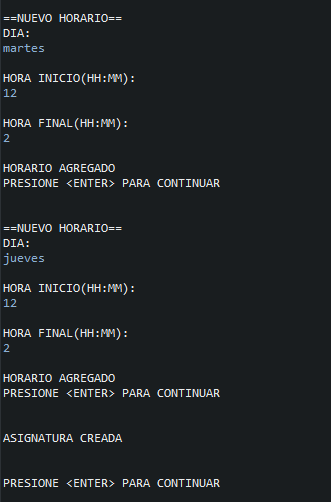
**Memoria Escrita  
Grupo 12**

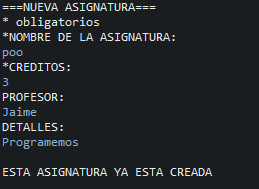
**Descripción general**

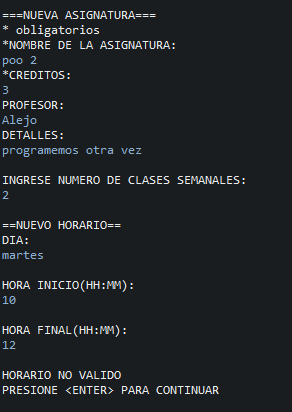
El objetivo general del proyecto es diseñar e implementar una aplicación de escritorio en Java que permita a estudiantes organizar sus horarios, asignaturas y calificaciones de la universidad de manera cómoda e intuitiva sin tener que recurrir a aplicaciones externas y consiguiendo funcionalidades útiles para el usuario tales como calcular su promedio por semestre, calcular su promedio ponderado acumulado y determinar el avance en su programa de estudios actual.  
Para esto, consideramos necesario crear 9 clases en el modelo lógico y desarrollar los métodos y funcionalidades en estas.

**Muestra de funcionalidades**

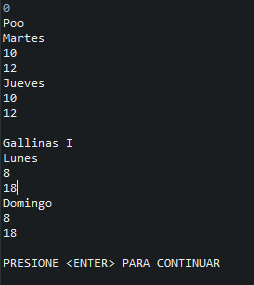
* ­Creación de asignaturas:







* Horarios:



* PAPA:



* Avance de carrera:



**Manual de Usuario:**

**Menú principal**Cada uno de los elementos del menú principal conduce a un submenú con los respectivos ajustes que indica su nombre y con la opción de mostrar las instancias que se han creado de cada uno de estos. A continuación los describiremos individualmente.

**Perfil**

* **Mostrar mi perfil:** Muestra el nombre, la cédula, el correo, el plan de estudios y el avance de carrera del estudiante
* **Editar mi perfil:** Permite modificar cada uno de los elementos listados anteriormente

**Asignaturas:**

* **Mis asignaturas:** Permite ver las asignaturas inscritas en el semestre en curso del sistema.
* **Editar asignaturas:** Permite editar los créditos, las notas, el horario, los detalles, el nombre de la asignatura y el nombre del profesor
* **Agregar y eliminar asignatura:** Permiten crear nuevas asignaturas y borrar asignaturas previamente creadas

**Notas:**

* **Mostrar notas:** Permite visualizar las notas de las asignaturas inscritas en el semestre en curso
* **Calcular PAPA:** Calcula el promedio aritmético ponderado de todos los semestres
* **Calculadora:** Permite hacer proyecciones con posibles notas actuales y futuras.

**Horarios:**

* **Mostrar horario de clases:** Muestra los horarios de cada clase inscrita en el semestre
* **Mostrar horario de asesorías:** Muestra los horarios de las asesorías

**Profesores:**

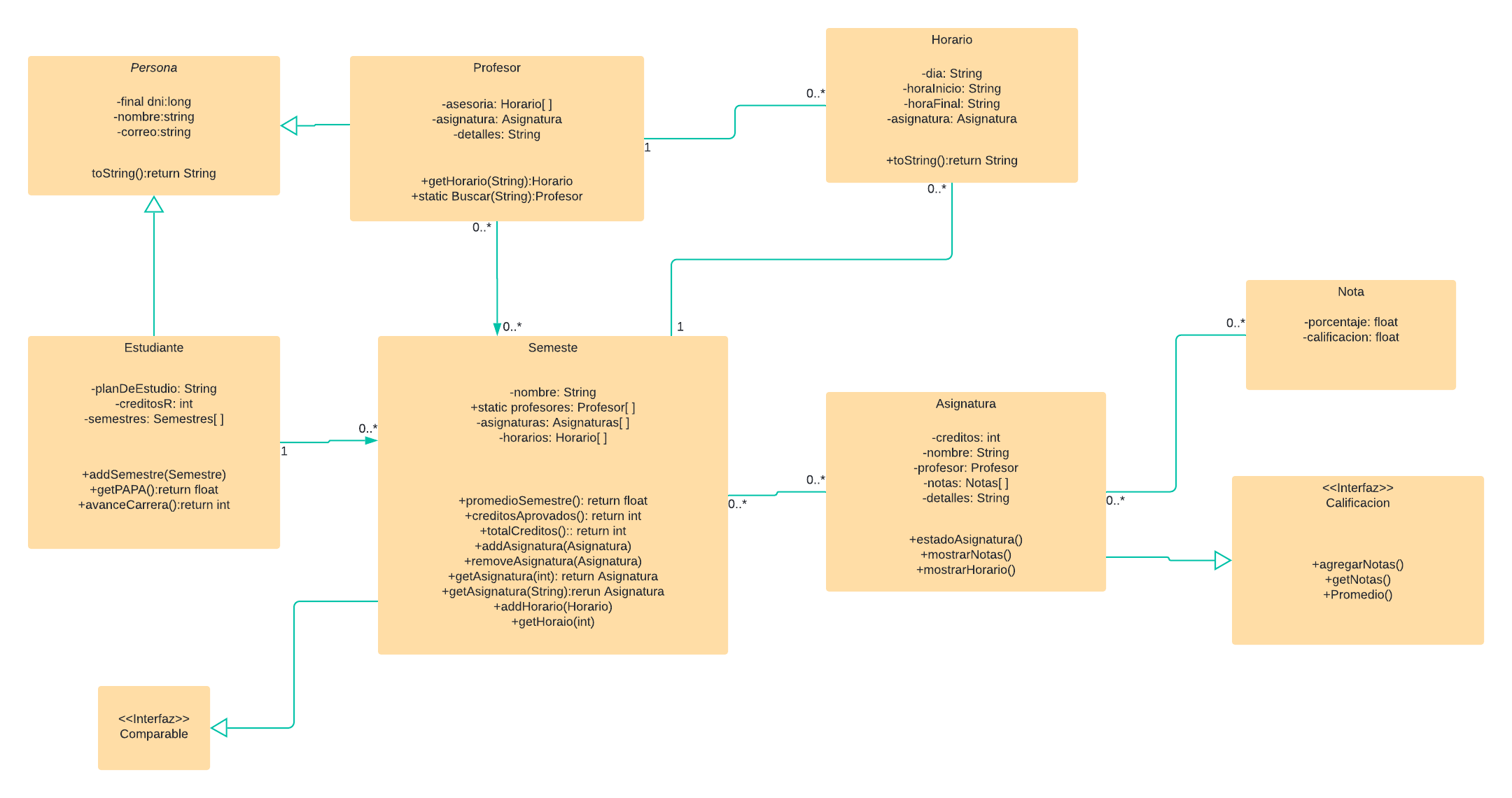
* **Lista de profesores:** Muestra os profesores de todas las asignaturas vistas
* **Editar profesores:** Permite crear y borrar profesores y modificar el nombre, correo, a asignatura que dicta, los horarios de asesoría y los detalles del profesor

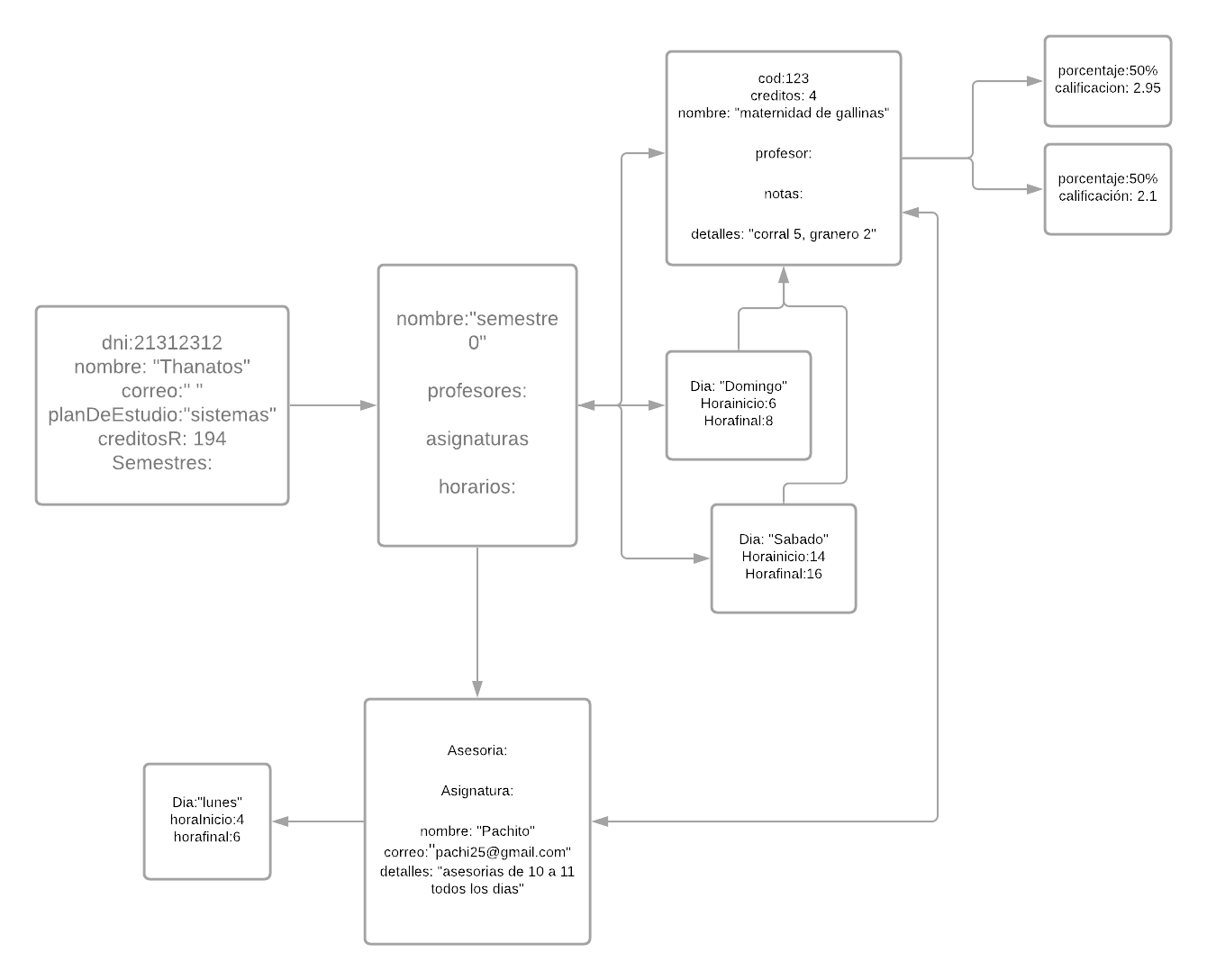
**Semestres:**

* **Nuevo semestre:** Crea un nuevo semestre
* **Editar semestre:** Permite cambiar el nombre del semestre
* **Eliminar semestre:** Permite borrar semestre previamente creados
* **Cambiar semestre:** Permite cambiar el semestre en curso en la aplicación

**Características de Poo:**

* **Clase y método abstracto:** Clase abstracta persona y método abstracto toString.
* **Interface:** Calificación.
* **Herencia:** Clases Estudiante y Profesor heredan de Persona
* **Atributo y método de clase:** ArrayList profesores en clase Semestre y método buscar en clase profesor
* **Uso de constante:** atributo dni en clase Persona
* **Encapsulamiento:** Todos los atributos de Persona tienen modificador protected, todos los de Asignatura son private y ArrayList profesores en Semestre es public
* **Sobrecarga de constructores:** Hay sobrecarga de constructores en las clases Profesor y Asignatura
* **Referencias this:** Usamos referencias this en los constructores de Profesor y Asignatura
* **Referencias this():** En las clases Profesor y Asignatura

****

****