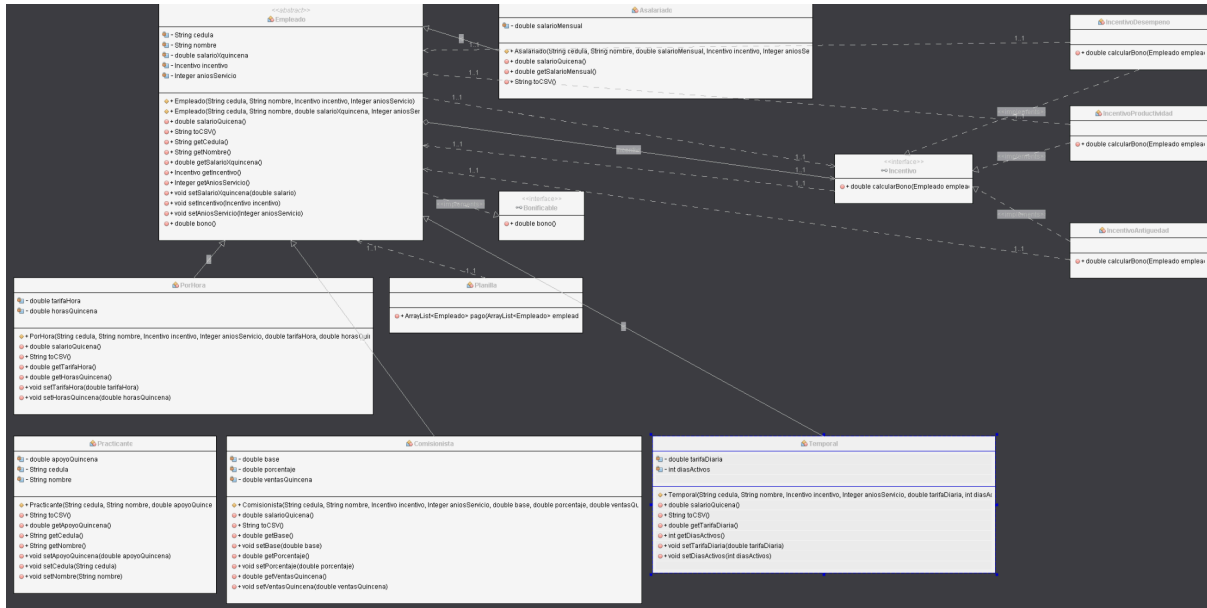


Diagrama UML



Decisiones de Diseño del Sistema

Herencia y Especialización:

- **Empleado abstracto y subtipos:** La clase Empleado es abstracta, lo que permite crear tipos específicos como PorHora, Asalariado, Comisionista, con comportamientos propios, como el cálculo de bonos.
- **Uso de interfaces:** La interfaz Incentivo permite flexibilidad para cambiar tipos de bonos sin alterar el código, implementando el patrón **Factory**.

Composición y Relaciones:

- **Asociación entre clases:** La clase Planilla tiene una relación con Empleado, centralizando el cálculo de pagos de todos los empleados de manera eficiente.
- **Composición en tipos de empleado:** PorHora, Asalariado y Comisionista tienen atributos específicos, reflejando diferentes formas de remuneración.

Cálculo de Bonos:

- **Métodos especializados:** Se delega el cálculo del bono en cada clase de incentivo, permitiendo flexibilidad para adaptar el cálculo a cada tipo de empleado.

- **Diversidad de incentivos:** Las clases IncentivoDesempeno o IncentivoAntiguedad personalizan el cálculo de bonos según características del empleado.

Estrategia de Pagos:

- **Método de pago flexible:** El método pago(ArrayList<Empleado> empleados) en Planilla permite manejar pagos de distintos tipos de empleados de manera escalable.

Optimización de Escalabilidad:

- **Escalabilidad con clases e interfaces:** El uso de clases concretas e interfaces facilita la extensión del sistema, permitiendo agregar nuevos tipos de empleados o incentivos sin complicar el diseño.

Abstracción y Generalización:

- **Uso de clases abstractas e interfaces:** La abstracción de Empleado y las interfaces Incentivo permiten un sistema modular, flexible y fácil de mantener.