

Campus Estado de México

Ricardo Cervantes Martínez A01745912

Josué Bernardo Villegas Nuño A01751694

Pensamiento computacional orientado a objetos (Gpo 302)

Proyecto Integrador

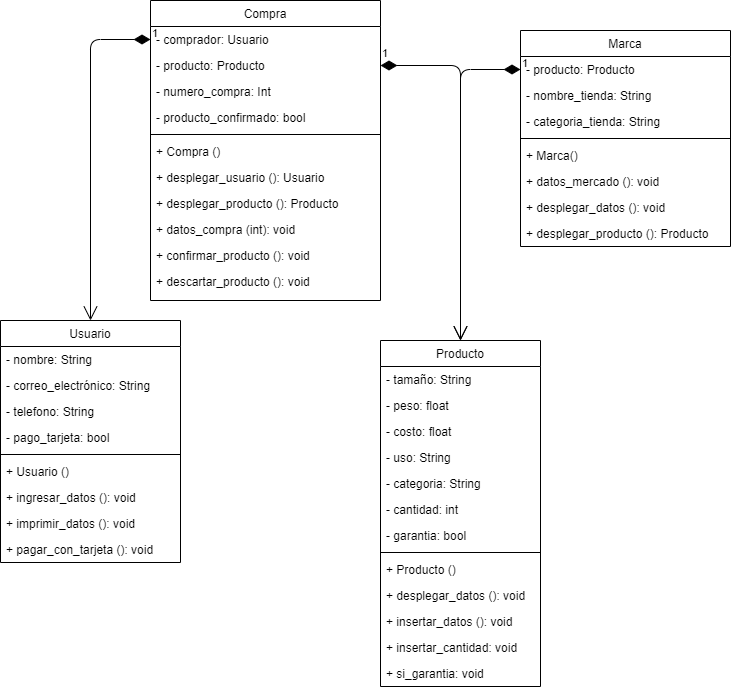
Profesor: Dr. Jorge Adolfo Ramírez Uresti

1/12/2020

*Propósito.*

Dentro del presente trabajo, se desarrolló un programa a través del lenguaje de programación C++, utilizando el entorno de programación Visual Basic, esto con el objetivo de poder desarrollar las competencias que se solicitan en la materia de Pensamiento computacional orientado a objetos. Dichas competencias conllevan un manejo a nivel básico del uso de la programación orientada a objetos (POO), la resolución de problemas a través de POO, la comprensión de problemas, la modelación de problemas y el desarrollo a nivel profesional de diseños gráficos que ayuden a entender el problema.

*Diagrama de clases.*



*Funcionalidad.*

El programa fue diseñado para poder facilitar el registro de compras en línea para personas que no puedan salir de casa por cuestiones de salud. A continuación, se mostrarán la explicación de los métodos utilizados en cada una de las clases del programa.

Usuario.

* *ingresar\_datos()*: Se utiliza para poder insertar los datos del usuario directamente en la consola para poder tener una mejor interacción por parte del usuario.
* *imprimir\_datos()*: Se utiliza para mostrar en la consola los datos que estén almacenados en los atributos que componen a la clase.
* *pagar\_con\_tarjeta()*: Es un método que cambia el estado del atributo pago\_tarjeta, para poder confirmar si el usuario al momento de la compra quiere realizar el pago con tarjeta.

Compra.

* *desplegar\_usuario()*: El método recibe un objeto de tipo “Usuario” y lo regresa conforme a los datos que tenga almacenado ese objeto, ya sea por el constructor de omisión u otro método que ajuste los datos de este.
* *desplegar\_producto()*: Este método recibe un objeto de tipo “Producto” y lo regresa dependiendo de los datos que se almacenen en el objeto, por definición del constructor de omisión o por otro método que cambie su valor.
* *datos\_compra()*: Este recibe un dato de tipo entero y se lo suma al número de compra para poder registrar el número de compras que el usuario ha realizado. Imprime en la pantalla el número de compra que lleva el usuario.
* *confirmar\_producto()*: El método permite almacenar un valor true en el atributo producto\_confirmado. Imprime en la pantalla el estado del producto, si está elegido o no.
* *descartar\_producto()*: Cambia el valor del atributo producto\_confirmado y lo cambia a falso en el caso de que no se quiera comprar el producto. Imprime en la pantalla el estado del producto, si está elegido o no.

Producto.

* *desplegar\_datos():* El método imprime los valores de los atributos que corresponden a la clase producto.
* *insertar\_datos():* El método permite insertar los valores que el usuario quiera para el producto y los almacena en cada uno de los atributos que lo componen.
* *insertar\_cantidad():* Este permite insertar la cantidad del producto elegido a comprar. Imprime la cantidad almacenada en el atributo cantidad.
* *si\_garantia():* Permite cambiar el estado de la garantía del producto, si tiene garantía se manda a llamar este método y se almacena un true en el atributo garantía.

Marca.

* *datos\_mercado()*: Permite ingresar el nombre de la marca de la que se va a comprar.
* *desplegar\_datos()*: Imprime los datos de la marca.
* *esplegar\_producto()*: Recibe un objeto de tipo Producto y regresa el valor del objeto que se instancia en la clase Marca con los valores que se le asignaron.

*Corrida de programa.*

Prueba 1.

Texto

Descripción generada automáticamente Texto

Descripción generada automáticamente

Prueba 2.

Texto

Descripción generada automáticamente Texto

Descripción generada automáticamente

Prueba 3.

Texto

Descripción generada automáticamente Texto

Descripción generada automáticamente

*Liga a Github (Equipo).*

<https://github.com/A01745912/Proyecto-Integrador-.git>