



**TECNOLÓGICO  
NACIONAL DE MÉXICO®**



# **INSTITUTO TECNOLÓGICO DE TLAXIACO**

## **INGENIERIA EN SISTEMAS COMPUTACIONALES FUNDAMENTOS DE INGENIERIA DE SOTFWARS**

### **REPORTE PRACTICA ARCHITECT**

#### **ALUMNO:**

- **ALVARO FROYLAN PACHECO OSORIO**

#### **PROFESOR: JOSE ALFREDO ROMAN CRUZ**



**SEPTIEMBRE DEL 2023**

# Introducción

Un diagrama de clases es una herramienta fundamental en el mundo de la programación orientada a objetos que se utiliza para visualizar y representar la estructura estática de un sistema de software. Proporciona una vista de alto nivel de las clases que componen el sistema, sus atributos, métodos y las relaciones entre ellas. Este tipo de diagrama es especialmente útil en el proceso de diseño y modelado de software, ya que ayuda a los desarrolladores a comprender la arquitectura del sistema y a comunicar de manera efectiva cómo interactúan las diferentes partes del mismo.

En este artículo, exploraremos en detalle cómo crear y utilizar un diagrama de clases, desde la notación básica hasta las mejores prácticas para su diseño. Además, veremos ejemplos de situaciones en las que un diagrama de clases puede ser una herramienta valiosa en el desarrollo de software, proporcionando una representación visual de la estructura de clases y sus relaciones en un sistema específico. A través de esta introducción, podrás adentrarte en el mundo de la modelación orientada a objetos y comprender la importancia de los diagramas de clases en el proceso de desarrollo de software.

## Desarrollo

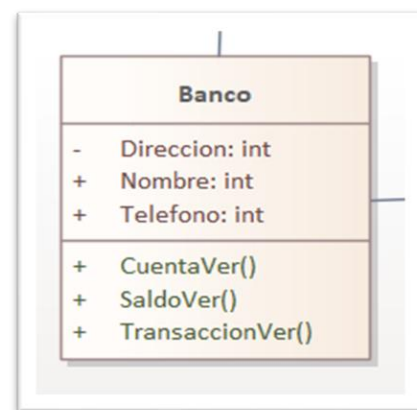
El objetivo central de un diagrama de clases es proporcionar una representación visual de las clases en un sistema de software y cómo están relacionadas entre sí. Esto ayuda a los equipos de desarrollo a planificar, comunicar y diseñar de manera efectiva, identificar posibles problemas y documentar la estructura del software, lo que mejora la eficiencia en el proceso de desarrollo y mantenimiento.

Empezaremos poniendo nuestras clases para hacer nuestro cajero automático

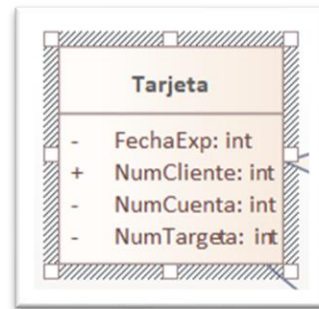
Ponemos una clase con el nombre de “cajero” con sus atributos, sus operaciones y su tipo de valor de la palabra



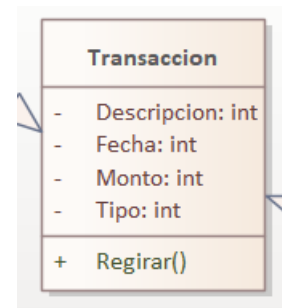
Luego creamos otra clase con el nombre de “banco” con sus atributos, sus operaciones y su tipo de valor de la palabra



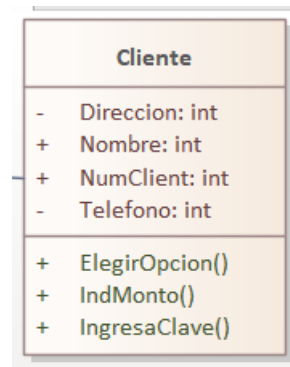
Luego creamos otra clase con el nombre de “Tarjeta” con sus atributos y su tipo de valor de la palabra



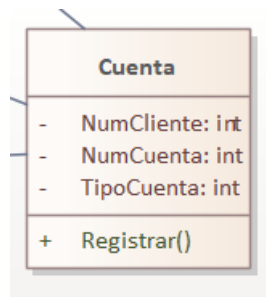
Creamos la siguientes clase llamada transacción y con sus atributos, su operación y su tipo de valor de la palabra



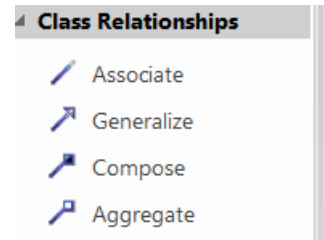
Creamos la clase llamada “Cliente” con sus atributos con sus operaciones y su tipo de valor de la palabra



Por ultimo creamos la clase llamada “cuenta” con sus atributos, sus operaciones y su tipo de valor de la palabra



Ya teniendo cada una de las clases que necesitamos tendremos que relacionar entre si las tablas



Relacionamos con la flecha de composición “banco” y “cajero”

Generalizamos “cajero” con “transacción”

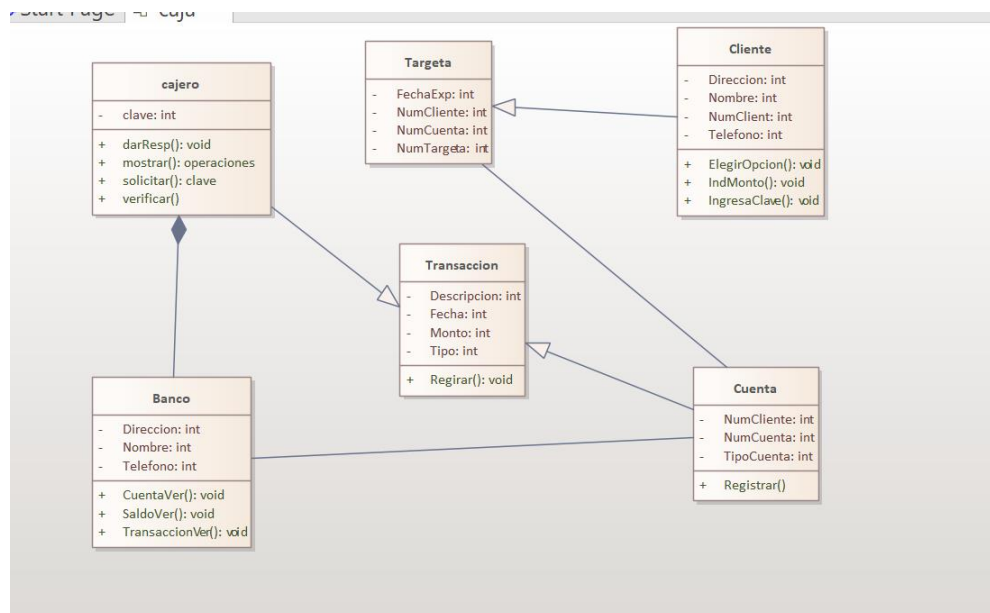
Relacionamos “cuenta” con “transacción”

Asociamos “cuenta” con “tarjeta”

“Cliente” con “tarjeta”

Asocio “banco” con “cuenta”

El cual nos quedaría con un diagrama de clases con relaciones



## Conclusión

En conclusión, el diagrama de clases de un cajero automático proporciona una representación clara y estructurada de las clases que componen el sistema, como "Cuenta Bancaria", "Cajero Automático", "Tarjeta de Débito", etc., junto con sus atributos y relaciones. Este diagrama es una herramienta esencial para el diseño y desarrollo de sistemas de este tipo, ya que ayuda a los diseñadores y desarrolladores a comprender cómo se organizan y relacionan las diferentes partes del cajero automático. Además, facilita la comunicación entre los miembros del equipo y sirve como una guía valiosa para el desarrollo, la identificación de posibles problemas y la documentación del sistema, contribuyendo en última instancia a la creación de un cajero automático funcional y eficiente.