

Análisis de la solución

Para la realización del proyecto final se empezó con la selección de una temática específica como base para la elaboración de la aplicación. Se optó por una tienda armamentística ficticia con los dos módulos requeridos, el del usuario o cliente y el del administrador o empleado. Ambos comparten una serie de atributos y métodos al heredar de una misma superclase. Sin embargo, la clase administrativa se le permite acceder a la información en la base de datos como lo son los productos, los usuarios con su información y las órdenes de compra realizadas. Por otra parte, la clase usuario se le permite usar el carrito de compras para realizar órdenes y un panel específico para la modificación de su información personal o la eliminación de su cuenta.

Inicialmente, la aplicación inicia el panel del catálogo con la opción de iniciar sesión o agregar productos al carrito de compras, previo a iniciar una sesión, aunque para acceder al mismo se redirige al panel para iniciar sesión y así continuar con las compras, sin perder el carrito anterior. Finalmente, desde el carrito se permite añadir, restar o remover completamente los productos seleccionados, para así al realizar el pedido en el carrito, realizar los cambios respectivos en la base de datos y generar la orden de compra.

En el caso de que en el inicio de sesión se opte por acceder como un administrador, se cambia el panel de cuenta por uno especial donde no solo se podrá modificar la información del empleado, sino que a su vez, podrá revisar, filtrar y modificar cierta información de los usuarios, productos y ventas a través de las tablas obtenidas directamente de la base de datos.

Para llevar a cabo las funciones anteriormente mencionadas, se implementó un Modelo Vista Controlador (MVC), para separar las funciones de cada código, clase y método, garantizando una alta cohesión y un bajo acoplamiento. Esto con el fin de controlar los diferentes procesos y sus variantes evitando editar una gran cantidad de líneas de código y dependencias entre las clases y la información obtenida. Desde el modelo se crearon diferentes clases para los objetos que se requieren tanto en tiempo de ejecución como para su persistencia en la base de datos. En la vista se

llevó a cabo todo control gráfico que implicaba el uso de interfaces para la creación de una *GUI* (interfaz gráfica de usuario) sencilla e intuitiva. Mientras que, en el controlador se desarrollaron todas las comunicaciones entre todo lo anterior sumado a la base de datos que genera la persistencia de los cambios realizados por el usuario o en su defecto, el administrador.