**TITULO DE LABORATORIO**

**PARCIAL (GESTOR DE NOTAS)**

**HANNA KATHERINE ABRIL GÓNGORA**

**KAROL ASLEY ORJUELA MAPE**

**UNIVERSIDAD MANUELA BELTRÁN**

**PROGRAMACIÓN ORIENTADA A OBJECTOS**

**DOCENTE**

**DIANA MARCELA TOQUICA RODRÍGUEZ**

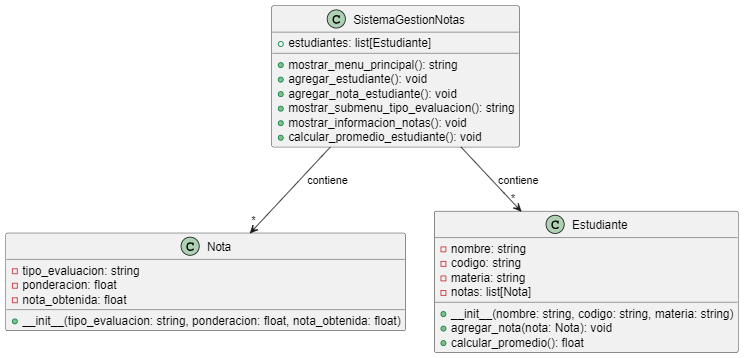
**BOGOTÁ DC VIERNES 22 DE MARZO**

1. **EXPLICACIÓN DEL CÓDIGO EN PYTHON**
2. **EXPLICACION DEL CODIGO UML**

**Texto

Descripción generada automáticamente**

1. **DIAGRAMA DEL CODIGO EN UML**

****

1. **EXPLICACION Y USO**

Este código es un sistema de gestión de notas para estudiantes. Primero, define dos clases: `Nota` para representar una nota en una evaluación, y `Estudiante` para representar a un estudiante con su nombre, código, materia y lista de notas. Luego, proporciona funciones para agregar estudiantes, agregar notas a los estudiantes, calcular el promedio de un estudiante y mostrar la información de las notas de un estudiante. Al ejecutar el programa, se muestra un menú con opciones para realizar estas acciones, como agregar un estudiante, agregar una nota a un estudiante, calcular el promedio de un estudiante, mostrar las notas de un estudiante y salir del sistema. Cada opción se ejecuta según la selección del usuario hasta que decide salir del programa.

1. **DEMOSTRACION DE PRUEBAS Y FUNCIONAMIENTO DEL CODIGO**
2. **DECISIÓN DE DISEÑO O CONSIDERACION IMPORTANTE**
3. **PILARES DE POO IMPLEMENTADOS**

**7.1. Encapsulamiento:** Se utilizan clases para encapsular la información y el comportamiento relacionado. Por ejemplo, la clase Nota encapsula la información sobre el tipo de evaluación, ponderación y nota obtenida. La clase Estudiante encapsula la información sobre el nombre, código, materia y las notas asociadas al estudiante.

**7.2. Abstracción:** Las clases Nota y Estudiante representan abstracciones de entidades del mundo real (notas y estudiantes, respectivamente). Se utilizan métodos como agregar\_nota, calcular\_promedio y mostrar\_informacion\_notas para abstraer el comportamiento relacionado con los estudiantes y las notas.

1. **CONCLUCIONES**

Al trabajar en este código, hemos profundizado en conceptos clave de programación orientada a objetos, como el uso de clases, objetos y métodos de instancia. Hemos aplicado estos conceptos para crear un sistema de gestión de notas que permite agregar estudiantes, asignar notas con ponderación, calcular promedios y mostrar información detallada de las notas y el desempeño de los estudiantes en cada materia.