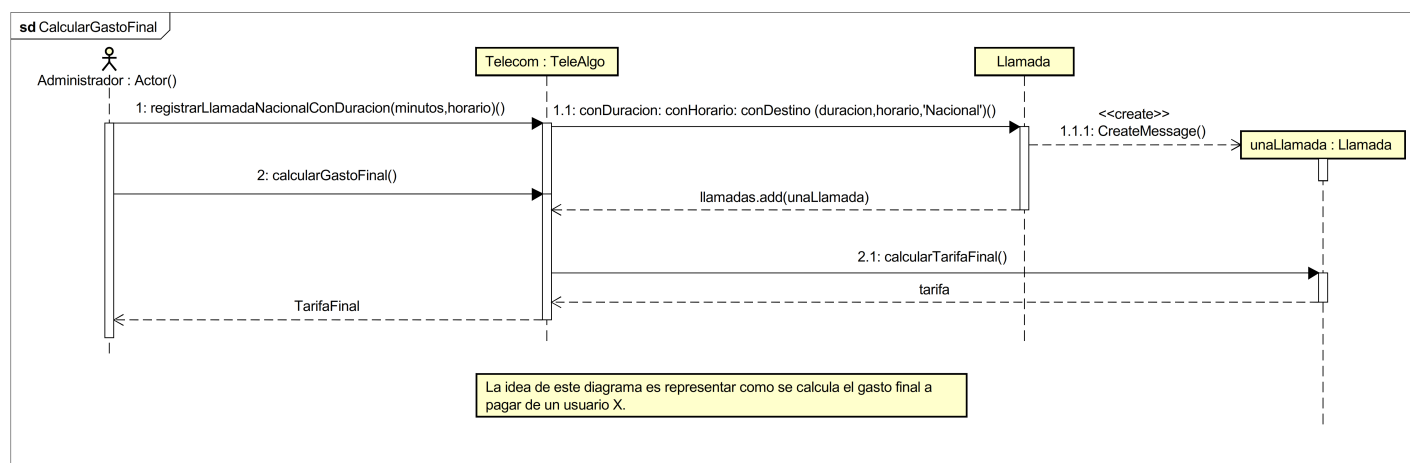


Figura 3: Consulta gasto final.