**Caso 1: Infracomp**

*Andrés Peña Diaz: 201913766*

*Santiago Campo Cotes: 201921995*

1. **Funcionamiento global de la aplicación**

En el Caso 1 de Infracomp se utiliza Java para poder desarrollar una apliación que permita que varios comensales coman en una mesa a la vez, no obstante se deben tener en cuenta ciertas condiciones como el hecho de que los comensales deben utilizar cubiertos para poder comer sus platos. Adicionalmente, estos cubiertos deben ser lavados en un fregadero y por un lavaplatos. Para modelar la interacción del caso se utiliza la clase Thread extendida en los comensales y el lavaplatos.

1. **Documentación del Código**

En el Código Fuente de la aplicación que resuelve el caso se encuentra una documentación detallada de lo que hacemos en cada paso. Adicionalmente, contamos con diferentes LOGs que informan en consola lo que está sucediendo en la aplicación, mientras que corren todos los Threads.

1. **Sincronización del Comensal**

Para la sincronización del Comensal se utilizó la mesa como un monitor, donde se pudiese ejcutar las diferentes acciones de recoger cubiertos, comer y dejar cubiertos. Adicionalmente utilizamos un CyclicBarrier que permitia evitar que algún commensal termine antes de la mitad de la cena.

1. **Sincronización del Fregadero**

Para la sincronización del Fregadero se utilizarón sleeps, que erán generados de momento aleatoria, los cuales modelaban la acción de “lavar” cubiertos. Adicionalmente, incluimos yield() como espera semi-activa para que se pudiese recoger y dejar cubiertos en el fregadero.

1. **Sincronización de la Mesa/Lavaplatos**

Para la sincronización de la Mesa se utilizarón esperas pasivas y métodos sincronizados que permitian que no se alterarán las variables de *numCubiertosT1 y numCubiertosT2.* Adicionalmente, se incluyerón sleeps que modelaban la acción de “comer”. También era importante un notify() que permitiera despertar los threads que está esperando un Comensal para poder recoger. Por el lado del Lavaplatos realmente no hay una sincronización directa puesto que es un loop infinito que se encarga de manejar la sincronización definida en Fregadero y Mesa.

A picture containing graphical user interface

Description automatically generated