

# Trabajo Práctico 1 — Smalltalk

[7507/9502] Algoritmos y Programación III  
Curso 2  
Primer cuatrimestre de 2021

Alumno:	QUINTERO, Nazareno Hernán
Número de padrón:	105296
Email:	nquintero@fi.uba.ar

## Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Supuestos</b>	<b>2</b>
<b>3. Diagramas de clase</b>	<b>2</b>
<b>4. Detalles de implementación</b>	<b>10</b>
4.1. Delegación . . . . .	10
4.2. Polimorfismo . . . . .	10
4.3. Encapsulamiento . . . . .	10
<b>5. Excepciones</b>	<b>11</b>
<b>6. Diagramas de secuencia</b>	<b>11</b>

## 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 circulación y salud en Pharo utilizando los conceptos del paradigma de la orientación a objetos vistos hasta ahora en el curso.

## 2. Supuestos

La clase AlgoVid tiene la responsabilidad de delegar a otras clases complementarias los mensajes que la misma expone. La idea es que se componga de las clases Personas, Burbujas y Colegios, se debe a que cada una de ellas pudiera existir por sí mismas de ser necesario. A sí mismo, cada una de ellas permite la existencia conjuntos de Persona, conjuntos de Burbuja, y conjunto de Colegio.

Cada conjunto de ".entidades", se hace cargo de las ".entidades" que contiene y de comunicarle los distintos requerimientos a cada una.

## 3. Diagramas de clase

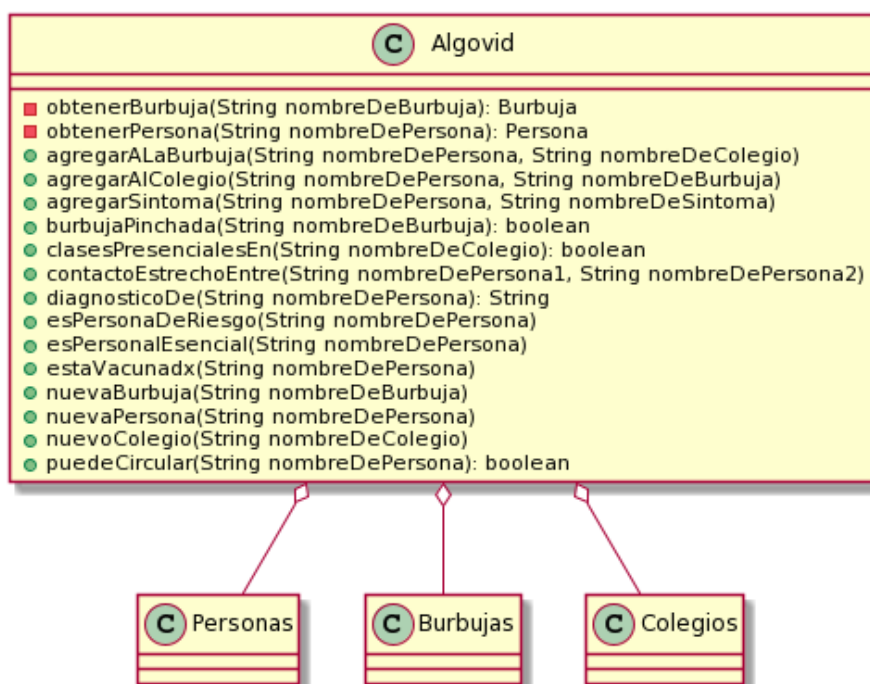


Figura 1: Diagrama de AlgoVid.

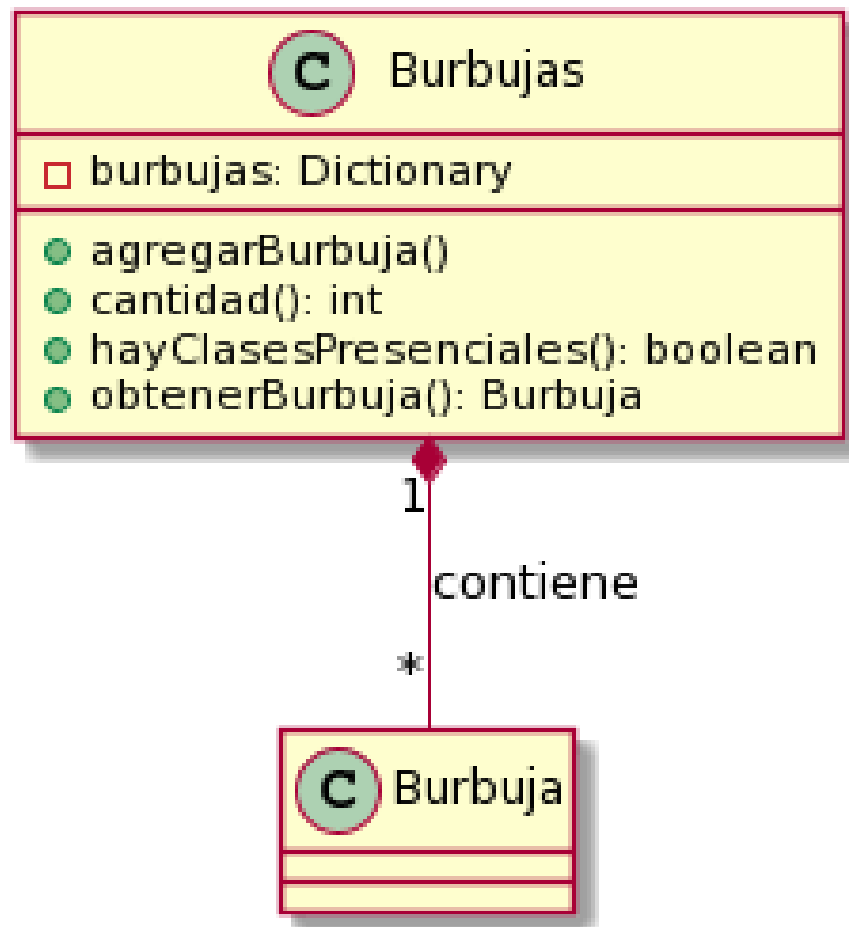


Figura 2: Diagrama de Burbujas.

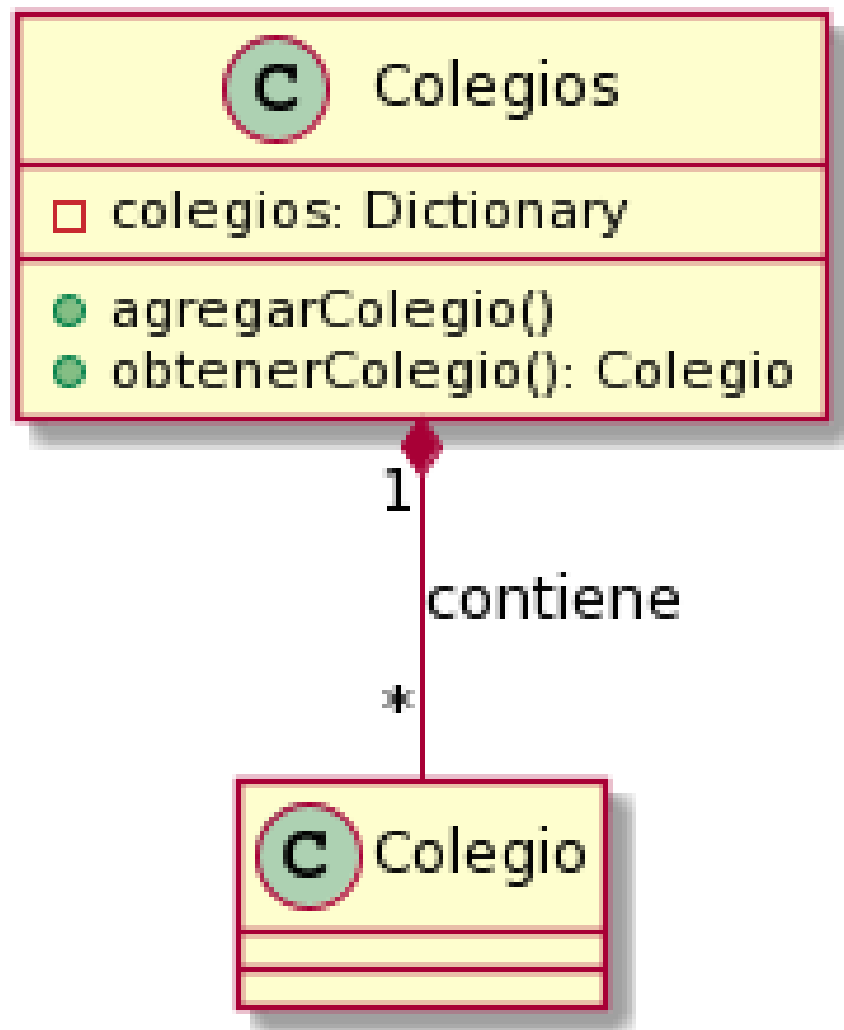


Figura 3: Diagrama de Colegios.

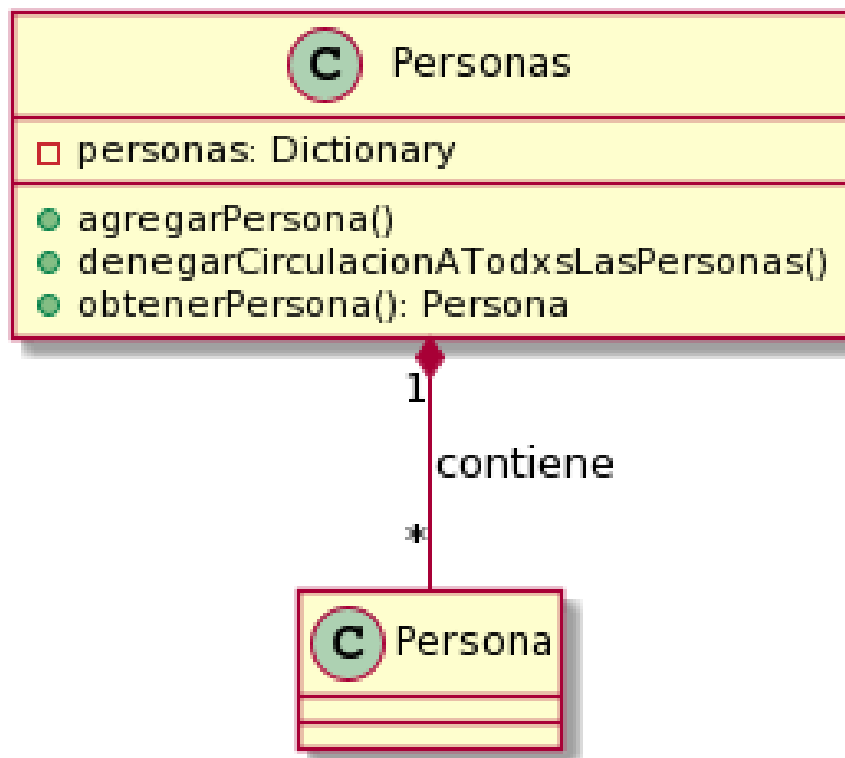


Figura 4: Diagrama de Personas.

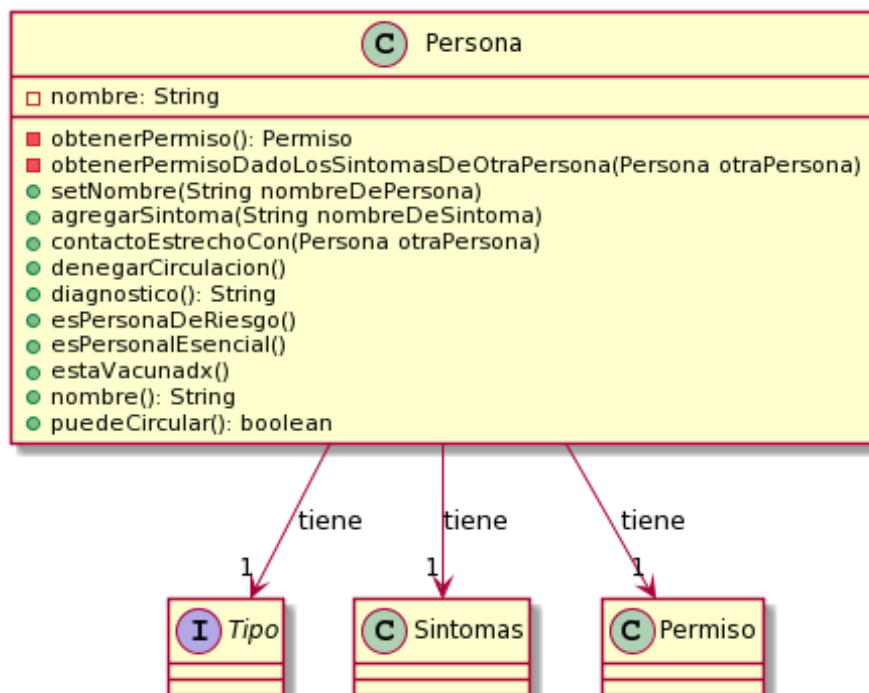


Figura 5: Diagrama de Persona.

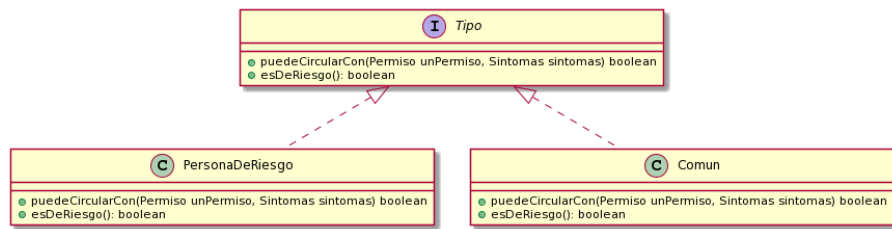


Figura 6: Diagrama de Tipo.

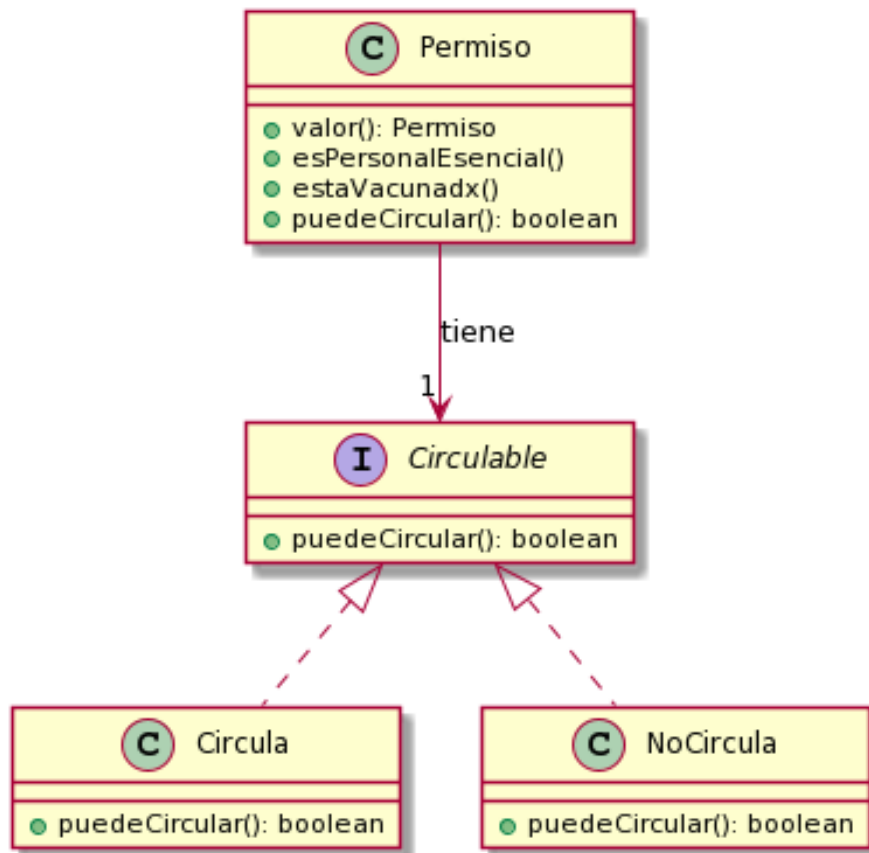


Figura 7: Diagrama de Permiso.

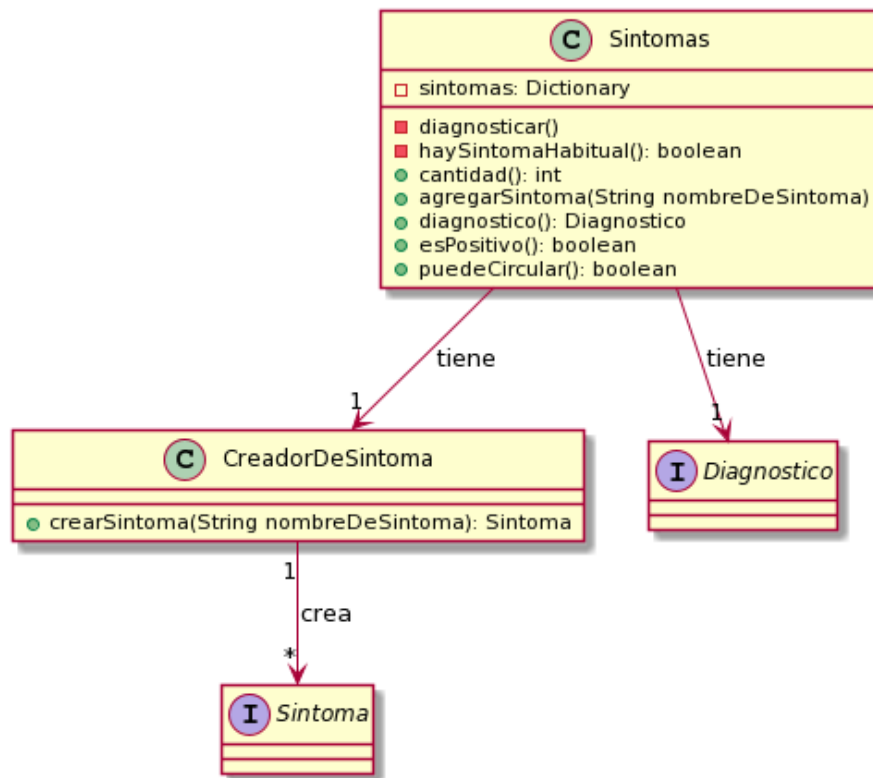


Figura 8: Diagrama de Sintomas.

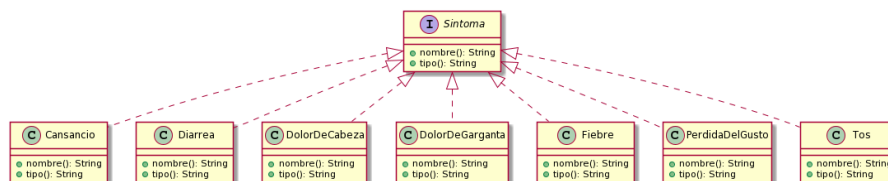


Figura 9: Diagrama de Sintoma.

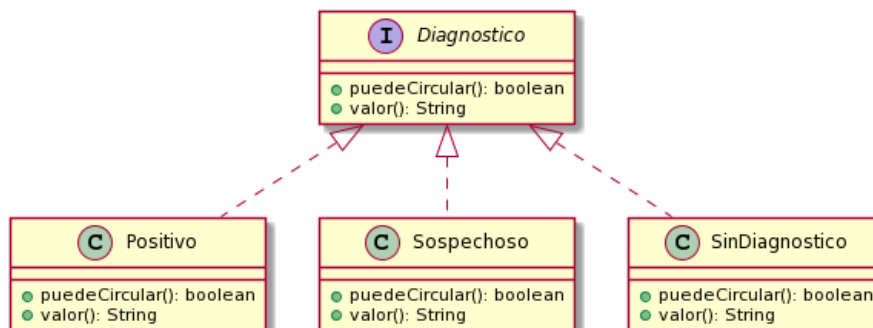


Figura 10: Diagrama de Diagnostico.



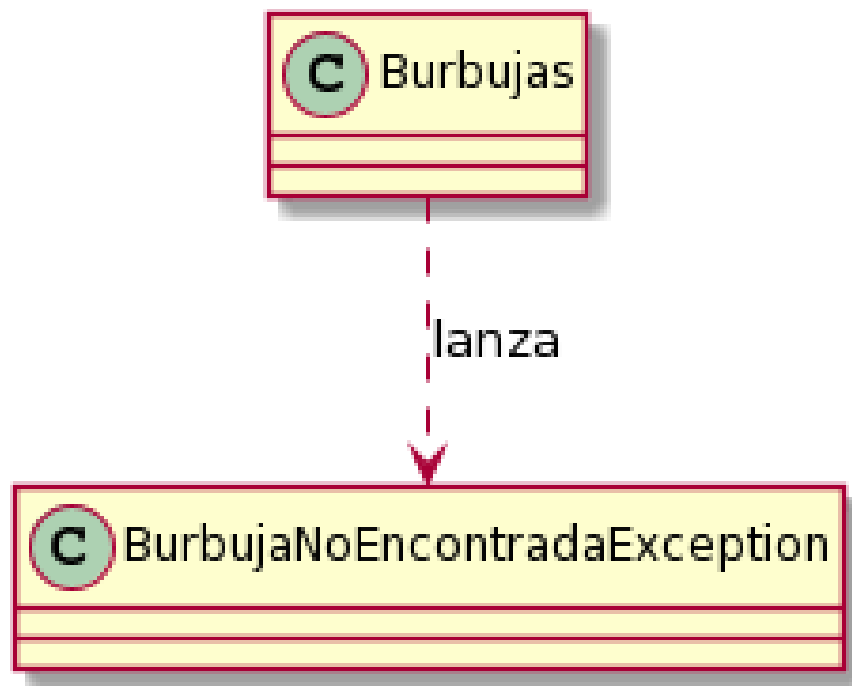


Figura 11: Excepcion lanzada por Burbujas.

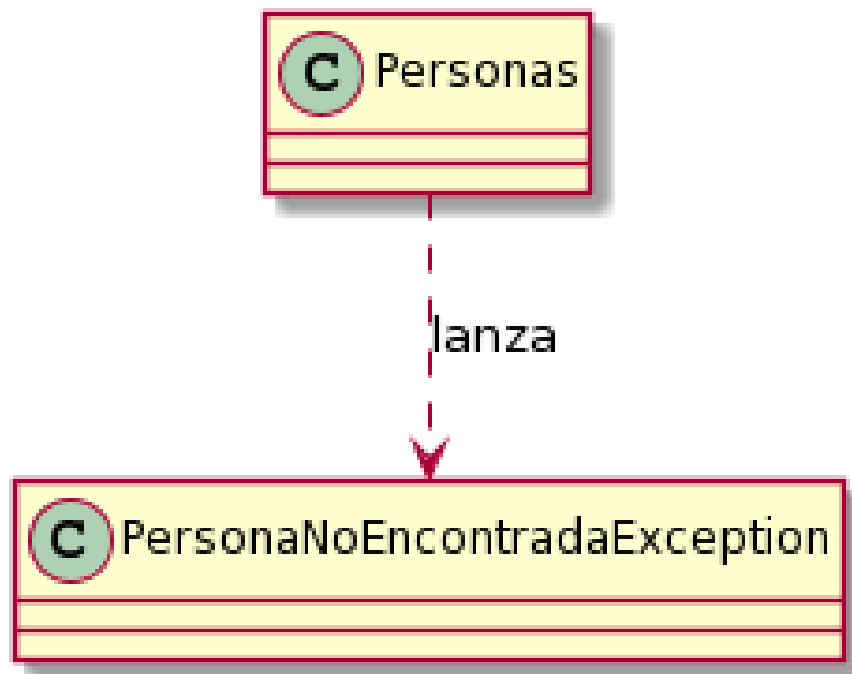


Figura 12: Excepcion lanzada por Personas.

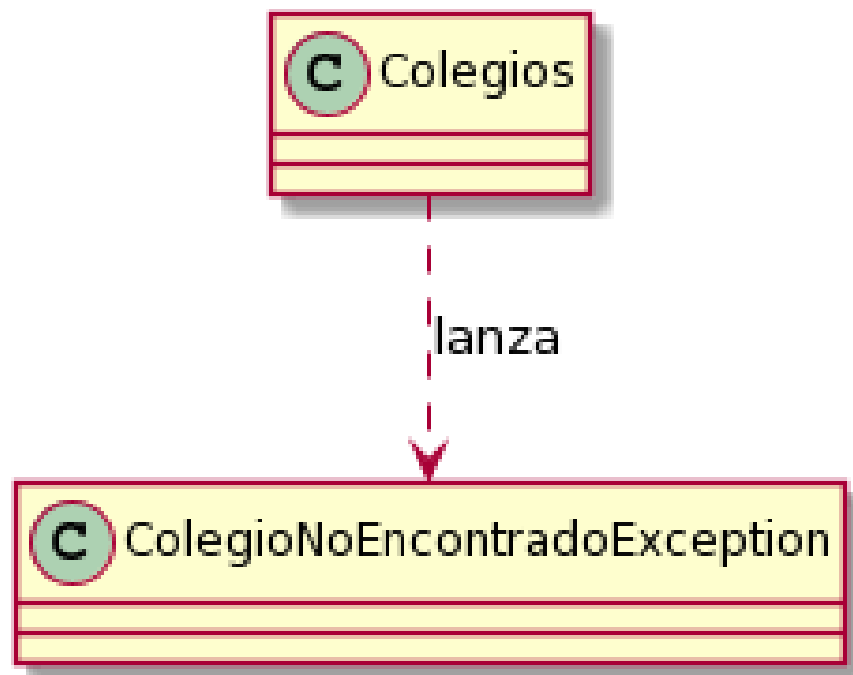


Figura 13: Excepcion lanzada por Colegios.

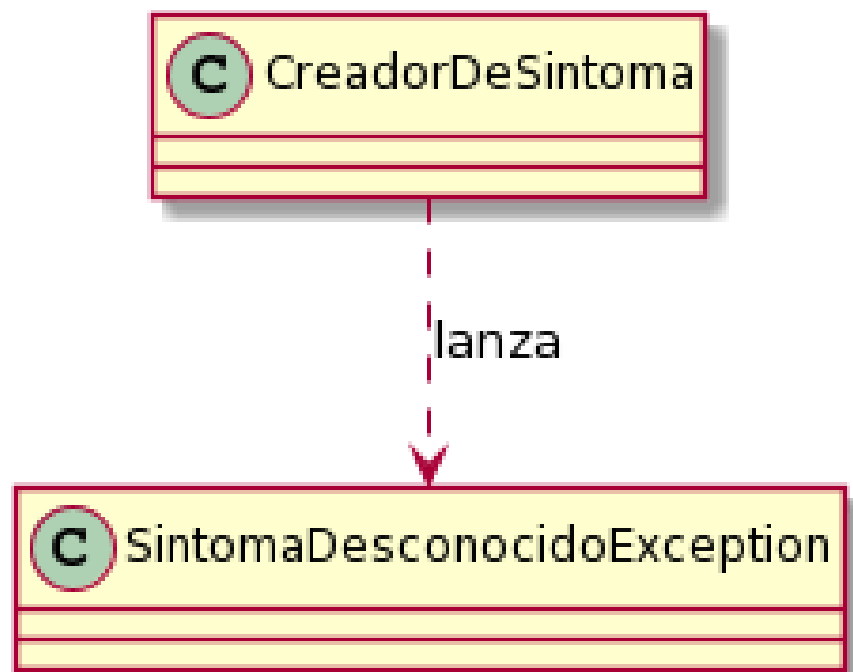


Figura 14: Excepcion lanzada por CreadorDeSintoma.

## 4. Detalles de implementación

### 4.1. Delegación

A partir de la idea de que la clase `Algovid` no tenga la responsabilidad de resolver los requerimientos (los mensajes que expone), es que se desprenden las diversas clases que conforman el sistema, favoreciendo la reutilización, y acotando el impacto que podrían ocasionar futuras modificaciones.

En el ejemplo de abajo, se puede evidenciar como la clase `Algovid` le delega a otra entidad, en este caso `'personas'` el agregado de la nueva persona:

Mensaje definido en `Algovid`:

```
nuevaPersona: unNombre  
personas agregarPersonaPorNombre: unNombre .
```

### 4.2. Polimorfismo

Si bien se presentan subclases, no fue optado el camino de la herencia en este sistema. El uso de las subclases es meramente a modo de agrupar en una especie de jerarquía a los ojos de quien mire cómo se encuentran agrupadas las clases. Es ejemplo de cómo se maneja este concepto en las clases `'Positivo'`, `'SinDiagnostico'` y `'Sospechoso'`, donde cada una de ellas cumple con la interfaz/contrato de entender los mensajes `'puedeCircular'` y `'valor'`, mensajes que cada clase definirá a su manera, dando distinto comportamiento a los objetos.

Mensaje definido en `Positivo`:

```
puedeCircular  
^false.
```

Otro mensaje definido en `Positivo`:

```
valor  
^'Positivo'
```

Mensaje definido en `Sospechoso`:

```
puedeCircular  
^false.
```

Otro mensaje definido en `Sospechoso`:

```
valor  
^'Sospechoso'
```

### 4.3. Encapsulamiento

Si bien se presentan subclases, no fue optado el camino de la herencia en este sistema. El uso de las subclases es meramente a modo de agrupar en una especie de jerarquía a los ojos de quien mire cómo se encuentran agrupadas las clases. Es ejemplo de cómo se maneja este concepto en las clases `'Positivo'`, `'SinDiagnostico'` y `'Sospechoso'`, donde cada una de ellas cumple con la interfaz/contrato de entender los mensajes `'puedeCircular'` y `'valor'`, mensajes que cada clase definirá a su manera, dando distinto comportamiento a los objetos.

## 5. Excepciones

**BurbujaNoEncontradaException** Esta excepción surge en el caso en que se intenta obtener una Burbuja dado su nombre, y la misma no se encuentre en el conjunto de burbujas existentes.

**PersonaNoEncontradaException** Esta excepción surge en el caso en que se intenta obtener una Persona dado su nombre, y la misma no se encuentre en el conjunto de personas existentes.

**ColegioNoEncontradoException** Esta excepción surge en el caso en que se intenta obtener un Colegio dado su nombre, y el mismo no se encuentra en el conjunto de colegios existentes.

**SintomaDesconocidoException** Esta excepción es lanzada por el 'CreadorDeSintoma' cuando se intenta crear un Sintoma a partir del nombre del mismo y este no se encuentra dentro del conjunto de síntomas que se pueden crear.

## 6. Diagramas de secuencia

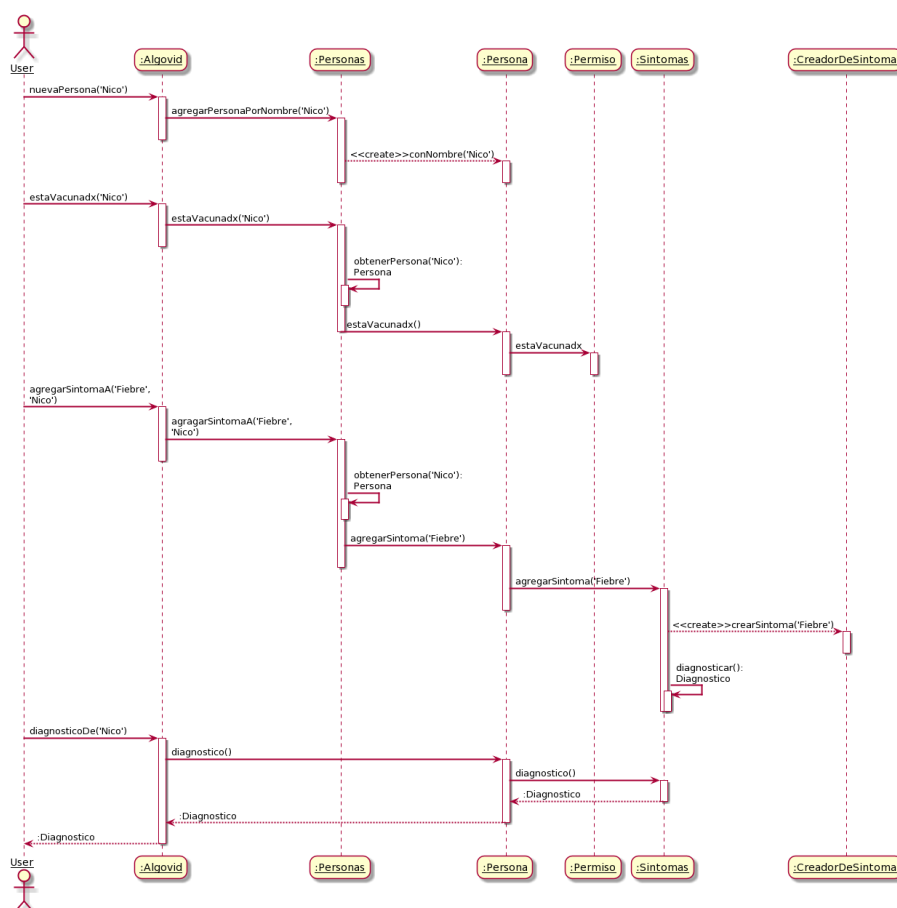


Figura 15: Diagrama de secuencia basado en el test 03 de Algovid entregado por la cátedra.

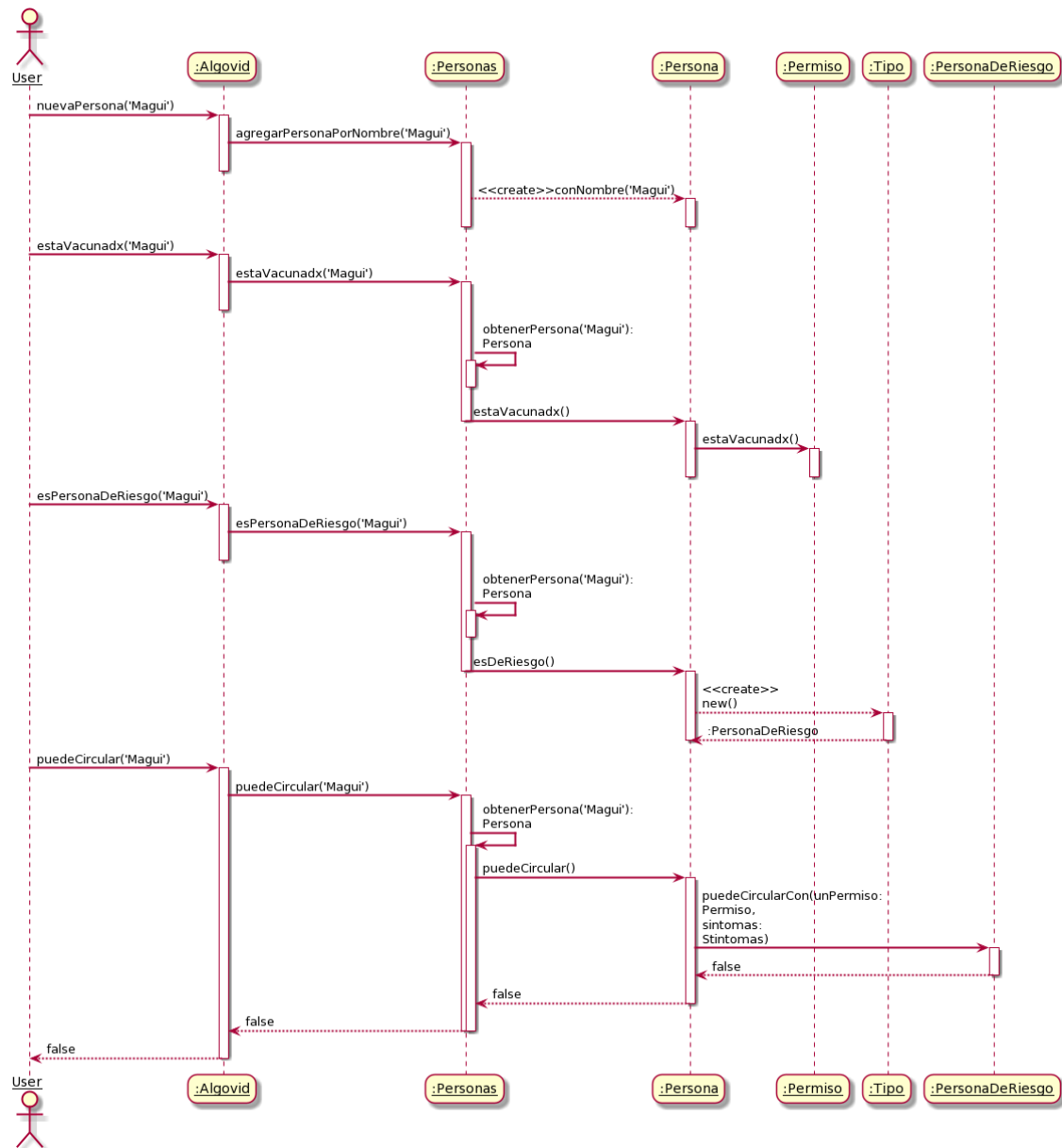


Figura 16: Diagrama de secuencia basado en el test 10 de Algovid entregado por la cátedra.