FACULTAD DE INGENIERIA
Universidad de Buenos Aires

Trabajo Práctico 1 — Smalltalk

[7507/9502] Algoritmos y Programación III
Curso 01 - Sánchez
Primer cuatrimestre de 2023

Alumno:	CABRERA, Isaías Augusto
Nro. de Padrón:	108885
Email:	iacabrera@fi.uba.ar

Índice

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 de viajes en Pharo utilizando los conceptos del paradigma de la orientación a objetos vistos hasta ahora en el curso.

2. Supuestos

Para este trabajo se tomó como idea principal el hecho de que esta es una empresa de Aerolíneas con base en Argentina, BsAs por lo que, cuando nos referimos a vuelos nacionales o locales, estamos haciendo referencias a vuelos que ocurren en el propio país y que parten desde Ezeiza. También consideramos que es una empresa nueva por lo que actualmente solo cuenta con el permiso de dos países además de sí mismo para poder transportar pasajeros. Estos países son Perú con el que forman parte del MERCOSUR, y Portugal que es considerado un país extranjero. En el futuro es probable que se asignen másvuelos a otros países, pero de momento esos son los únicos disponibles. Otra consideraciónque se hizo es que, si la nacionalidad de la que proviene el pasajero no es reconocida como como país por parte de los estatutos de la empresa, este misma se considerará como extranjera de acuerdo a los criterios de nuestra aerolínea

3. Modelo de dominio

Las clases creadas durante este programa se hicieron con la finalidad de establecer un sistema básico que permitiera a un individuo registrar un vuelo por medio de nuestra aerolínea hacia un determinado destino. Dentro de este contexto, sedesarrollaron formas por las cuales poder establecer una tarifa de vuelo según el destino a que se desee viajar, y diferentes criterios para establecer estas tarifas en casode que se desee realizar múltiples vuelos por medio de la aerolínea.

En primer lugar, se creó la clase AlgoViajes que tiene como finalidad hacer de puente entre los datos que se reciben del usuario y los demás procesos que se puedanefectuar con ellos.

Luego tenemos clases como Pasaje que almacena los datos provistos por el usuarioque luego nos resultan útiles para obtener la tarifa de viaje, el destino al que se desea viajar y los datos del pasajero.

Las demás clases se hicieron con el objetivo de favoreces el funcionamiento de estasdos clases, dividiendo las responsabilidades entre cada una de ellas.

4. Diagramas de clase

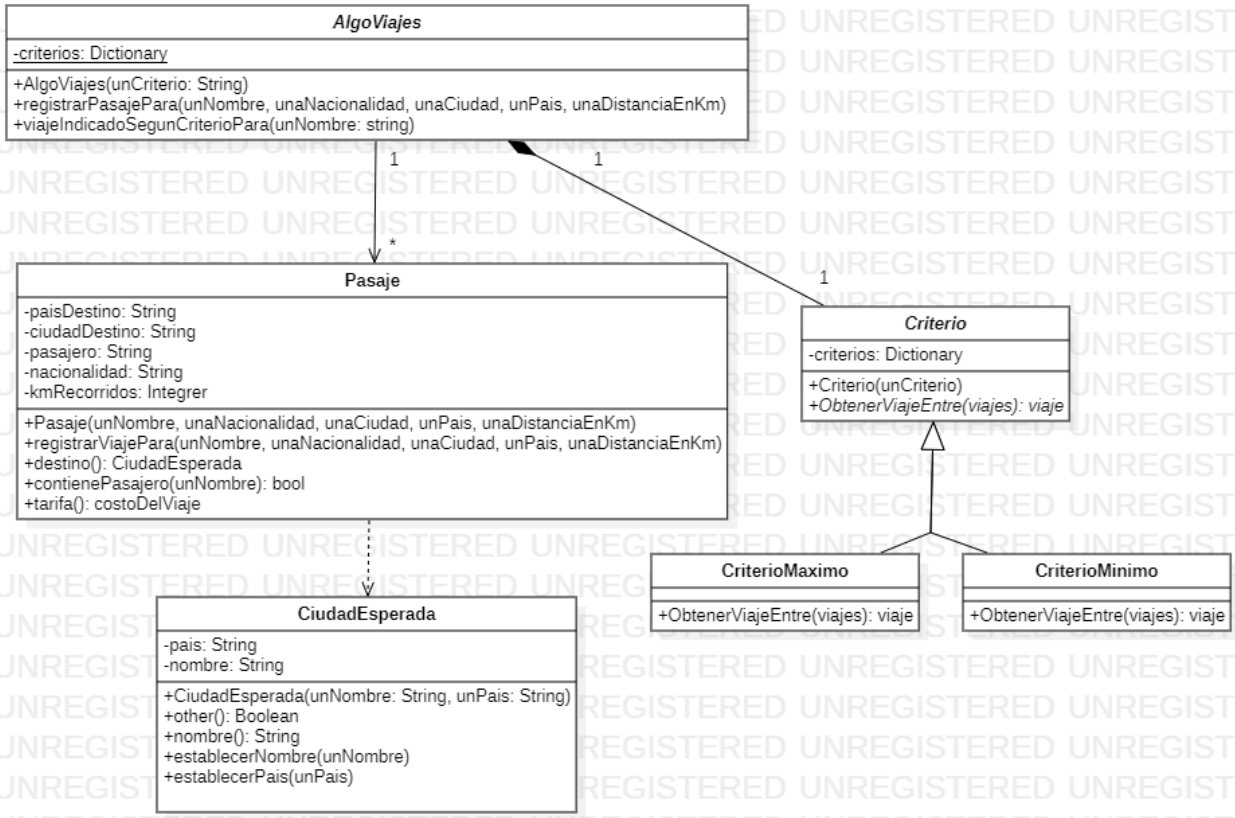

Figura 1: Diagrama de clases general.

En el diagrama que se presenta anteriormente, se puede apreciar la relación de herencia que presenta Criterio y la asociación de esta con la clase AlgoViajes.

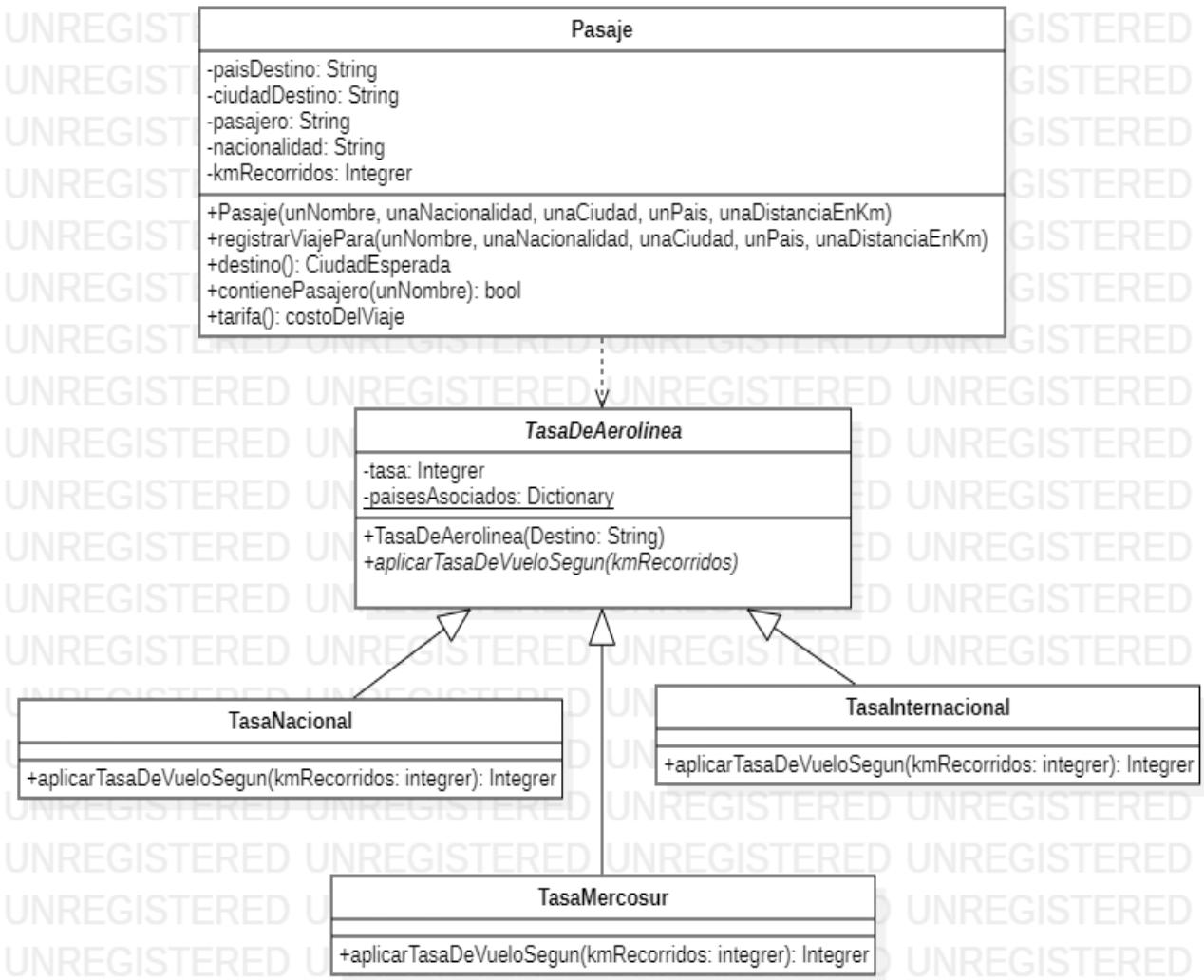

Figura 2: Dependencia con TasaDeAerolinea.

En el diagrama que se muestra arriba se refleja la relación de herencia que se produce con respecto a la clase TasaDeAerolineas, y a su vez como Pasaje depende de TasaDeAerolineas ya que es esta la que le permite obtener la tarifa del pasaje.

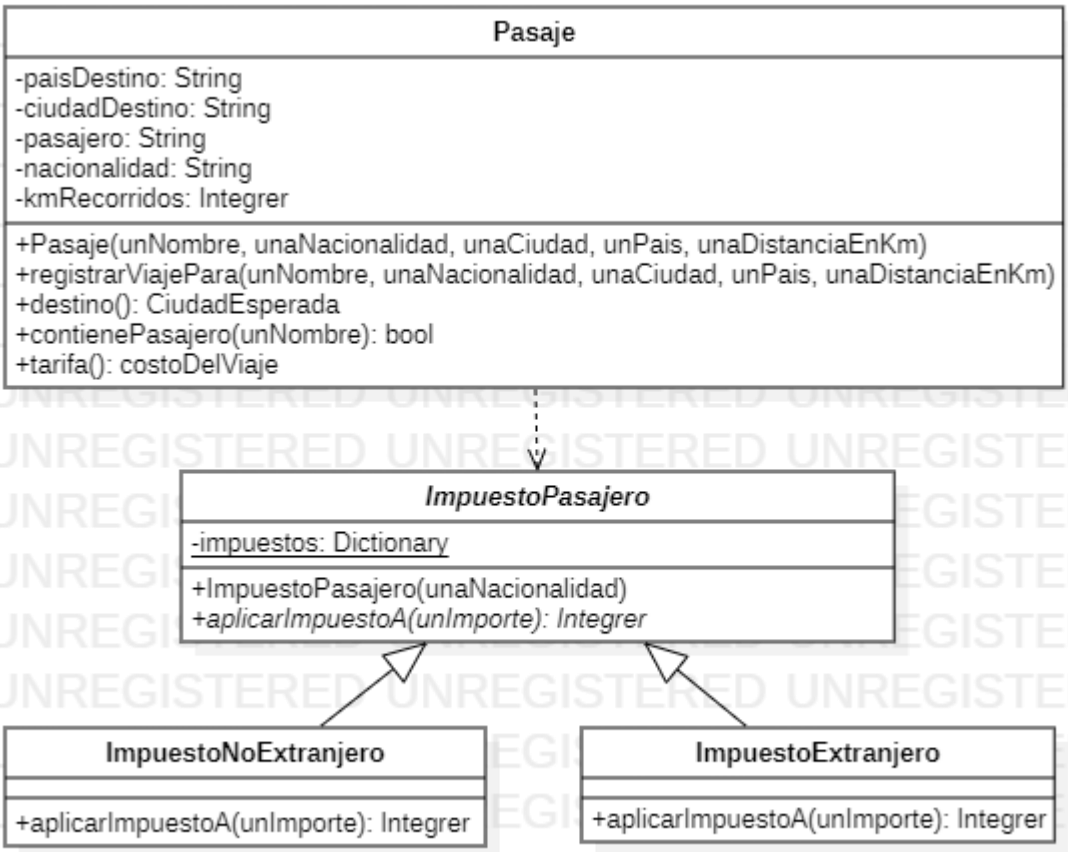

Figura 3: Dependencia con ImpuestoPasajero.

Al igual que antes, se muestra una relación de herencia a partir de las clase ImpuestoPasajero así como también la dependencia que la clase Pasaje tiene de esta ya que es la que le permite obtener la tarifa del pasaje.

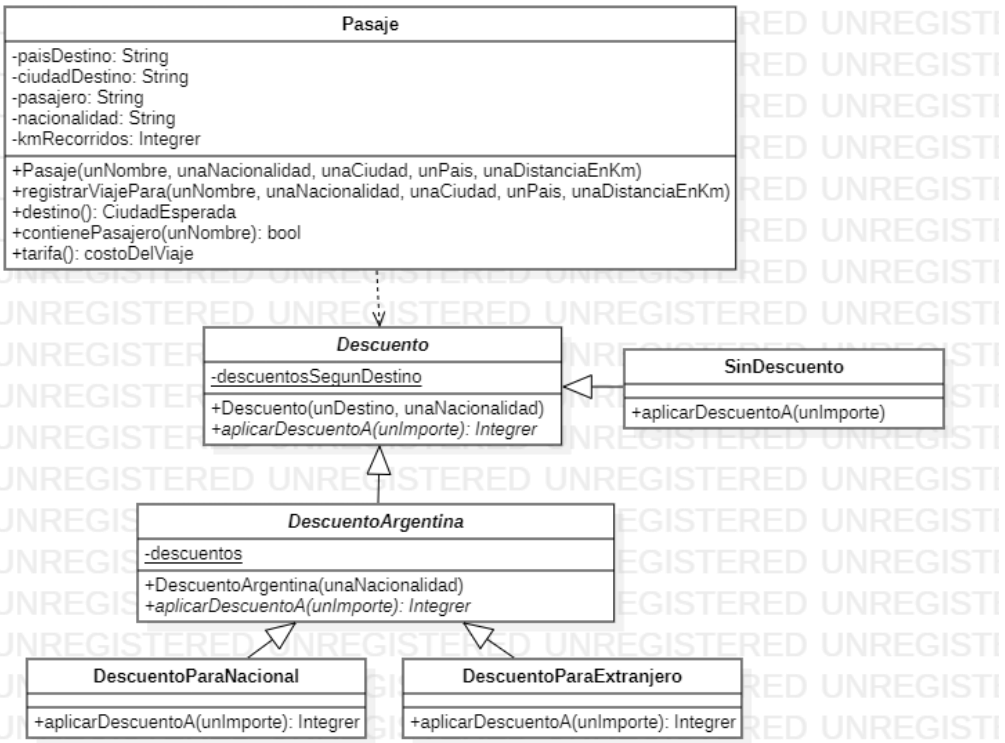

Figura 4: Dependencia con Descuento.

Al igual que antes, se muestra una relación de herencia a partir de las clase Descuento así como también la dependencia que la clase Pasaje tiene de esta ya que es la que le permite obtener la tarifa del pasaje.

5. Detalles de implementación

5.1. AlgoViajes

AlgoViajes es una clase creada por ustedes que representa el sistema de registro de viajes. En un primer momento era una clase que contaba con 2 métodos de instancia, los cuales son: **registrarPasajeParaDeNacionalidadconDestinoAyPaisaKms** y **viajeIndicadoSegunCriterioPara**. Luego está el método de clase **ConCriterio**, el cual fue utilizado como constructor de instancias para AlgoViajes.

Dentro de las mismas se almacena una base de datos en forma de colección de los viajes en donde en caso de necesitar un viaje de determinado pasajero, por medio del método viajeIndicadoSegunCriterioPara se puede obtener el viaje solicitado (si existe), a partir de esa colección.

```
AlgoViajes:
   viajeIndicadoSegunCriterioPara: unNombre

   | viajesRegistrados|

   viajesRegistrados := viajes select: [: pasaje | pasaje contienePasajero: unNombre ].

   viajesRegistrados ifEmpty: [ PasajeroInexistente signal].

   ^ criterioDeViaje obtenerViajeEntre: viajesRegistrados.
```

5.2. CiudadEsperada

Ciudad Esperada es una clase proporcionada por ustedes y representa la ciudad destino esperada por el viaje. La clase que cuenta con un constructor llamado **nombreen** al cual se le envía el nombre de la ciudad a la que se espera ir y luego el paísal que esta pertenece. Como métodos de instancia se implementan **asignarNombre, asignarPais y nombre**. Los dos primeros se encargan de recibir los valores obtenidos mediante el constructor y almacenarlos en el interior de la instancia creada. El método nombre se encarga de devolver el nombre de la ciudad que ha sido anteriormente almacenado al crear la instancia.

5.3 TasaDeAeropuerto

TasaDeAeropuerto es una clase creada con el fin de contener los métodos necesariospara aplicar las tasas de embarque requeridas para vuelos tanto nacionales como internacionales. La clase cuenta con un atributo de instancia denominado **paisesAsociados** en donde se guardan el nombre de los países a los cuales la aerolínea tiene permitido volar, y por lo tanto establecer tasas de vuelo según los kilómetros recorridos desde el país origen al país destino.

La clase tiene un constructor llamado **ConDestino**, el cual se encarga de crear todas las instancias de **TasaDeAeropuerto** en caso de que sea invocado. Además, la clase cuenta con el método **aplicarTasaDeVueloSegun** el cual como se comentó anteriormente, calcula la tasa en base a los kilómetros. Este método es abstracto y cede la responsabilidad de ejecutarse a sus clases hijas denominadas: **TasaInternacional, TasaNacional, TasaMercosur** en donde cada una responde al mensaje de diferente forma.
TasaDeAeropuerto además cuenta con un atributo de instancia denominado **tasa**, elcual se inicializa con un determinado valor según el importe establecido por la aerolínea como “tasa de embarque”.

5.4. ImpuestoPasajero

Impuesto Pasajero es una clase creada con el fin de contener los métodos necesariospara aplicar los costes asociados a los recargos que aplica la aerolínea según la nacionalidad del pasajero, así como también descuentos en ciertas ocasiones como extranjeros con vuelos nacionales o pasajeros argentinos que viajan por el país. La clasecuenta con un constructor denominado **ConNacionalidad** el cual crea una instancia de la clase a partir de la nacionalidad recibida. Al crearse las instancias se define un atributo de clase denominado **impuestos**, el cual no es más que un diccionario que de momento solo almacena a Argentina como referencia, pero eventualmente si se desea establecer vuelos con demás países, deberá contener si es que se desea aplicar el impuesto correspondiente. En caso de que la aerolínea no tenga conocimiento de la nacionalidad u origen especificado del pasajero, se le atribuirá un impuesto extranjero asu viaje. La clase tiene dos hijas que responden al mensaje de **aplicarImpuestoAconDestin**o el cual es un método de instancia donde se aplica el impuesto en base al importe que recibe por parámetro.

5.5. Pasaje

Pasaje es una clase creada con la finalidad de almacenar los datos provistos por el usuario de forma compacta y poder hacer operaciones con ellos de ser necesario. La clase cuenta con un constructor llamado **RegistrarParDeNacionalidadConDestinoACiudadDelPaisAKms** en el cual se crea la instancia de pasaje y a la misma se le asignan los atributos correspondientes que luego necesitara para realizar la diversas operaciones de la clase.

La instancia de la clase está compuesta por los atributos **pasajero, paisDestino, nacionalidad, ciudadDestino y kmRecorridos**, los cuales nos permiten obtener la información necesaria acerca del viaje del individuo. Dentro de esta clase tenemos los métodos **tarifa, destino**, **reservarViajeParaDeNacionalidadconDestinoAyPaisaKm** y **contienePasajero**. El de reservarViajePara se encarga de asignar los valores correspondientes al vuelo en la instancia pasaje. Luego, contienePasajero es un método que tiene la finalidad de poder encontrar luego el pasaje del pasajero correcto a la hora de buscarlo entre las bases de datos, y destino es un método que crea una instancia de otra clase denominada **CiudadEsperada** a la cual se le asigna el destino al que se deseallegar.

Finalmente tenemos el método tarifa, el cual se encarga de calcular el costo total del viaje aplicando los impuestos, descuentos y tasas de aerolinea correspondientes.

5.6 Criterio

Criterio es una clase creada con la finalidad de definir el comportamiento al obtener un viaje según el criterio ingresado a la hora de crear AlgoViajes. La clase cuenta con un atributo denominado **criterios**, el cual guarda en su interior los criterios validos.

Criterio cuenta con dos subclases denominadas **CriterioMaximo** y **CriterioMinimo**, las cuales por medio del método ObtenerViajeEntre se encargan de seleccionar el viaje mas costoso o el menos costoso según el criterio de cada una. Ademas, al momento de crear un criterio, esta clase determina si el criterio ingresado es valido o no, en donde en el caso en el que no sea valido se lanzara una excepción.

CriterioMinimo:
ObtenerViajeEntre: viajes

```
^ viajes detectMin: [: elemento |elemento tarifa]
```

CriterioMaximo:
ObtenerViajeEntre: viajes

```
^ viajes detectMax: [: elemento |elemento tarifa]
```

5.7 Descuento

La clase descuentos es una clase abstracta que cuanta con atributo de clase denominado **descuentosSegunDestino**, el cual almacena dentro de si los diferentes destinos a los que se le puede aplicar descuento. La clase se crea por medio del constructor denominado **AlImporteConDestinoConNacionalidad**, en donde según los parámetros enviados, creara la instancia de descuento necesaria.
Inicialmente solo podemos aplicarle descuento a los vuelos con destino a Argentina por lo que descuento cuenta con una subclase denominada **SinDescuento** y con **DescuentoArgentina**, la cual a su vez cuenta con dos hijas llamadas **DescuentoParaNacional** y **DescuentoParaExtranjero** según la nacionalidad del pasajero que desee viajar a Argentina.
El unico metodo de esta clase es **aplicarDescuentoA**, en donde recibe un importe y al mismo le aplica el descuento necesario en caso de tener que recibir uno.

6. Excepciones

Criterio Inexistente: Al invocar al constructor de la clase Algo Viajes, si a este no se le envíauno de los criterios validados para crear las instancias de la misma, entonces se lanza esta excepción.

Distancia Inexistente: Cuando estamos registrando el viaje, si los kilómetros a recorrer tienen un valor negativo o nulo, entonces se lanza este error indicando que eso no es posible.

País Inexistente: Cuando se intenta generar la tasa de vuelo con destino a un país que no se encuentra asociado con la aerolínea (y por lo tanto no existen vuelos actuales que vayan hastaallí), entonces se lanza esta excepción notificando que el país no existe dentro de la base de datos de la aerolínea.

Pasajero Inexistente: Cuando se intenta obtener el viaje de un pasajero al cual no se le ha registrado aun en el sistema, se lanzará esta excepción indicando que el mismo no se encuentra en la base de datos.

7. Diagramas de secuencia

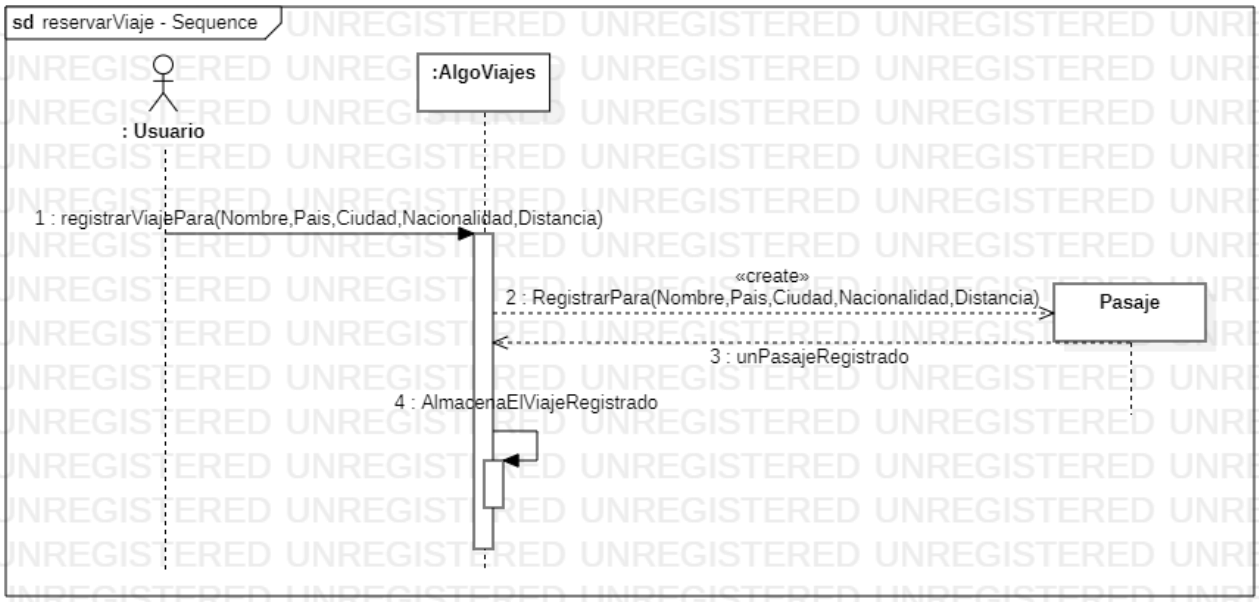

Figura 1 – Reservar un viaje.

En el diagrama se representa como el usuario a partir de una instancia de AlgoViajes ya creada, es capaz de reservar un viaje que luego es almacenado en la base de datos de AlgoViajes.

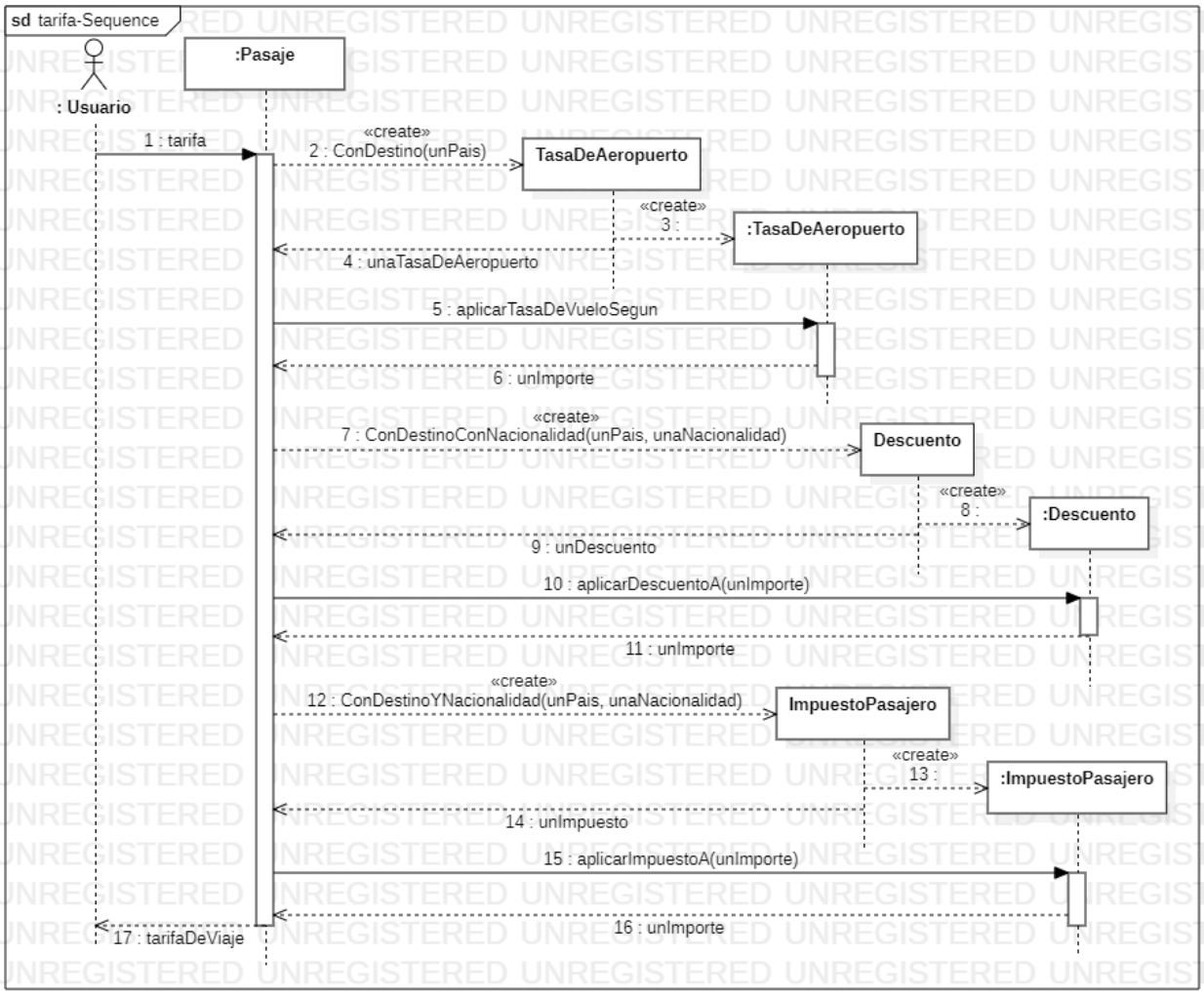

Figura 2 – Obtener tarifa.

En el diagrama se representa como el usuario partiendo de un viaje ya obtenido (su pasaje), envía el mensaje tarifa a una instancia de pasaje para que esta, por medio de las clases TasaDeAeropuerto, Descuento e ImpuestoPasajero, pueda calcular el valor del viaje.

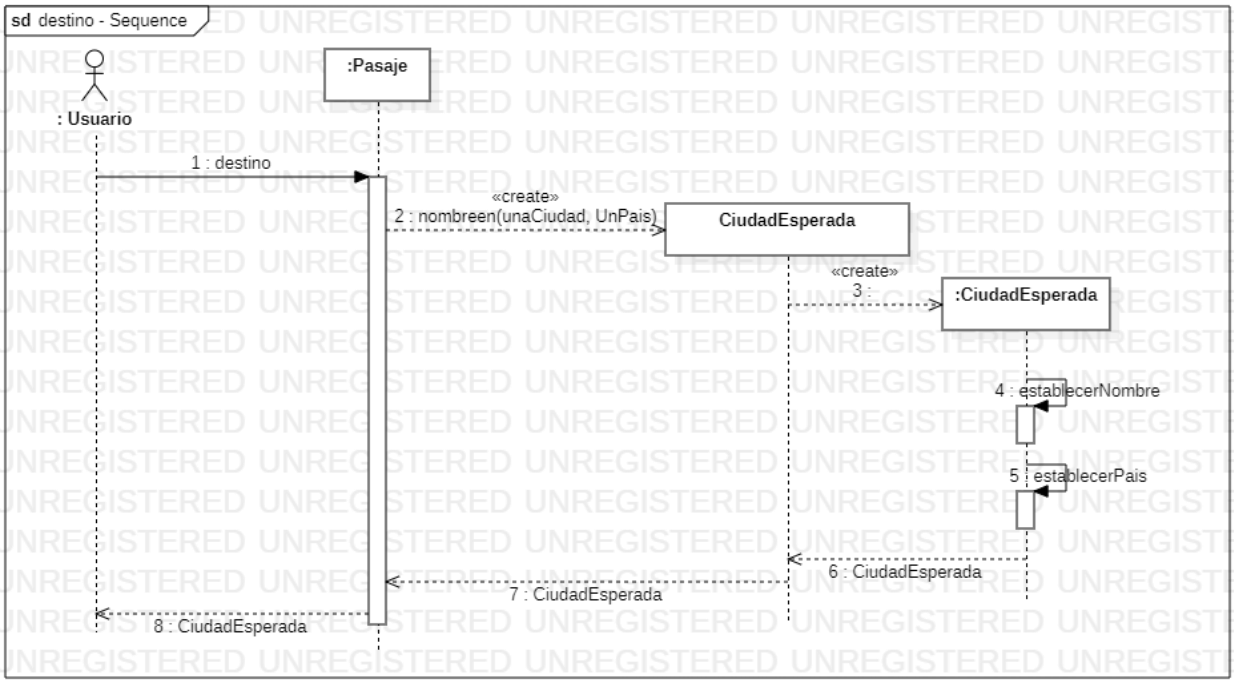

Figura 3 – Obtener destino

En el diagrama se representa como partiendo nuevamente de un viaje ya obtenido, se envía el mensaje destino a la instancia Pasaje que se encuentra almacenada en la basede datos de AlgoViajes, y esta por medio de la clase CiudadEsperada devuelve una instancia de si misma con la ciudad y el país al que se desea viajar.

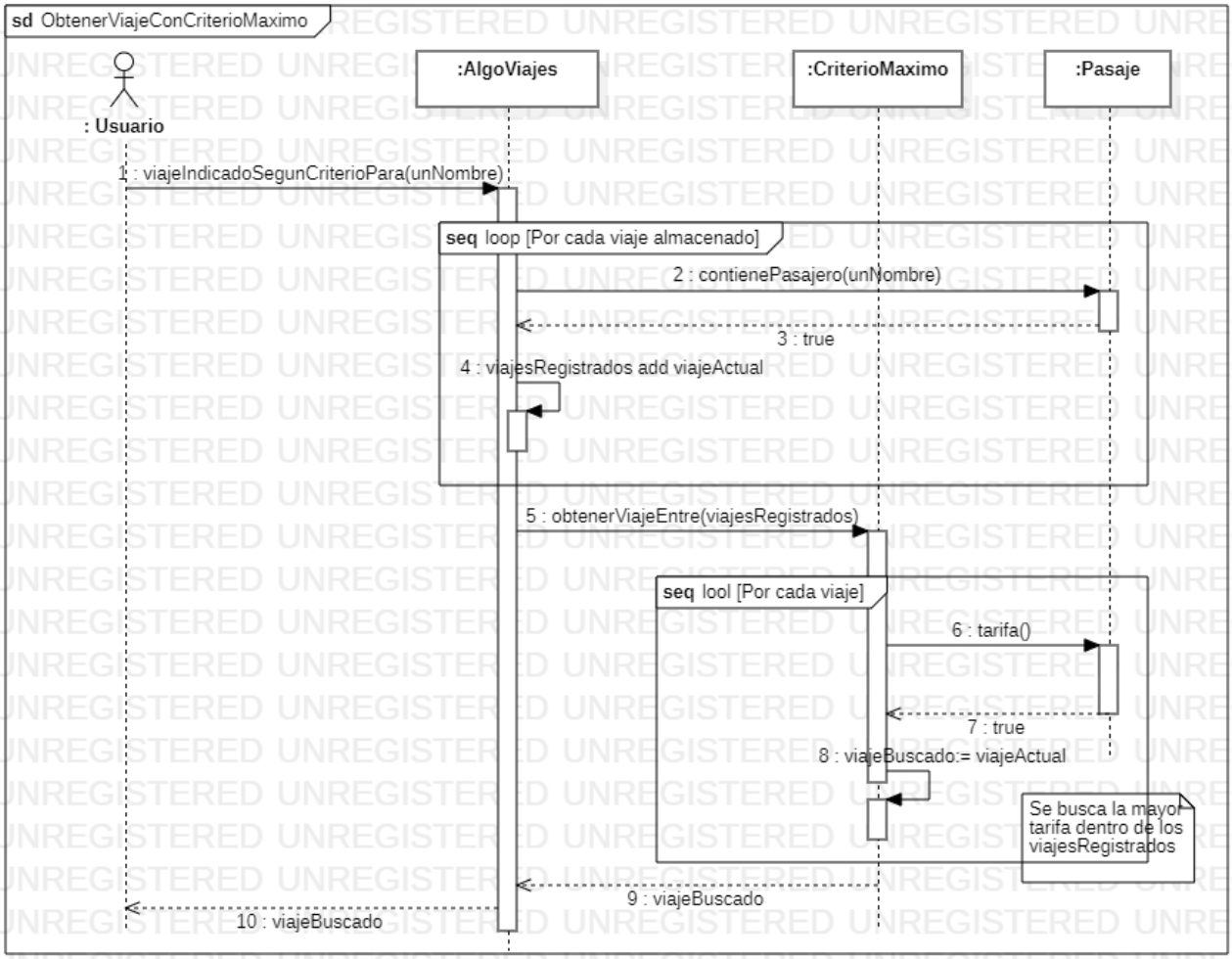

Figura 4 – Obtener un viaje con criterio MAXIMO.

En el diagrama se representa como obtener un viaje con el criterio máximo a partir de la base de datos propuesta en AlgoViajes. Se envía el mensaje viajeObtener a la instancia ya generada de AlgoViajes en donde luego se seleccionan de la misma los pasajeros que coincidan con el buscado, y luego con todos los hallados delega la responsabilidad a la instancia de CriterioMaximo para que busque el pasaje que posea la tarifa más alta.

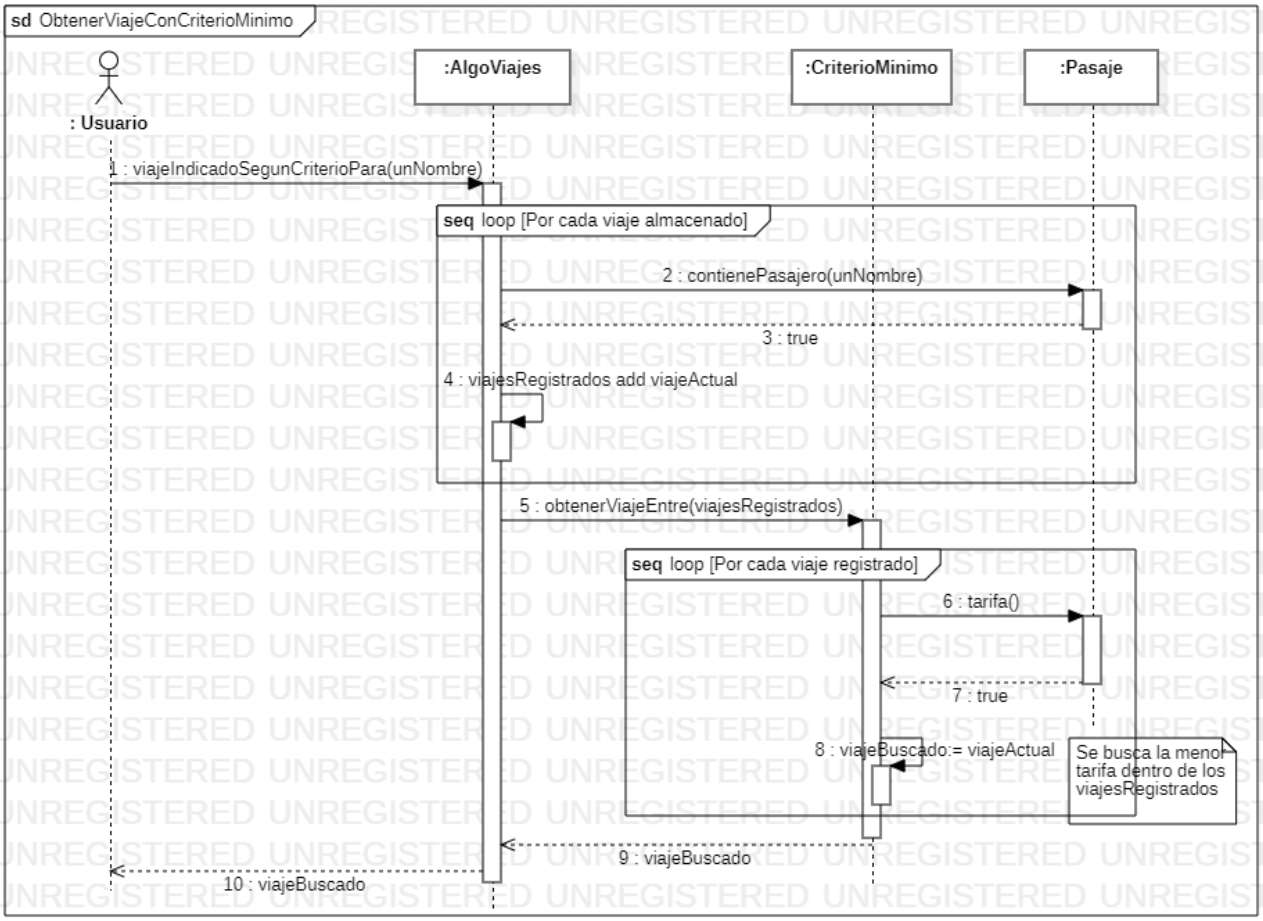

Figura 5 – Obtener un viaje con criterio MINIMO.

En el diagrama se representa de forma similar lo que ocurre en el caso en el que lo que estemos buscando sea con criterio Mínimo.

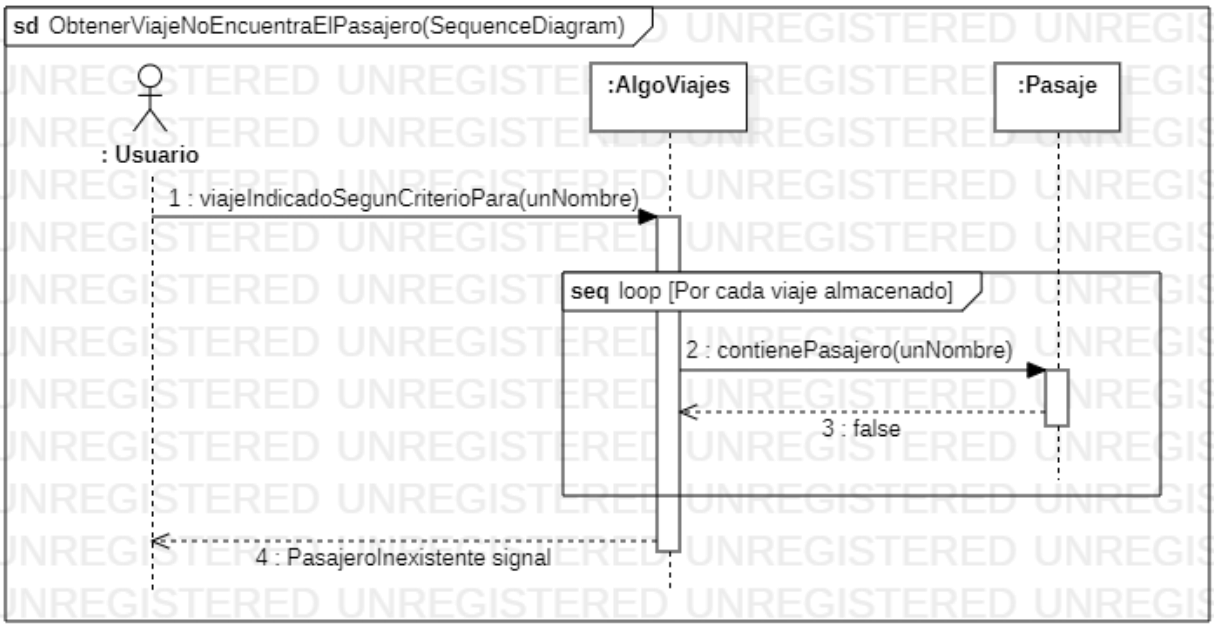

Figura 6 – Obtener un viaje de un pasajero no registrado.

En el diagrama se representa como obtener un viaje para el cual no hay un pasajero registrado. Se envía el mensaje viajeObtener a la instancia ya generada de AlgoViajes en donde luego se seleccionan de la misma los pasajeros que coincidan con el buscado, y al no coincidir ninguno con lo esperado, se genera el fallo “PasajeroInexistente”, el cual indica que no hay una reserva asociada al nombre introducido.