**Diseño y Programación O.O**

**Documento de Diseño proyecto 2**

Sergio Arango Arango 201921814

Juan José Sierra Alarcón 202013642

Hubo un cambio de diseño importante con respecto al proyecto 1 y es que la estructura de datos registro de Compras ahora es un mapa de la forma: <Cedula de Cliente, Lista de compras del cliente>. Antes solo era una lista de compras. Esto se hace porque el propósito de tener las compras registradas es porque nos interesa saber las compras de un cliente en específico para conocer su actividad.

El resto de los cambios con respecto al proyecto 1 (obviamente exceptuando la interfaz) son correcciones inmensas de implementación, no de diseño.

Diseño de la interfaz:

Para diseñar la interfaz no se realizó ni un solo cambio de diseño con respecto al diseño del proyecto 1. Es decir que nuestro programa sigue siendo completamente desacoplado de la interfaz y funciona perfectamente con la interfaz gráfica o por consola. Es decir: nunca se envían objetos de las clases del modelo o procesamiento a la interfaz, toda la mensajería que ocurre es de tipo simple o como máximo estructuras de java.util. La mensajería entre la interfaz y el programa es simple para reducir el acoplamiento entre las dos. Esta decisión se tomó como prevención a lo desconocido cuando no se sabía todavía cómo sería el diseño y desarrollo de interfaces gráficas y además para que el programa siguiera siendo compatible con la interfaz por consola del proyecto 1. En retrospectiva esto resulta tal vez innecesario ya que la interfaz gráfica construida será permanente y no habrá más cambios con respecto a esto en el futuro y se conoce perfectamente en términos prácticos cómo funcionan las interfaces gráficas. Además, esto quita flexibilidad de cómo se desea mostrar la información en la interfaz. Ya que casi todo llega como String, siempre se debe buscar donde poner el mensaje con el texto con la información sin más opciones. En el caso que se usa un mapa como respuesta cuando se piden los lotes y cantidades de un producto al Supermercado, se evidencia que hay más flexibilidad para poder mostrar un componente deslizante, esto fue gracias a que en el proyecto 1 ya se enviaba esta respuesta como mapa y se convertía a texto en la interfaz de consola.

Con respecto al diseño interno de las ventanas.

Siempre se fijan las ventanas mismas como sus propios ActionListeners, estas luego tienen métodos internos que ejecutan la acción deseada que puede llamar o no (seguramente) métodos del supermercado. Cuando se crean subventanas se les pasa como parámetro el objeto supermercado para que de ahí puedan llamarle los métodos para obtener la información que se deba mostrar.