



# **Universidad Técnica de Manabí**



**Facultad de Ciencias Informáticas  
Escuela de Ingeniería en Sistemas Informáticos**

## ***Programación Avanzada***

**Docente:**

**Ing. Jorge Párraga**

**Responsables:**

**ARRAY ARANA ISABEL CRISTINA**

**AVELLÁN VALDÉS SHABELY**

**ÁVILA BRIONES WILSON ALEXANDER**

**HOLGUÍN INTRIAGO CRISTOPHER AGUSTÍN**

**Paralelo:**

**“A”**

**Periodo Académico:**

**Mayo 2020 – Octubre 2020**

## Índice

Caracterización del problema .....	3
¿Qué problema resolverá? .....	3
¿Por qué es importante resolver ese problema? .....	3
Propuesta de solución.....	5
Alcance de la propuesta.....	8
¿Qué hará el sistema de información?.....	8
¿Qué no hará el sistema de información?.....	8

## Caracterización del problema

### ¿Qué problema resolverá?

La empresa objeto de estudio es Tres Hermanos, que arrancó sus operaciones en el año 2010 y se acoge como un ejemplo de una situación representativa para la mayoría de caso dentro del rubro. Alrededor de 20 distribuidores visitan la tienda cada semana, pero



como es de esperarse los dueños (aquellos que realizan los pedidos), no se encuentran todo el tiempo en el local, por lo cual suelen haber encargados (los cuales no tienen un total conocimiento de que pedidos realizar). Sabiendo que los distribuidores pasan en días

definidos pero no en horas exactas y que los tenderos no saben en qué momentos llegarán los mismos, se vuelve una situación problemática para ambos lados, debido a que el distribuidor puede perder



potencialmente un pedido y el tendero perdería los productos para esa semana. Otra situación bastante común es que el tendero se olvide de ciertos productos en el pedido, ya sea por prisas o por falta de tiempo, ocasionando que ambas partes pierdan. Una de las soluciones poco viables que afectan al lado del cliente (tendero) por su falta de comodidad sería la creación de sistemas individuales por cada empresa. Por todo esto y más vimos necesario resolver la problemática que presentan este tipo de relaciones de negocio.

### ¿Por qué es importante resolver ese problema?

En la actualidad las empresas están enfocadas en obtener los mejores resultados, anhelan lograr excelencia operativa, son muy exigentes en sus objetivos, buscan adaptarse al



entorno, a la situación actual del país, a las políticas que rigen y así poder lograr su desarrollo y permanencia en el tiempo. Es por esto que es importante ayudar a centralizar la forma en la que se hacen los pedidos, es decir, mejorar

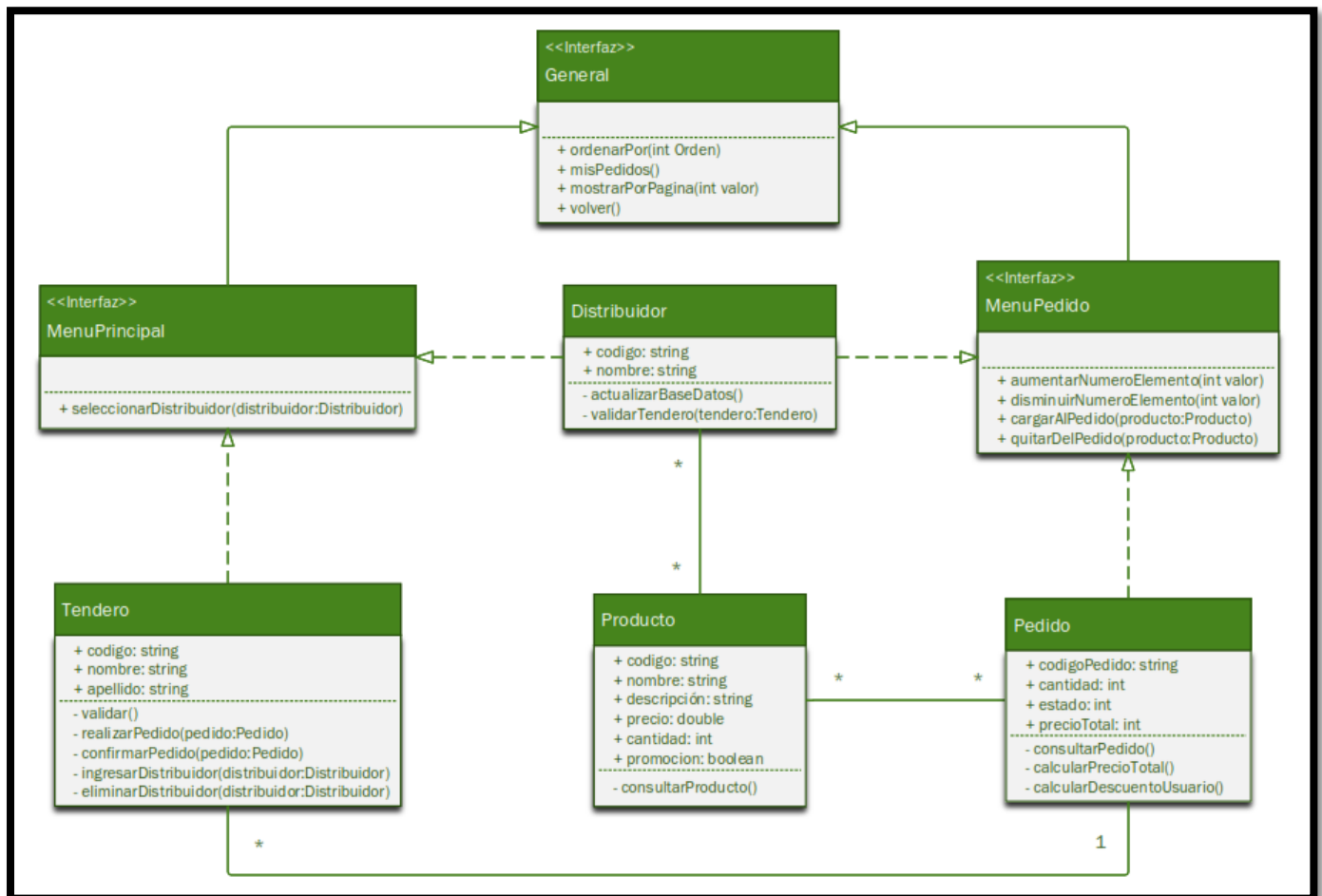
la eficiencia de la realización de pedidos sin intermediarios y así apoyar a la pequeña y

mediana empresa nacional. Si les proporcionamos métodos que faciliten su trabajo y un mejor acceso a las TI estas pequeñas empresas lograrán una forma más fácil y directa de lograr sus objetivos. Al no tener a los intermediarios (que son parte de la empresa), ya que no son necesarios, se contribuye a mover fuerzas laborales a campos que tengan mayor importancia.

## Propuesta de solución

Crear un sistema que sirva como intermediario digital entre las empresas y los usuarios (los tenderos). Los usuarios tendrán un solo sistema y se proporcionará un portal que permitirá a las empresas vender sus productos de manera digital sin necesidad de invertir en un sistema propio.

Para tener una mejor comprensión del problema a solucionar se presenta el siguiente diagrama de clases que manifiesta de forma clara el comportamiento en el sistema.

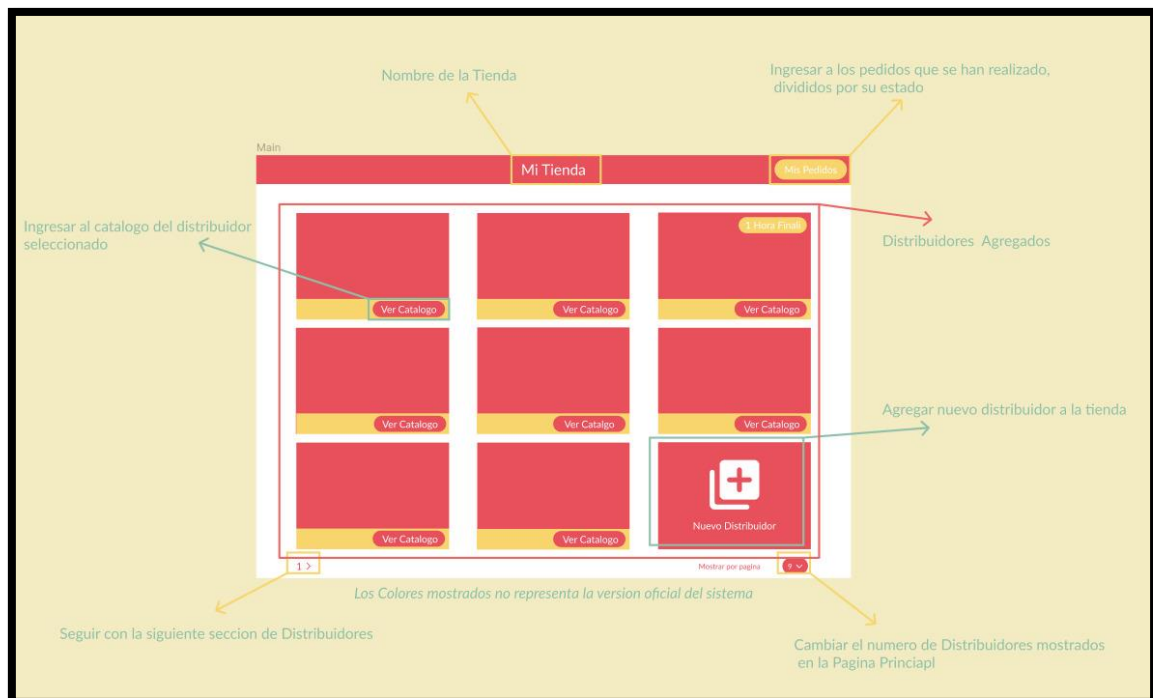


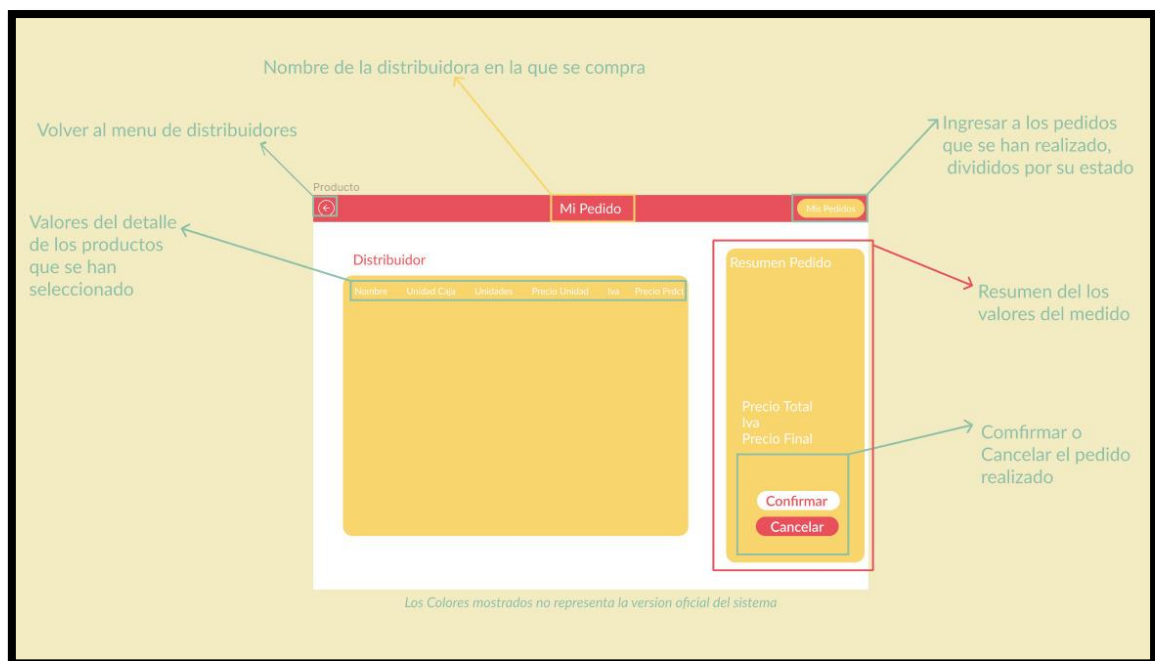
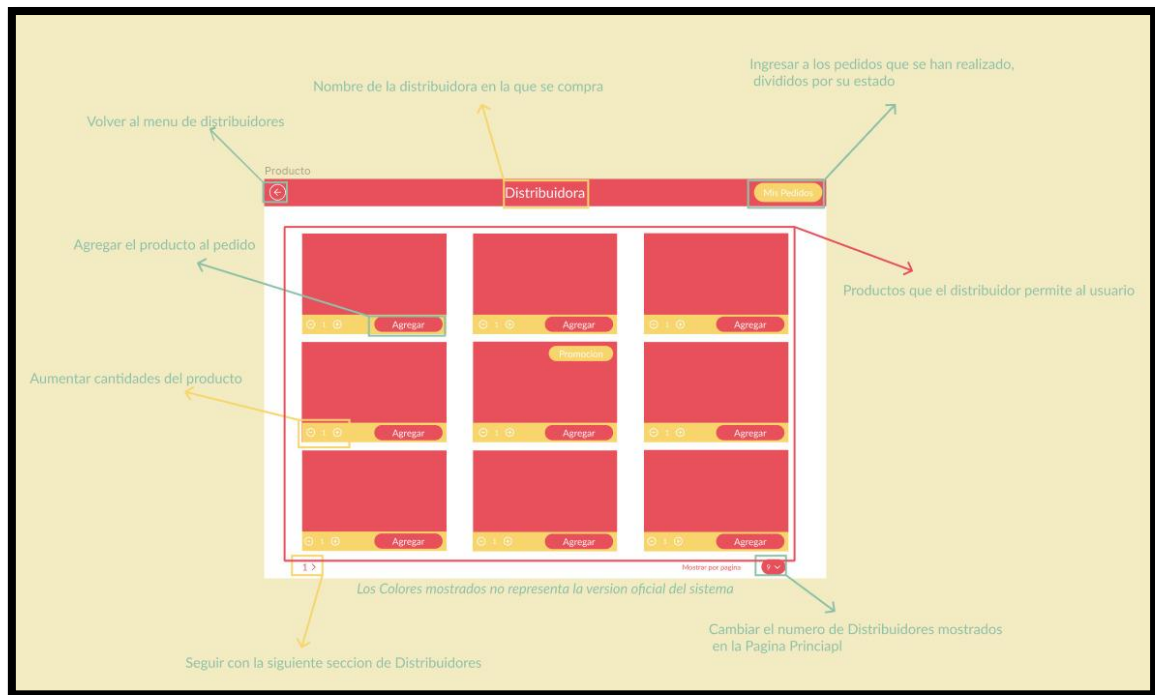
En el diagrama podemos encontrar como tratan los diferentes objetos y vistas que tendrá el sistema. Algunas interacciones importantes que podemos nombrar son:

- ❖ `MenuPedido` tiene una relación directa tanto con `distribuidor` como con `pedido` ya que esta sea reutilizable por sus métodos de edición al pedido.
- ❖ `MenuPrincipal` y `MenuPedido` son hijos de `General` ya que sus métodos de manejo de interfaz y ordenamiento así como acceso directo a `misPedidos()` hacen que este manejo sea más eficiente.

- ❖ Debido a que cada distribuidor tiene su propio manejo de base de datos interno la clase Distribuidor solo necesita el método actualizarBaseDatos() para obtener cualquier cambio realizado.

Una propuesta de modelo para las interfaces y su explicación breve de que hace cada cosa.





## Alcance de la propuesta.

### ¿Qué hará el sistema de información?

- ❖ El sistema permite realizar pedidos de diferentes distribuidores
- ❖ Permite ver las promociones que cuentan los diferentes proveedores.
- ❖ Permite revisar el catálogo de cada distribuidor.
- ❖ Permite tener un histórico de los pedidos realizados.
- ❖ Permite que la empresa solo tenga que actualizar la base de datos.
- ❖ Permite agregar productos a favoritos.
- ❖ Permite realizar los pedidos de acuerdo a los distribuidores.
- ❖ Tiene un proceso de facturación.
- ❖ El programa cuenta con un sistema de alerta de cierre de pedidos.

### ¿Qué no hará el sistema de información?

- ❖ El sistema no administra los productos una vez ingresados.
- ❖ No administra la creación de convenios del usuario con la empresa.
- ❖ No cuenta con una gestión de pagos.
- ❖ No permite crear borradores de los pedidos.
- ❖ No permite crear listas de compra o de pedido.
- ❖ No permite realizar pedidos mezclando a los distribuidores.