

Facultad de Ingeniería,  
universidad de buenos aires

# Trabajo práctico 1

## Algoritmos y Programación III

Alumno	María Carolina Silvestri
Padrón:	99175
Correo electrónico	silconito@gmail.com

## Introducción

El siguiente informe detalla la solución a un modelo de una agencia de viajes utilizando el paradigma orientado a objetos. Esta solución responde a la consigna del trabajo práctico de la materia Algoritmos y Programación 3: aplicando los conceptos explicados en la materia hasta el momento y, la metodología TDD desarrollar un modelo de clases de un sistema de agencia de viajes.

Se utilizará Pharo con el lenguaje Smalltalk para la resolución del mismo.

## Supuestos

Si bien en la implementación no fue necesario suponer algo a gran escala, si lo fue a pequeña escala para cada clase.

Nuestro cliente viaja, entonces va a tener una ida y una vuelta, además, si reservamos un hotel tenemos entrada y salida. Esto quiere decir que: vamos a tener máximo dos vuelos y mínimo una noche en el hotel. Por otro lado, la agencia ofrece paquetes de viaje entonces podemos especificar que dentro de este vamos a tener el la máxima cantidad de vuelos ( y que siempre vamos a tener vuelos ) y siempre con hotel. Sin embargo, el paquete no contempla la comida especial así que se decidió agregarla al paquete al igual que en el viaje.

## Modelo del dominio

Las clases principales son las siguientes:

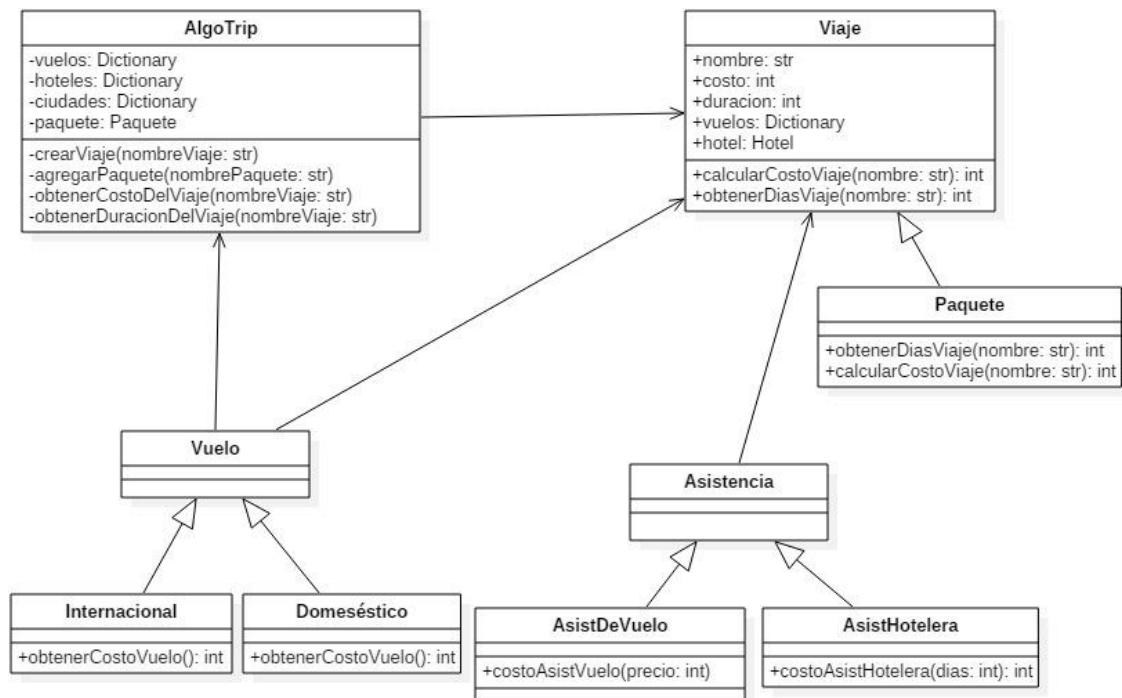
AlgoTrip: esta clase es la que moldea la agencia de viajes, es responsable de intermediar entre las otras clases para tener la información de lo que esta ocurriendo y por lo tanto, tener un registro de las cosas que se hacen. Es decir, todas las cosas que se agreguen o se modifiquen pasan por esta clase.

Viaje: esta clase moldea el viaje de nuestro cliente. Es responsable de juntar toda la información con respecto al mismo y generar un “informe” es decir calcular su precio total y su duración.

Paquete: esta clase es hija de viaje ya que realiza el mismo trabajo pero con ciertos cambios. Es tratada por AlgoTrip como una clase Viaje.

Vuelo: esta clase es una clase abstracta ya que sus hijas Internacional y Doméstico se encargan del mismo problema pero de distinta manera. Este problema es: calcular el precio de un Vuelo.

## Diagramas de clase



## Detalles de implementaci n

### AlgoTrip y Paquete

La clase **AlgoTrip** maneja una cierta cantidad de diccionarios para poder acceder r pidamente a la informaci n que necesita. El m s importante es el registro de viajes cuyas claves son los nombres de los viajes y sus valores una instancia de la clase **Viaje**.

Como fue mencionado anteriormente la clase **Paquete** es hija de la clase **Viaje** y **AlgoTrip** la trata como una clase **Viaje**. Esto se explica de la siguiente manera: el paquete es un viaje completo. Por ende, cuando llegue el momento de agregar un paquete al viaje, nuestra clase **AlgoTrip** que tiene un registro de todos los viajes creados, reemplazar  la clase **Viaje** que se cre  por la clase **Paquete** y as  le dar  el mismo trato.

### Vuelos y Asistencia

Como se mencion  anteriormente, **vuelo** es una clase importante ya que calcula el costo de nuestro vuelo. Existen dos tipos de vuelos: **Internacional** y **Dom stico**, y cada uno tiene un costo por kil metro distinto. Por ende, **vuelo** termina siendo una clase abstracta y le delega el trabajo a sus hijas para que cada una haga los c lculos con sus valores.

Por otro lado, a la clase **Asistencia** le ocurre lo mismo al haber dos tipos de asistencias que calculan con distintos valores, la clase **Asistencia** termina siendo una clase abstracta que delega.

## Excepciones

Las excepciones creadas para detectar posibles errores en el funcionamiento del programa son:

- HotelPrecioNegativo: un precio no puede ser negativo por ende esta excepción será lanzada para que no rompa con el modelo planteado.
- AlgoTripViajeInexistente: si se intentan agregar elementos a un viaje que no existe, se lanzará esta excepción.
- AlgoTripCiudadNoDisponible: si no se agrego la ciudad previamente a la agencia, entonces esa ciudad no esta disponible.
- ViajeDiasNegativos: la cantidad de días de un viaje no puede ser un valor negativo, no tiene sentido y por ende si llega a ocurrir se lanza la excepción.