PROGRAMACIÓN ORIENTADA A OBJETOS

TEMA 1.1 Introducción al paradigma de la programación orientada a objetos.

ACTIVIDAD 1.1

GRUPO: 2g2A

NOMBRE DE LOS ALUMNOS:

- ENRIQUEZ MONTALVO RIGOBERTO
- GIL RODRIGUEZ JONATHAN
- GUTIERREZ CRUZ ÁNGEL DE JESÚS
- MORALES VAZQUEZ JUAN DIEGO
- SOSA FIGUEROA BENJAMÍN DE JESÚS

PROFESORA: PATRICIA QUITL GONZALEZ

FECHA DE ENTREGA: 02/03/2023

HORA DE CLASE: 11:00-12:00

PROBLEMA 1.

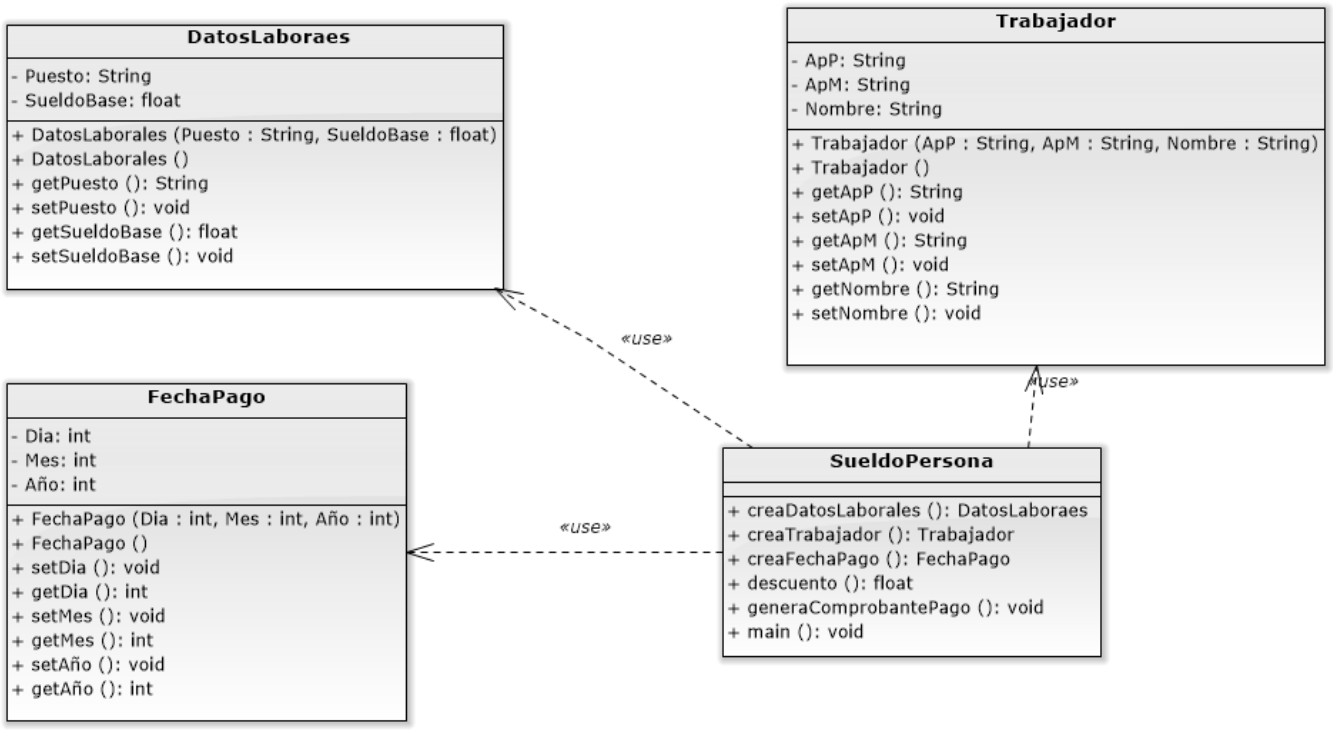
Definición del problema

Elaborar un programa que permita mostrar el comprobante de pago de un trabajador, el cual muestre la siguiente información:

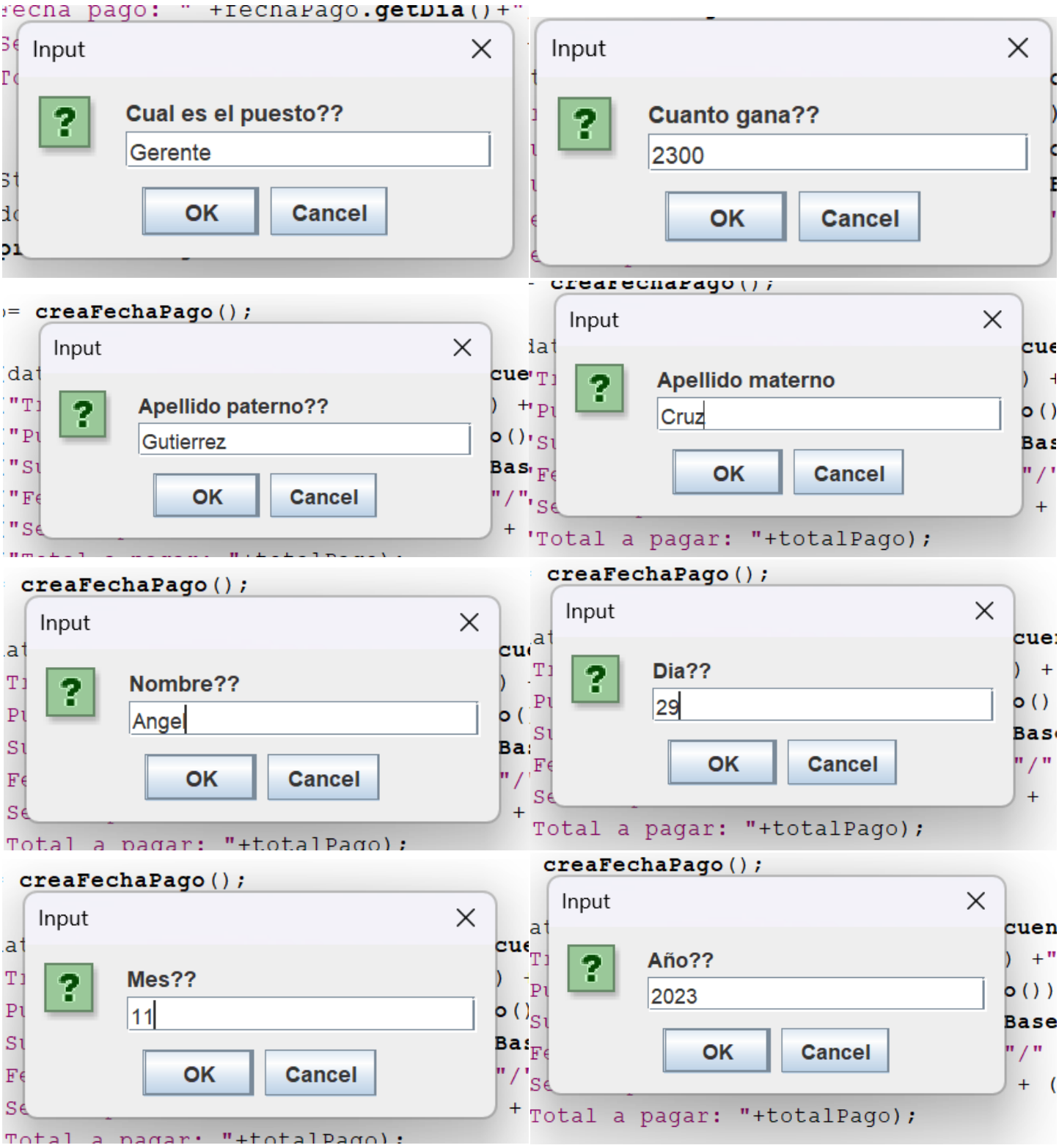
Fecha de pago Nombre del trabajador: ap am nombre Puesto: sssssss Sueldo base: $xxxxx.xx Descuento: $xxxx.xx Total a pagar: $xxxxx.xx	El descuento se calculará de acuerdo con la siguiente información:

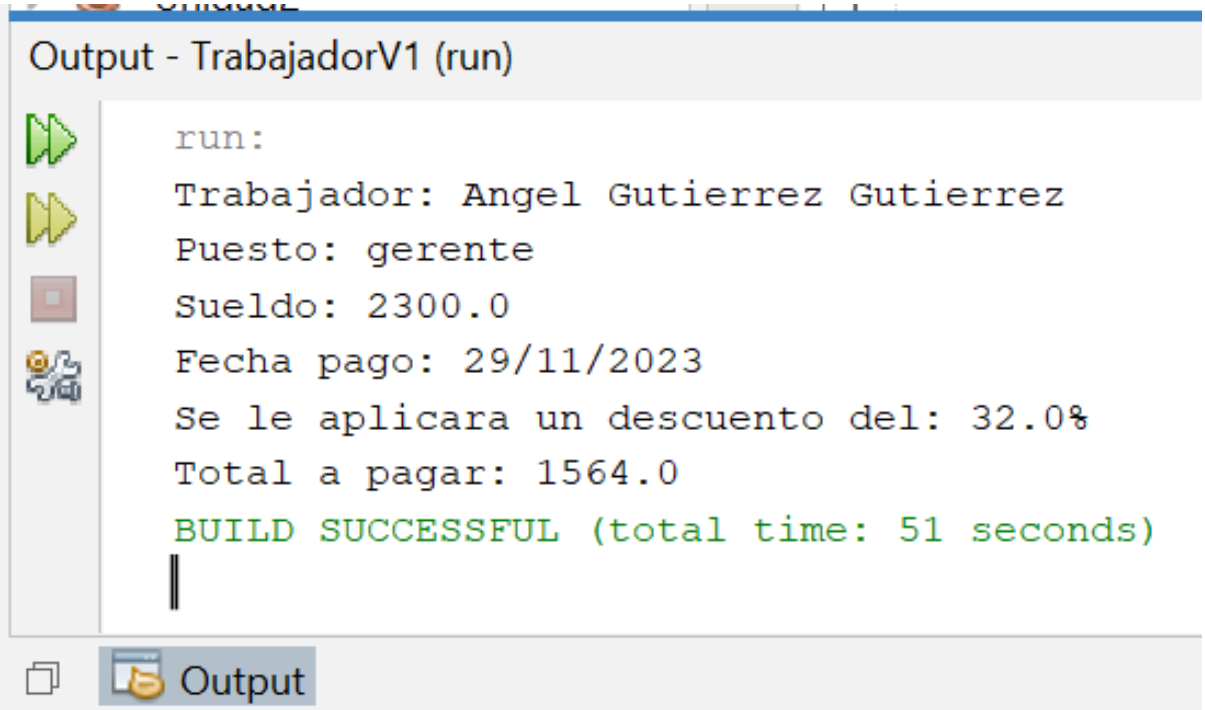
Puesto	Descuento
Gerente	32%
Subgerente	25%
Supervisor	20%
Operativo	16%

Diseño de la solución mediante diagrama de clases (UML)



Casos de prueba

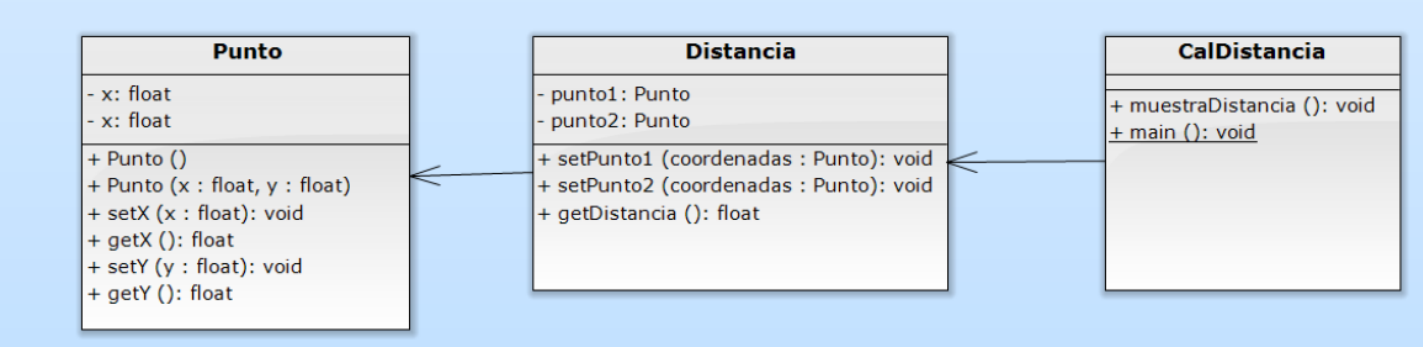




PROBLEMA 2.

Definición del problema

Calcular la distancia entre dos puntos, consultar el siguiente diseño:



En la clase Distancia el método getDistancia deberá contener la fórmula del cálculo de la distancia.

entre el punto1 y el punto2 que devuelva un valor de retorno.

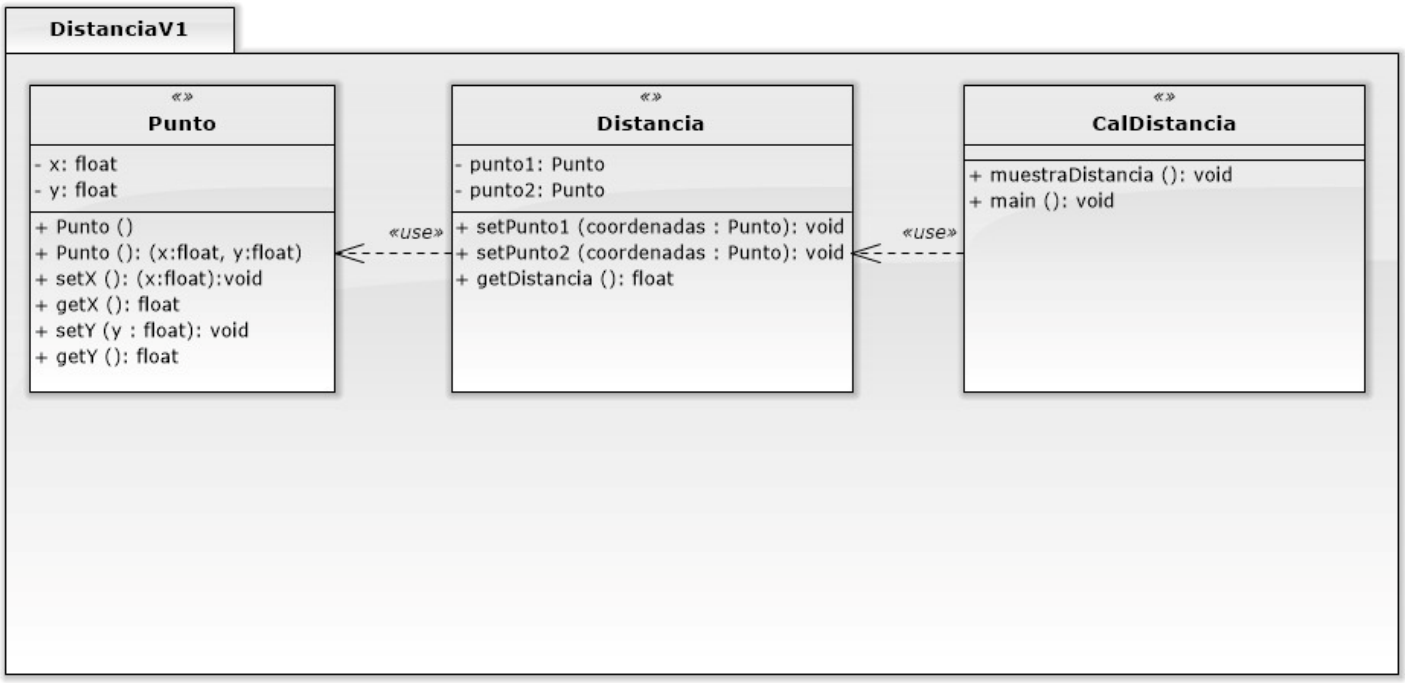
En el método muestraDistancia:

- Crear un objeto puntoA de la clase Punto.
- Crear otro objeto puntoB de la clase Punto.
- Crear un objeto distancia de la clase Distancia.

- Enviar los objetos puntoA y puntoB a través de los métodos setPunto1 y setPunto2.
- Mostrar la distancia a través del método getDistancia.

En la clase CalDistancia main invocará al método muestraDistancia.

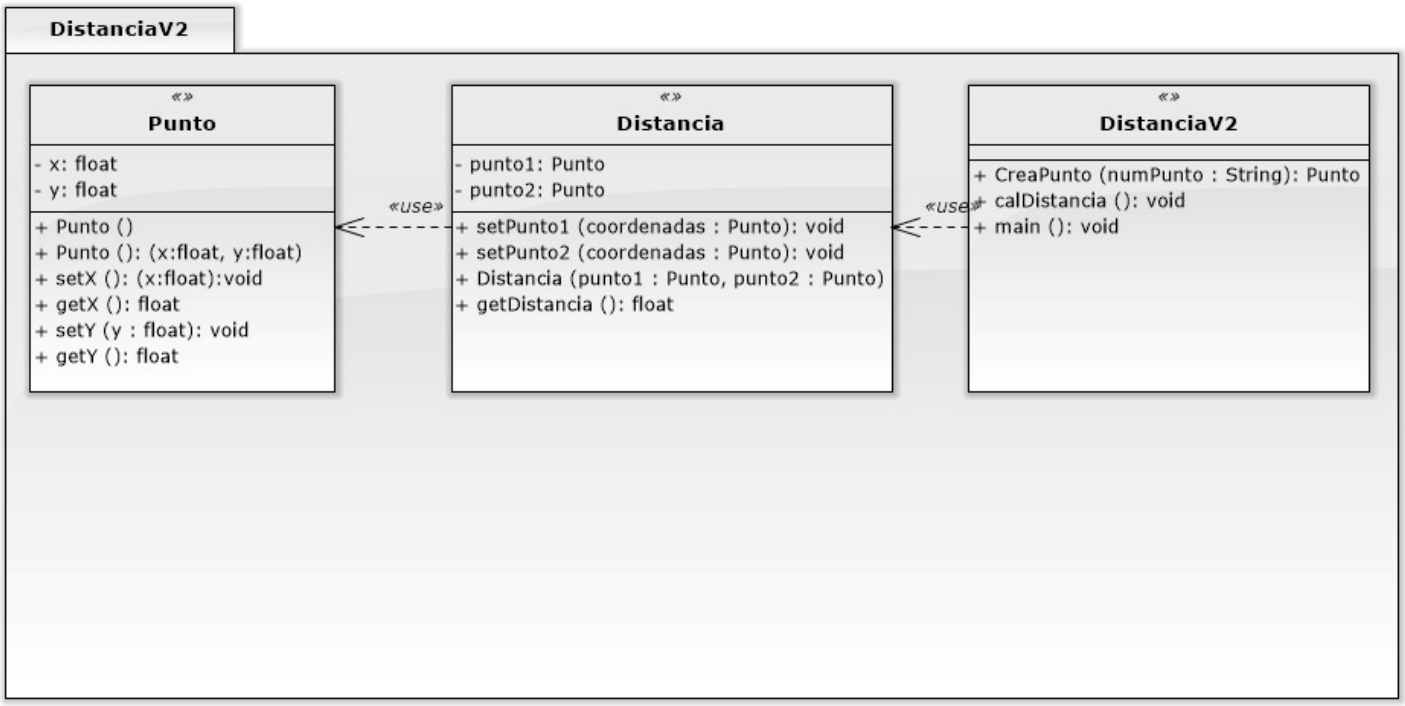
Diseño de la solución mediante diagrama de clases (UML)



Casos de prueba

PROBLEMA 2 VERSIÓN 2.

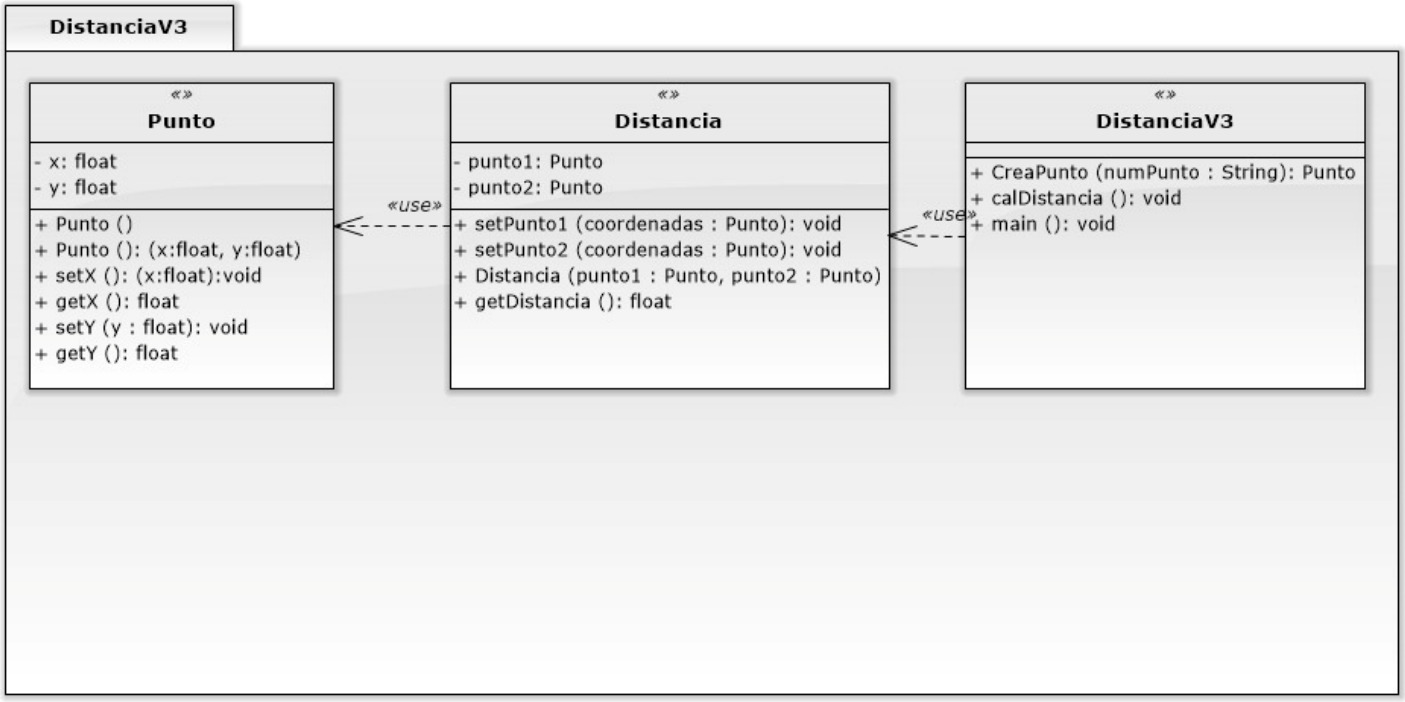
Diseño de la solución mediante diagrama de clases (UML)



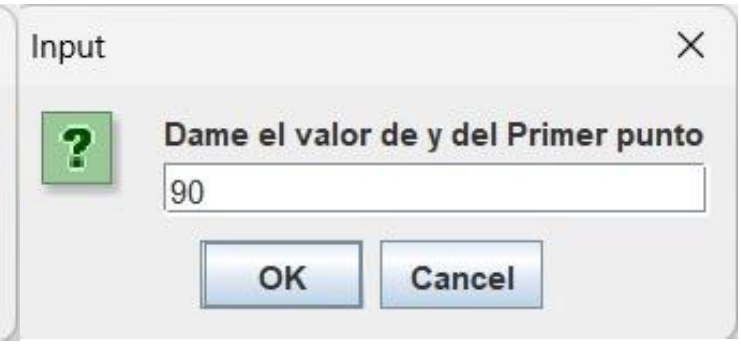
Casos de prueba

PROBLEMA 2 VERSIÓN 3.

Diseño de la solución mediante diagrama de clases (UML)



Casos de prueba



PROBLEMA 3.

Definición del problema

Un centro cultural se dedica al préstamo de dos tipos de materiales: Revista y Libros.

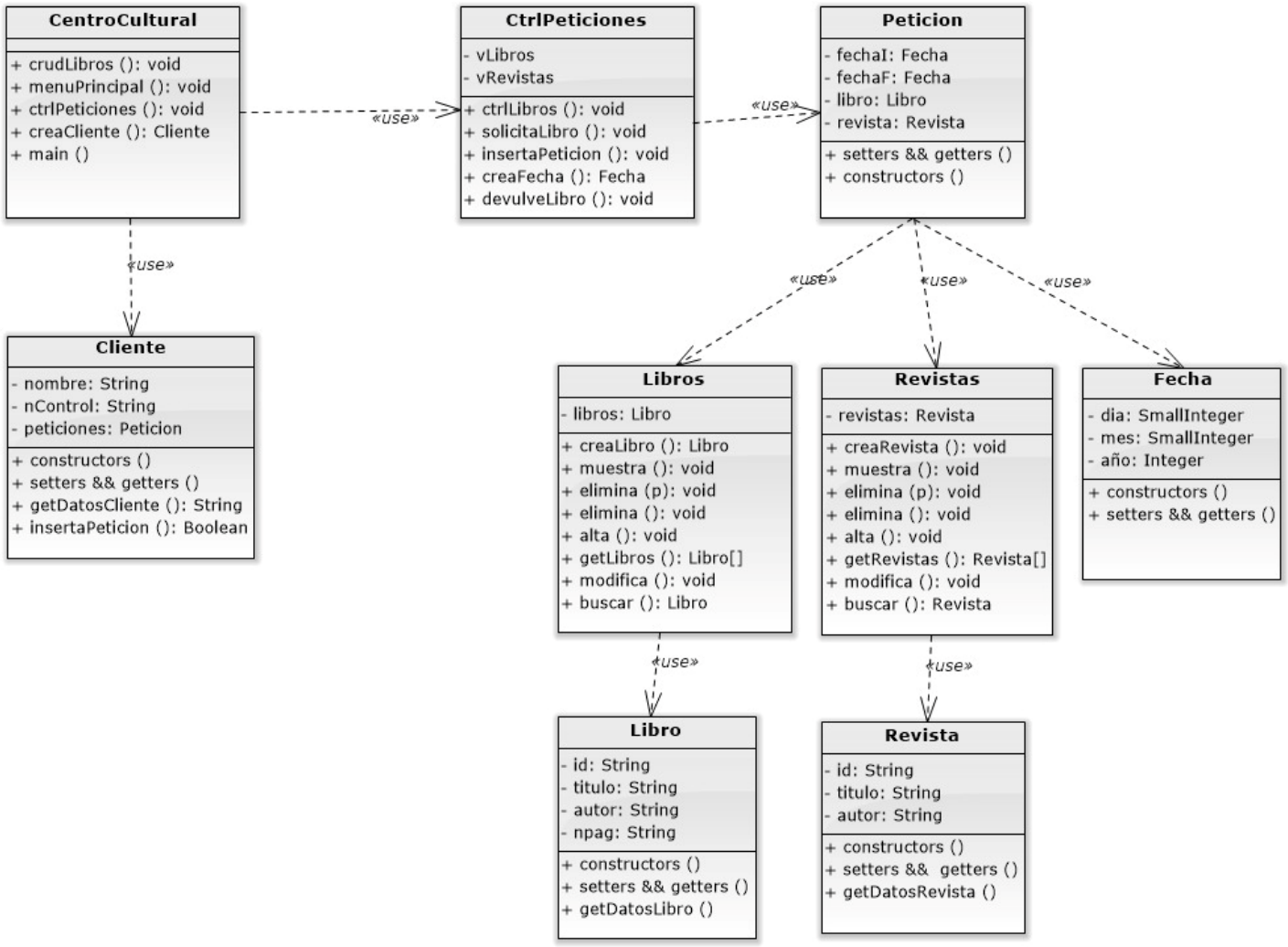
Para los dos se guarda información general, como su código identificativo, el titulo y el autor. En el caso de los libros, almacenamos también su número de páginas, y para los discos el nombre de la discografía.

Al centro cultural acuden una serie de clientes (de los que se pregunta su NC y Nombre), que realizan una serie de peticiones de revistas o libros (como mucho hasta 5 peticiones). Para cada peticion se guarda la fecha de inicio y su fin de préstamo.

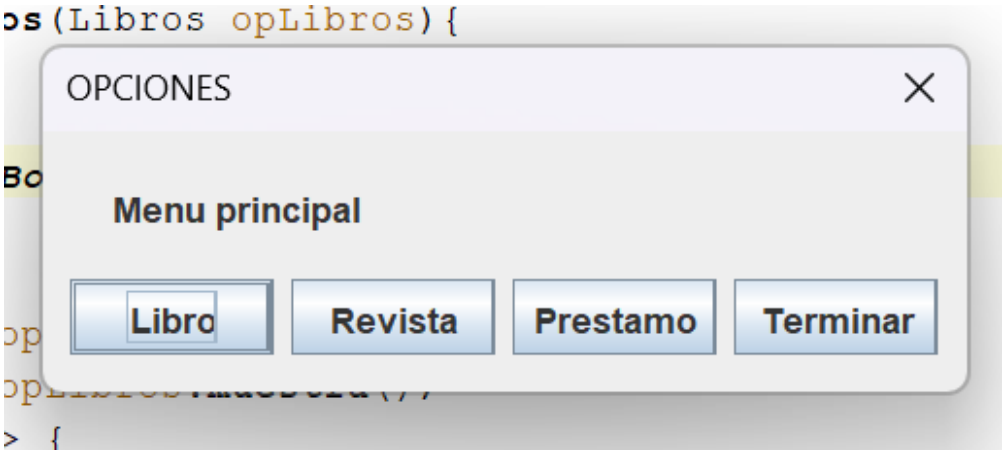
Diseñar y construir las clases apropiadas para el supuesto planteado anteriormente.

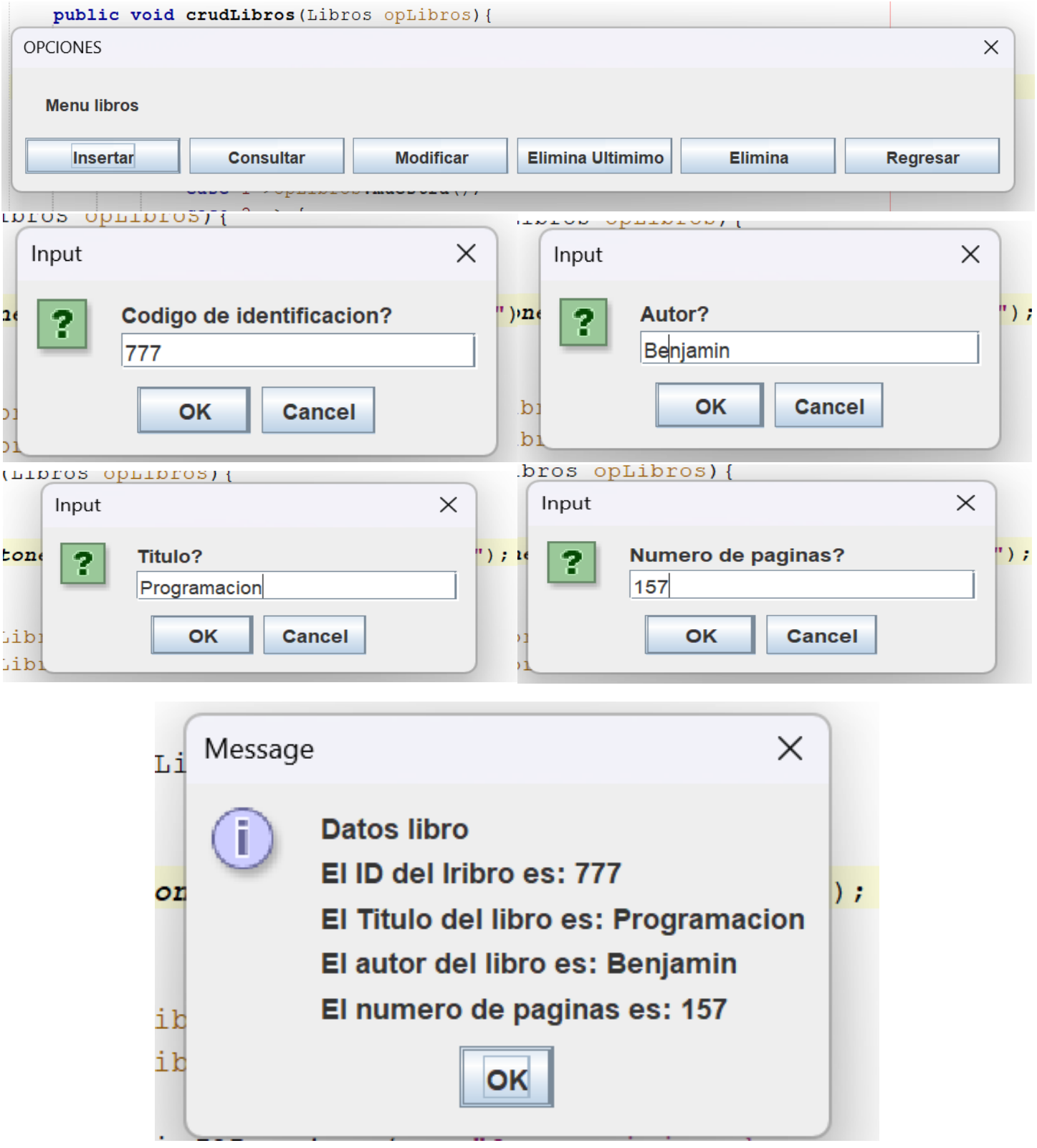
Crear todas estas clases en un paquete préstamos.

Diseño de la solución mediante diagrama de clases (UML)



Casos de prueba



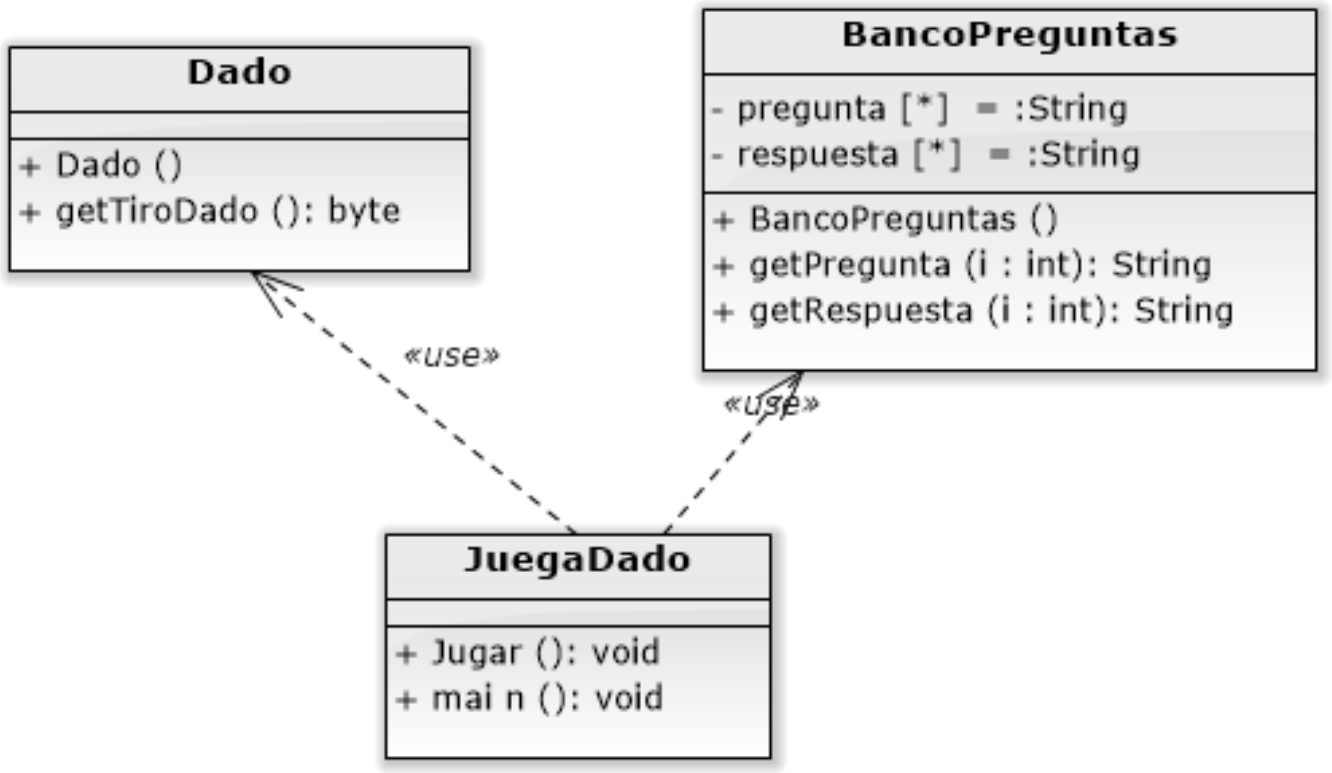

public void crudLibros(Libros opLibros){
OPCIONES
Menu libros
Insertar
Consultar
Modificar
Elimina Ultimimo
Elimina
Regresar
Input
Codigo de identificacion?
777
OK
Cancel
Input
Autor?
Benjamin
OK
Cancel
Input
Titulo?
Programacion
OK
Cancel
Input
Numero de paginas?
157
OK
Cancel
Message
Datos libro
El ID del lribro es: 777
El Titulo del libro es: Programacion
El autor del libro es: Benjamin
El numero de paginas es: 157
OK

PROBLEMA 4.

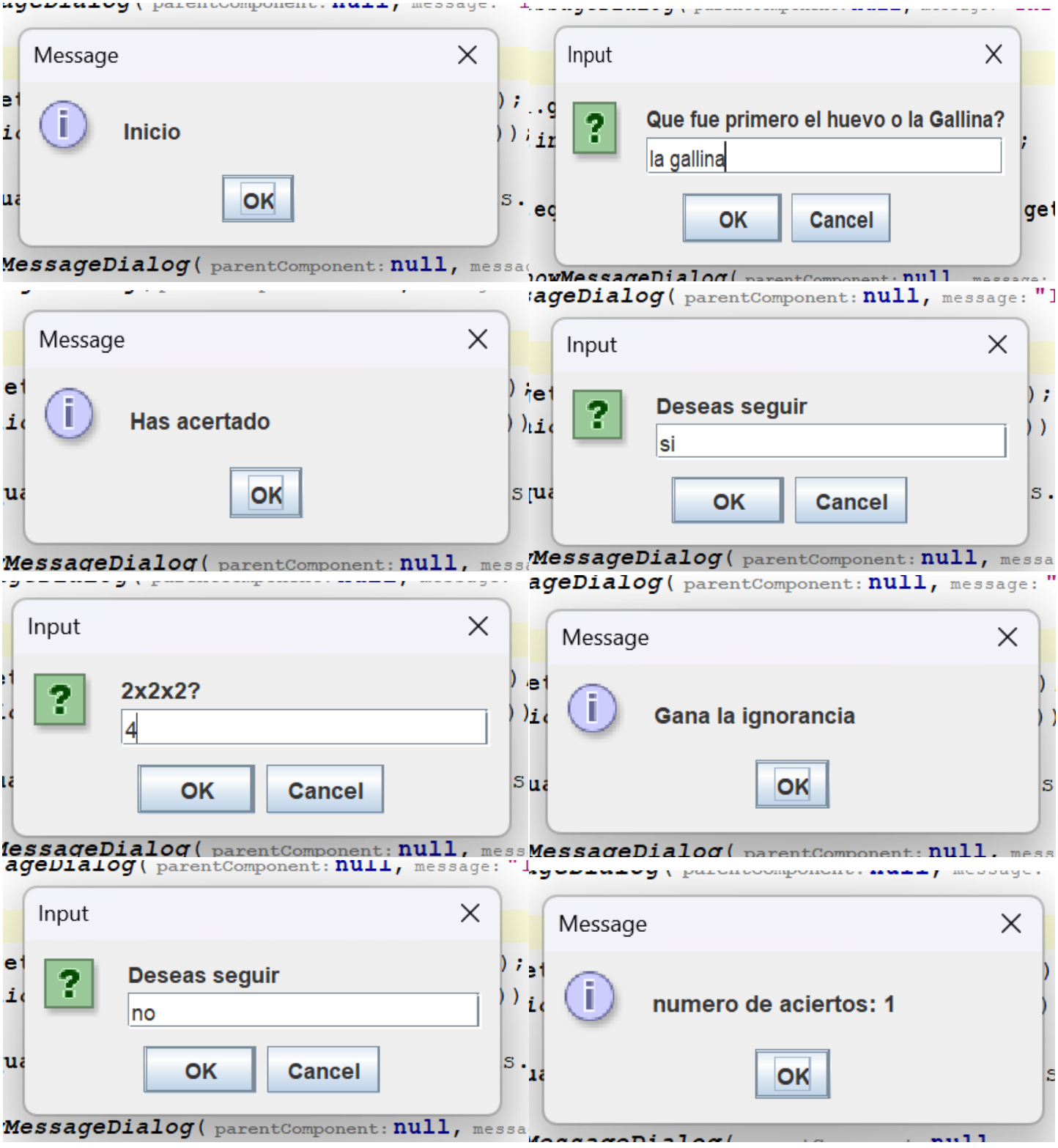
Definición del problema

Elaborar un programa en el paradigma orientado a objetos que permita mostrar un juego de preguntas a través de un numero al azar de 2 dados.

Diseño de la solución mediante diagrama de clases (UML)



Casos de prueba



Conclusiones

Enríquez Montalvo Rigoberto: Durante esta unidad he comprendido el uso y significado del paradigma orientado a objetos, cuyo significado está basado en el concepto de abstracción (resaltar los atributos o características de un objeto en especial); en sí, este patrón de programación consiste en dividir el programa en subprogramas con el fin de simplificar las líneas de código que uno ocupa en una sola clase para hacer todo el proyecto. Estos subprogramas contienen atributos y constructores, así como métodos de tipo getters y setters que por medio de parámetros permiten su invocación desde otras clases (no todos los métodos ocupan parámetros) para acceder a sus datos u operaciones que guardan. Gracias al paradigma orientado a objetos, el programa en general queda en mejor orden y permite detectar errores de una manera más específica y clara.

Gil Rodriguez Jonathan: En los temas y actividades vistos y realizados en clase he aprendido que la programación orientada a objetos no es más que una forma distinta de programar, la cual se orienta a diseñar la solución al problema creando clases las cuales puedan tener atributos y métodos, esto con el fin de darle una forma un poco más parecida a la realidad, también aprendí que en POO se utilizan también los métodos constructores, estos nos ayudan a construir nuestro objeto con los atributos que debe llevar, los métodos getters y setters, nos ayudan a capturar datos y mostrarlos de métodos encapsulados.

Gutierrez Cruz Ángel de Jesus: En esta unidad he aprendido que la programación orientada a objetos (POO) es una forma diferente de programación, en la que se generan diferentes clases y se entrelazan entre ellas a través del paso de parámetros y las invocaciones, esto con el fin de poder tener líneas de código mejor organizadas, en las que, si surge un error este mismo se pueda localizar más rápido, todo esto acompañado de una nueva forma de guardar y pedir datos (setter y getter).

Morales Vázquez Juan Diego: Estudiando programación utilizando el paradigma orientado a objetos, en el tema de abstracción y encapsulamiento, llego a la conclusión de que es una nueva forma de ver, analizar y resolver problemas informáticos, un método que hace ver la importancia de la información y como nosotros los programadores la usamos para crear algoritmos basados en nuestra vida diaria, sin duda es una forma de programar que abre muchas

puertas para la creación de programas enfocados a un concepto más amplio de realidad y así mejorar nuestro desempeño programando.

Sosa Figueroa Benjamín de Jesus: Tras venir sin conocimiento alguno respecto a lo que engloba la programación orientada a objetos, puedo decir que se una nueva forma de resolver un problema, con el paradigma orientado a objetos, he comprendido que en este paradigma lo principal es dividir/separar el cogido para mejorar el rendimiento que se tiene al programar, de esta forma los errores pueden ser encontrados fácilmente. Se utiliza un proceso de abstracción donde se lleva un objeto del mundo a código, mediante la aplicación de constructores, atributos que tiene el objeto y sus respectivos métodos, esto se plasma en los diferentes ejercicios realizados durante esta unidad, desarrollando una nueva forma de programar y una dificultad diferente, ya que en muchas ocasiones el paradigma procedural parece ser el más fácil de realizar en comparación con el orientado a objetos, solo que en términos de optimización en un código muy grande es mucho más efectivo el fragmentar el código en subprogramas.