Guía 1 programación orientada a objetos

Velandia Castellanos Oscar Andrés

Universidad EAN

Desarrollo de Software

Dilsa Enith Triana Martinez

19 de agosto de 2024

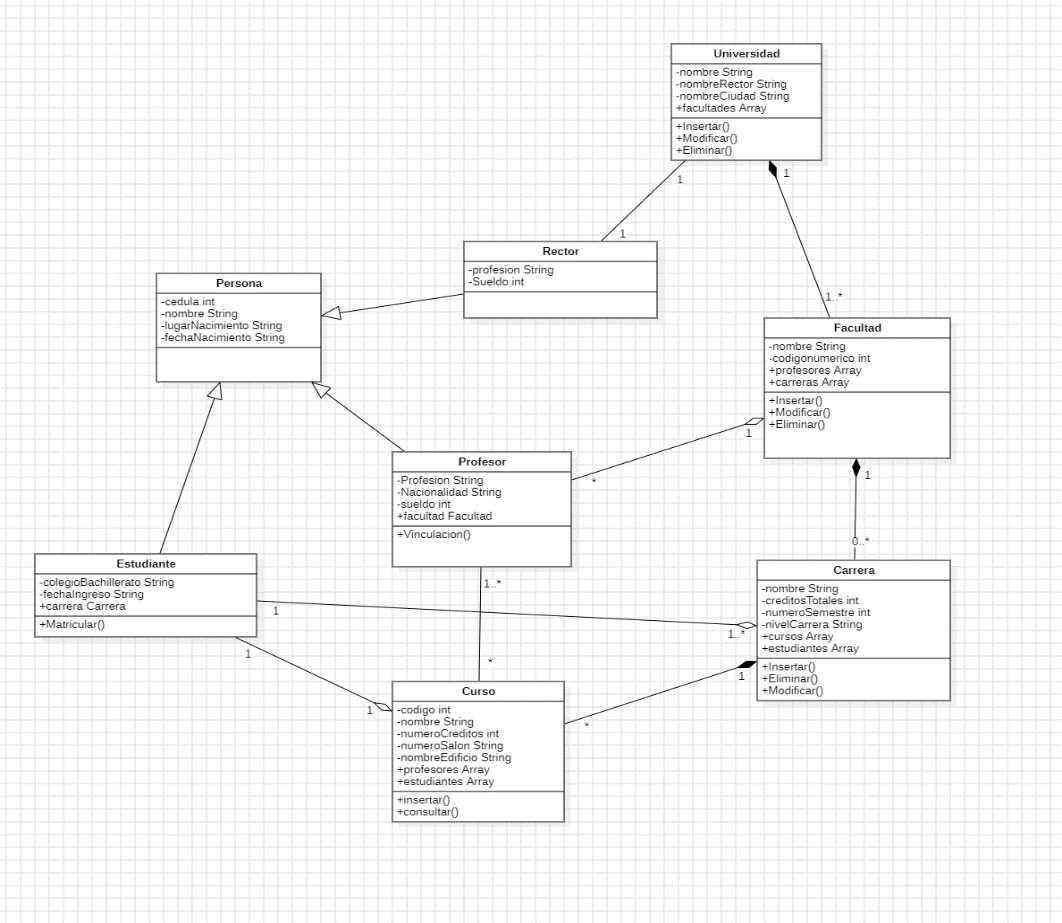
**LINK actividad 1, 2, 3**

<https://github.com/elgordowsky/GUIA1Desarrollo-de-Software.git>

**Actividad 1**

**Análisis**

El entregable de esta actividad consta de un programa que gestiona diferentes datos relevantes dentro del ecosistema de una universidad, este ha sido diseñado usando la herramienta StarUML para poder establecer cual es la estructura adecuada, destacando también la importancia del desarrollo en paradigma como lo es programación orientada a objetos.



Este UML consta de ocho clases sin contar con la clase Main que nos sirven para la estructuración del proyecto, algunos conceptos a tener en cuenta son los siguientes.

* La herencia de la clase persona, hacia las clases que identifica como sujeto.
* En el diseño se tiene en cuenta cual va ser el encapsulamiento de los atributos que estamos manipulando.
* La composición que está dada desde la clase Universidad pasando por las clases Facultad y Carrera.
* La agregación de estudiante a las clases Curso y carrera.
* También están dadas las relaciones por la cantidad de interacciones entre las diferentes clases participantes estableciendo las relaciones (1 a 0, 1 a 1, \* a 1, \* a 0, etc..).

A computer screen shot of a computer

Description automatically generated

Para la generación del código en StarUML fue necesario instalar un plugin de java y seleccionando el proyecto generaba una estructura en código que tiene en cuenta el diseño del proyecto que estamos trabajando.

A screenshot of a computer program

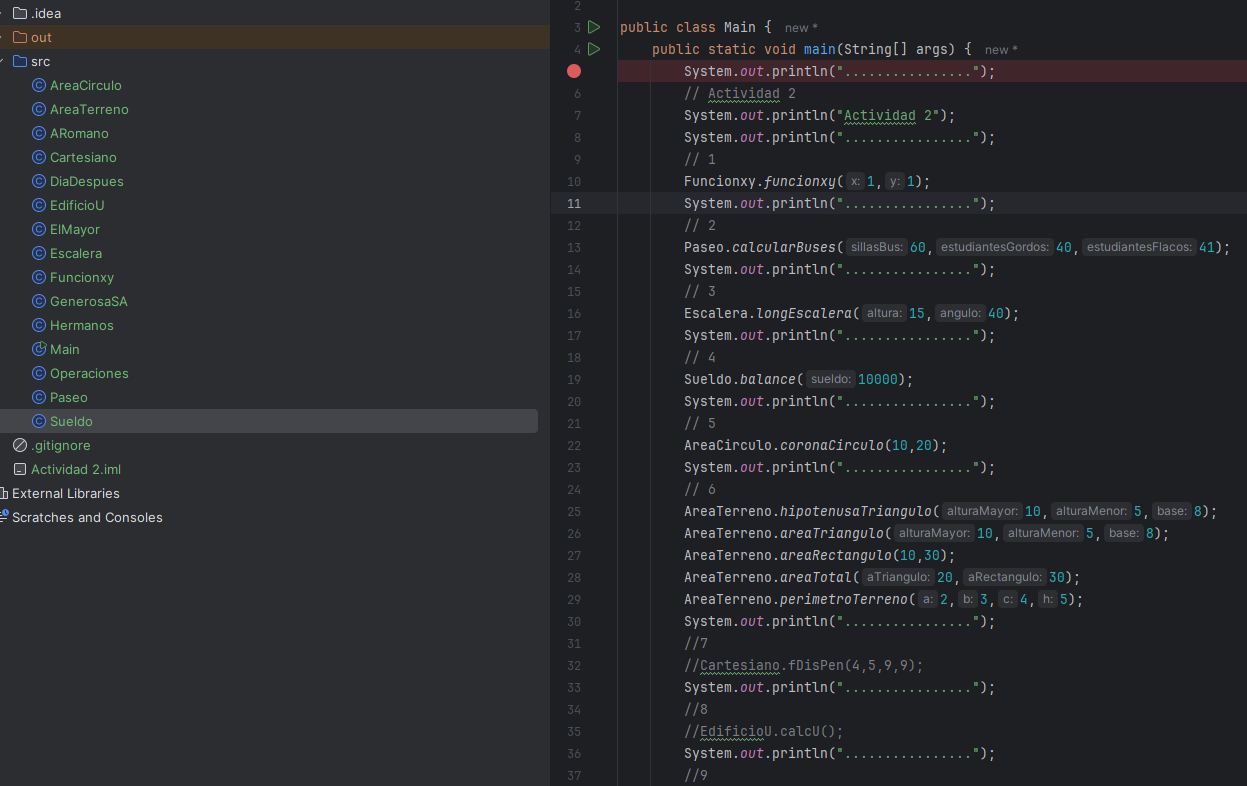
Description automatically generated

Una vez descargada la estructura es necesario implementar los métodos necesarios para el funcionamiento del programa; también es necesario anexar los métodos constructores y respectivos getters y setters de cada una de las variables de las clases.

**Actividad 2**

Los programas de la actividad 2, han sido añadidos al repositorio Github con los siguientes nombres.

1. Funcionxy
2. Paseo
3. Escalera
4. Sueldo
5. ÁreaCirculo
6. AreaTerreno
7. Cartesiano
8. EdificioU
9. ElMayor
10. Operaciones
11. ARomano
12. DiaDespues
13. GenerosaSA
14. Hermanos
15. Main



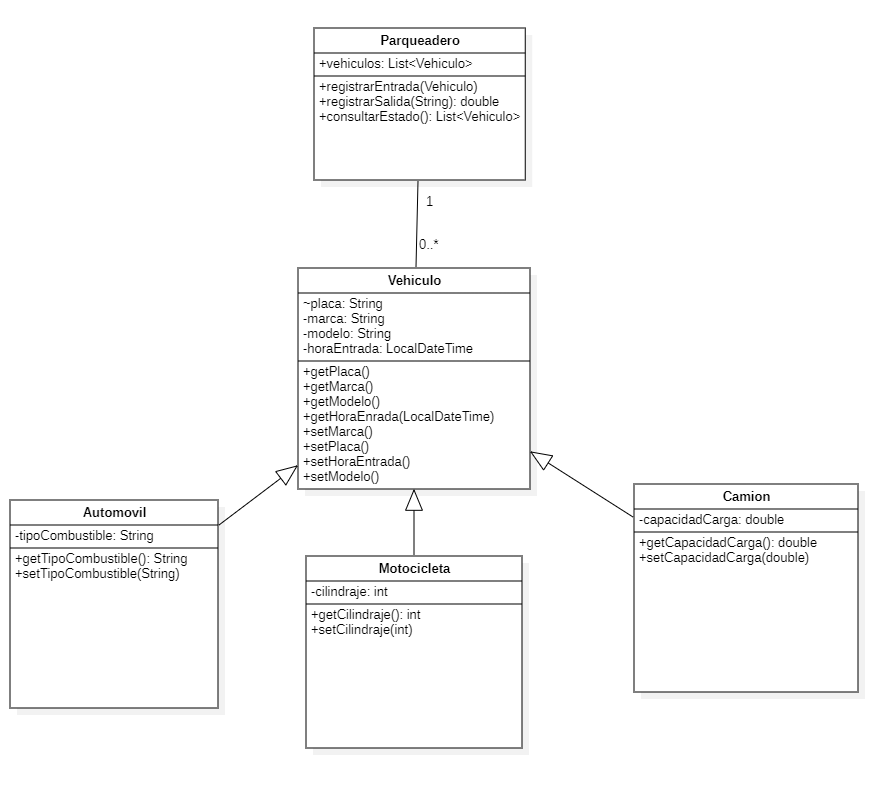
A diagram of a computer

Description automatically generated with medium confidence

En general en los ejercicios de la actividad 2, encontramos que solo usamos métodos para la realización de las tareas encomendadas en cada uno de los puntos y están descritas en el diagrama UML que esta adjuntado con anterioridad; en el también podemos apreciar los tipos de datos involucrados en cada uno de los parámetros de cada método en el proyecto del paseo empresarial.

Para la ejecución de los métodos se estableció una asociación con la clase Main para ejecutar los métodos que existen dentro de las clases.

**Actividad 3**

****

El proyecto de parqueadero lo podemos dividir en las siguientes clases:

1. Clase Vehículo:

Es la clase que tiene la función de heredar la mayor parte de los atributos para todos los vehículos

2. Clases que heredan:

Las clases que usan atributos de vehículo son Automóvil, Motocicleta, y Camión; estas clases añaden atributos específicos según el tipo de vehículo.

3. Clase Parqueadero

Es la clases que está encargada de gestionar la entrada y salida de vehículos, también está encargada de manejar una lista de objetos Vehículo; la clases parqueadero es la que calcula el costo del parqueo que está basado en el tipo de vehículo y el tiempo de estadía.

References

Deitel, P., Deitel, H., Romero Elizondo, A. V., & Fuenlabrada Velázquez, S. (2016). Cómo programar en Java (10a edición). Capítulos 8, 9 y 10. México, México: Pearson Educación.

Joyanes Aguilar, L., & Zahonero Martínez, I. (2014). Programación en C, C++, Java y UML (2a ed.). Capítulo 13. México, México: McGraw-Hill / Interamericana.