

### Parte 1

- a. Discuta y defina las responsabilidades y que hace cada módulo.
- b. Haga un boceto (Journey Map) de cómo sería el proceso de compra utilizando este modelo.

### Parte 2

Analicen si este diseño inicial cumple con los principios de bajo acoplamiento y alta cohesión. Si no es el caso, deben proponer una solución que cumpla con estos principios.

Para el análisis, considere estas preguntas:

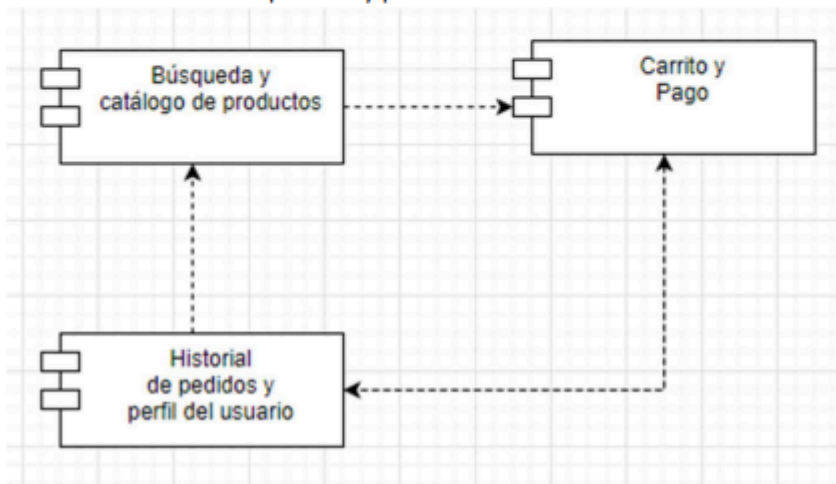
- ¿Los módulos tienen responsabilidades claramente definidas y relacionadas entre sí?
- ¿Existen dependencias innecesarias entre los módulos?
- ¿Se puede modificar un módulo sin afectar significativamente a los otros?

## UNIDAD TEMÁTICA 1 – Diseño y UML – Trabajo de Aplicación 2

### EJERCICIO 1

Supongamos que estamos desarrollando una aplicación de comercio electrónico. La aplicación permite a los usuarios buscar productos, añadirlos al carrito, realizar el pago y ver el historial de pedidos. Desde el área de arquitectura nos proponen esta distribución de módulos:

- Módulo de búsqueda y catálogo de productos
- Módulo de carrito y pago
- Módulo de historial de pedidos y perfil del usuario



### Parte 1:

a)

- i) Búsqueda y catálogo de productos:

Gestionar las búsquedas del usuario, incluyendo el uso de filtros, y presentar los productos generales y filtrados.

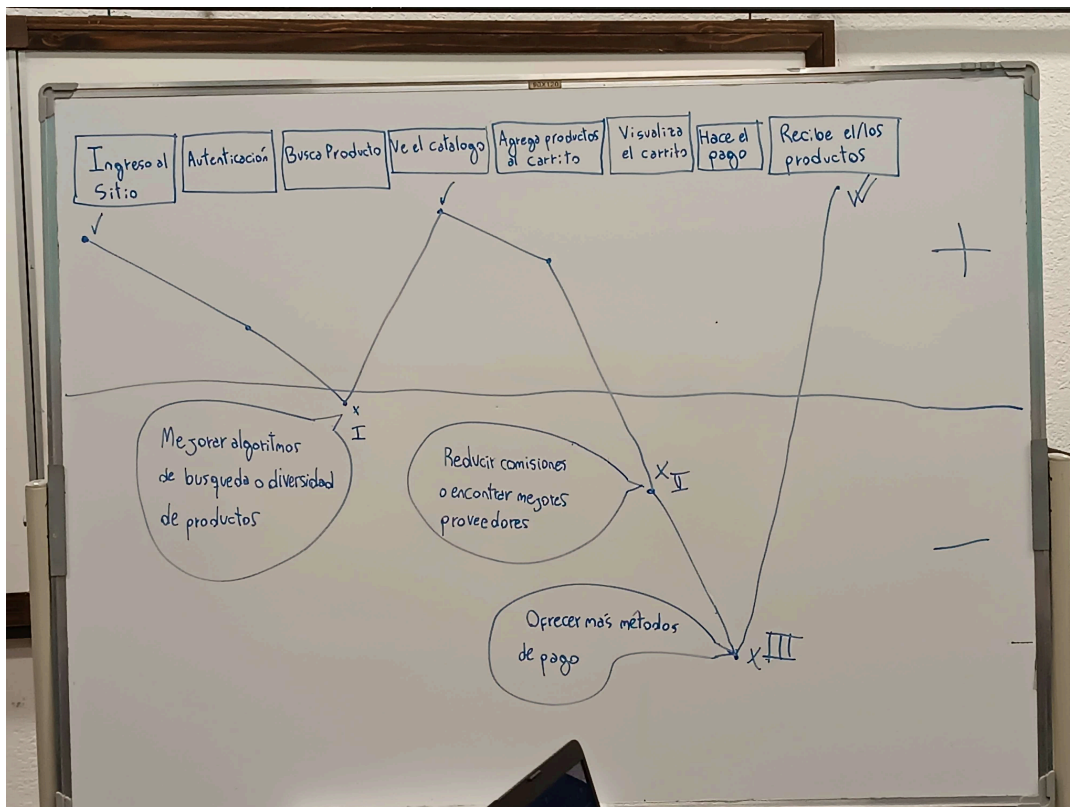
- ii) Carrito y Pago

Permite al usuario mantener un registro de varios productos que quiere comprar, seleccionar un método de pago o ingresar un nuevo método de pago, posteriormente realizar la transacción.

iii) Historial de pedidos y perfiles de usuario

Mantiene un registro de todos los pedidos de cada usuario, ver, registrar y cambiar los datos del mismo.

b)



Puntos Críticos:

I) La bajada en este punto crítico se debe a que la búsqueda de los productos no conduce a lo que el usuario en verdad desea, o le toma demasiado tiempo encontrarlo.

II) Al ir al carrito de compras el usuario se da cuenta de todas las comisiones e impuestos que se le añaden a los precios de los productos, aumentando casi al doble el monto total.

III) El usuario se puede enojar porque los métodos de pago no coinciden con los que él realmente posee, o porque no hay descuentos con ninguna tarjeta, o no se le permite hacer el pago en cuotas.

Parte 2:

¿Los módulos tienen responsabilidades claramente definidas y relacionadas entre sí?

- Si hay responsabilidades y poco relacionadas entre ellas
- Módulo de carrito debe poder acceder a datos de los productos para enlistarlos al carrito.

¿Existen dependencias innecesarias entre los módulos?

- La búsqueda y catálogo de productos no requiere de información del historial ni del usuario.

¿Se puede modificar un módulo sin afectar significativamente a los otros?

- No, todos los módulos dependen significativamente de algún otro módulo.
- Si se modifica el módulo de búsqueda y catálogo de productos, se afecta el carrito y pago
- Si se modifican el carrito y pago, se afecta el historial y perfil
- Si se modifica el historial y perfil, se afecta a carrito y pagos.

Hay un alto acoplamiento en el módulo de historial de pedidos y perfil de usuario, dado que ambos datos son distintos y separables.

Aparte de eso, hay una alta cohesión en el sistema, dado que el carrito y el sistema de pago van mucho de la mano.

Solución:

Como propuesta para mejorar al sistema se podría separar las clases, quedando como clases que cumplan una única responsabilidad cada una, bajando la dependencia del sistema

(Lucas Alegre, Franco De Stefano, Santiago Ferraro, Franco Robotti)