



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

## Reporte Práctica 0

ALUMNO

López Diego Gabriela - 318243485

PROFESOR

Gilde Valeria Rodríguez Jiménez

AYUDANTES

Luis Angel Leyva Castillo

Rogelio Alcantar Arenas

Gibrán Aguilar Zuñiga

ASIGNATURA

Cómputo Concurrente

7 de febrero de 2024

# 1. UML

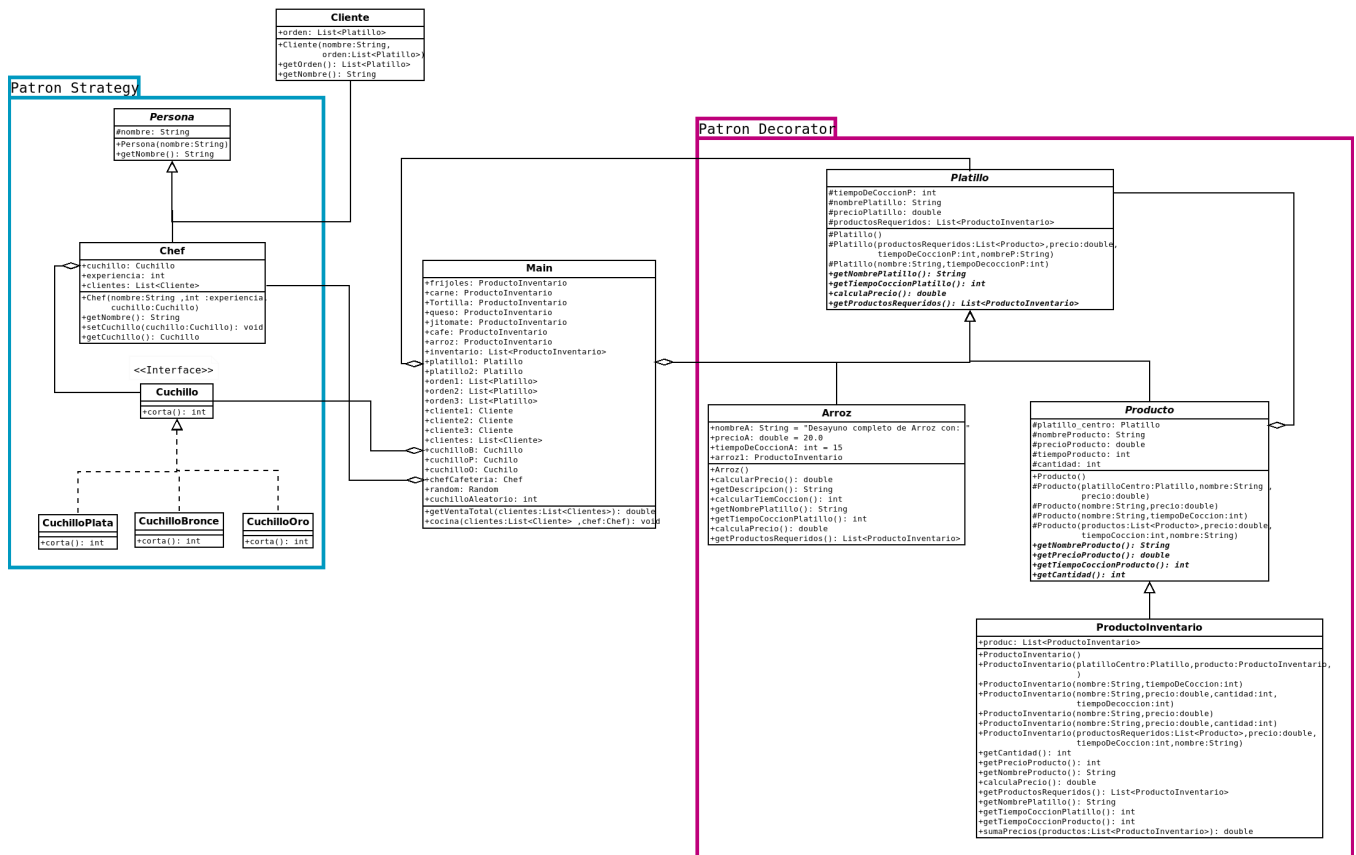


Figura 1: UML de la práctica 0

## 2. Patrones de diseño

1. Patrón de diseño Strategy. Decidimos utilizar este patrón de diseño ya que nos permite y nos facilita cambiar de herramienta a nuestro chef. Que es este caso, se trata del cuchillo que utiliza para preparar los diferentes platillos que nos llegan a la cafetería. Cada cuchillo tiene consigo, diferente poder de cortar los ingredientes. Tenemos el cuchillo de bronce, de plata o bien, de oro. Dependiendo del tipo de cuchillo que se utilice podemos disminuir en unidades de tiempo el tiempo TOTAL de elaboración de los platillos de cada uno de nuestros clientes.

Por ello, tenemos la interfaz Cuchillo (representa la clase strategy), 3 clases que implementan de Cuchillo (representan las diferentes tipos de estrategias), y la clase Chef (representa la clase contexto). Finalmente, en el main asignamos de forma aleatoria el tipo de cuchillo que utilizara nuestro chef.

2. Patrón de diseño Decorator. En nuestra cafetería, el ingrediente base o que siempre se servirá, ya que se trata de una guarnición perfecta para un desayuno equilibrado. Nuestros clientes, tienen la opción de elegir e ir agregando los productos que deseen. El precio del platillo y el tiempo de cocción se definen por todos los ingredientes que fueron agregados. De este modo, el patrón de diseño que quedaba a la perfección era decorator.

Por ello, la clase Platillo se convierte en nuestra clase componente. Arroz en nuestro componente base, la clase Producto en nuestra clase decorator y finalmente ProductoInventario en nuestra clase que representa los diferentes tipos de ingredientes que se pueden ir agregando a nuestro Platillo final.

### 3. SonarLint

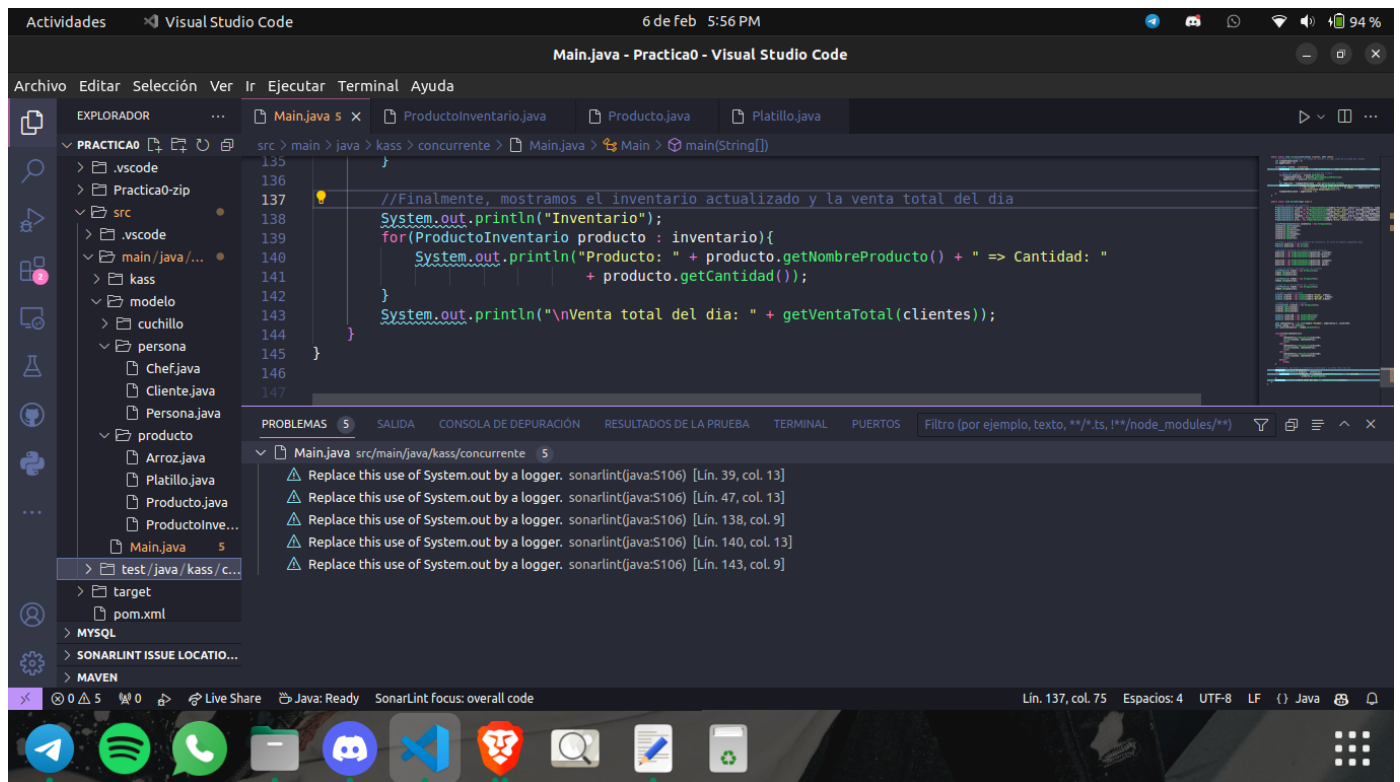


Figura 2: Captura que muestra los resultados del plugin Sonarlint en la práctica 0