



---

# ABARROTES TIZIMÍN

---

Diseño de software



LICENCIATURA EN INGENIERÍA DE SOFTWARE

23/05/2023

DR. JOSÉ LUIS LÓPEZ MATINEZ

*INTEGRANTES:*

- Emir Alejandro Bellos Cruz
- Andrés de Jesús Turriza Euan
- Luis Javier Quintana Olivera
- Abraham Raymundo Cruz Colli

## Indice

|                                  |    |
|----------------------------------|----|
| Introducción.....                | 2  |
| Objetivo .....                   | 3  |
| Glosario de palabras .....       | 4  |
| Requerimientos funcionales ..... | 5  |
| Diagramas.....                   | 7  |
| Diagrama de clase .....          | 7  |
| Diagrama de secuencia.....       | 8  |
| Diagrama de estado.....          | 9  |
| Diagrama de colaboración.....    | 10 |
| Diagrama de actividad.....       | 11 |
| Diagrama de casos de uso.....    | 12 |
| Vistas del proyecto .....        | 13 |
| Ejemplo de funcionamiento .....  | 21 |
| Enlace video funcionamiento..... | 28 |
| Enlace repositorio GitHub.....   | 28 |
| Conclusiones.....                | 29 |

## Introducción

El presente documento tiene presente el software desarrollado para el gerente de la tienda "Abarrotes Tizimín", el cual ha sido diseñado para brindar apoyo en el proceso de ventas. Este software cumple con una serie de requerimientos funcionales que permiten registrar clientes, artículos, realizar compras y generar tickets de venta. Durante el desarrollo de este proyecto, se emplearon los patrones de diseño MVC, Observer y Singleton, que desempeñaron un papel fundamental en la estructura y funcionalidad del programa.

En relación con los artículos, se han definido los siguientes atributos: Identificador, nombre del artículo, precio al público, precio del proveedor y cantidad total de existencia. Estos datos permiten gestionar adecuadamente el inventario de productos. De igual manera, agregamos el sistema de puntos, los cuáles se agregan al momento de agregar un producto, nosotros decidimos los puntos que genera ese artículo, y al momento de la compra, al final todos los puntos de los artículos comprados se suman, los puntos se usan al pasar cierto mínimo de compra.

Para brindar una interfaz amigable y accesible, se ha utilizado javafx como librería gráfica, para desplegar un menú intuitivo que facilita la navegación y utilización de las funcionalidades mencionadas.

Los patrones de diseño MVC, Observer y Singleton fueron de vital importancia en este proyecto. Proporcionaron una estructura clara y modular, facilitando el desarrollo, mantenimiento y escalabilidad del software. Estos patrones permitieron la separación de responsabilidades, una comunicación eficiente entre componentes y una gestión adecuada de las instancias, contribuyendo al éxito y la calidad del programa desarrollado.

A través de este software, el gerente de "Abarrotes Tizimín" contará con una herramienta eficiente y confiable para gestionar sus ventas, agilizando los procesos y mejorando la atención al cliente.

## Objetivo

El objetivo de este documento es presentar el software desarrollado para satisfacer las necesidades del gerente de la tienda "Abarrotes Tizimín" en cuanto a la gestión de ventas. El software cumple con los requerimientos funcionales establecidos, los cuales incluyen la capacidad de registrar clientes, artículos y realizar compras. Además, el sistema genera un ticket al finalizar cada compra, mostrando los detalles de los productos adquiridos, verificando su existencia y calculando el importe total.

El documento proporciona capturas de pantalla del software, acompañadas de sus descripciones, para ilustrar las distintas funcionalidades y características implementadas.

Con este documento, se busca brindar al gerente de "Abarrotes Tizimín" una herramienta efectiva y eficiente para optimizar sus procesos de venta, mejorar la gestión de clientes y artículos, y agilizar la generación de tickets de compra.

## Glosario de palabras

**Java:** Es un lenguaje de programación orientado a objetos que se utiliza ampliamente en el desarrollo de aplicaciones de software. Java es conocido por su portabilidad y se utiliza en una amplia gama de aplicaciones, desde aplicaciones de escritorio hasta aplicaciones web y móviles.

**JavaFX:** Biblioteca gráfica de Java utilizada para desarrollar interfaces de usuario (UI) ricas y atractivas. JavaFX proporciona una amplia gama de controles y herramientas para la creación de interfaces de usuario interactivas.

**MVC (Modelo-Vista-Controlador):** Patrón de diseño arquitectónico utilizado en el desarrollo de aplicaciones de software. MVC divide una aplicación en tres componentes principales: el Modelo, que representa los datos y la lógica de la aplicación; la Vista, que es responsable de la presentación de la interfaz de usuario; y el Controlador, que maneja las interacciones del usuario y coordina la comunicación entre el Modelo y la Vista.

**Singleton:** Patrón de diseño creacional que garantiza la existencia de una única instancia de una clase en un programa. El patrón Singleton se utiliza cuando se requiere un acceso global a una instancia única de una clase y se utiliza ampliamente en situaciones donde se necesita compartir información o recursos comunes.

**Observer:** Patrón de diseño de comportamiento que establece una relación de dependencia uno a muchos entre objetos. En este patrón, un objeto llamado Observable (sujeto) notifica automáticamente a varios objetos Observers (observadores) cuando cambia su estado. El patrón Observer permite la actualización y sincronización de información en tiempo real y se utiliza en situaciones donde se requiere una comunicación desacoplada entre componentes de un programa.

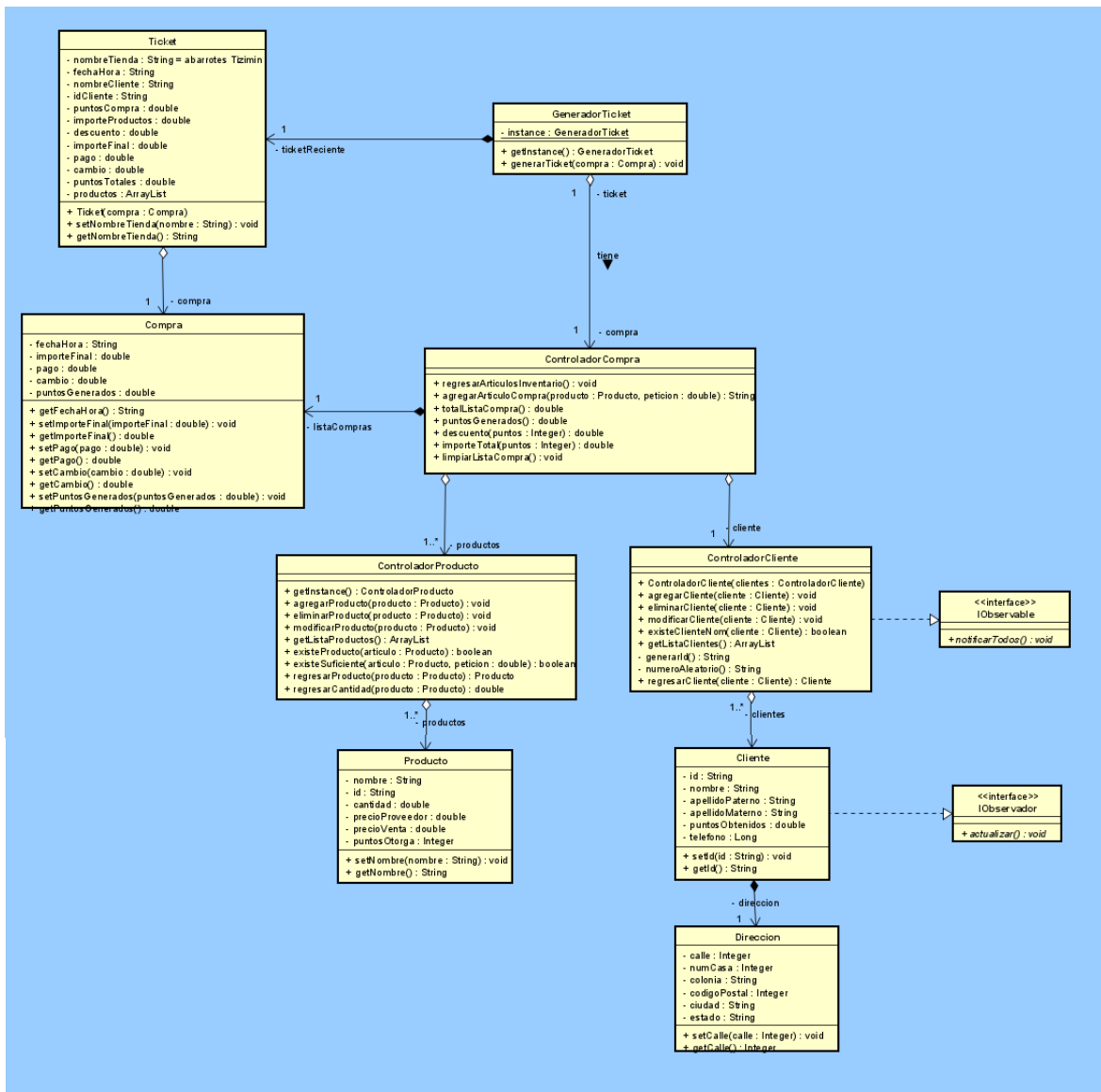
## Requerimientos funcionales

| No.  | MoSCoW     | Requerimiento funcional                   | Descripción  |
|------|------------|---|--|
| RF01 | Most have  | Registro de clientes                      | La aplicación debe permitir al usuario Registrar un Cliente con los siguientes datos: <ul style="list-style-type: none"> <li>• Identificador del Cliente</li> <li>• Nombre</li> <li>• Apellido paterno</li> <li>• Dirección (Calle, Número, Colonia, CP, Ciudad, Estado, Teléfono)</li> </ul>                                    |
| RF02 | Most have  | Sistema de generado de ticket de venta    | La aplicación debe proporcionar un Ticket al finalizar la compra, el Ticket debe mostrar: <ul style="list-style-type: none"> <li>• Nombre del Cliente</li> <li>• Identificador del Cliente</li> <li>• Nombre del Artículo</li> <li>• La Cantidad y el Precio de lo adquirido</li> <li>• El importe total de la compra</li> </ul> |
| RF03 | Most have  | Sistema de carga de productos disponibles | La aplicación debe permitir al usuario realizar una venta, en donde se proporcione el o los artículos comprados y la cantidad.   |
| RF04 | Most have  | Registro y control de artículos           | La aplicación debe permitir al usuario Registrar un Artículo con los siguientes datos: <ul style="list-style-type: none"> <li>• Nombre del Artículo</li> <li>• Precio al publico</li> <li>• Cantidad total en existencia.</li> </ul>   |
| RF05 | Most have  | Registro y control de clientes            | El sistema contará con un apartado para cargar clientes frecuentes de tienda.  |
| RF06 | Could have | Menú principal                            | El sistema desplegará un menú principal donde se seleccionará si quiere realizar la venta, control de artículos y el control de clientes clientes.   |
| RF07 | Most have  | Interfaz gráfica                          | El programa debe tener una interfaz gráfica amigable y fácil de usar para que el usuario pueda usarlo de la mejor manera posible.  |
| RF08 | Could have | Sistema de puntos para clientes           | El sistema dará puntos a los clientes por cada compra, cada artículo tiene un valor en los puntos diferente y al   |

|  |  |  |   |
|--|--|--|---|
|  |  |  | canjear los puntos se le aplicará un descuento al total de su compra. |
|--|--|--|---|

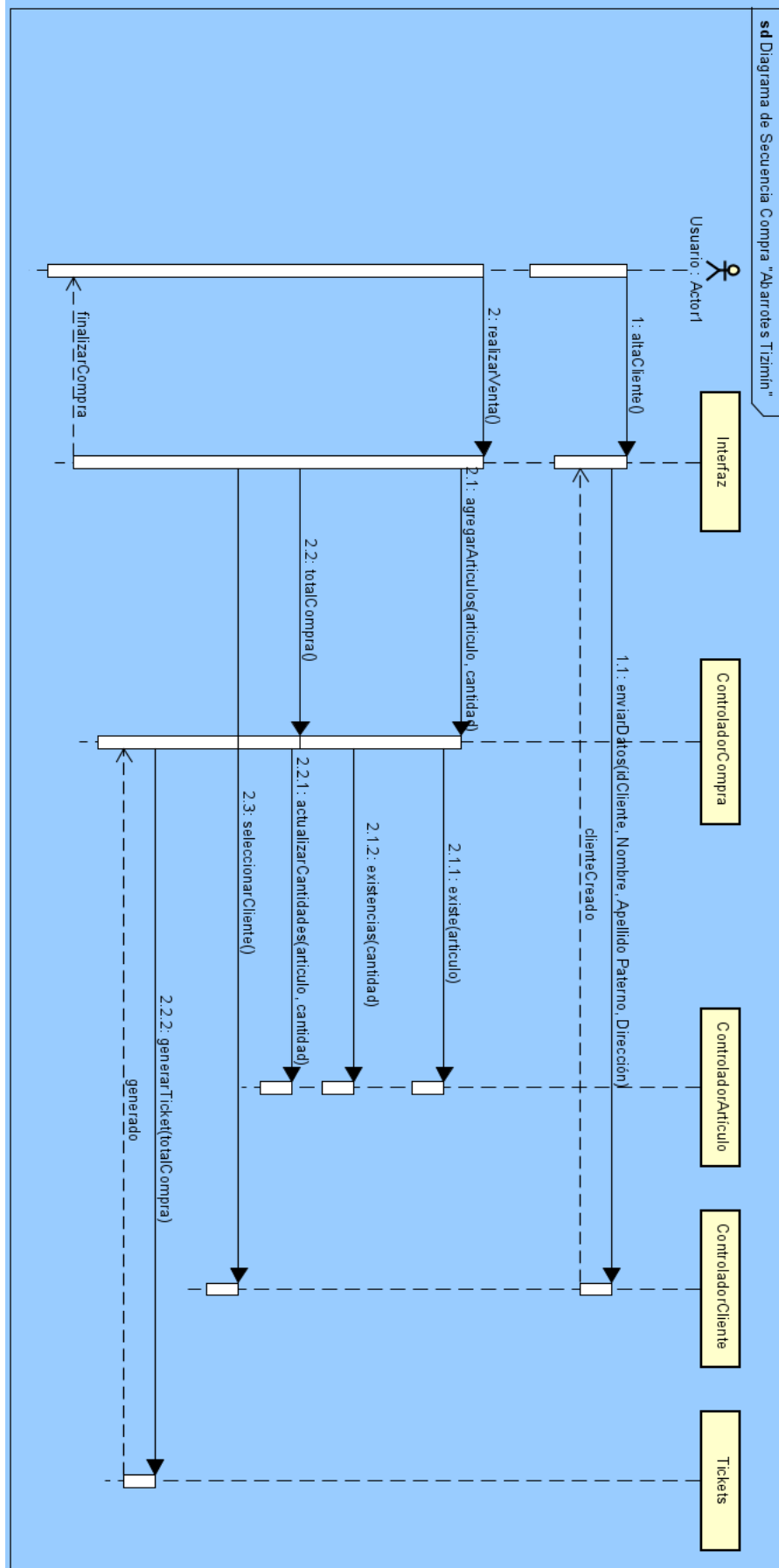
# Diagramas

## Diagrama de clase





### Diagrama de secuencia. Compra y agregado de clientes



## Diagrama de estado. Compra

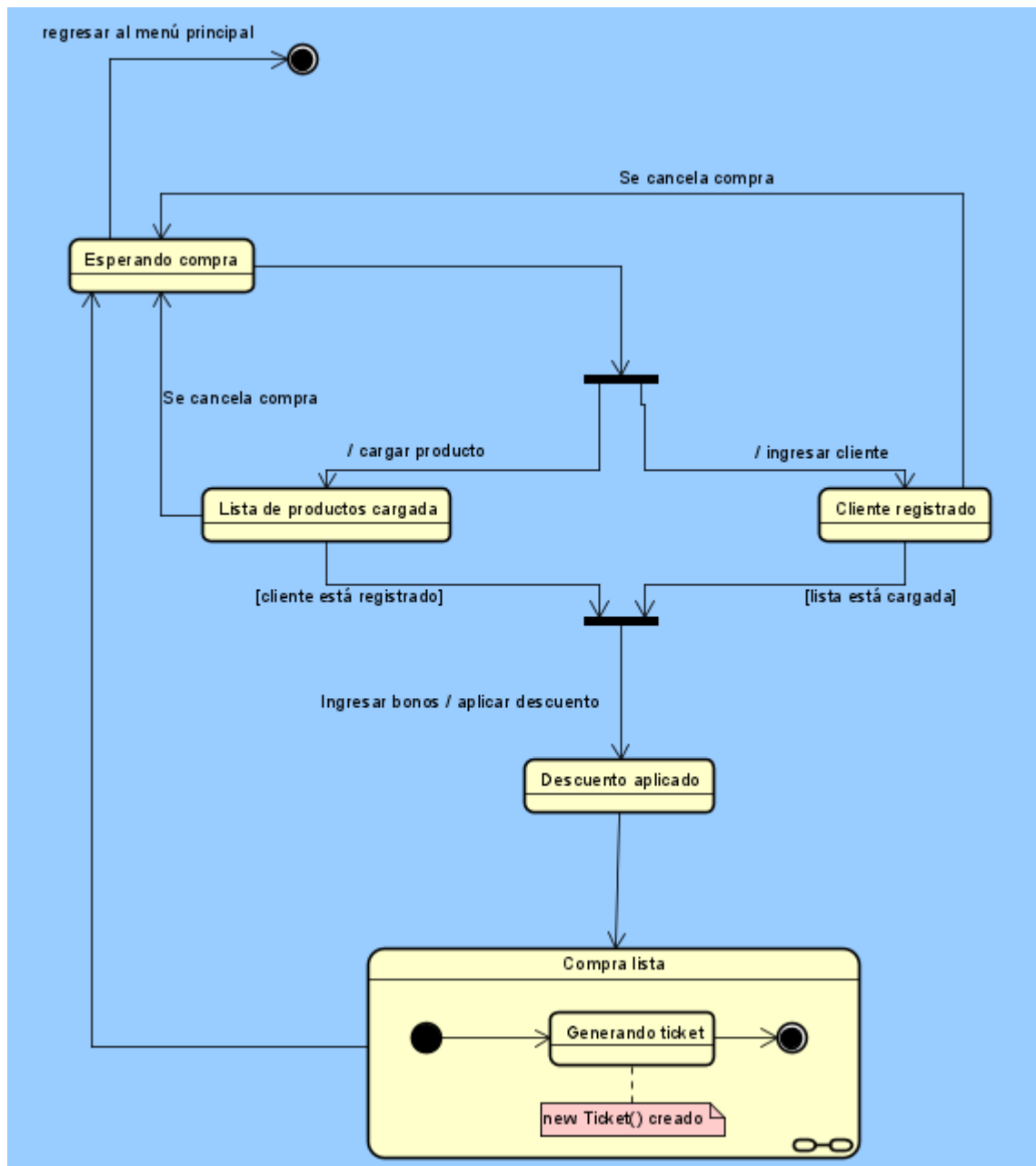
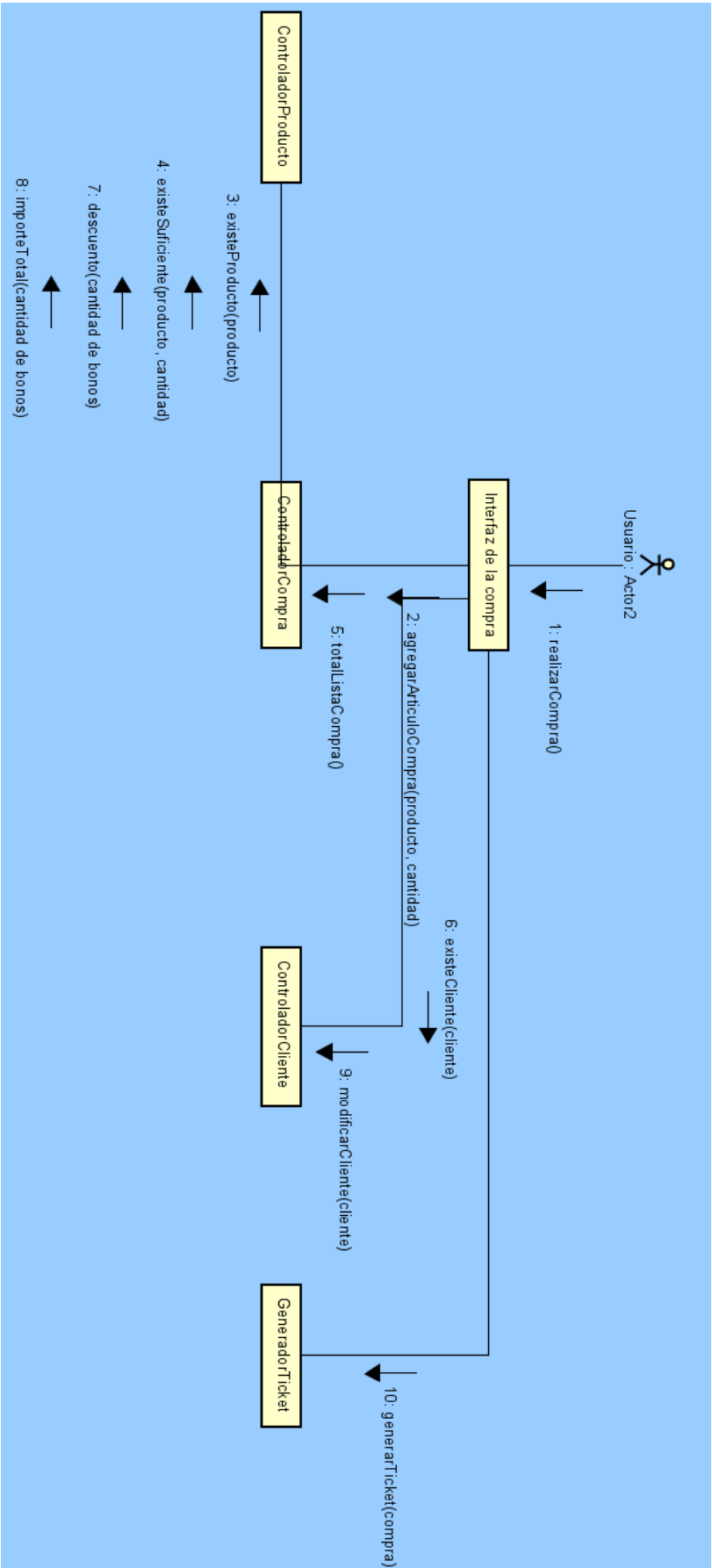
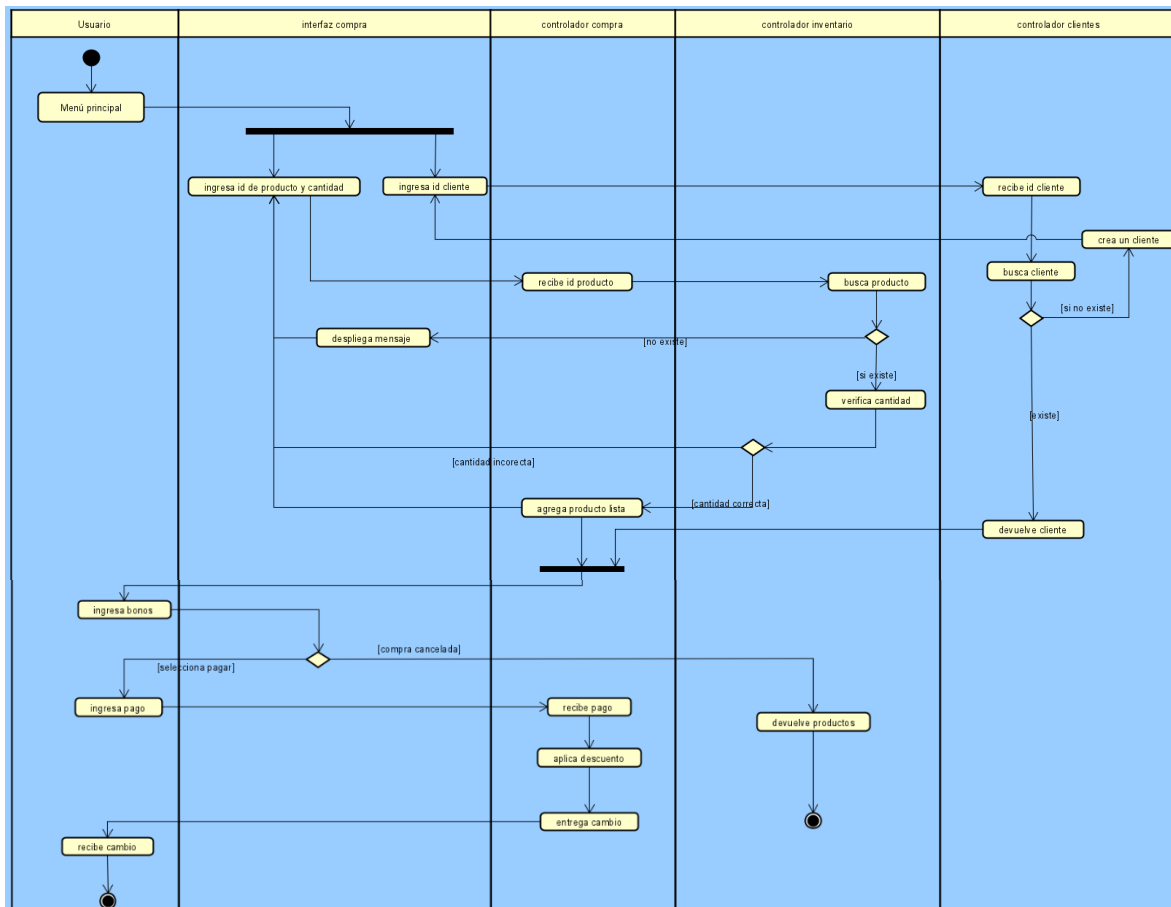


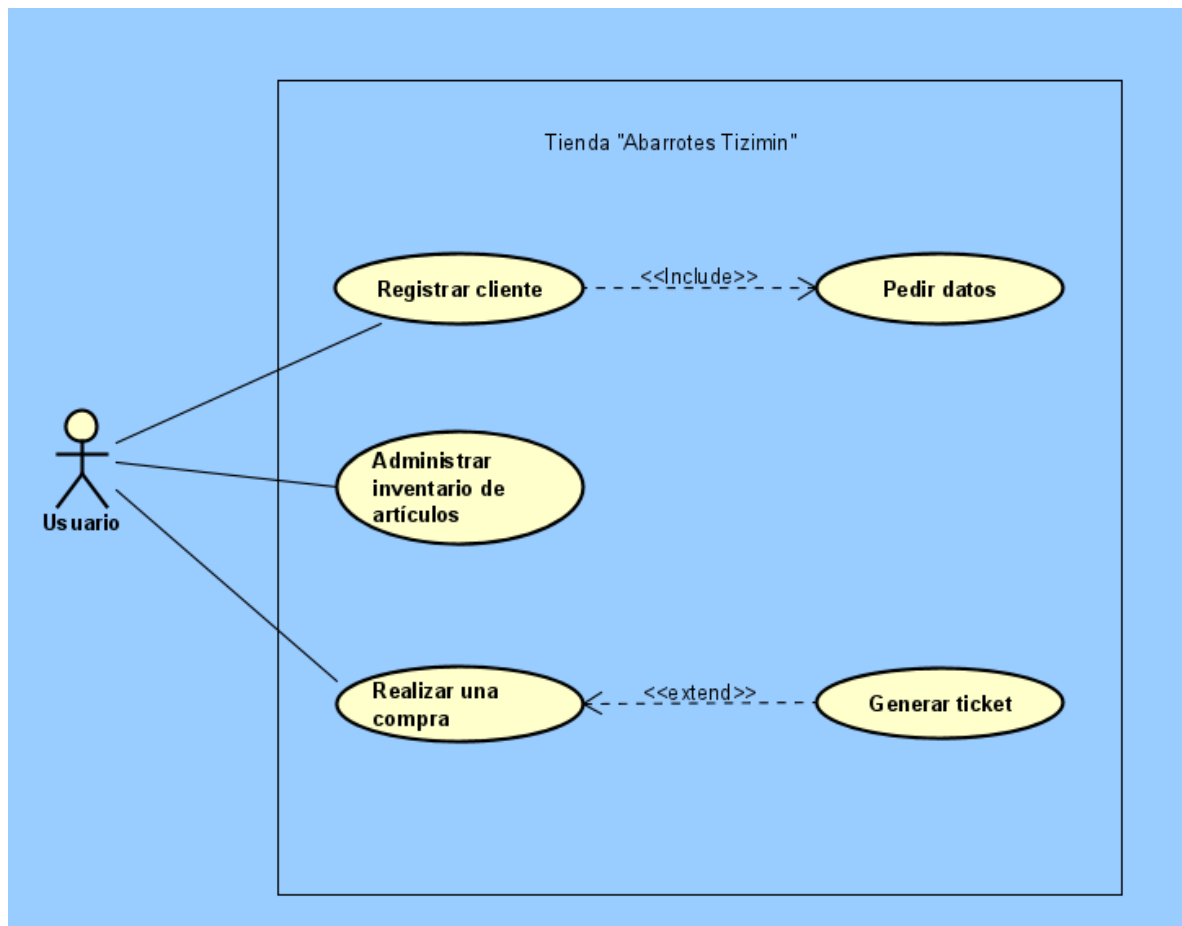
Diagrama de colaboración. Compras



## Diagrama de actividad. Compra

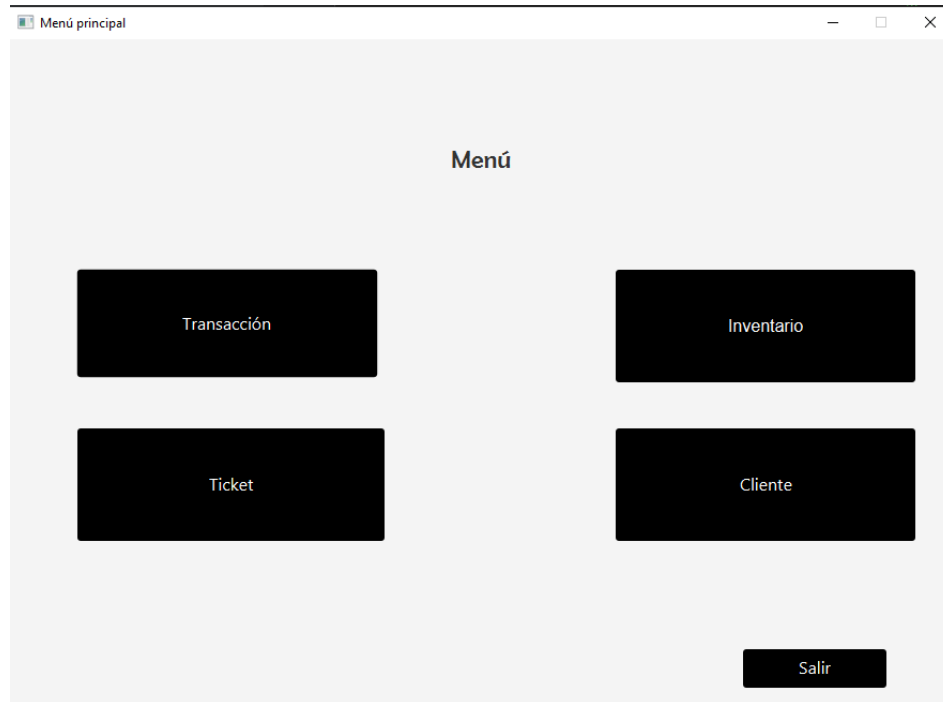


## Diagrama de casos de uso



## Vistas del proyecto


### Vista del Menú:



### Vista Inventario:



## Vista Agregar Producto:

 Agregar producto

— □ ×

Agregar producto

Nombre:

ID producto:

Cantidad:

Precio proveedor:


Precio venta:

Puntos:

Guardar

Regresar

## Vista Modificar Producto:

 Modificar producto

— □ ×

Modificar producto

ID producto:

Buscar

Nombre:

Cantidad:

Precio proveedor:

Precio venta:

Puntos:

Guardar

Regresar

**Vista Eliminar Producto:**

Eliminar producto

ID producto:

Buscar

Producto

Nombre:

Cantidad:

Precio proveedor:

Precio venta:

Puntos:

Eliminar

Regresar

**Vista Ver Inventario:**[illegible]



### Vista Compra:

[illegible]

## Vista Ticket:



## Vista Administrador Cliente:

Realizar compra

Administrador cliente

Agregar Cliente

Modificar cliente

Eliminar cliente

Ver clientes

Notificar clientes

Menú

## Vista Agregar Cliente:

Agregar cliente

Nombre:

Apellido Paterno:

Apellido materno:

Teléfono:

Calle:

Num :

Colonia:

CP :

Ciudad:

Estado:

Guardar

Regresar

### Vista Modificar Cliente:

Modificar cliente

ID Cliente:

Nombre:

Apellido paterno:

Apellido materno:

Teléfono:

Calle:

Num:

Colonia:

C.P. :

Ciudad:

Estado:

### Vista Eliminar Cliente:

Eliminar cliente

ID Cliente:

Nombre:

Apellido paterno:

Apellido mater...

Teléfono:

Calle:

N...

Colonia:

Ciudad:

Estado:

C.P. :

### Vista Lista de Clientes:

[illegible]

## Ejemplo de funcionamiento

### Agregar un producto:

## Agregar producto

|               |                                   |                   |                                 |
|---------------|-----------------------------------|-------------------|---------------------------------|
| Nombre:       | <input type="text" value="coca"/> | ID producto:      | <input type="text" value="2"/>  |
| Cantidad:     | <input type="text" value="5"/>    | Precio proveedor: | <input type="text" value="10"/> |
| Precio venta: | <input type="text" value="12"/>   | Puntos:           | <input type="text" value="1"/>  |

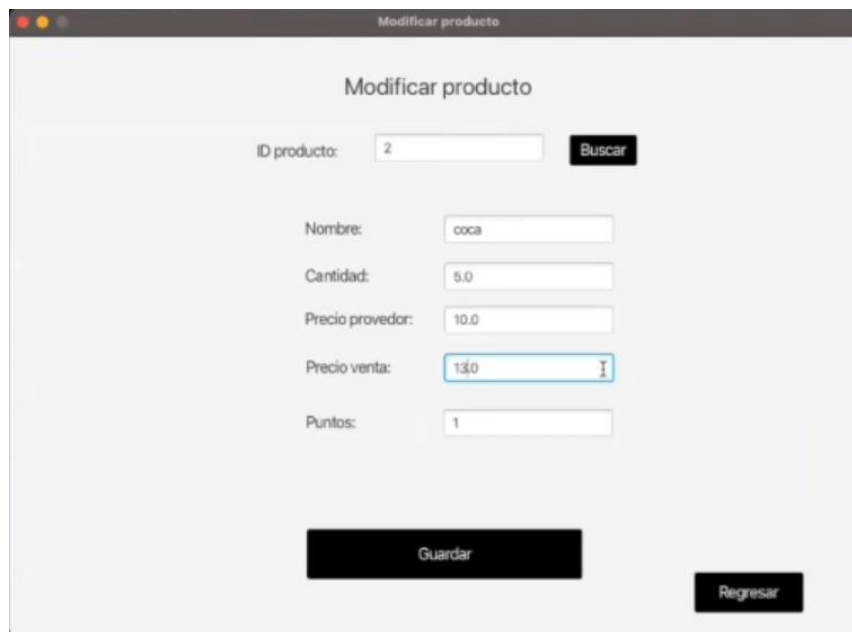
Guardar

Regresar

## Checar el inventario de los productos agregados

[illegible]

## Modificación de un producto:



A screenshot of a web application window titled "Modificar producto". The window has a light gray background and a dark gray title bar with standard macOS window controls. The main content area is titled "Modificar producto" in a bold, dark font. Below the title, there is a form with several input fields and buttons. The "ID producto:" field contains the number "2", and a "Buscar" button is to its right. Below this, there are five more input fields: "Nombre:" with "coca", "Cantidad:" with "5.0", "Precio proveedor:" with "10.0", "Precio venta:" with "13.0", and "Puntos:" with "1". At the bottom of the form, there is a large "Guardar" button and a smaller "Regresar" button to its right.

Modificar producto

ID producto: 2 **Buscar**

Nombre: coca

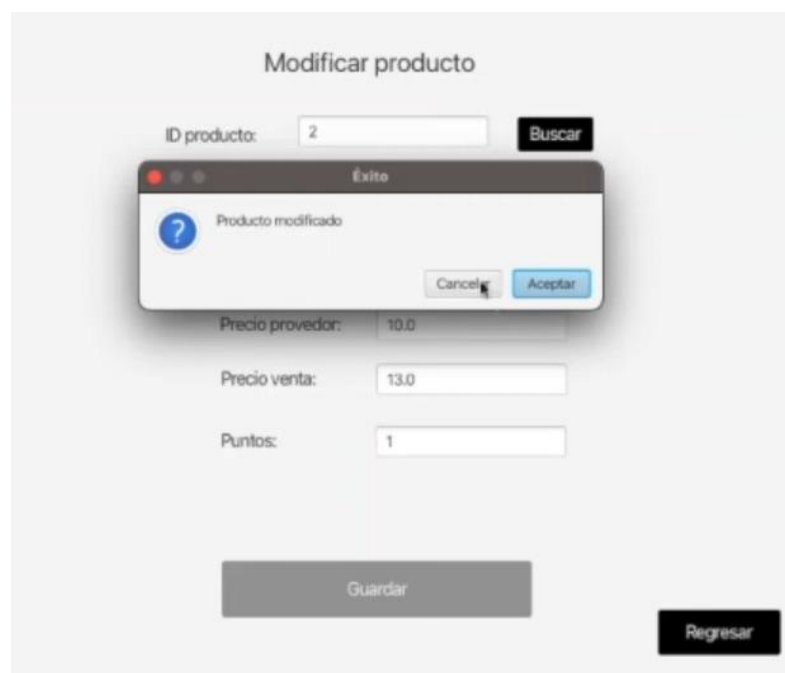
Cantidad: 5.0

Precio proveedor: 10.0

Precio venta: 13.0

Puntos: 1

**Guardar** **Regresar**



A screenshot of the same "Modificar producto" form, but with a success dialog box overlay. The dialog box is titled "Éxito" and contains a blue question mark icon and the text "Producto modificado". It has "Cancelar" and "Aceptar" buttons. The "Guardar" button in the background is now disabled (grayed out), and the "Regresar" button remains active. The "ID producto:" field still contains "2", and the "Buscar" button is still present. The other input fields ("Nombre:", "Cantidad:", "Precio proveedor:", "Precio venta:", "Puntos:") are still filled with their respective values.

Modificar producto

ID producto: 2 **Buscar**

**Éxito**

Producto modificado

Cancelar Aceptar

Precio proveedor: 10.0

Precio venta: 13.0

Puntos: 1

Guardar **Regresar**

## Checamos inventario nuevamente y vemos que realmente se haya modificado

[illegible]

## Eliminación de un producto

Eliminar producto

Eliminar producto

ID producto:

Producto

Nombre: coca

Cantidad: 5.0

Precio proveedor: 10.0

Precio venta: 13.0

Puntos: 1

Eliminar

Regresar



### Agregar un cliente:

Agregar cliente

Nombre:

abraham

Apellido Paterno:

cruz

Apellido materno:

coll

Teléfono:

9861021373

Calle:

40

Num :

380

Colonia:

centro

CP:

97700

Ciudad:

tizmin

Estado:

yucatan

Guardar

Regresar

[illegible]

[illegible]

### Modificamos un Cliente, posteriormente lo eliminamos y verificamos en la lista de Clientes

The screenshot displays a web application interface for modifying a client. At the top, a dark header bar contains the text "Modificar cliente". Below this, the main content area has a title "Modificar cliente" centered. A modal dialog box is open in the center, titled "Éxito" (Success), with a blue question mark icon and the text "Cliente modificado" (Client modified). The dialog has "Cancelar" (Cancel) and "Aceptar" (Accept) buttons. In the background, the "Modificar cliente" form is visible, featuring a "Buscar" (Search) button and input fields for "ID Cliente:" (2115623), "Nombre:" (partially visible), "Apellido paterno:" (partially visible), "Apellido materno:" (olivera), "Teléfono:" (2222222222), "Calle:" (1), "C.P.:" (97700), "Ciudad:" (Cancun), and "Estado:" (Quintana Roo). A "Guardar" (Save) button is at the bottom center, and a "Regresar" (Return) button is at the bottom right.

Eliminar cliente

ID Cliente: 2115623

Buscar

Nombre: Luis

Apellido paterno: quintana

Apellido materno: olivera

Teléfono: 222222222

Éxito

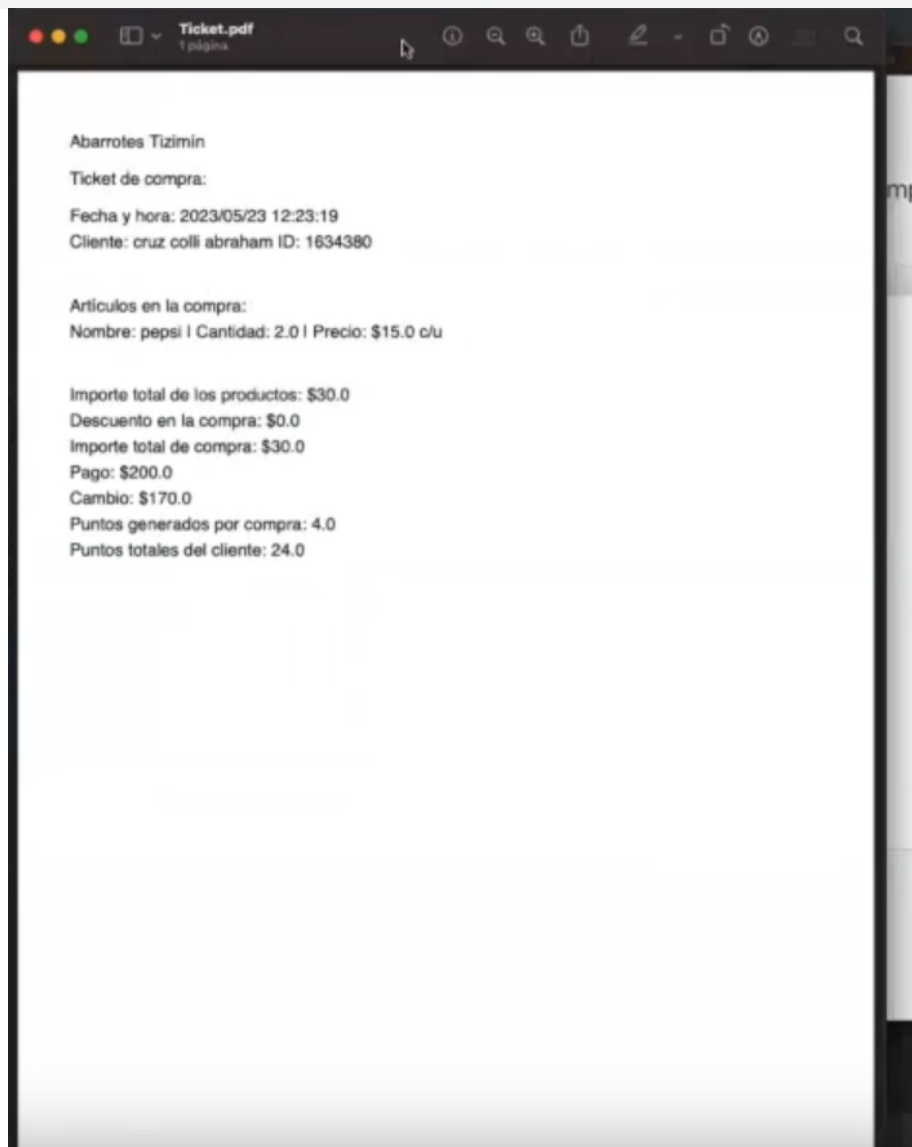
Cliente eliminado

Cancelar Aceptar

Eliminar

Regresar

[illegible]



Enlace video funcionamiento

<https://www.youtube.com/watch?v=3qmqbNSowoc>

Enlace repositorio GitHub

<https://github.com/AndyTue/Abarrotes-Tizim-n>

## Conclusiones

En conclusión, el software desarrollado cumple con los requerimientos establecidos, brindando al gerente de "Abarrotes Tizimín" una herramienta poderosa y confiable para gestionar eficientemente las ventas de su tienda.

Con este software, el gerente de "Abarrotes Tizimín" puede mejorar la eficiencia y la precisión en la gestión de clientes y artículos, así como obtener una visión clara de las transacciones realizadas. Al automatizar y simplificar estas tareas, se proporciona un mayor nivel de servicio al cliente y se fomenta un ambiente de trabajo más productivo.