



Universidad Veracruzana

Área Económico-administrativa

Facultad de Estadística E Informática

“PROYECTO: CAFETERIA DE LA FEI ”

Francisco Xavier Aguilar Martinez

Fausto Mendez Peralta

Yasser Yahid Gapi Perez

Xalapa Enríquez, Ver. 24 de febrero de 2023

COFFIE FEI



1. Introducción	3
1.1. Propósito	3
2. Alcance del proyecto	4
2.2. Clases de Usuario y Características	5
3. Ambiente operativo.	5
4. Requisitos funcionales.....	5
5. Requisitos no funcionales	6
6. CASOS DE USO	8
7. ESTRATEGIA DE ENTREVISTAS Y ENCUESTAS	9
8. ESTRATEGIA DE ESTUDIO DE MERCADO	15
SOLUCIONES DE SOFTWARE SIMILARES:	15
DELIVERYS	16
Formas de monetización	17

1. Introducción

1.1. Propósito

El propósito de este documento es adjuntar todos los requisitos que va a tener el sistema, así como su propósito, alcance y el ambiente en el que será utilizado.

Lectores al que está destinado:

- Usuarios.
- Desarrolladores.

1.2. Convenciones

A continuación, en esta sección abordaremos las siglas, abreviaturas y lenguajes utilizados, esto con el fin de que se pueda entender el sistema con mayor facilidad.

NOMBRE	SIGNIFICADO	DESCRIPCION
FEI	Facultad de estadística e informática	Facultad de la Universidad Veracruzana
UV	Universidad Veracruzana	Universidad de carácter público

1.2. ANTECEDENTES

En la actualidad la cafetería de la FEI cuenta con un gran problema, derivado del gran número de estudiantes y el poco tiempo con el que se dispone para realizar la compra de alimentos, aunado a que el proceso de compra y elaboración es muy tardado en la actualidad este proceso puede llegar a tener una duración de 30 minutos, por lo cual muchos estudiantes prefieren no consumir los alimentos que se venden en dicho establecimiento

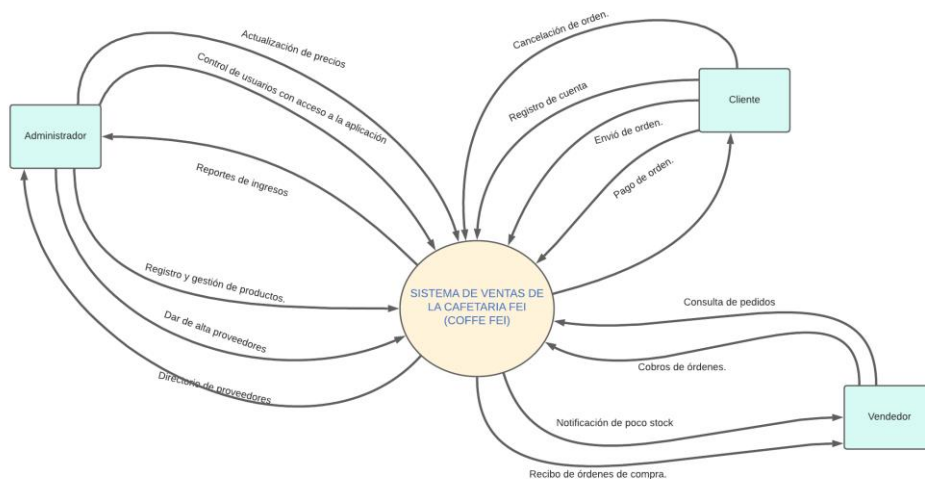
La cafetería escolar de la FEI solicita un sistema que ayude en la gestión de los productos que estarán a la venta, el cobro de órdenes, la creación y control de acceso de usuarios y sobre todo la creación y envío de órdenes de compra de los estudiantes para mejorar la productividad de dicha cafetería.

Este producto tiene como objetivo cumplir esa solicitud y será hecho desde cero.

2. Alcance del proyecto

El Sistema para la administración de la cafetería es un producto de software destinado a integrantes del proceso de compra de alimentos en la cafetería de la Facultad de Estadística e Informática; es decir, a administradores de cafetería, vendedor y estudiantes de la facultad, que permitirá la gestión de los pedidos con mayor rapidez. Con base a lo descrito en la problemática la excesiva inversión de tiempo y esfuerzo destinada a actividades administrativas, el retraso en la entrega de los pedidos de comida y el poco tiempo que tienen los estudiantes para pasar por la cafetería evidencia del seguimiento a aumentar la eficiencia de la cafetería, por los estudiantes que se dirigen a comer afuera y piden a domicilio suponen oportunidades de negocio para el desarrollo de un producto de software adecuado. Permitirá a los usuarios correspondientes administrar, de forma general, el control de pedidos para entrega de comida durante el periodo escolar en curso, facilitando la rápida administración y evitando la aglomeración en la cafetería y el consecuente aumentando oferta en comida de interés para los estudiantes y eficaz seguimiento de ventas de alimentos durante el quehacer escolar de la organización en cuestión. A diferencia de mantener a la cafetería de un proceso manual que aunque no es consumido por la solicitud de pedidos, mejorar el tiempo que se lleva en la entrega de comida, el producto de software proporcionará a los encargados la digitalización del monitoreo del proceso interno a través de la rápida y óptima construcción de medios para la consulta de comida basados en la posesión de la información válida sobre los alimentos ofertados del día, agilizando el pago de lo que consumen los estudiantes y la espera que llegan hacer para recoger su pedido y proponiendo una votación para elecciones de platillos para cada dos semanas

Se incluye el siguiente diagrama de contexto:



2.2. Clases de Usuario y Características

Administrador

Es la persona responsable de las acciones de gestión de personal, contratación e incorporación de vendedores, así como formarlos, además de llevar el control de personal y nomina, la realización de reportes y planeación de nuevas estrategias para mejorar los ingresos debe coordinarse con los proveedores y hacer pedidos cuando sea necesario, así como mantener registros actualizados de los ingresos y gastos diarios, semanales y mensuales.

Vendedor

Es la persona encargada de surtir las órdenes de compra, en su caso puede cancelar, debe actualizar el stock y el menú, así como realizar las entregas.

Cliente

Es la comunidad universitaria conformada por estudiantes, docentes y administrativos, los cuales consumen los productos de COFFIE FEI, ellos tienen a cargo la compra y el pago de los productos.

3. Ambiente operativo.

1. El sistema para la capa de aplicación debe estar creado con WPF.
2. Para la capa de base de datos de estar creada en Microsoft SQL Server.
3. La arquitectura en la cual se desarrollará el sistema debe ser Cliente-Servidor.
4. Para otorgar créditos es necesario que los clientes hayan sido registrados en el sistema previamente.
5. La API será creada en Java Script, y ejecutado con nodeJS.
6. La conexión de la API y Microsoft SQL server se hace con el uso de mssql.

4. Requisitos funcionales

1. El sistema debe administrar todos los productos ingresados a la cafetería.
2. El sistema debe poder modificar la descripción o eliminar un producto del stock.
3. El sistema deberá permitir a los clientes poderse registrarse.
4. El sistema deberá permitir al administrador registrar a nuevo personal de la cafetería.
5. El sistema debe notificar cuando un producto esté a punto de terminarse.
6. El sistema debe administrar los precios de los alimentos como bebidas.
7. El sistema debe realizar el cobro de las órdenes a través de diferentes medios de pago.
8. El sistema debe notificar cuando exista una nueva orden de compra.
9. El sistema debe permitir a los compradores tanto como al vendedor poder cancelar una compra.
10. El sistema debe mostrar el stock de alimentos de manera ordenada en un menú.
11. El sistema Deberá proporcionar filtros como precio o categorías.
12. El sistema debe crear un reporte diario sobre los ingresos que se hayan obtenido.
13. El sistema debe poder administrar los anuncios cuando se tiene colaboración con otros negocios de comida.

7. Requisitos no funcionales

Atributos de calidad externos

5.1. Disponibilidad

DIS01: El sistema estará disponible a partir de las 6:00 de la mañana hasta las 6:00 de la tarde.

DIS02: El sistema estará utilizando una nube para cumplir el 99% de tiempo de disponibilidad.

5.2. Integridad

INT01: El sistema resguardara la información introducida en una base de datos

5.3. Facilidad de Instalación

IFI01: El sistema solo necesitará el uso de un instalador.

5.4. Desempeño

DES01: La carga de las ventanas dependerá de la velocidad de internet del usuario, pero se planea que logre cargar en un máximo de 20 segundos, si no se logra, el sistema enviará un mensaje indicando al usuario que recargue la página.

DES02: La carga de cada pestaña no deberá tomar más de 10 segundos

5.6. Confiabilidad

CONF01: El sistema es estable y sencillo por lo que es poco probable que exista una falla, si se presenta una falla, esta deberá ser después de aproximadamente un año, o alguna actualización.

5.7. Seguridad (física)

SEGF01: Dependa de la configuración del equipo del usuario.

5.9. Seguridad

RNF01: La visualización de los datos debe estar protegida contra la adición, eliminación o modificación no autorizada de datos mediante el uso de la autenticación.

5.10. Usabilidad

USA01: Es un sistema practico, sencillo y factible, por lo que es muy fácil usarlo, una persona nueva puede aprender a usarlo en un plazo muy corto. Si un usuario se ausenta y vuelve después de mucho tiempo no sería necesario recordar o practicar el uso, pues solo hay pocas funciones sencillas para que hacen al software poco complejo.

5.11. Exactitud

EXAC01: El sistema cubre las necesidades del personal y facilita la administración y organización del proceso de compra - venta, además es cómodo para los usuarios.

Atributos de Calidad Internos

1. Eficiencia

EFI01: El sistema al ser sencillo no requiere un computador con gran potencial, usará los recursos necesarios para su funcionalidad.

2. Facilidad de modificación

FMOD01: Cualquier cambio deseado o nueva actividad se podrá hacer por personal autorizado, el sistema puede alterarse y cambiarse si así se desea.

3. Facilidad de reutilización

FREU01: El sistema se puede usar en computadoras de escritorio, portátiles, smartphones y tabletas.

4. Escalabilidad

ESC01: No hay un límite de usuarios, el sistema crecerá como sea necesario, no importa cuantos usuarios aparezcan se pueden registrar.

5. Facilidad de verificación

FVER01: Como el sistema lleva a cabo tareas simples para una institución, es fácil de probar y verificar que cumpla con las expectativas.

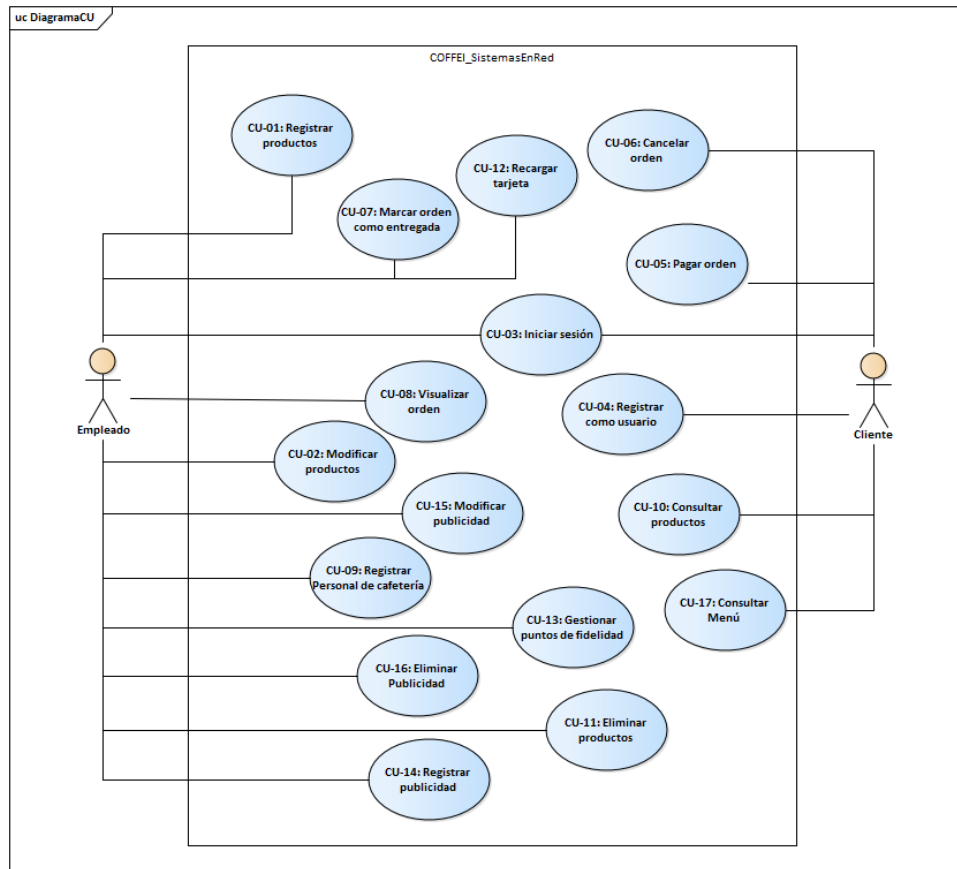
6. CASOS DE USO

Registro y gestión de productos.

- Reportes de ingresos
- Notificación de poco stock
- Registro de usuarios.
- Actualización de precios.
- Recibo de órdenes de compra.
- Cobros de órdenes.
- Envío de orden.
- Pago de orden.
- Cancelación de orden.
- Control de usuarios con acceso a la aplicación.
- Registro y gestión de productos.
 - CU-01: Registrar productos
 - CU-02: Modificar productos
 - CU-03: Iniciar sesión
 - CU-04: Registrar como usuario
 - CU-05: Pagar orden
 - CU-06: Cancelar orden
 - CU-07: Marcar orden como entregada
 - CU-08: Visualizar orden
 - CU-09: Registrar Personal de cafetería
 - CU-10: Consultar productos
 - CU-11: Eliminar productos
 - CU-17: Consultar Menú
 - CU-12: Recargar tarjeta
 - CU-13: Gestionar puntos de fidelidad
 - CU-14: Registrar publicidad
 - CU-15: Modificar publicidad
 - CU-16: Eliminar Publicidad

Diagrama de Casos de Uso.

Cliente WPF



7. ESTRATEGIA DE ENTREVISTAS Y ENCUESTAS

COFFE FEI ENTREVISTA

Para la entrevista que se llevó a cabo se manejó preguntas que estuvieran relacionadas a la opinión de los usuarios y determinaran si es necesario mejorar el rendimiento que ofrece la cafetería de la FEI, así como la información que tiene relación con el presupuesto y formas de pago, también se dieron algunas propuestas que podrían traer el sistema a diseñar y pueda cumplir con las expectativas de los desarrolladores, aunque esta entrevista tuvo poca asistencia a la opinión, los participantes aumentaran durante este periodo de análisis y retroalimentación de los requerimientos iniciales de este proyecto. Este sistema estará destinado a todos los estudiantes de facultad de informática y economía, así como maestros de la facultad y personal de mantenimiento de las dos facultades.

RESULTADOS DE LA ENTREVISTA

Consideras que es necesario optimizar los procesos de la cafeteria de la facultad

[Más detalles](#)



2. ¿Cuanto tiempo dispones para comer en la facultad?

22 Respuestas

ID ↑	Nombre	Respuestas
1	anonymous	10 m
2	anonymous	10 minutos
3	anonymous	10 min
4	anonymous	10 a 15 minutos
5	anonymous	1 hora
6	anonymous	10 minutos
7	anonymous	20 minutos
8	anonymous	20 minutos
9	anonymous	10 minutoe
10	anonymous	5 min

3. ¿Cuál es tu presupuesto para comer en la facultad?

22 Respuestas

1	anonymous	40
2	anonymous	50 pesos
3	anonymous	50
4	anonymous	30
5	anonymous	70 pesos
6	anonymous	50
7	anonymous	\$ 50
8	anonymous	50 pesos
9	anonymous	50
10	anonymous	20
11	anonvmous	50

4. ¿Usarías una app que te permita ordenar tu comida y solo pasar a recogerla a la cafetería?

22 Respuestas

1	anonymous	Ci
2	anonymous	Tal vez
3	anonymous	Si
4	anonymous	mmm chi
5	anonymous	Si
6	anonymous	si
7	anonymous	Si
8	anonymous	si
9	anonymous	Si
10	anonymous	si
11	anonymous	si

5. ¿Te gustaría poder consultar la comida de la cafetería sin tener que salir de tu salón?

22 Respuestas

1	anonymous	Ci
2	anonymous	Sí
3	anonymous	Si
4	anonymous	Chi, aunque ya no hay cafetería xd
5	anonymous	Si
6	anonymous	si
7	anonymous	Si
8	anonymous	si
9	anonymous	Si
10	anonymous	si
11	anonymous	si

¿Pasas a comer a otro lugar cuando la cafetería esta llena o tiene una fila larga?


[Más detalles](#)


● si	17
● no	4
● a veces	7



¿Te gustaría tener la opción de poder votar para elegir la comida del día?

[Más detalles](#)


 Información



 si	16
 no	6
 tal vez	6



10. ¿Te parecería bien pagar diario o semanalmente lo que consumes de comida?

[Más detalles](#)


 Información

 diario	21
 semanalmente	7



11. ¿Te interesaría una función que te permita saber el promedio de calorías que consumes?

[Más detalles](#)

 Información

 Si	11
 No	13



12. ¿Padeces alguna discapacidad visual, auditiva o motriz?

[Más detalles](#)

Visual	9
Auditiva	0
Motriz	0
Otras	5



13. ¿Comes, comerías o te gustaria saber el menu de la cafeteria de otras facultades?

[Más detalles](#)

[Información](#)

Si	15
No	9



14. ¿Te gustaria usar o usarias paypal o algun medio electronico para pagar en cafeteria?

[Más detalles](#)

[Información](#)

Si	21
No	3



8. ESTRATEGIA DE ESTUDIO DE MERCADO

Público objetivo: Tomando en cuenta el rango de edad de los estudiantes de la FEI debido a que son los principales consumidores de la cafetería deducimos que este rango va de los 18 a los 26 años, si bien, hay excepciones este rango es el que abarca la mayoría de los estudiantes de la Facultad de Estadística e Informática.

Como rango socioeconómico tomamos en cuenta los resultados de nuestra estrategia de levantamiento de requerimientos y averiguamos que el promedio que los alumnos de la FEI pueden gastar en la cafetería va de los 40 a los 70 pesos, a pesar de ser un poco disparejo la cafetería cuenta con diferentes precios. Si bien con los resultados demostrados en la encuesta la mayoría de los estudiantes no cuentan con ninguna discapacidad, hubo una respuesta donde alguien marcaba que contaba con una, así que deben ser tomados en cuenta para la aplicación para ser amigable el uso con cualquier tipo de usuario.

El público objetivo es un público joven, familiarizado con la tecnología el cual puede facilitar el desarrollo y diseño de la aplicación pudiendo implementar opciones como pagar con PayPal, u otros métodos de pago para así hacer más eficaz la aplicación.

Este sistema será escalable, una vez que el sistema presente un buen desempeño del punto de origen que es la facultad de economía y informática se presentará a integrar a las facultades en cercanía para poder tener un menú más amplio y variado que se le puede ofrecer a los estudiantes como maestros y personal de la uv en el mantenimiento.

Luego de eso paso el sistema pasara a establecer en toda la uv aumentando el incremento de los usuarios en el sistema así como más puntos donde podría pasar el estudiante por su comida.

SOLUCIONES DE SOFTWARE SIMILARES:

Actualmente en el mercado de software para cafeterías y de repartición de comida rapida existen muchas alternativas, entre las cuales se encuentran las siguientes, para terminales de puntos de venta, se encuentran los softwares de Poster, Sicar, Venty, Loyverse, SoftRestaurant, MRC software y los cuales son software privativo, para sistemas operativos de Windows 10 y en su mayoría no poseen aplicación móvil. Por otro lado en el mundo de distribución de comida rápida se encuentran las alternativas ya conocidas de Rappi, Ubber Eats, DoorDash, DidiFood y 7Now de la compañía 7eleven.

Si bien todas estas alternativas representan una gran competencia a la propuesta de solución de COFFIE para el software de cafetería, la diferencia principal radica en que la propuesta desarrollada por el equipo de desarrollo, es que esta integrara ambos en un mismo producto, es decir será un sistema POS y un sistema de delivery enfocado 100% en el entorno de cafeterías de la Universidad Veracruzana.

Es decir es un producto de exclusividad para la comunidad estudiantil para el modelo de negocio de las cafeterías de dicha entidad estudiantil, a diferencia de las aplicaciones de la competencia anteriormente mencionadas, las cuales no están diseñadas ni enfocadas en dicha exclusividad de producto universitario, eso sin contar que, contara con ciertas funcionalidades exclusivas que los otros productos no cuentan, como es una votación para el menú del día siguiente de la cafetería,

además de ser de un rango de alcance 100% a nivel de facultades y no fueras de ellas, siendo esto posible mediante el uso de las matrículas estudiantiles.

Fuentes de referencia:

TERMILES DE PUNTOS DE VENTA

Poster

<https://joinposter.mx/>

Sicar

<https://www.sicar.mx/>

Vendty

<https://www.vendty.com/>

Loyverse

<https://loyverse.com/es>

SoftRestaurant

<https://softrestaurant.com/>

MRCSoftware

<https://www.mrcomanda.com/web/index.php>

DELIVERYS

Rappi

<https://play.google.com/store/apps/details?id=com.grability.rappi>

UbberEats

<https://play.google.com/store/apps/details?id=com.ubercab.eats>

7 Now : Food Delivery & Alcohol

https://play.google.com/store/apps/details?id=com.seven.eleven.phoenix&hl=es_MX&gl=US

DiDiFood

<https://play.google.com/store/apps/details?id=com.xiaojukeji.didi.global.customer>

DoorDash

<https://play.google.com/store/apps/details?id=com.dd.doordash>

FORMAS DE MONETIZACIÓN

El sistema contará con dos métodos de pago, los cuales son efectivo, PayPal y tarjeta de prepago propia de la cafetería.

El usuario podrá seleccionar dicho método de pago y llenar los campos correspondientes en PayPal o la tarjeta de prepago, en el caso del pago en efectivo este se realizará en las instalaciones de la cafetería.

La tarjeta de prepago es una tarjeta la cual se asigna al momento del registro en el sistema de la cafetería y se va recargando con dinero electrónico el cual disminuirá después de cada compra o aumentará después de cada recarga.

El sistema verificará que los datos sean correctos y que se cuente con un saldo disponible, posteriormente se le informará al usuario si procede o no su compra.

9. MODELO DE DOMINIO

class DiagramaDominio

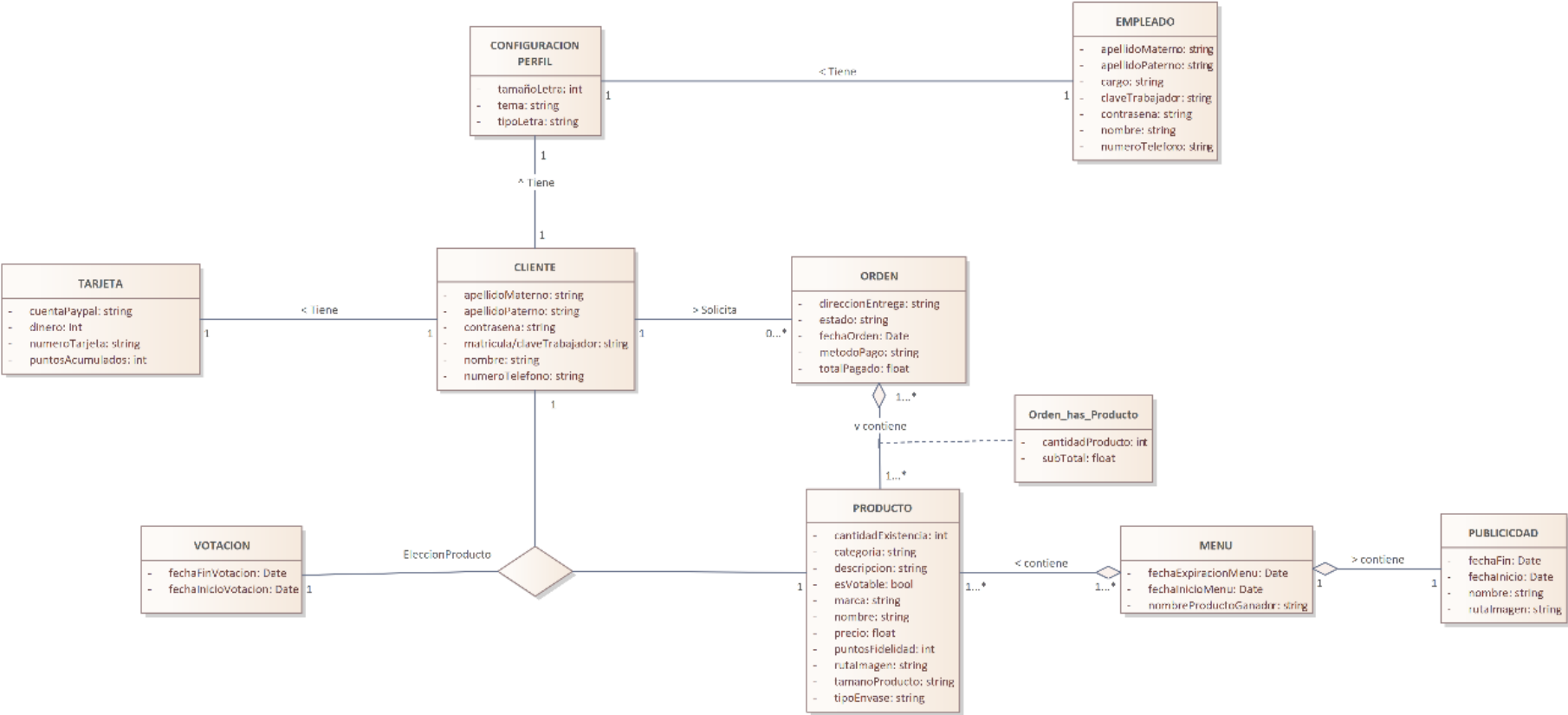
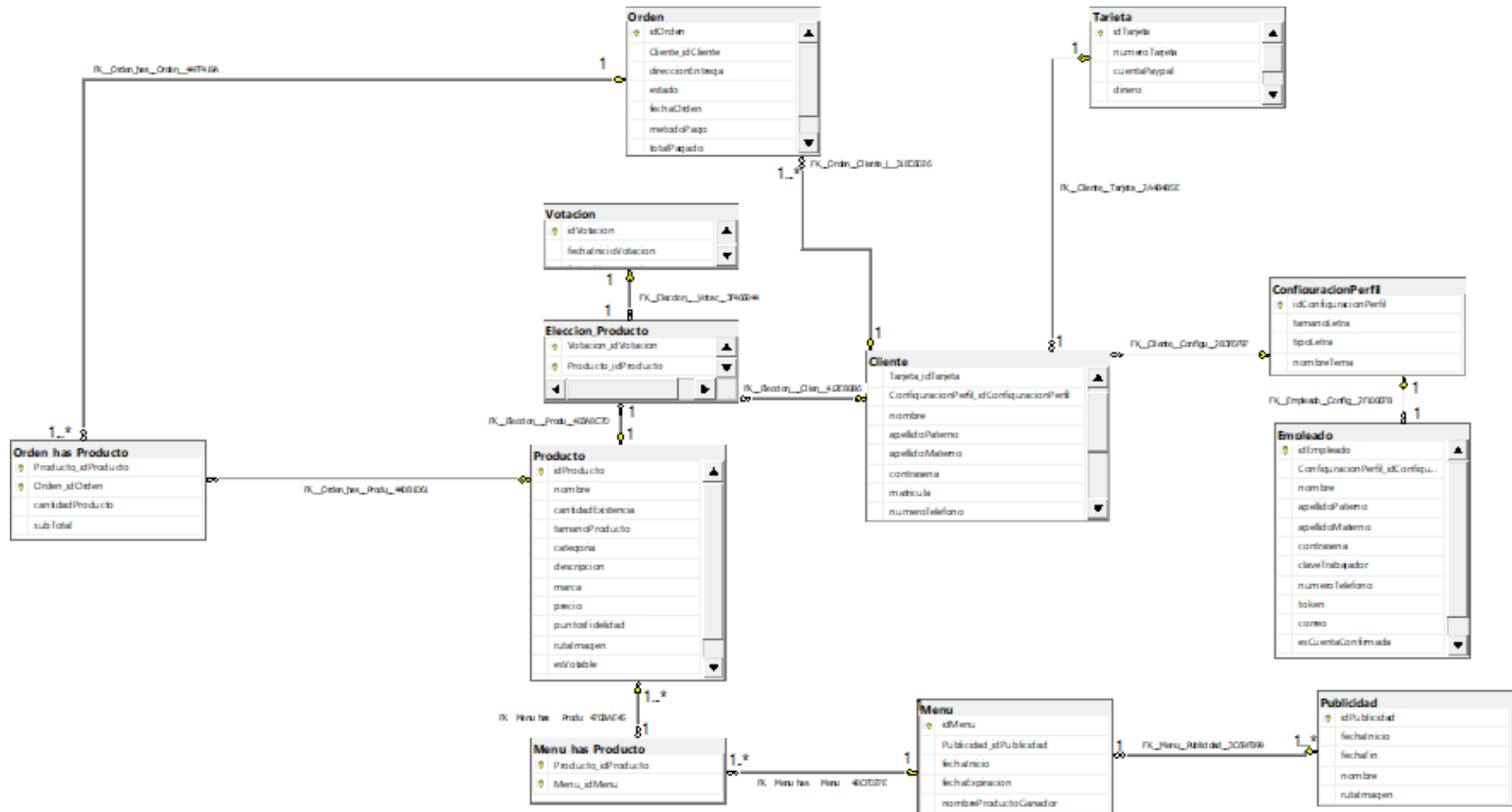


DIAGRAMA RELACIONAL DE LA BASE DE DATOS



10. DESCRIPCIONES DE CASOS DE USO

ID:	CU-05
Nombre:	Pagar orden
Responsable:	Francisco Xavier Aguilar Martínez
Fecha de actualización:	05/03/2023
Descripción:	Se paga una orden
Actor(es):	Comprador
Disparador:	Seleccionar botón
Precondiciones:	PRE-1: La orden no debe estar vacía
Flujo Normal:	<p>API Base de Datos:</p> <ol style="list-style-type: none">1. El sistema recopila de la base de datos los precios de los ítems escogidos. <p>API Front-End:</p> <ol style="list-style-type: none">2. El sistema muestra los precios de los ítems escogidos en pantalla. El sistema muestra diferentes opciones de pago y los botones "Pagar" y "Salir" (ver EX1) (ver FA 1.1). <p>API Base de Datos:</p> <ol style="list-style-type: none">3. El actor selecciona el método de pago "Efectivo". <p>API Front-End:</p> <ol style="list-style-type: none">4. El actor selecciona el botón "Pagar". <p>API Base de Datos:</p> <ol style="list-style-type: none">5. El sistema realiza la suma total del total a pagar. <p>API Front-End:</p> <ol style="list-style-type: none">6. El sistema muestra el total a pagar en pantalla. <p>API Base de Datos:</p> <ol style="list-style-type: none">7. El sistema agrega la orden en órdenes pendientes para ser visualizadas por el personal de cafetería.

	<p>API Front-End:</p> <p>8. El actor selecciona el botón "Salir". Termina el caso de uso.</p>
Flujos Alternos:	<p>FA 1.1 Cancelación de operación</p> <p>API Front-End:</p> <p>1.El sistema despliega una ventana flotante que muestra la leyenda "¿Estás seguro de que quiere salir?" con opciones "Sí" y "No".</p> <p>API Base de Datos:</p> <p>2. El actor selecciona la opción "Sí" y termina el caso de uso.</p> <p>API Front-End:</p> <p>3. Si el actor selecciona la opción "No", vuelve al flujo normal en el paso 2</p> <p>FA 2.1 Selección de botón Salir</p> <p>Termina el caso de uso.</p>
Excepciones:	<p>EX1. No hay conexión con la base de datos</p> <p>API Front-End:</p> <p>1. El sistema muestra el mensaje "No hay conexión con la base de datos".</p> <p>API Base de Datos:</p> <p>2. El actor selecciona la opción "Cerrar".</p> <p>3. Termina el caso de uso.</p>
Postcondiciones:	POST-1 Se agrega la orden en ordenes pendientes
Incluye:	Ninguno
Extiende:	Ninguno

GUI CU05 PagarOrden

Pagar Orden

Resumen de compra

Metodo de pago

Pagar

Cancelar

Cantidad

Nombre

Precio

Total

Prueba

sd Pagar Orden

Interacción Actor y Front-End

```

sequenceDiagram
    actor Comprador
    participant Front-End

    Comprador->>Front-End: Seleccionar ítems
    activate Front-End
    Front-End->>Front-End: Mostrar precios en pantalla
    deactivate Front-End
    Front-End->>Front-End: Mostrar opciones de pago y botones "Pagar" y "Salir"
    deactivate Front-End
    Comprador->>Front-End: Seleccionar método de pago "Efectivo"
    activate Front-End
    Comprador->>Front-End: Seleccionar botón "Pagar"
    deactivate Front-End
    Front-End->>Front-End: Mostrar total a pagar en pantalla
    deactivate Front-End
    opt [Flujos Alternos]
        Comprador->>Front-End: Seleccionar botón "Salir"
        deactivate Front-End
    end
    opt [Excepciones]
        Front-End->>Front-End: Mostrar mensaje "No hay conexión con la base de datos"
        deactivate Front-End
        Comprador->>Front-End: Seleccionar opción "Cerrar"
        deactivate Front-End
    end
  
```

Diagrama de secuencia CU “Pagar Orden” Interacción Actor con Front-End

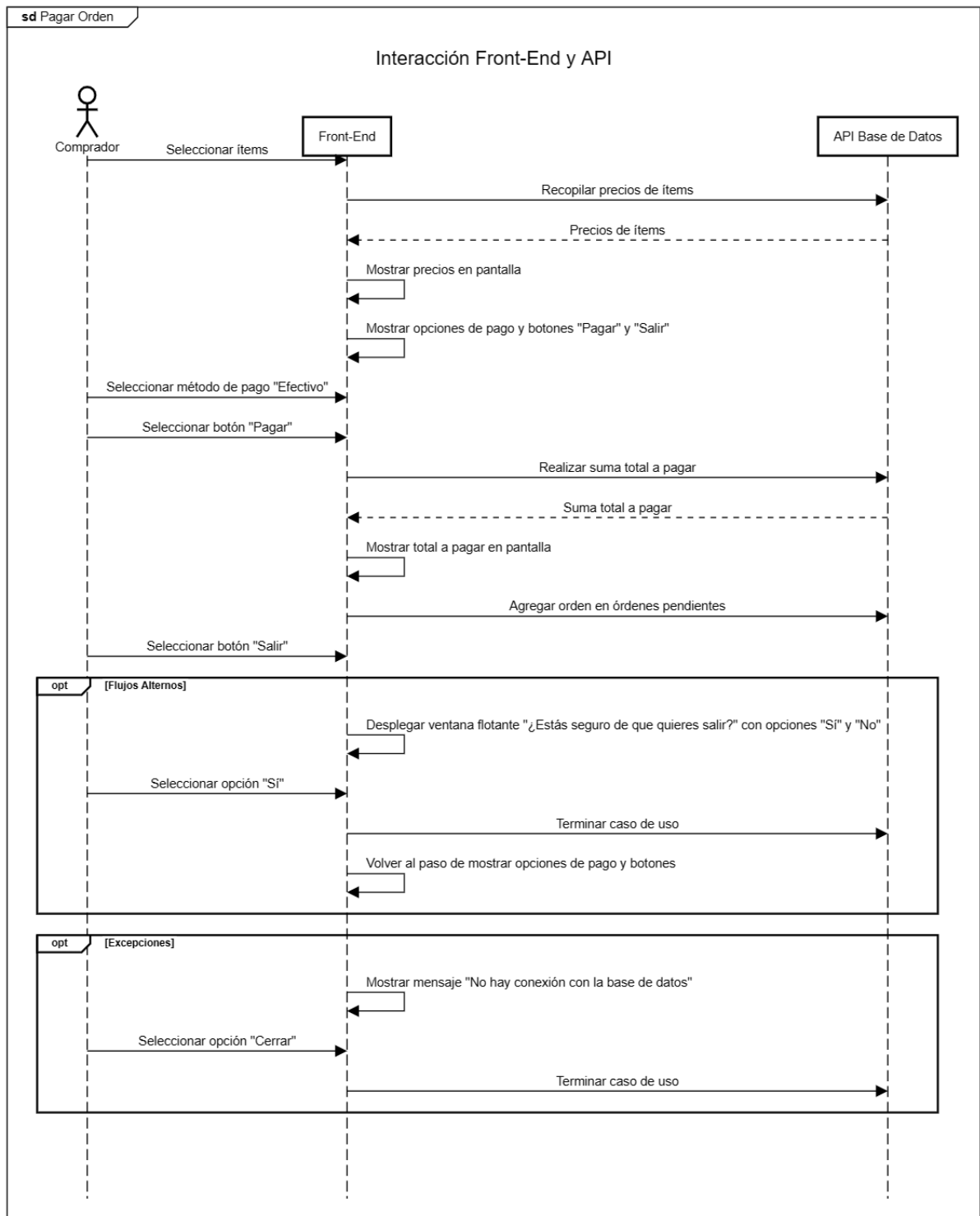


Diagrama de secuencia CU "Pagar Orden" Interacción API con Front-End

ID del caso de uso:	CU-06
Caso de uso:	Cancelar orden
Responsable:	Francisco Xavier Aguilar Martinez
Descripción:	El cliente decide de lo que eligió en su orden no le gusto y puede cancelar su orden
Fecha de modificación	16/06/2023
Actor (es):	Cliente
Precondiciones:	PRE-01: Los productos seleccionados por el cliente deben estar disponibles.
Disparador:	El cliente ingresa a la ventana de orden
Flujo normal.	<p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana "GUI Orden" con el panel lateral derecho del carrito que muestra los productos seleccionados por el cliente, incluyendo nombre, descripción, contenido, precio y total. 2. El sistema muestra el panel de productos disponibles con el botón "Agregar a orden" en cada elemento. El sistema muestra los botones "Cancelar orden" y "Efectuar Pago". (ver FA01, FA02, EX01) <p>Cliente:</p> <ol style="list-style-type: none"> 3. El cliente hace clic en "Cancelar orden". <p>Front-End:</p> <ol style="list-style-type: none"> 4. El sistema muestra el mensaje "¿Seguro de cancelar orden?" con los botones "Si" y "No". <p>Cliente:</p> <ol style="list-style-type: none"> 5. El cliente hace clic en el botón "Si". <p>Front-End:</p> <ol style="list-style-type: none"> 6. El sistema cierra el panel del carrito y regresa al menú principal. 7. Termina el caso de uso.
Flujos alternos.	<p>FA01: El cliente hace clic en el botón "Agregar a orden".</p> <p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema agrega el producto al carrito y regresa al flujo normal 1. <p>FA02: El cliente hace clic en el botón "Efectuar Pago".</p> <p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema pasa a la siguiente ventana de pago y muestra la opción de pago con tarjeta o físicamente, junto con el botón "Continuar". <p>Cliente:</p> <ol style="list-style-type: none"> 2. El cliente hace clic en la opción de pago con tarjeta y luego en el botón "Continuar". <p>Front-End:</p> <ol style="list-style-type: none"> 3. El sistema muestra una ventana emergente con el mensaje "Se ha efectuado el pago, puede ir a recoger su pedido en 20 minutos" y el botón de "aceptar". <p>Cliente:</p> <ol style="list-style-type: none"> 4. El cliente hace clic en el botón "aceptar". <p>Front-End:</p> <ol style="list-style-type: none"> 5. El sistema regresa al menú principal. Termina el caso de uso.

Excepciones.	<p>EX1: No hay conexión con la base de datos.</p> <p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje "No se pudo conectar con la base de datos. Por favor, inténtelo más tarde" y el botón "Aceptar". <p>Cliente:</p> <ol style="list-style-type: none"> 2. El cliente hace clic en el botón "Aceptar". Termina el caso de uso y regresa al menú principal.
Postcondición(es).	POST-1: El cliente debe tener guardado el código de su orden.
Incluye.	Ninguno.
Extiende.	CU05-Pagar orden

Pagar Orden

Resumen de compra

Cantidad	Nombre	Precio

Método de pago

Total

Prueba

Pagar

Cancelar

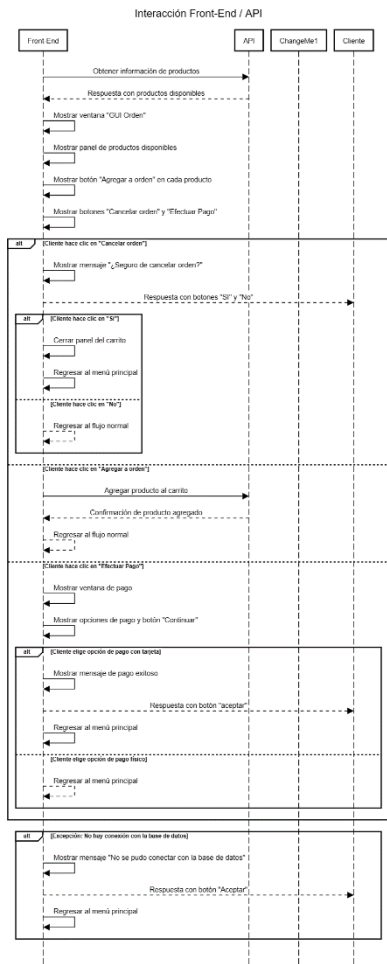


Diagrama de secuencia CU “Pagar Orden” Interacción Actor con Front-End

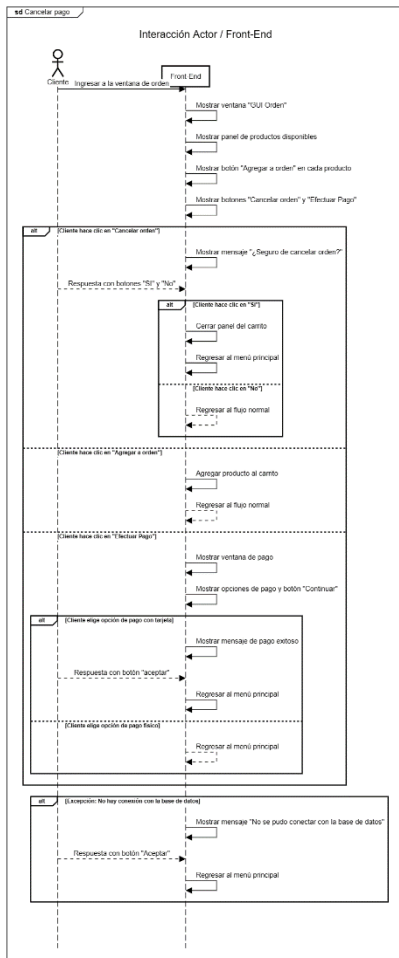
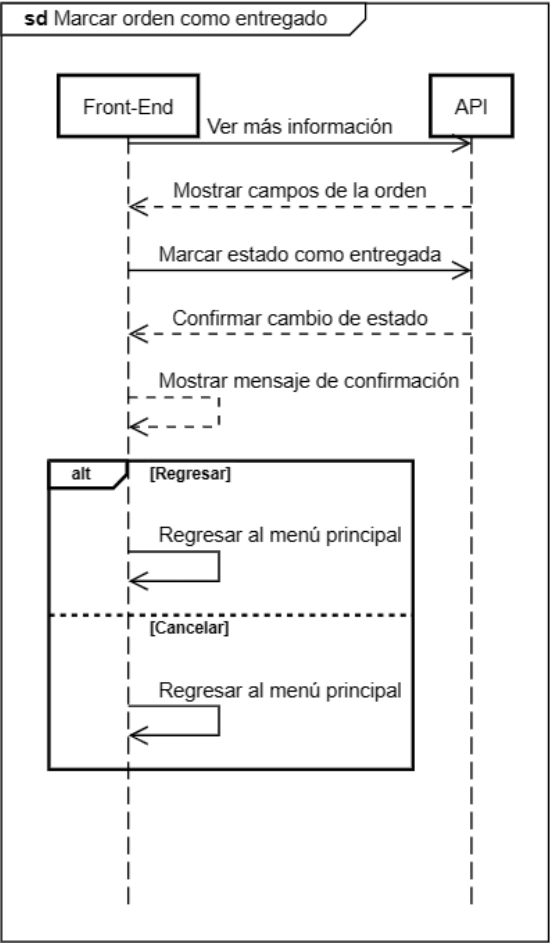
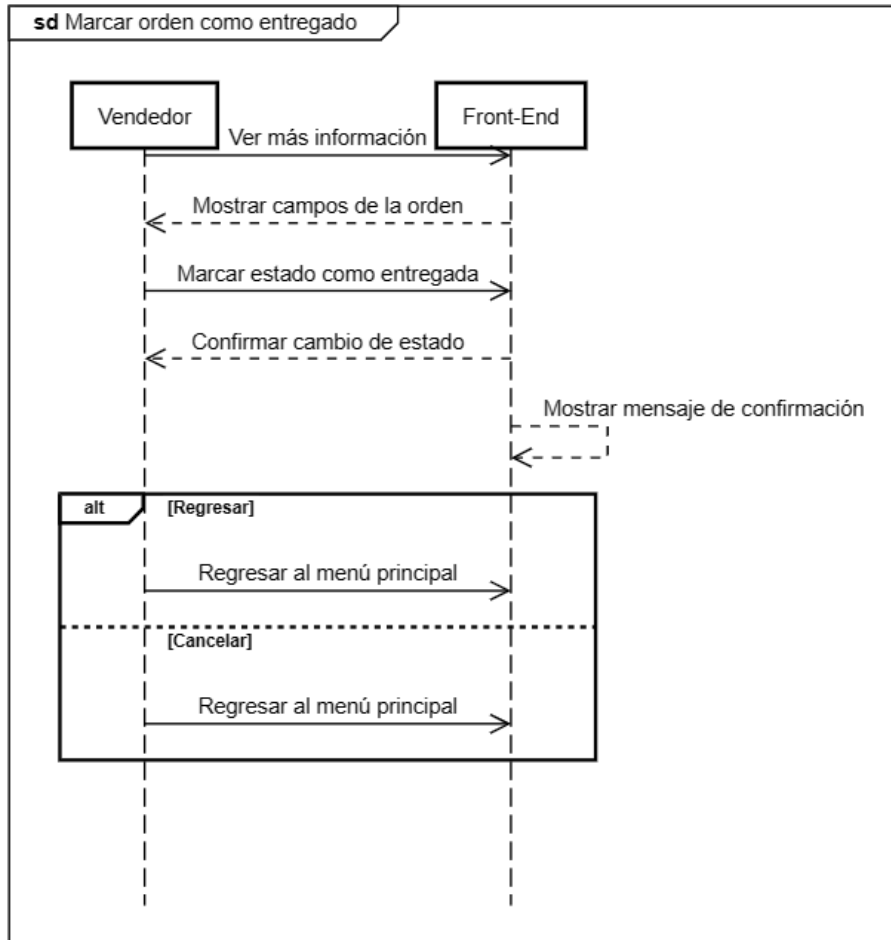


Diagrama de secuencia CU "Pagar Orden" Interacción Actor con Front-end

ID del caso de uso:	CU-07
Caso de uso:	Marcar orden como entregada
Responsable:	Francisco Xavier Aguilar Martinez
Descripción:	El vendedor puede cambiar el estado de la orden
Fecha de modificación	16/06/2023
Actor (es):	Vendedor
Precondiciones:	PRE-01: Debe sido pagada la orden que el sistema registro
Disparador:	El vendedor ingresa a la página de ordenes
Flujo normal.	<p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema muestra la ventana "GUI Ordenes a atender" con una lista ordenada de órdenes que incluye el nombre del cliente, la fecha, el estado, el monto total y la forma de pago. Se muestra el botón de enlace "Ver más información" y el botón "Regresar". (ver FA01, EX01) <p>Vendedor:</p> <ol style="list-style-type: none"> 2. El vendedor hace clic en el enlace "Ver más información". <p>Front-End:</p> <ol style="list-style-type: none"> 3. El sistema muestra los campos de la orden y los botones "Guardar", "Cancelar" y "Mandar a cocinar". (ver FA02, FA03) <p>Vendedor:</p> <ol style="list-style-type: none"> 4. El vendedor marca el estado de la orden como entregada y hace clic en el botón "Guardar". <p>Front-End:</p> <ol style="list-style-type: none"> 5. El sistema muestra una ventana emergente con el mensaje "Se han guardado los cambios". Termina el caso de uso.
Flujos alternos.	<p>FA01: El vendedor hace clic en el botón "Regresar".</p> <p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema regresa al menú principal. Termina el caso de uso. <p>FA02: El vendedor hace clic en el botón "Cancelar".</p> <p>Front-End: El sistema regresa al menú principal. Termina el caso de uso.</p> <p>FA03: El vendedor hace clic en el botón "Mandar a cocinar".</p> <p>Front-End: El sistema envía la información al cocinero y muestra un mensaje emergente con el texto "Se ha enviado con éxito la orden". Termina el caso de uso.</p>
Excepciones.	<p>Excepciones:</p> <p>EX1: No hay conexión con la base de datos.</p> <p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje "No se pudo conectar con la base de datos. Por favor, inténtelo más tarde" y el botón "Aceptar". <p>Vendedor:</p> <ol style="list-style-type: none"> 2. El vendedor hace clic en el botón "Aceptar". Termina el caso de uso y regresa al menú principal.

Postcondición(es).	POST-1: El vendedor debe actualizar la información de las ordenes.
Incluye.	Ninguno
Extiende.	Ninguno






ID:	CU-01
Nombre:	Registrar producto.
Responsable:	Yasser Yaid Gapi Pérez.
Fecha de actualización:	25 de marzo del 2023.
Descripción:	Permite al administrador registrar algún PRODUCTO en la base de datos del sistema.
Actor:	Administrador.
Disparador:	El Administrador, da clic en el botón "Registrar Producto".
Precondición:	PRE-01. Conexión del sistema a la base de datos. PRE-02. Administradores registrados en la base de datos.
Flujo Normal:	Front-End: <ol style="list-style-type: none"> 1. El sistema muestra la ventana "GUI-Registrar/Modificar producto" con los campos

	<p>necesarios para capturar la información relevante al registro del PRODUCTO, incluyendo etiquetas, cuadros de texto, componentes de lista desplegable y botones. (ver EX-01)</p> <p>Administrador:</p> <ol style="list-style-type: none"> 2. El Administrador ingresa la información requerida en los campos correspondientes y presiona el botón "Guardar". (ver FA-01, FA-02, FA-03) <p>API:</p> <ol style="list-style-type: none"> 3. 3. El API recibe los datos del Front-End y realiza la conexión con el servidor.El API almacena la información del PRODUCTO en la base de datos. Termina caso de uso.
Flujo Alternativo:	<p>FA-01: El administrador presiona el botón "Cancelar".</p> <ol style="list-style-type: none"> 1. Termina caso de uso sin guardar información en la base de datos. <p>FA-02: El administrador presiona el botón "Seleccionar imagen".</p> <p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema abre una ventana emergente del explorador de archivos del sistema operativo para seleccionar la imagen del producto. <p>Administrador:</p> <ol style="list-style-type: none"> 2. El administrador selecciona la imagen y presiona el botón "Guardar" en la ventana del explorador de archivos. <p>Front-End:</p> <ol style="list-style-type: none"> 3. El sistema muestra la ruta de la imagen anexa al producto. 4. Retorna al paso 2 del flujo normal. <p>FA-03: El administrador presiona el botón "Guardar" dejando campos vacíos o sin seleccionar el tipo de producto.</p> <p>Front-End:</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con un mensaje de error. 2. El administrador hace clic en el botón "Aceptar". 3. Retorna al paso 2 del flujo normal.
Excepciones:	<p>EX-01: Si no se puede realizar la conexión con la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con un mensaje de error.

	<ol style="list-style-type: none">2. El administrador hace clic en el botón "Aceptar".3. Termina el caso de uso.
Postcondición:	El sistema registra un PRODUCTO en la base de datos correctamente.

 GUI_CU01_RegistrarProducto

Ingrese nombre del producto

Ingrese la marca del producto

Ingrese la descripcion del producto

Seleccione la categoria del producto

Ingrese el precio del producto

Puntos de fidelidad que da producto

Ingrese el tamano del producto

Seleccione el tipo de envase

Ingrese la cantidad en existencia

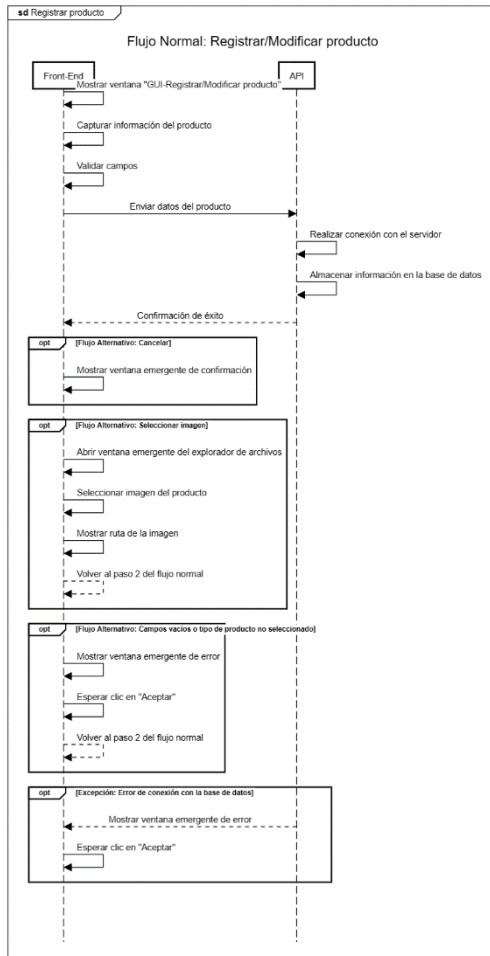
Ruta de la imagen

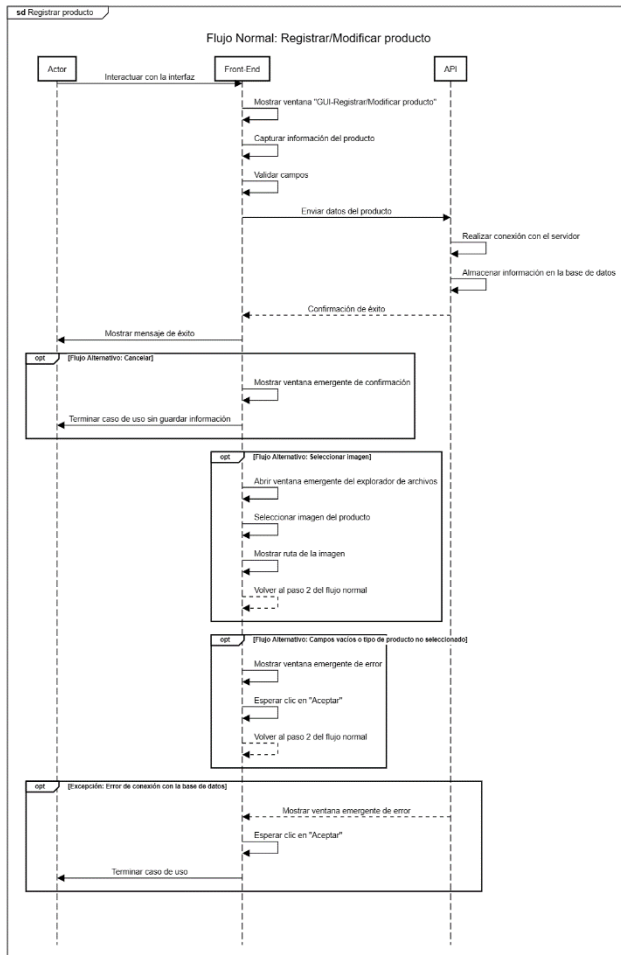
Seleccionar Imagen

Guardar

Cancelar



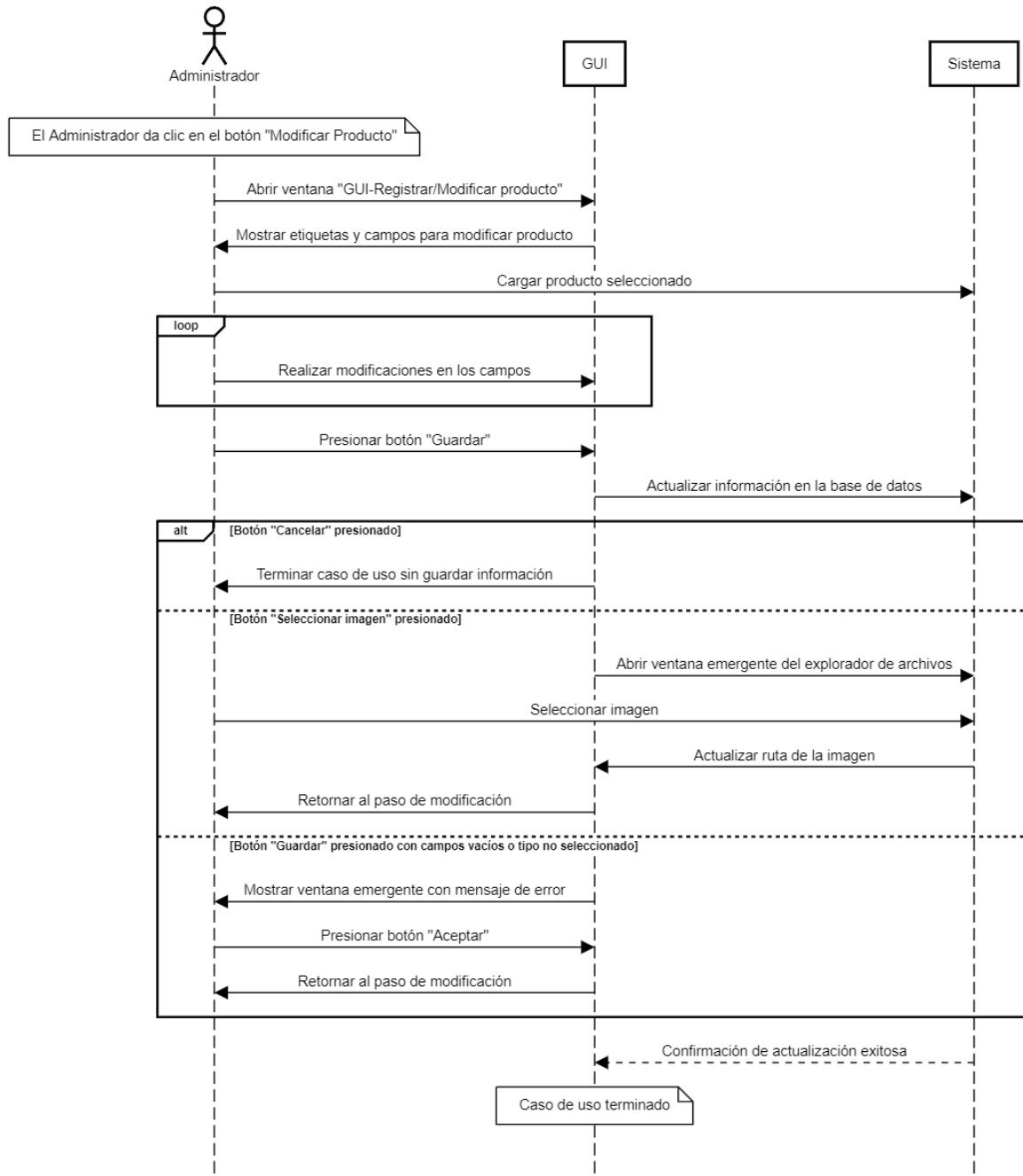




ID:	CU-02
Nombre:	Modificar producto.
Responsable:	Yasser Yaid Gapi Pérez.
Fecha de actualización:	25 de marzo del 2023.
Descripción:	Permite al administrador modifica la información de algún PRODUCTO en la base de datos del sistema.
Actor:	Administrador.
Disparador:	El Administrador, da clic en el botón “Modificar Producto”.
Precondición:	PRE-01. Conexión del sistema a la base de datos. PRE-02. Administradores registrados en la base de datos. PRE-03 PRODUCTOS registrados en la base de datos del sistema.
Flujo Normal:	FN: <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI-Registrar/Modificar producto” con 9 etiquetas de color azul que guían al administrador para poder modificar la información almacenada de algún PRODUCTO, 1 etiqueta de color negro para mostrar la ruta de la imagen anexa al PRODUCTO 6 cuadros de texto para capturar las modificaciones del nombre, marca, descripción, precio del producto, cantidad de producto y cantidad en existencia, 2 componentes de lista desplegable que permite modificar el tipo de PRODUCTO y tipo de envase, 3 botones de “Seleccionar imagen”, “Guardar” y “Cancelar”. (ver EX-01) 2. El sistema carga el producto anteriormente consultado y seleccionado en el “CU-Consultar producto” por el administrador para su modificación. 3. El Administrador ingresa los cambios deseados de nombre, marca, descripción, precio de producto, cantidad de producto o cantidad en existencia en los respectivos cuadros de texto y selecciona la modificación al tipo de PRODUCTO o tipo de envase que es de los componentes de lista desplegable y presiona el botón “Guardar” (ver FA-01, FA-02, FA-03). 4. El sistema se conecta con el servidor y actualiza la información en la base de datos. 5. Termina caso de uso.
Flujo Alterno:	<p>FA-01: El administrador presiona el botón “cancelar”.</p> <ol style="list-style-type: none"> 1. Termina caso de uso sin guardar información en base de datos. <p>FA-02: El administrador presiona el botón “seleccionar imagen”.</p> <ol style="list-style-type: none"> 1. El sistema abre una ventana emergente del explorador de archivos del sistema operativo para poder seleccionar la imagen a cargar. 1. El administrador selecciona la imagen de la ventana emergente del explorador de archivos del sistema operativo y presiona el botón guardar de la ventana emergente del sistema. 2. El sistema actualiza la ruta de la imagen anexa al producto en la etiqueta correspondiente. 3. Retorna al paso número 2 del flujo normal. <p>FA-03 El administrador presiona el botón guardar dejando campos vacíos en los cuadros de texto o no seleccionando el tipo de bebida.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “No se puede registrar un producto sin seleccionar el tipo o con información vacía, por favor ingrese toda la información necesaria” y un botón de “aceptar”. 2. El administrador hace clic en el botón “aceptar”. 3. Retorna al paso número 2 del flujo normal.

Excepciones:	<p>EX-01 Si no se puede realizar la conexión con la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “Error de conexión con la base de datos” y un botón de “aceptar”. 2. El administrador hace clic en el botón “aceptar”. 3. Termina el caso de uso.
Postcondición:	El sistema registra un PRODUCTO en la base de datos correctamente.

Modificar producto



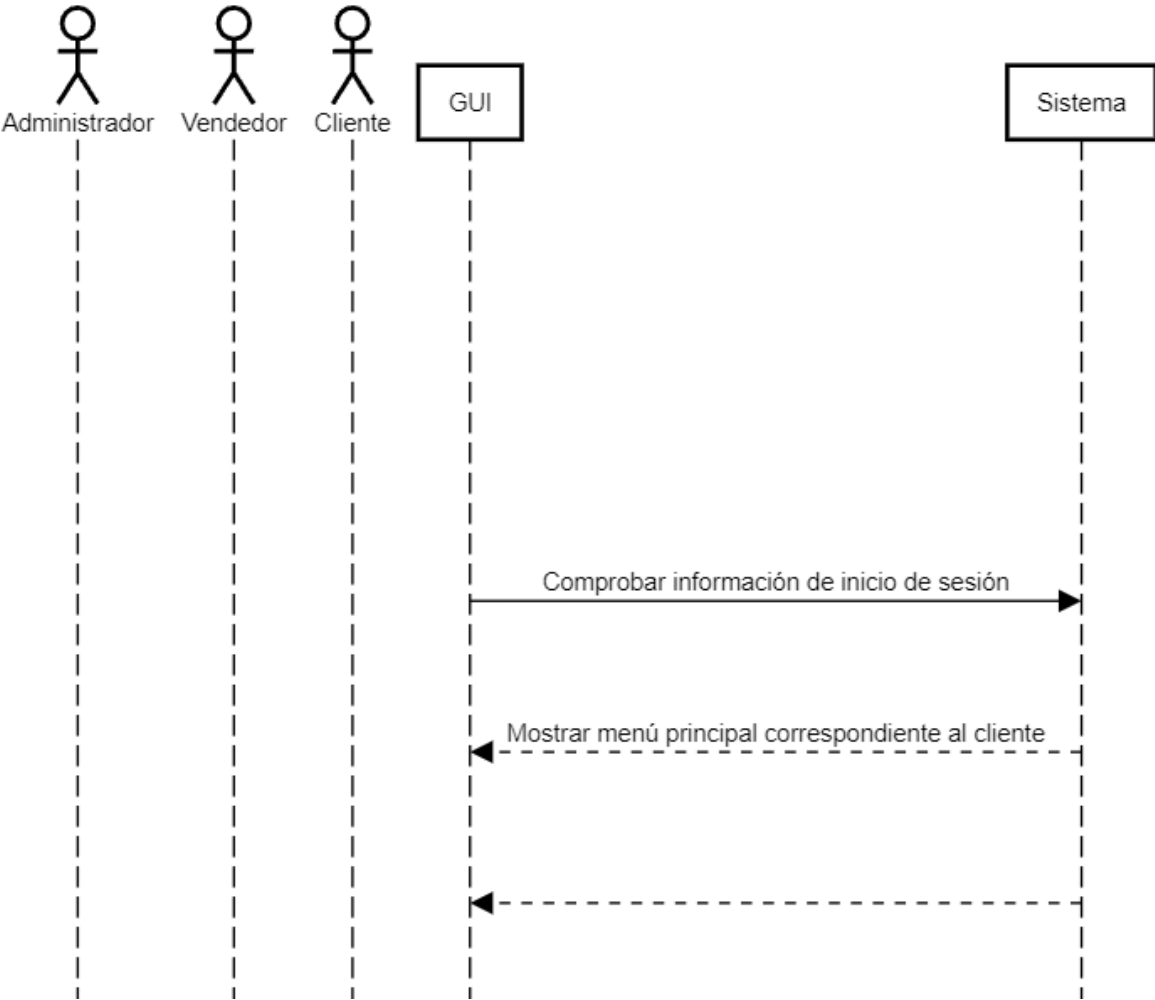
ID:	CU-03
Nombre:	Iniciar sesión
Responsable:	Yasser Yaid Gapi Pérez.
Fecha de actualización:	25 de marzo del 2023.
Descripción:	Permite al administrador, vendedor o cliente iniciar sesión en el sistema.
Actor:	Administrador, vendedor, cliente.
Disparador:	El Administrador, da clic en el botón “Iniciar sesión”.
Precondición:	PRE-01. Conexión del sistema a la base de datos. PRE-02. Administradores, vendedores y clientes registrados en la base de datos.
Flujo Normal:	FN: <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI-Iniciar sesión” con 2 etiquetas de color azul y dos cuadros de texto que guían al administrador, vendedor, o cliente en la inserción de su matrícula o clave de trabajador y contraseña, para poder acceder al sistema, 1 etiqueta de color negro para invitar al cliente a crear una cuenta 1 etiqueta de color azul para permitir al cliente registrarse como cliente en el sistema, 2 botones de “Aceptar” y “Cancelar”. (ver EX-01) 2. El Administrador, vendedor o cliente ingresa la matrícula o clave de trabajador y la contraseña en sus respectivos cuadros de texto, y presiona el botón “Aceptar” (ver FA-01, FA-02, FA-03). 3. El sistema se conecta con el servidor compara si la información ingresada en los cuadros de texto es coincidente con la información registrada de los USUARIOS en la base de datos y compara el inicio de la clave de trabajador/matricula para mostrar el menú principal correspondiente (ver FA-04, FA-05, FA-06, EX-02) 4. Termina caso de uso.
Flujo Alterno:	<p>FA-01: El administrador presiona el botón “cancelar”.</p> <ol style="list-style-type: none"> 1. Termina caso de uso sin guardar información en base de datos. <p>FA-02 El administrador presiona el botón aceptar dejando campos vacíos en los cuadros de texto.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “No se puede ingresar al sistema, ingrese su matricula/clave de trabajador y contraseña” y un botón de “acceptar”. 2. El administrador, cliente o vendedor hace clic en el botón “acceptar”. 3. Retorna al paso número 2 del flujo normal. <p>FA-03: El cliente presiona la etiqueta “registrarse”.</p> <ol style="list-style-type: none"> 1. Extiende al “CU-04 Registrarse como cliente”. 2. Termina caso de uso “CU-03 Iniciar sesión”. <p>FA-04: El administrador, cliente o vendedor ingreso una matricula/clave de trabajador con inicio de ZS.</p> <ol style="list-style-type: none"> 1. El sistema muestra el menú principal correspondiente al cliente. 2. Termina caso de uso. <p>FA-05: El administrador, cliente o vendedor ingreso una matricula/clave de trabajador con inicio de VEN.</p> <ol style="list-style-type: none"> 1. El sistema muestra el menú principal correspondiente al vendedor. 2. Termina caso de uso. <p>FA-06: El administrador, cliente o vendedor ingreso una matricula/clave de trabajador con inicio de ADM.</p> <ol style="list-style-type: none"> 1. El sistema muestra el menú principal correspondiente al administrador.

	2. Termina caso de uso.
Extend:	CU-04 Registrarse como cliente.
Excepciones:	<p>EX-01 Si no se puede realizar la conexión con la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: "Error de conexión con la base de datos" y un botón de "aceptar". 2. El administrador, cliente o vendedor hace clic en el botón "aceptar". 3. Termina el caso de uso. <p>EX-02 Si la matricula/clave de trabajador no coincide</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: "El nombre de usuario o contraseña no coincide con un usuario registrado, por favor ingrese nuevamente su información" y un botón de "aceptar". 2. El administrador, cliente o vendedor hace clic en el botón "aceptar". 3. Retorna al paso numero 2 del flujo normal.
Postcondición:	El sistema registra un PRODUCTO en la base de datos correctamente.

The screenshot shows a web application window titled "MainWindow". It contains a login form with the following elements:

- A label "Ingrese matricula o clave de trabajador" followed by a text input field.
- A label "Ingrese la contraseña" followed by a password input field.
- A link labeled "¿Aun no tiene cuenta de usuario?" with a sub-link "Registrarse" below it.
- A button labeled "IniciarSesion" at the bottom.

Iniciar sesión



REGISTRARSE COMO CLIENTE DE COFFEI

Ingrese su nombre :

Juan

Ingrese su Apellido Paterno:

Lopez

Ingrese su Apellido Materno:

Suarez

Ingrese su Matricula :

ZS20182096

Ingrese su Numero de telefono:

2283145688

Ingrese su contraseña :

Confirme la contraseña :

SUBIR COPIA DE CREDENCIAL

ACEPTAR

CANCELAR

ID:	CU-04
Nombre:	Registrarse como cliente.
Responsable:	Yasser Yaid Gapi Pérez.
Fecha de actualización:	25 de marzo del 2023.
Descripción:	Permite al cliente crear una cuenta de cliente en el sistema.
Actor:	Cliente.
Disparador:	El Cliente, da clic en “Registrarse”.
Precondición:	PRE-01. Conexión del sistema a la base de datos.
Flujo Normal:	FN: <ol style="list-style-type: none"> El sistema muestra la ventana “GUI-Registrarse como cliente de COFFEI” con 7 etiquetas de color azul y 7 cuadros de texto que guían al cliente en la inserción de su nombre, apellido paterno, apellido materno, matrícula, número de teléfono, contraseña y su confirmación, para poder acceder al sistema, 3 botones de “Subir copia de credencial”, “Aceptar” y “Cancelar”. (ver EX-01) El cliente completa el llenado del formulario ingresando en los cuadros de texto su nombre, apellido paterno, apellido materno, matrícula, número de teléfono, contraseña y su confirmación, presiona el botón “subir copia de credencial”. El sistema abre una venta emergente del explorador del sistema operativo. El cliente selecciona la imagen de la copia de su credencial, y presiona el botón “guardar” de la ventana emergente. El sistema carga la imagen seleccionada para su uso, cierra la ventana emergente y retorna el control a la “GUI-Registrarse como cliente de COFFEI”.

	<ol style="list-style-type: none"> 6. El cliente presiona el botón “Aceptar” (ver FA-01, FA-02, FA-03). 7. El sistema compara que la información ingresada en los cuadros de texto de contraseña y confirmación sean la misma, compara que la matrícula ingresada comience con ZS y posteriormente este seguida de solo caracteres numéricos, y que la extensión de la matrícula sea de 10 caracteres, se conecta al servidor para subir la imagen, asociar la dirección de esta a la cuenta y almacenar el resto de información de la nueva cuenta de cliente (ver EX-02, EX-03). 8. Termina caso de uso.
Flujo Alternativo:	<p>FA-01: El cliente presiona el botón “cancelar”.</p> <ol style="list-style-type: none"> 1. Termina caso de uso sin crear la cuenta del cliente en la base de datos. <p>FA-02 El cliente presiona el botón “aceptar” dejando campos vacíos en algún cuadro de texto.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “No se puede registrar la cuenta en el sistema con campos vacíos, por favor llene todos los campos” y un botón de “aceptar”. 2. El cliente hace clic en el botón “aceptar”. 3. Retorna al paso número 2 del flujo normal. <p>FA-03: El cliente presiona el botón “aceptar” sin “subir la copia de credencial”.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “No se puede registrar la cuenta en el sistema sin subir una copia de su credencial” y un botón de “aceptar”. 2. El cliente hace clic en el botón “aceptar”. 3. Retorna al paso número 2 del flujo normal.
Excepciones:	<p>EX-01 Si no se puede realizar la conexión con la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “Error de conexión con la base de datos” y un botón de “aceptar”. 2. El cliente hace clic en el botón “aceptar”. 3. Termina el caso de uso. <p>EX-02 Si las contraseñas introducidas no coinciden.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “Las contraseñas introducidas no coinciden, por favor verifique que las contraseñas coincidan” y un botón de “aceptar”. 2. El cliente hace clic en el botón “aceptar”. 3. Retorna al paso número 2 del flujo normal. <p>EX-03 Si la matrícula introducida no comienza con ZS y posteriormente no esta seguida de solo números, o no es de extensión de 10 caracteres.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “La matrícula debe de ser de extensión de 10 caracteres, comenzar con ZS y estar seguida de solo 8 caracteres numéricos, por favor ingrese una matrícula valida” y un botón de “aceptar”. 2. El cliente hace clic en el botón “aceptar”. 3. Retorna al paso número 2 del flujo normal.
Postcondición:	El sistema registra una nueva cuenta de cliente en la base de datos correctamente.

REGISTRARSE COMO CLIENTE DE COFFEI



Ingrese su nombre :

Juan

Ingrese su Apellido Paterno:

Lopez

Ingrese su Apellido Materno:

Suarez

Ingrese su Matricula :

ZS20182096

Ingrese su Numero de telefono:

2283145688

Ingrese su contraseña :

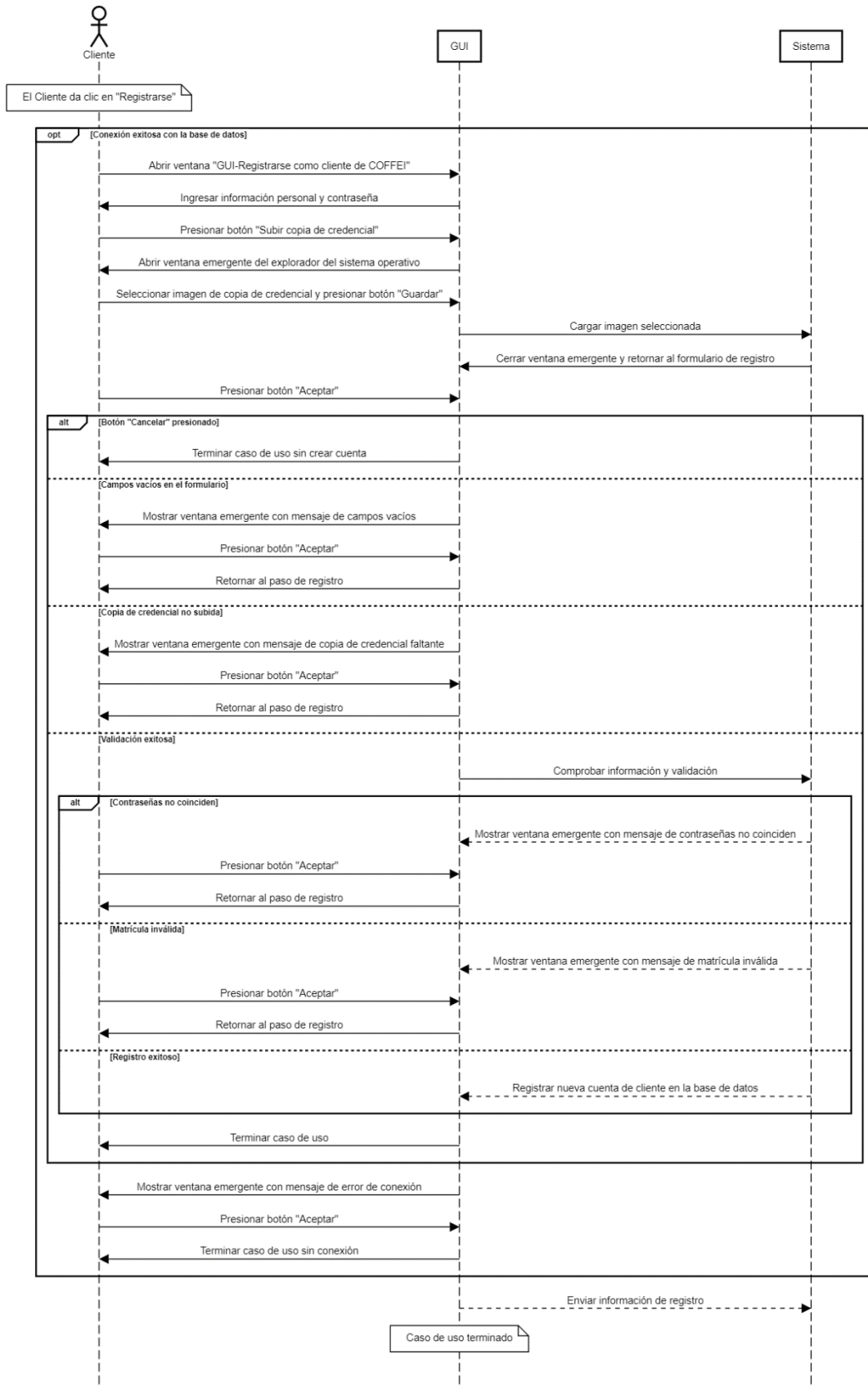
Confirme la contraseña :

SUBIR COPIA DE
CREDENCIAL

ACEPTAR

CANCELAR

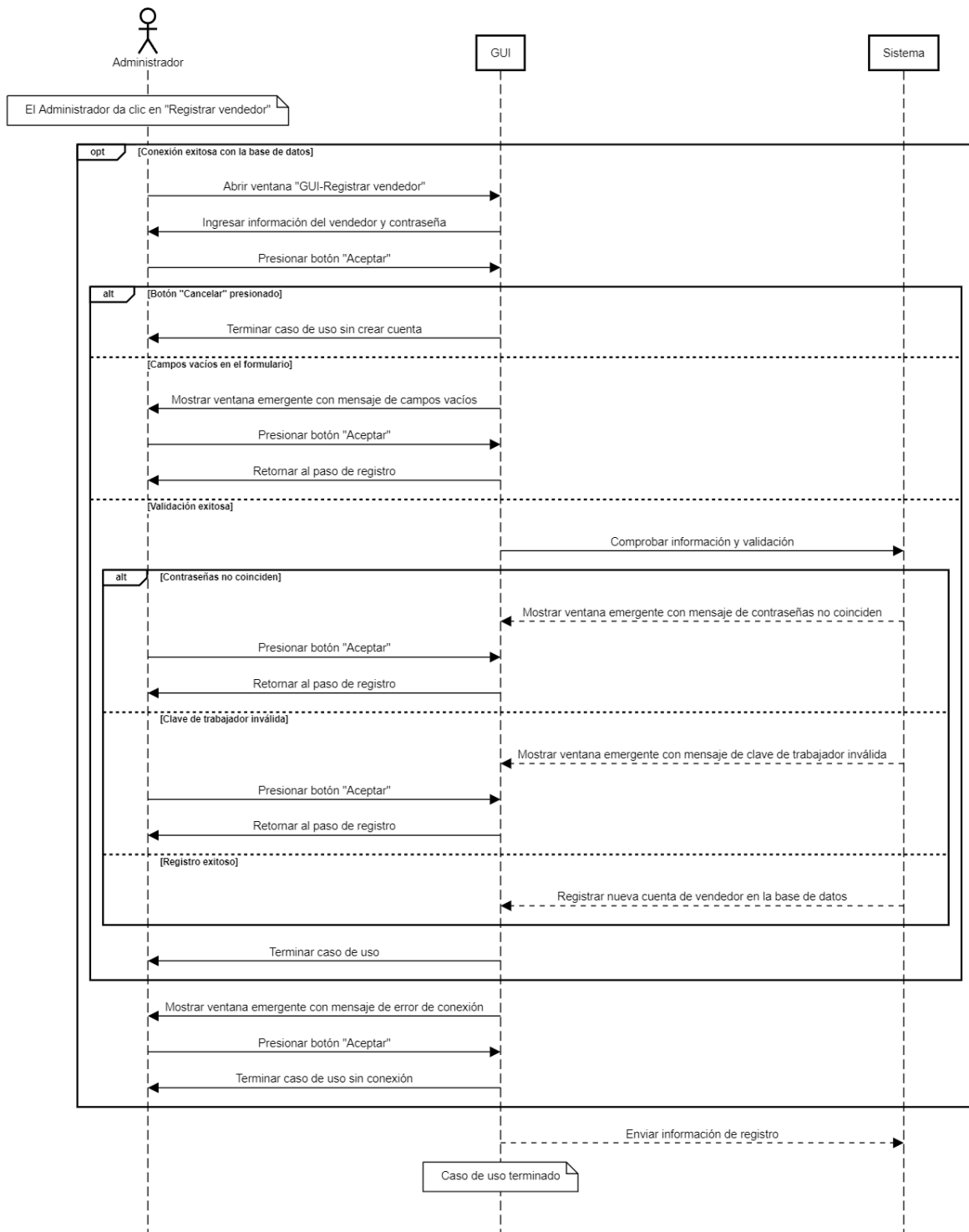
Registrarse como cliente



ID:	CU-05
Nombre:	Registrar vendedor.
Responsable:	Yasser Yaid Gapi Pérez.
Fecha de actualización:	25 de marzo del 2023.
Descripción:	Permite al administrador crear una cuenta de vendedor en el sistema.
Actor:	Administrador.
Disparador:	El Administrador, da clic en “Registrar vendedor”.
Precondición:	PRE-01. Conexión del sistema a la base de datos. PRE-02. Administrador registrado en la base de datos.
Flujo Normal:	FN: <ol style="list-style-type: none"> 1. El sistema muestra la ventana “GUI-Registrar vendedor” con 7 etiquetas de color azul y 7 cuadros de texto que guían al administrador en la inserción del nombre, apellido paterno, apellido materno, matrícula, número de teléfono, contraseña y su confirmación, para poder crear la cuenta del vendedor, 2 botones de “Aceptar” y “Cancelar”. (ver EX-01) 2. El administrador completa el llenado del formulario ingresando en los cuadros de texto el nombre, apellido paterno, apellido materno, matrícula, número de teléfono, contraseña y la confirmación de la contraseña del vendedor, presiona el botón “Aceptar” (ver FA-01, FA-02). 3. El sistema compara que la información ingresada en los cuadros de texto de contraseña y confirmación sean la misma, compara que la clave de trabajador ingresada comience con VEN y posteriormente este seguida de solo caracteres numéricos, y que la extensión de la clave sea de 10 caracteres, se conecta al servidor almacena la información de la nueva cuenta del vendedor (ver EX-02, EX-03). 4. Termina caso de uso.
Flujo Alterno:	<p>FA-01: El administrador presiona el botón “cancelar”.</p> <ol style="list-style-type: none"> 1. Termina caso de uso sin crear la cuenta del vendedor en la base de datos. <p>FA-02 El administrador presiona el botón “aceptar” dejando campos vacíos en algún cuadro de texto.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “No se puede registrar la cuenta en el sistema con campos vacíos, por favor llene todos los campos” y un botón de “aceptar”. 2. El cliente hace clic en el botón “aceptar”. 3. Retorna al paso número 2 del flujo normal.
Excepciones:	<p>EX-01 Si no se puede realizar la conexión con la base de datos.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “Error de conexión con la base de datos” y un botón de “aceptar”. 2. El cliente hace clic en el botón “aceptar”. 3. Termina el caso de uso. <p>EX-02 Si las contraseñas introducidas no coinciden.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “Las contraseñas introducidas no coinciden, por favor verifique que las contraseñas coincidan” y un botón de “aceptar”. 2. El cliente hace clic en el botón “aceptar”. 3. Retorna al paso número 2 del flujo normal. <p>EX-03 Si la clave de trabajador introducida no comienza con VEN y posteriormente no está seguida de solo números, o no es de extensión de 10 caracteres.</p>

	<ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “La matrícula debe de ser de extensión de 10 caracteres, comenzar con VEN y estar seguida de solo 8 caracteres numéricos, por favor ingrese una clave valida” y un botón de “aceptar”. 2. El administrador hace clic en el botón “aceptar”. 3. Retorna al paso número 2 del flujo normal.
Postcondición:	El sistema registra una nueva cuenta de vendedor en la base de datos correctamente.

Registrar vendedor



REGISTRAR VENDEDOR

Ingrese el nombre :

Ramiro

Ingrese elApellido Paterno:

Ortiz

Ingrese elApellido Materno:

Perez

Ingrese la Matricula :

VEN20182096

Ingrese el Numero de telefono:

2282039410

Ingrese la contraseña :

Confirme la contraseña :

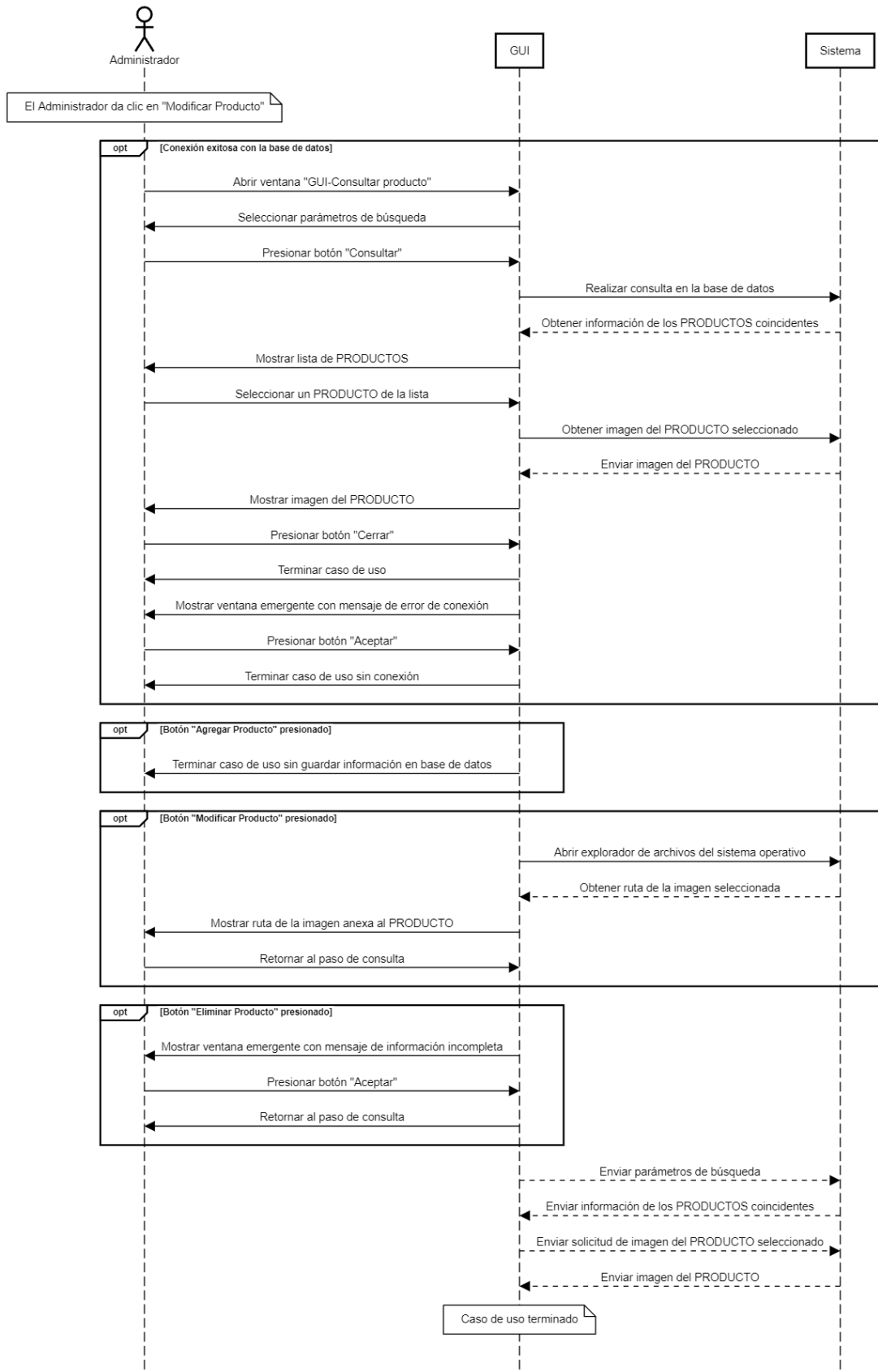
ACEPTAR

CANCELAR

ID:	CU-02
Nombre:	Consultar producto.
Responsable:	Yasser Yaid Gapi Pérez.
Fecha de actualización:	25 de marzo del 2023.
Descripción:	Permite al administrador consulta la información de los PRODUCTOS en la base de datos del sistema.
Actor:	Administrador.
Disparador:	El Administrador, da clic en el botón “Modificar Producto”.
Precondición:	PRE-01. Conexión del sistema a la base de datos. PRE-02. Administradores registrados en la base de datos. PRE-03 PRODUCTOS registrados en la base de datos del sistema.
Flujo Normal:	FN: <ol style="list-style-type: none"> El sistema muestra la ventana “GUI-Consultar producto” con 5 etiquetas de color azul que guían al administrador para poder consultar la información almacenada de algún PRODUCTO, 5 componentes de lista desplegable que permite al administrador filtrar y consultar los productos por: tipo de PRODUCTO, marca, cantidad de producto, rango de precio o estado de producto, si es agotado o existente. 1 tabla para mostrar la lista de PRODUCTOS coincidentes con los parámetros en los componentes de lista desplegable, 1 componente de cuadro de imagen para mostrar la imagen relacionada al producto, 5 botones de “Consultar”, “Agregar Producto” y “Modificar Producto”, “Eliminar Producto”, “Cerrar”. (ver EX-01) El Administrador selecciona el tipo de producto, marca, rango de la cantidad de producto, rango de precio, estado de existencia del producto y presiona el botón “Consultar” (ver FA-01, FA-02, FA-03). El sistema se conecta con el servidor realiza la consulta y obtiene la información de la base de datos y la muestra en la tabla de PRODUCTOS encontrados.

	<ol style="list-style-type: none"> 4. El administrador selecciona un PRODUCTO de la tabla. 5. El sistema actualiza el componente de imagen y muestra la imagen anexada al PRODUCTO 6. El administrador presiona el botón “Cerrar” (Ver FA-01, Ver FA-02, Ver, FA-03). 7. Termina caso de uso.
Flujo Alternativo:	<p>FA-01: El administrador presiona el botón “Agregar Producto”.</p> <ol style="list-style-type: none"> 1. Termina caso de uso sin guardar información en base de datos. <p>FA-02: El administrador presiona el botón “Modificar Producto”.</p> <ol style="list-style-type: none"> 1. El sistema abre una ventana emergente del explorador de archivos del sistema operativo para poder seleccionar la imagen a cargar. 4. El administrador selecciona la imagen de la ventana emergente del explorador de archivos del sistema operativo y presiona el botón guardar de la ventana emergente del sistema. 5. El sistema actualiza la ruta de la imagen anexa al producto en la etiqueta correspondiente. 6. Retorna al paso número 2 del flujo normal. <p>FA-03: El administrador presiona el botón “Eliminar Producto”.</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con el mensaje: “No se puede registrar un producto sin seleccionar el tipo o con información vacía, por favor ingrese toda la información necesaria” y un botón de “aceptar”. 4. El administrador hace clic en el botón “aceptar”. 5. Retorna al paso número 2 del flujo normal. <p>FA-04: El administrador presiona el botón consultar sin haber utilizado todos los componentes de lista desplegable.</p>
Excepciones:	<p>EX-01 Si no se puede realizar la conexión con la base de datos.</p> <ol style="list-style-type: none"> 4. El sistema muestra una ventana emergente con el mensaje: “Error de conexión con la base de datos” y un botón de “aceptar”. 5. El administrador hace clic en el botón “aceptar”. 6. Termina el caso de uso.
Postcondición:	El sistema registra un PRODUCTO en la base de datos correctamente.

Consultar producto



CONSULTAR PRODUCTO

Seleccione tipo de producto :
Bebida

Seleccione estado del producto :
En existencia

Seleccione la marca :
Todas


Seleccione el rango de precio:
\$10 - \$20

Seleccione cantidad producto:
Todas

CONSULTAR

PRODUCTOS ENCONTRADOS

Tipo	Marca	Nombre	Cantidad Producto	Envase	Precio	Cantidad en existencia
Bebida	Pepsi	Mirinda	400 Mililitros	Lata	8	8
Bebida	Coca-Cola	Fanta	600 Mililitros	Vidrio	12	10
Bebida	Pepsi	Sprite	1.5 Litros	Botella Plastico	20	3



AGREGAR PRODUCTO

MODIFICAR PRODUCTO

ELIMINAR PRODUCTO

CERRAR

ID	CU-18.
Nombre.	Recargar tarjeta.
Responsable.	Fausto Mendez Peralta
Fecha de actualización.	24/04/2023.
Descripción.	Permite al actor recargar una TARJETA
Actor(es).	Vendedor.
Disparador.	El vendedor da clic en el botón “Recargar tarjeta” ubicado en la página principal.
Precondición(es).	La API tiene conexión con la base de datos.
Flujo normal.	<ol style="list-style-type: none"> 1. El sistema front-end establece conexión con la API de servicios web y muestra GUI “Recargar tarjeta”, junto con los campos número de TARJETA, monto y el botón Recargar. (EX-1) 2. El vendedor ingresa el número de TARJETA y el monto y da clic en el botón Recargar. 3. El sistema front-end valida que los datos sean válidos y envía una solicitud a la API de servicios web con los parámetros número de TARJETA y monto. (FA-3.1, FA-3.2) 4. La API de servicios web recibe la solicitud y verifica que los parámetros sean válidos y que cumplan con las precondiciones: la API tiene conexión con la base de datos, el número de TARJETA existe en la base de datos y el monto es un valor positivo. (EX-2, EX-3, EX-4) 5. La API de servicios web consulta el saldo actual de la TARJETA en la base de datos y le suma el monto recibido. 6. La API de servicios web actualiza el saldo de la TARJETA en la base de datos y registra la transacción con la fecha, el número de TARJETA y el monto. 7. La API de servicios web envía una respuesta al sistema front-end con el código 200 (OK) y el mensaje “Recarga exitosa”. 8. El sistema front-end muestra el mensaje “Recarga exitosa” 9. El vendedor da clic en el botón de “Aceptar”. 10. Termina el caso de uso.
Flujos alternos.	<p>FA-3.1. El vendedor da clic en el botón de “Cancelar”.</p> <ol style="list-style-type: none"> 1. El sistema front-end muestra una ventana emergente con el mensaje “Cancelar la recarga” y los botones de “Aceptar” o “Cancelar”. 2. El vendedor da clic en el botón “Aceptar”. 3. Termina el caso de uso. <p>FA-3.2. El vendedor ingresa valores incorrectos.</p> <ol style="list-style-type: none"> 1. El sistema front-end muestra una ventana emergente con el mensaje “Revisa los datos introducidos” y los botones de “Aceptar” o “Cancelar”. 2. El vendedor da clic en el botón “Aceptar”. 3. Termina el caso de uso.
Excepciones.	<p>EX-1. No hay conexión con la API de servicios web.</p> <ol style="list-style-type: none"> 1. En el paso 1 del flujo normal, el sistema front-end no logra conectarse a la API de servicios web. 2. El sistema front-end muestra una ventana emergente con el mensaje “No hay conexión con la API” y los botones de “Aceptar” o “Cancelar”. 3. El vendedor da clic en el botón “Aceptar”. 4. Termina el caso de uso.

	<p>EX-2. Los parámetros son inválidos.</p> <ol style="list-style-type: none"> 1. En el paso 4 del flujo normal, la API de servicios web detecta que los parámetros no tienen el formato o el tipo correcto. 2. La API de servicios web envía una respuesta al sistema front-end con el código 400 (Bad Request) y el mensaje "Parámetros inválidos". 3. El sistema front-end muestra una ventana emergente con el mensaje "Parámetros inválidos" y los botones de "Aceptar" o "Cancelar". 4. El vendedor da clic en el botón "Aceptar". 5. Termina el caso de uso. <p>EX-3. El número de TARJETA no existe en la base de datos.</p> <ol style="list-style-type: none"> 1. En el paso 2 del flujo normal, la API detecta que el número de TARJETA no se encuentra en la base de datos. 2. La API envía una respuesta al sistema front-end con el código 404 (Not Found) y el mensaje "TARJETA no encontrada". 3. Termina el caso de uso. <p>EX-4. El monto es un valor negativo o cero.</p> <ol style="list-style-type: none"> 1. En el paso 4 del flujo normal, la API de servicios web detecta que el monto es un valor negativo o cero. 2. La API de servicios web envía una respuesta al sistema front-end con el código 422 (Unprocessable Entity) y el mensaje "Monto inválido". 3. El sistema front-end muestra una ventana emergente con el mensaje "Monto inválido" y los botones de "Aceptar" o "Cancelar". 4. El vendedor da clic en el botón "Aceptar". 5. Termina el caso de uso.
Postcondición(es).	POST-1. El sistema recarga de manera correcta una TARJETA.
Regla(s) de negocio.	
Incluye.	Ninguno.
Extiende.	Ninguno.

Introduzca el número de tarjeta:

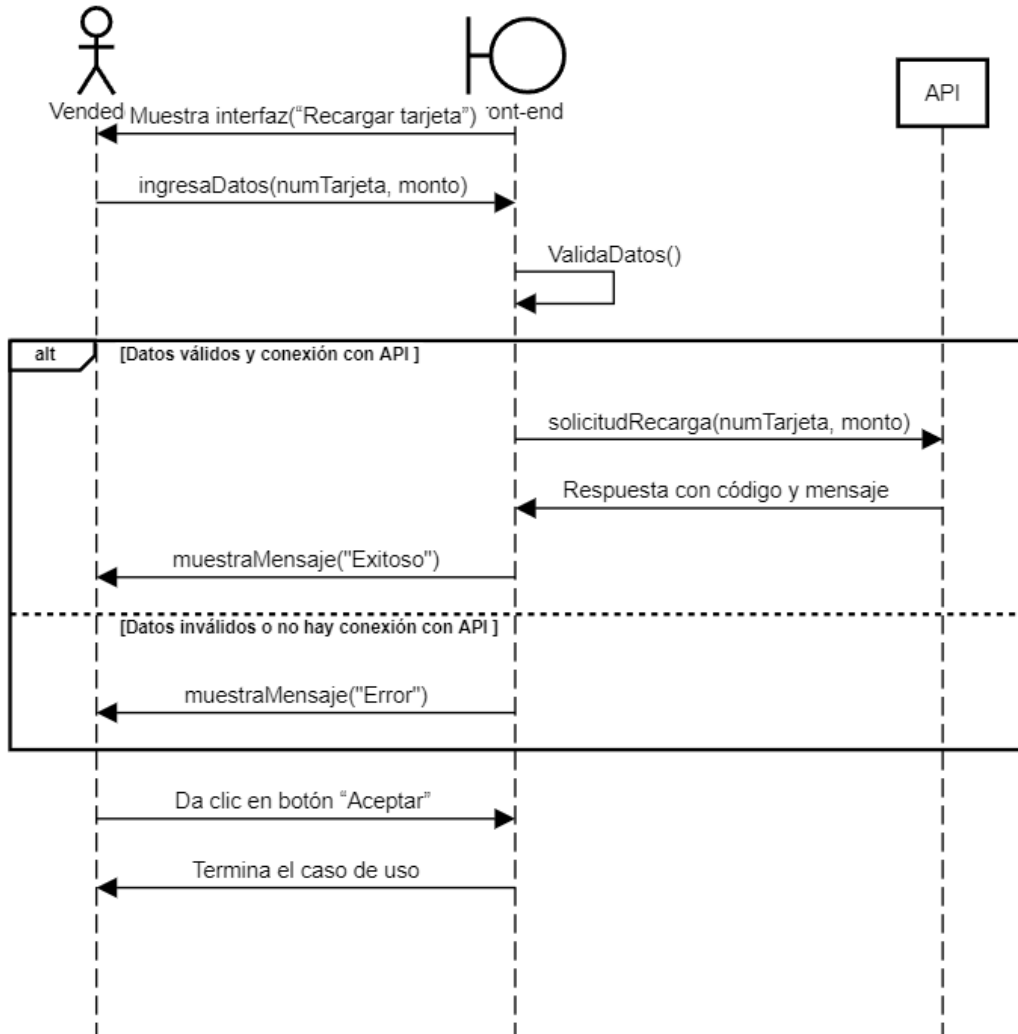
<input type="text"/>	Validar
----------------------	---------

Introduzca el monto a recargar:

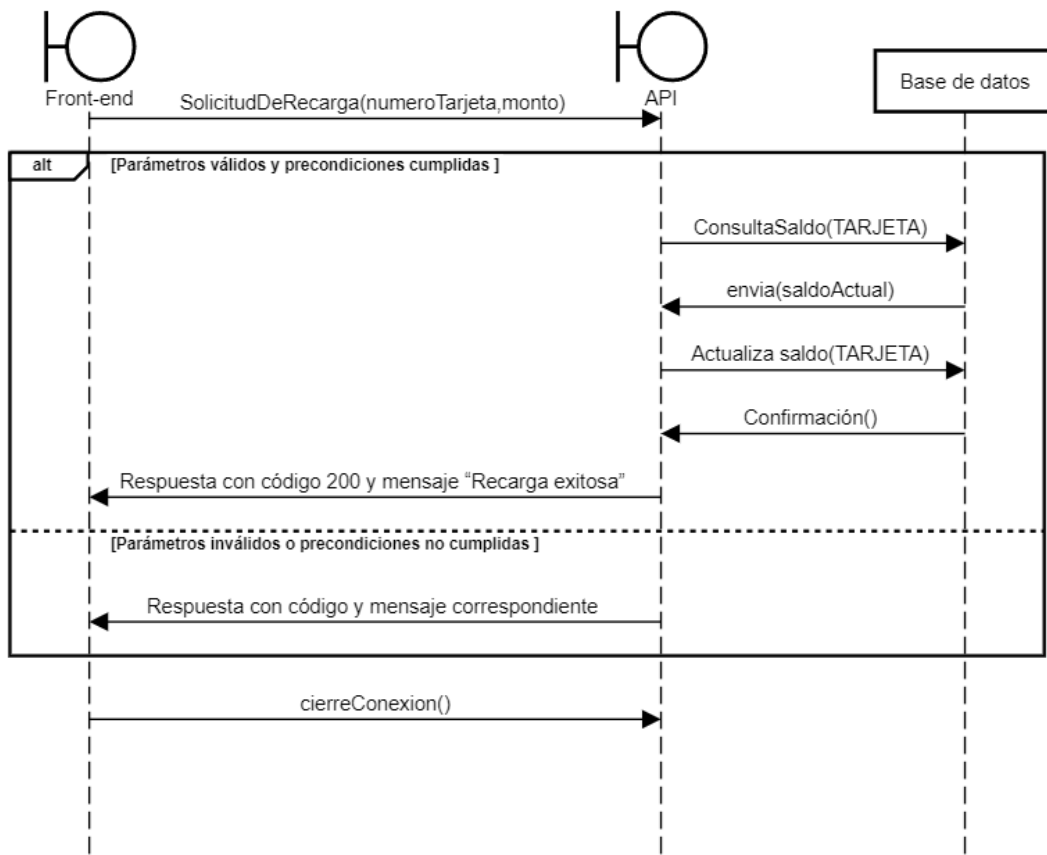
<input type="text"/>	\$0.00
----------------------	--------

Cancelar	Pagar
----------	-------

Recarga de TARJETA



Recarga de TARJETA



ID:	CU-17
Nombre:	Consultar menú
Responsable:	Francisco Xavier Aguilar Martínez
Fecha de actualización:	05/03/2023
Descripción:	Se visualiza el menú de la cafetería
Actor(es):	Comprador
Disparador:	Seleccionar botón “Menú”
Precondiciones:	PRE-1: Debe existir al menos un ítem registrado en la base de datos
Flujo Normal:	<ol style="list-style-type: none"> 1. El sistema recopila de la base de datos los ítems del menú del día con (activo==true) y los muestra en pantalla 2. El comprador selecciona el botón salir 3. Termina el caso de uso
Flujos Alternos:	<p>FA1 No hay productos disponibles</p> <ol style="list-style-type: none"> 1. El sistema muestra una ventana emergente con la leyenda (“No hay productos disponibles todavía, inténtelo más tarde”) 2. El comprador da click aceptar 3. Regresa al FN1
Excepciones:	<p>EX1. No hay conexión con la base de datos</p> <ol style="list-style-type: none"> 1. El sistema muestra el mensaje “No hay conexión con la base de datos” 2. El actor selecciona la opción cerrar. 3. Termina el caso de uso
Postcondiciones:	Ninguno.
Incluye:	Ninguno
Extiende:	Ninguno

COFFE FEI

Menú

Comida

Torta de jamón

Torta de choriqueso

Cuernito

Chilaquiles

Bebidas

Coca cola 600ml

Sprite 600ml

Agua de horchata

Agua de jamaica

Cancelar

Agregar a la orden

ID	CU-18.
Nombre.	Gestionar Puntos de fidelidad.
Responsable.	Fausto Mendez Peralta
Fecha de actualización.	24/04/2023.
Descripción.	Permite al actor gestionar PUNTOS DE FIDELIDAD
Actor(es).	Vendedor y cliente.
Disparador.	El vendedor o cliente da clic en el botón “Pagar” ubicado en la página de cobro.
Precondición(es).	La API tiene conexión con la base de datos.
Flujo normal.	<ol style="list-style-type: none"> El sistema front-end establece conexión con la API de servicios web y realiza el cálculo de puntos obtenidos. (EX-1) El sistema front-end muestra el mensaje “XX puntos acumulados”. El vendedor o cliente acepta el mensaje. El sistema front-end invoca el método de recuperar productos canjeables de la API de servicios web y muestra las opciones de productos que puede obtener con sus puntos, y los botones de canjear y omitir. El vendedor o cliente selecciona el producto y da clic en el botón de “Canjear”. (FA-3.1, FA-3.2) El sistema front-end invoca el método de restar puntos utilizados de la API de servicios web. La API de servicios web recibe la solicitud del sistema front-end con los parámetros número de TARJETA y monto. La API de servicios web verifica que los parámetros sean válidos y que cumplan con las precondiciones: la API tiene conexión con la base de datos, el número de TARJETA existe en la base de datos y el monto es un valor positivo. (EX-2, EX-3, EX-4) La API de servicios web consulta el saldo actual de la TARJETA en la base de datos y le resta el monto recibido. La API de servicios web actualiza el saldo de la TARJETA en la base de datos y registra la transacción con la fecha, el número de TARJETA y el monto. La API de servicios web envía una respuesta al sistema front-end con el código 200 (OK) y el mensaje “Canje exitoso”. El sistema front-end muestra el mensaje “Canje exitoso” El vendedor o cliente da clic en el botón de “Aceptar”.

	14. Termina el caso de uso..
Flujos alternos.	<p>FA-3.1. El vendedor o cliente da clic en el botón de “Omitir”.</p> <ol style="list-style-type: none"> 1. El sistema front-end muestra GUI “Página principal”. 2. Termina el caso de uso. <p>FA-3.2. El vendedor o cliente no cuenta con los puntos necesarios.</p> <ol style="list-style-type: none"> 1. El sistema front-end muestra una ventana emergente con el mensaje “Puntos insuficientes” y los botones de “Aceptar” o “Cancelar”. 2. El vendedor o cliente da clic en “Aceptar”. 3. Termina el caso de uso.
Excepciones.	<p>EX-1. No hay conexión con la API de servicios web.</p> <ol style="list-style-type: none"> 1. En el paso 1 o 4 o 6 o 7 del flujo normal, el sistema front-end no logra conectarse a la API de servicios web. 2. El sistema front-end muestra una ventana emergente con el mensaje “No hay conexión con la API” y los botones de “Aceptar” o “Cancelar”. 3. El vendedor o cliente da clic en el botón “Aceptar”. 4. Termina el caso de uso. <p>EX-2. Los parámetros son inválidos.</p> <ol style="list-style-type: none"> 1. En el paso 8 del flujo normal, la API de servicios web detecta que los parámetros no tienen el formato o el tipo correcto. 2. La API de servicios web envía una respuesta al sistema front-end con el código 400 (Bad Request) y el mensaje “Parámetros inválidos”. 3. El sistema front-end muestra una ventana emergente con el mensaje “Parámetros inválidos” y los botones de “Aceptar” o “Cancelar”. 4. El vendedor o cliente da clic en el botón “Aceptar”. 5. Termina el caso de uso. <p>EX-3. El número de TARJETA no existe en la base de datos.</p> <ol style="list-style-type: none"> 1. En el paso 8 del flujo normal, la API de servicios web detecta que el número de TARJETA no se encuentra en la base de datos. 2. La API de servicios web envía una respuesta al sistema front-end con el código 404 (Not Found) y el mensaje “TARJETA no encontrada”. 3. El sistema front-end muestra una ventana emergente con el mensaje “TARJETA no encontrada” y los botones de “Aceptar” o “Cancelar”. 4. El vendedor o cliente da clic en el botón “Aceptar”. 5. Termina el caso de uso. <p>EX-4. El monto es un valor negativo o cero.</p> <ol style="list-style-type: none"> 1. En el paso 8 del flujo normal, la API de servicios web detecta que el monto es un valor negativo o cero. 2. La API de servicios web envía una respuesta al sistema front-end con el código 422 (Unprocessable Entity) y el mensaje “Monto inválido”. 3. El sistema front-end muestra una ventana emergente con el mensaje “Monto inválido” y los botones de “Aceptar” o “Cancelar”. 4. El vendedor o cliente da clic en el botón “Aceptar”. 5. Termina el caso de uso.
Postcondición(es).	POST-1. El sistema gestiona los puntos de fidelidad del CLIENTE.
Regla(s) de negocio.	RN-3.
Incluye.	Ninguno.
Extiende.	Ninguno.

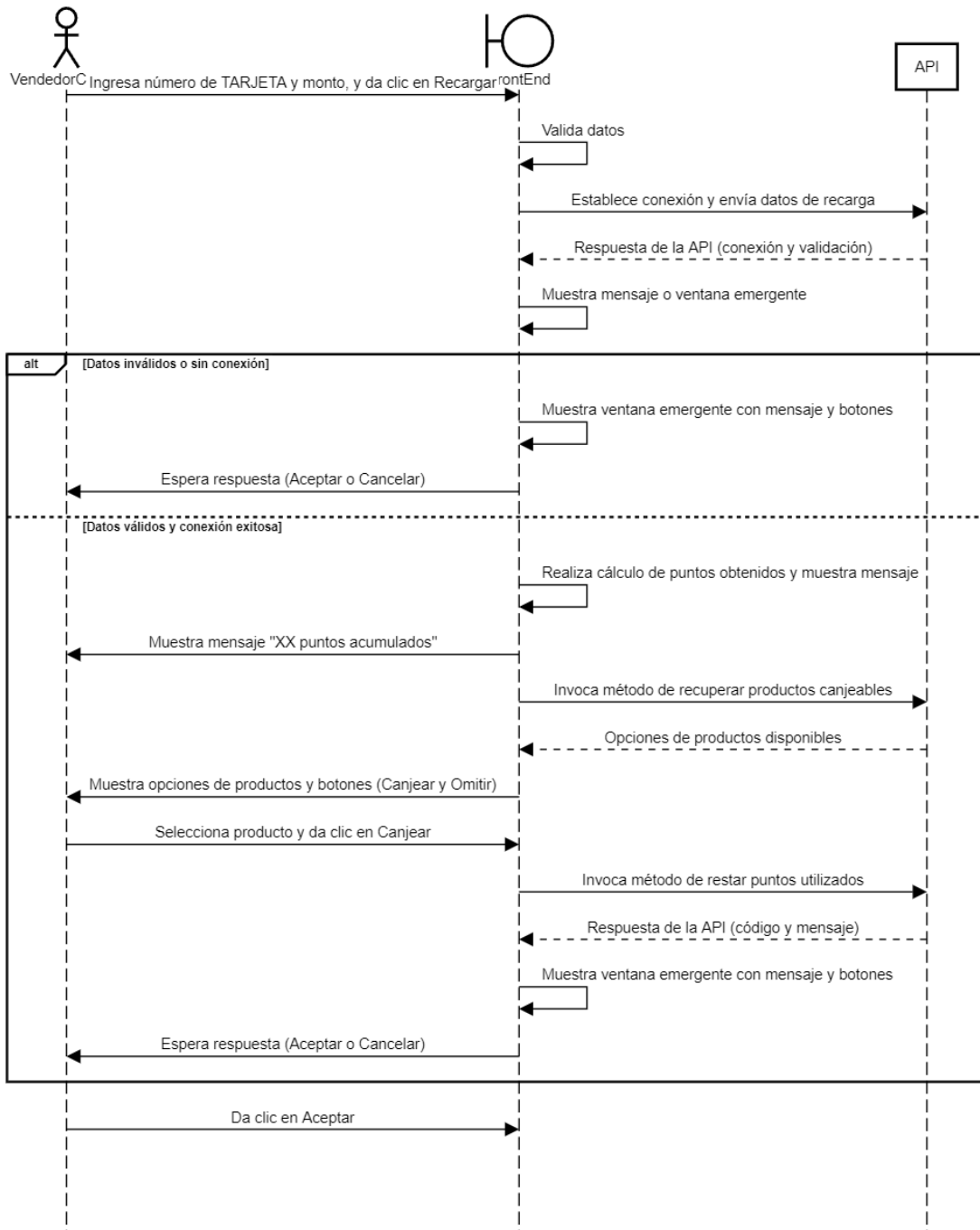
Puntos De Fidelidad

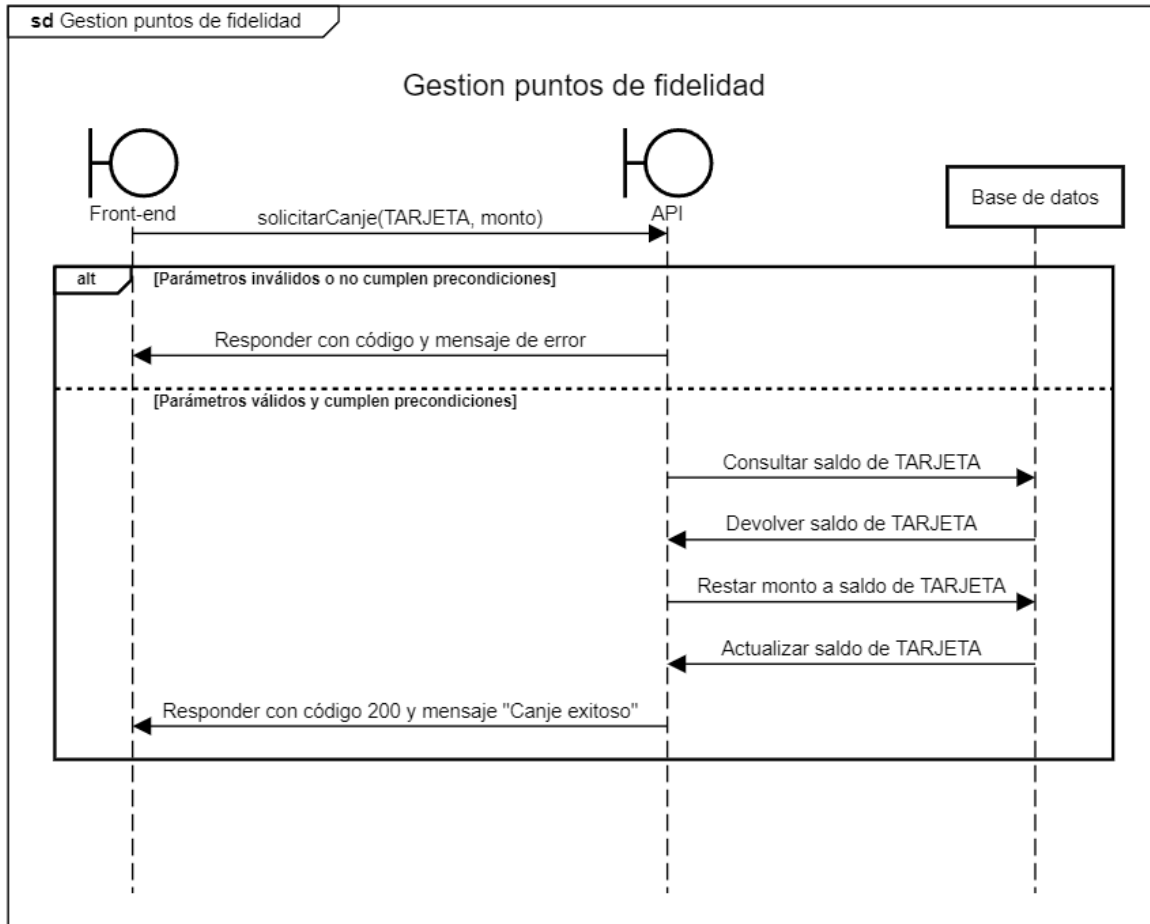
Cuentas con: puntos, que puedes canjear por:

Omitir

Canjear

"Gestionar puntos de fidelidad"





ID	CU-20.
Nombre.	Gestionar Publicidad.
Responsable.	Fausto Mendez Peralta
Fecha actualización. de	24/04/2023.
Descripción.	Permite al actor gestionar PUBLICIDAD
Actor(es).	Administrador.
Disparador.	El administrador da clic en el botón "Publicidad" ubicado en la página principal.
Precondición(es).	Ninguna.

Flujo normal.	<ol style="list-style-type: none"> 1. El sistema front-end establece conexión con la API de servicios web y muestra GUI “Gestión de publicidad”, junto con una tabla con las columnas “Id”, “Nombre”, “Fecha de inicio”, “Fecha de conclusión” y “Banner”, junto con los botones de “Nuevo”, “Modificar” y “Eliminar”. (EX-1) 2. El sistema front-end invoca el método recuperar publicidad de la API de servicios web y muestra la información de la PUBLICIDAD recuperada de la base de datos y los ordena en la tabla. (FA-2.1, FA-2.2, FA-2.3) 3. El administrador da clic en el botón de “Regresar”. 4. Termina el caso de uso.
Flujos alternos.	<p>FA-2.1. El administrador da clic en el botón de “Nuevo”.</p> <ol style="list-style-type: none"> 1. El sistema front-end muestra GUI “Nueva publicidad”, junto con los campos de nombre, fecha de inicio, fecha de fin, un apartado para seleccionar el banner y los botones guardar y cancelar. 2. El administrador llena los campos y da clic en el botón “Guardar”. 3. El sistema front-end invoca el método nueva publicidad de la API de servicios web con los parámetros nombre, fecha de inicio, fecha de fin y banner. 4. La API de servicios web recibe la solicitud del sistema front-end con los parámetros nombre, fecha de inicio, fecha de fin y banner. 5. La API de servicios web verifica que los parámetros sean válidos y que cumplan con las precondiciones: la API tiene conexión con la base de datos y el nombre no está repetido en la base de datos. (EX-2, EX-3) 6. La API de servicios web guarda la información de la nueva PUBLICIDAD en la base de datos y envía una respuesta al sistema front-end con el código 200 (OK) y el mensaje “Publicidad creada”. 7. Termina el caso de uso. <p>FA-2.2. El administrador da clic en el botón de “Eliminar”.</p> <ol style="list-style-type: none"> 1. El administrador selecciona la publicidad que desea eliminar y da clic en el botón eliminar. 2. El sistema front-end invoca el método eliminar publicidad de la API de servicios web con el parámetro id. 3. La API de servicios web recibe la solicitud del sistema front-end con el parámetro id. 4. La API de servicios web verifica que el parámetro sea válido y que cumpla con las precondiciones: la API tiene conexión con la base de datos y el id existe en la base de datos. (EX-4, EX-5) 5. La API de servicios web elimina la información de la PUBLICIDAD con el id correspondiente de la base de datos y envía una respuesta al sistema front-end con el código 200 (OK) y el mensaje “Publicidad eliminada”. 6. Termina el caso de uso. <p>FA-2.3. El administrador da clic en el botón de “Modificar”.</p> <ol style="list-style-type: none"> 1. El administrador selecciona la PUBLICIDAD que desea modificar y da clic. 2. El sistema front-end muestra GUI “Modificar”, junto con los campos de nombre, fecha de inicio, fecha de fin, un apartado para seleccionar el banner y los botones guardar y cancelar. 3. El administrador modifica los campos de nombre, fecha de inicio, fecha de fin, selecciona el banner y da clic en el botón “Guardar”. 4. El sistema front-end invoca el método modificar publicidad de la API de servicios web con los parámetros id, nombre, fecha de inicio, fecha de fin y banner. 5. La API de servicios web recibe la solicitud del sistema front-end con los parámetros id, nombre, fecha de inicio, fecha de fin y banner. 6. La API de servicios web verifica que los parámetros sean válidos y que cumplan con las precondiciones: la API tiene conexión con la base de datos, el id existe en la base de datos y el nombre no está repetido en la base de datos (excepto por el mismo id). (EX-4, EX-5, EX-6) 7. La API de servicios web actualiza la información de la PUBLICIDAD con el id correspondiente en la base de datos y envía una respuesta al sistema front-end con el código 200 (OK) y el mensaje “Publicidad modificada”. 8. Termina el caso de uso.

<p>Excepciones.</p>	<p>EX-1. No hay conexión con la API de servicios web.</p> <ol style="list-style-type: none"> 1. En el paso 1 o 2 o 3 o 4 del flujo normal o en cualquier paso del flujo alterno donde se invoque un método de la API, el sistema front-end no logra conectarse a la API de servicios web. 2. El sistema front-end muestra una ventana emergente con el mensaje “No hay conexión con la API” y los botones de “Aceptar” o “Cancelar”. 3. El administrador da clic en el botón “Aceptar”. 4. Termina el caso de uso. <p>EX-2. Los parámetros son inválidos.</p> <ol style="list-style-type: none"> 1. En cualquier paso del flujo alterno donde se reciban parámetros desde el sistema front-end, la API de servicios web detecta que los parámetros no tienen el formato o el tipo correcto. 2. La API de servicios web envía una respuesta al sistema front-end con el código 400 (Bad Request) y el mensaje “Parámetros inválidos”. 3. El sistema front-end muestra una ventana emergente con el mensaje “Parámetros inválidos” y los botones de “Aceptar” o “Cancelar”. 4. El administrador da clic en el botón “Aceptar”. 5. Termina el caso de uso. <p>EX-3. El nombre está repetido en la base de datos.</p> <ol style="list-style-type: none"> 1. En el paso 5 del flujo alterno FA-2.1, la API de servicios web detecta que el nombre recibido ya existe en la base de datos. 2. La API de servicios web envía una respuesta al sistema front-end con el código 409 (Conflict) y el mensaje “Nombre repetido”. 3. El sistema front-end muestra una ventana emergente con el mensaje “Nombre repetido” y los botones de “Aceptar” o “Cancelar”. 4. El administrador da clic en el botón “Aceptar”. 5. Termina el caso de uso. <p>EX-4. El id no existe en la base de datos.</p> <ol style="list-style-type: none"> 1. En el paso 4 del flujo alterno FA-2.2 o en el paso 6 del flujo alterno FA-2.3, la API de servicios web detecta que el id recibido no se encuentra en la base de datos. 2. La API de servicios web envía una respuesta al sistema front-end con el código 404 (Not Found) y el mensaje “Id no encontrado”. 3. El sistema front-end muestra una ventana emergente con el mensaje “Id no encontrado” y los botones de “Aceptar” o “Cancelar”. 4. El administrador da clic en el botón “Aceptar”. 5. Termina el caso de uso. <p>EX-5. No hay conexión con la base de datos.</p> <ol style="list-style-type: none"> 1. En cualquier paso del flujo alterno donde se acceda a la base de datos, la API de servicios web no logra conectarse a la base de datos. 2. La API de servicios web envía una respuesta al sistema front-end con el código 500 (Internal Server Error) y el mensaje “No hay conexión con la base de datos”. 3. El sistema front-end muestra una ventana emergente con el mensaje “No hay conexión con la base de datos” y los botones de “Aceptar” o “Cancelar”. 4. El administrador da clic en el botón “Aceptar”. 5. Termina el caso de uso. <p>EX-6. El nombre está repetido en la base de datos (excepto por el mismo id).</p> <ol style="list-style-type: none"> 1. En el paso 6 del flujo alterno FA-2.3, la API de servicios web detecta que el nombre recibido ya existe en la base de datos para otro id distinto al recibido. 2. La API de servicios web envía una respuesta al sistema front-end con el código 409 (Conflict) y el mensaje “Nombre repetido”. 3. El sistema front-end muestra una ventana emergente con el mensaje “Nombre repetido” y los botones de “Aceptar” o “Cancelar”. 4. El administrador da clic en el botón “Aceptar”.
----------------------------	--

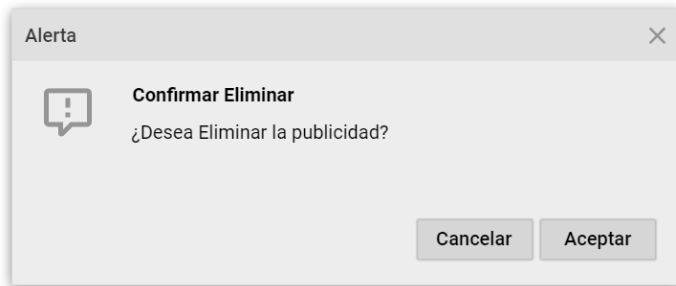
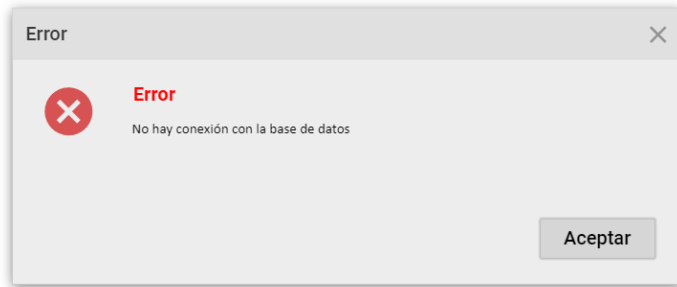
	5. Termina el caso de uso.
Postcondición(es).	POST-1. El sistema gestiona de manera correcta la PUBLICIDAD
Regla(s) de negocio.	RN-3.
Incluye.	Ninguno.
Extiende.	Ninguno.

ID	NOMBRE	FECHA INICIO	FECHA FIN	BANER
001	Dulces Axel	20-06-2023	28-06-2023	001.png
002	Galletas	21-06-2023	22-06-2023	002.png
003	Helados	01-07-2023	05-07-2023	003.png

Eliminar

Modificar

Nuevo



Nombre

Fecha de inicio


08/13/2018



Fecha de fin

08/13/2018



 Baner

Cancelar

Guardar

Nombre

Fecha de inicio


08/13/2018



Fecha de fin

08/13/2018

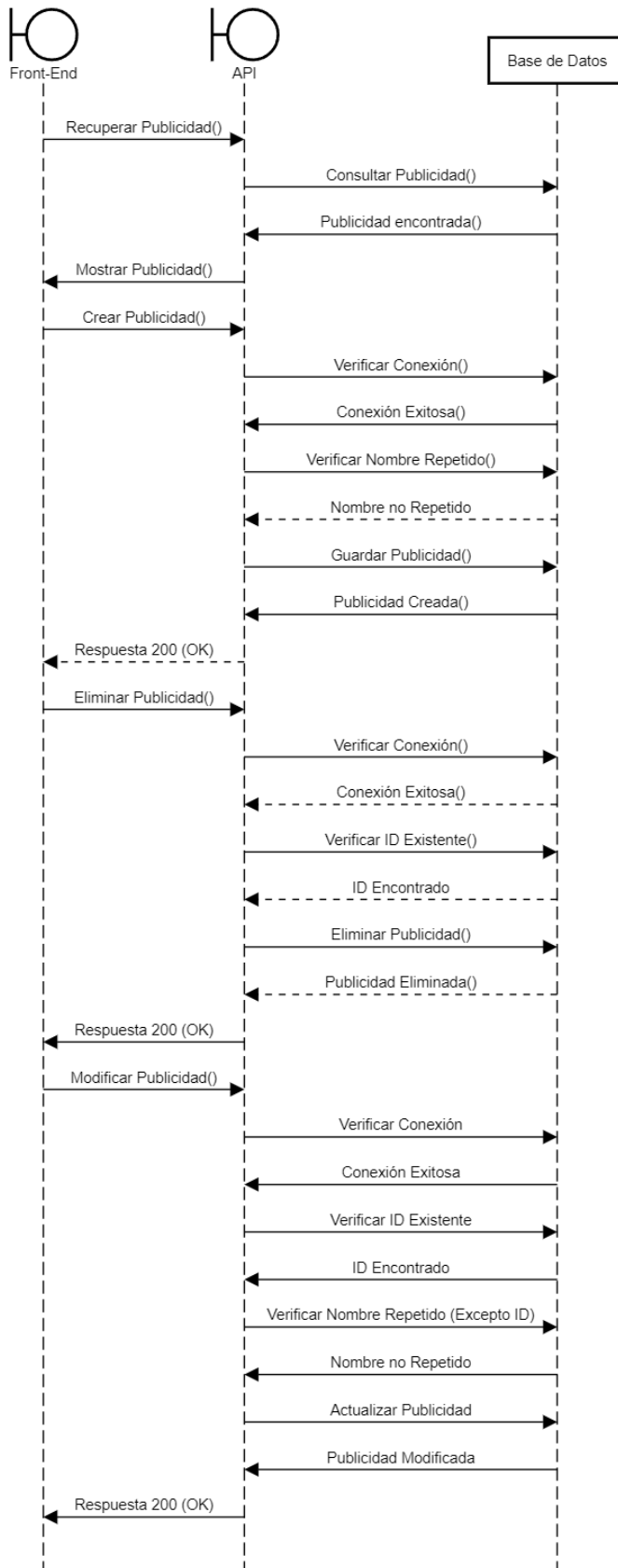


 Baner

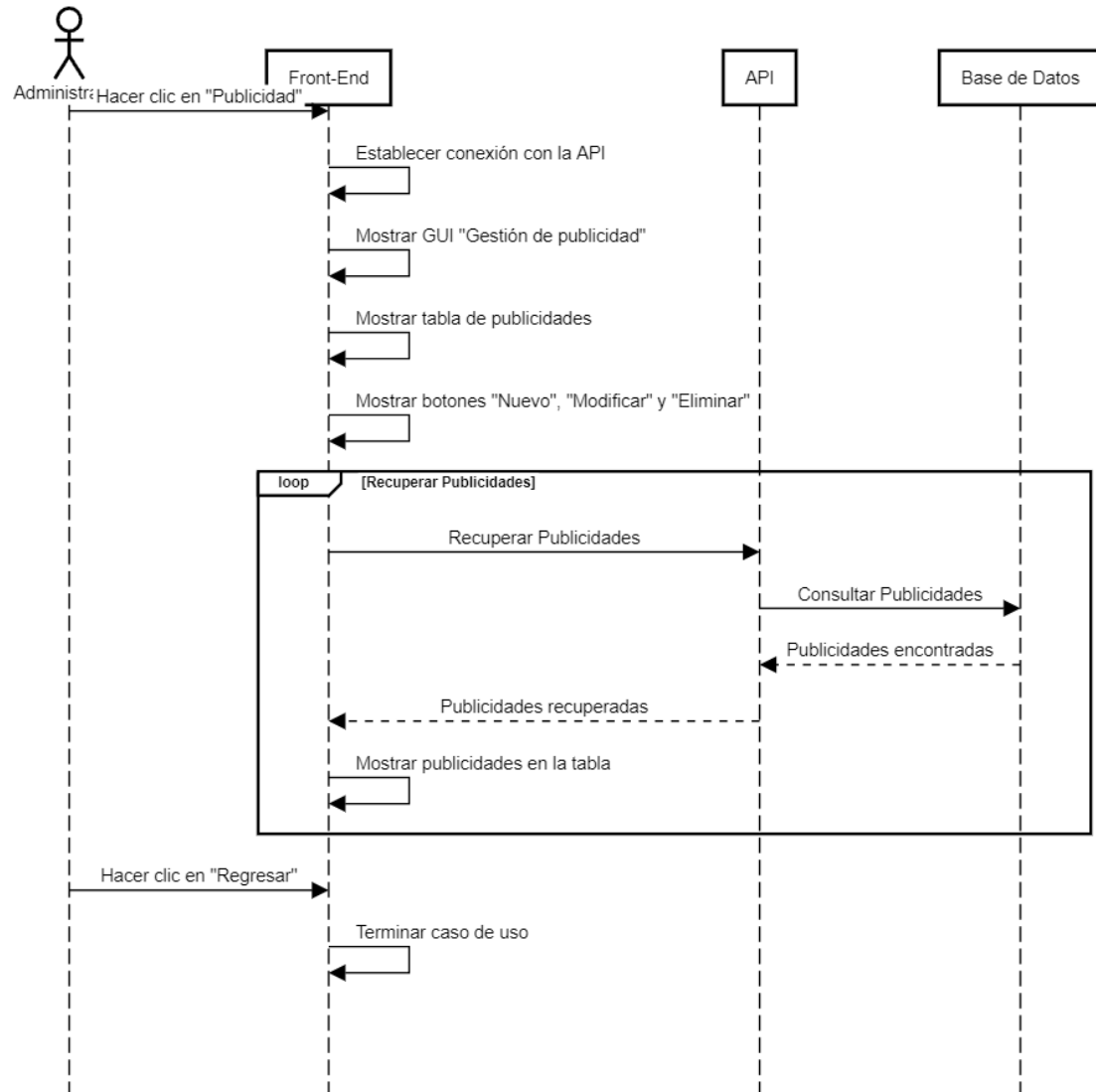
Cancelar

Guardar

Gestionar Publicidad



Gestionar Publicidad



Diagrama

de

Componentes

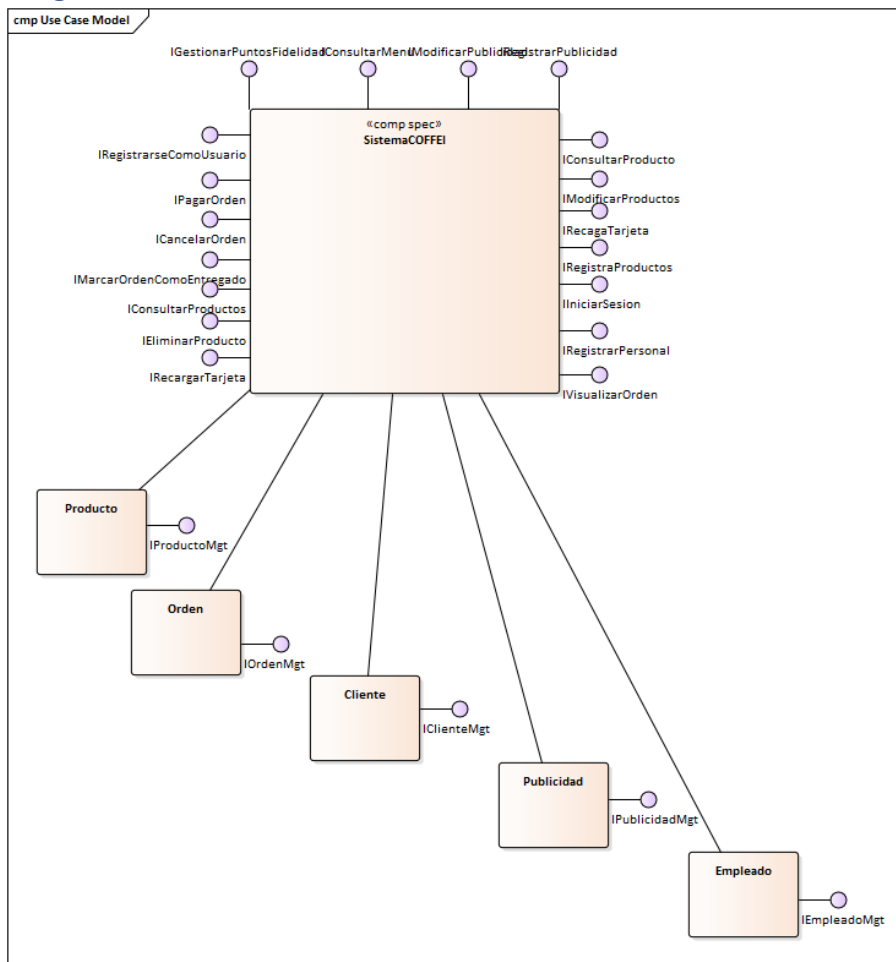
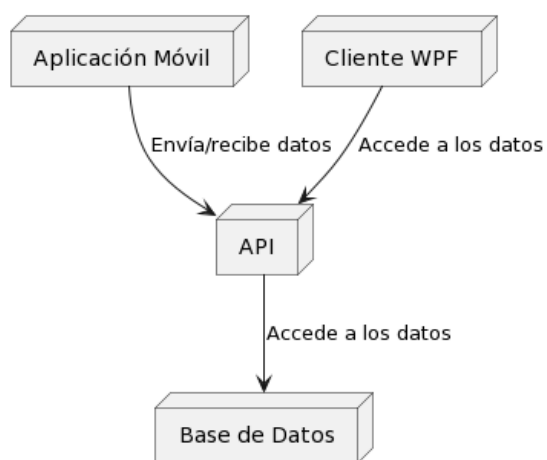


Diagrama de Despliegue:



Plan de pruebas.

1. Registrar producto

- Enviar una solicitud POST a la API con los detalles del producto.
- Verificar que el producto se haya registrado correctamente en la base de datos.

2. Modificar producto

- Enviar una solicitud PUT a la API con los detalles actualizados del producto.
- Verificar que el producto se haya modificado correctamente en la base de datos.

3. Consultar producto

- Enviar una solicitud GET a la API con el ID del producto.
- Verificar que la respuesta contenga los detalles correctos del producto consultado.

4. Eliminar producto

- Enviar una solicitud DELETE a la API con el ID del producto.
- Verificar que el producto se haya eliminado correctamente de la base de datos.

5. Iniciar sesión

- Enviar una solicitud POST a la API con las credenciales de inicio de sesión.
- Verificar que la API responda con un token de autenticación válido.

6. Registrarse como cliente

- Enviar una solicitud POST a la API con los detalles de registro del cliente.
- Verificar que el cliente se haya registrado correctamente en la base de datos.

7. Registrar empleado

- Enviar una solicitud POST a la API con los detalles de registro del empleado.
- Verificar que el empleado se haya registrado correctamente en la base de datos.

8. Modificar puntos de fidelidad

- Enviar una solicitud PUT a la API con el ID del cliente y los puntos de fidelidad actualizados.
- Verificar que los puntos de fidelidad del cliente se hayan modificado correctamente en la base de datos.

9. Modificar dinero

- Enviar una solicitud PUT a la API con el ID de la tarjeta y la cantidad de dinero actualizada.
- Verificar que la cantidad de dinero en la tarjeta se haya modificado correctamente en la base de datos.

10. Agregar tarjeta

- Enviar una solicitud POST a la API con los detalles de la tarjeta.
- Verificar que la tarjeta se haya agregado correctamente a la base de datos.

11. Agregar, modificar, eliminar publicidad

- Enviar solicitudes POST, PUT y DELETE a la API para agregar, modificar y eliminar publicidad respectivamente.
- Verificar que la publicidad se haya agregado, modificado o eliminado correctamente en la base de datos.

12. Consultar menú

- Enviar una solicitud GET a la API para obtener el menú actual.
- Verificar que la respuesta contenga los detalles correctos del menú.

13. Pagar orden

- Enviar una solicitud POST a la API con los detalles de la orden y el método de pago.
- Verificar que la orden se haya pagado correctamente y que el total pagado se haya actualizado en la base de datos.

14. Cancelar orden

- Enviar una solicitud DELETE a la API con el ID de la orden a cancelar.
- Verificar que la orden se haya cancelado correctamente en la base de datos.

No.	Método	Ruta	Parámetros	Resultado Esperado
1	GET	/productos	N/A	Lista de productos obtenida correctamente
2	POST	/productos	Detalles del producto nuevo	Producto registrado correctamente
3	GET	/productos/{id}	ID del producto	Detalles correctos del producto consultado
4	PUT	/productos/{id}	ID del producto, Detalles actualizados	Producto modificado correctamente
5	DELETE	/productos/{id}	ID del producto	Producto eliminado correctamente
6	POST	/login	Credenciales de inicio de sesión	Token de autenticación válido
7	POST	/clientes	Detalles de registro del cliente	Cliente registrado correctamente
8	POST	/empleados	Detalles de registro del empleado	Empleado registrado correctamente
9	PUT	/clientes/{id}	ID del cliente, Puntos de fidelidad	Puntos de fidelidad modificados correctamente
10	PUT	/tarjetas/{id}	ID de la tarjeta, Cantidad de dinero	Cantidad de dinero modificada correctamente

11	POST	/tarjetas	Detalles de la tarjeta	Tarjeta agregada correctamente
12	POST	/publicidad	Detalles de la publicidad	Publicidad agregada correctamente
13	PUT	/publicidad/{id}	ID de la publicidad, Detalles actualizados	Publicidad modificada correctamente
14	DELETE	/publicidad/{id}	ID de la publicidad	Publicidad eliminada correctamente
15	GET	/menu	N/A	Detalles correctos del menú obtenidos
16	POST	/orden	Detalles de la orden, Método de pago	Orden pagada correctamente
17	DELETE	/orden/{id}	ID de la orden	Orden cancelada correctamente

Conclusiones:

Durante el proceso de desarrollo de una API REST en JavaScript utilizando Node.js y la conexión con clientes WPF y Android, hemos adquirido varios aprendizajes significativos y hemos enfrentado diversos retos que nos han permitido crecer como estudiantes de ingeniería de software.

A continuación, resumimos nuestras conclusiones sobre esta experiencia:

En cuanto a los aprendizajes, hemos comprendido la importancia de seguir buenas prácticas de diseño y desarrollo de API REST para garantizar su eficiencia, escalabilidad y seguridad. Hemos adquirido conocimientos sobre la estructura básica de una API REST, cómo definir y gestionar rutas, y cómo realizar operaciones CRUD en la base de datos desde la API. Además, hemos aprendido a utilizar bibliotecas y frameworks populares en Node.js, como Express, para simplificar el desarrollo y facilitar la creación de endpoints.

También hemos comprendido la importancia de la documentación clara y detallada de la API, ya que esto facilita la comunicación y colaboración entre los equipos de desarrollo de la API y los clientes. La documentación precisa de los endpoints, los parámetros requeridos, las respuestas esperadas y los posibles errores brinda a los desarrolladores de los clientes la información necesaria para consumir correctamente la API y manejar situaciones inesperadas de manera adecuada.

Otro desafío ha sido lograr una buena arquitectura y estructura de código en la API, especialmente al manejar una mayor complejidad en las rutas y operaciones. Hemos aprendido a dividir el código en módulos y capas, utilizando el patrón MVC (Modelo-Vista-Controlador) u otros patrones arquitectónicos adecuados para mantener el código organizado, legible y fácil de mantener.

La conexión con diferentes clientes, como WPF y Android, también ha presentado desafíos adicionales. Ha sido necesario comprender las particularidades de cada plataforma y adaptar la comunicación con la API a las necesidades de cada cliente. Esto ha implicado trabajar con diferentes formatos de datos, como JSON, y asegurarse de que los clientes puedan enviar y recibir datos correctamente.

En cuanto a las dificultades y retos del trabajo en equipo, hemos experimentado la necesidad de una comunicación clara y constante entre los miembros del equipo de desarrollo. Coordinar esfuerzos, definir responsabilidades y mantener una buena colaboración son aspectos fundamentales para lograr un proyecto exitoso. Además, hemos enfrentado desafíos relacionados con la sincronización de las tareas y la gestión eficiente de los recursos y tiempos.

En conclusión, la experiencia de desarrollar una API REST en JavaScript con Node.js y conectarla con clientes WPF y Android ha sido enriquecedora y desafiante, tanto en términos técnicos como en el ámbito del trabajo en equipo. Los aprendizajes obtenidos sobre el diseño de API REST, las prácticas de seguridad, la documentación y la estructura de código, junto con las habilidades adquiridas para enfrentar los desafíos

Bibliografia:

Fielding, R. T. (2000). Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, University of California, Irvine. URL: <https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>

Richardson, L., & Amundsen, M. (2013). RESTful Web APIs: Services for a Changing World. O'Reilly Media.

Almasri, E., & Wats, D. (2019). Mastering Node.js: Develop robust and scalable real-time server-side web applications. Packt Publishing.

Fractal, L. (2020). API Development with Node.js. Apress.

Newman, M. (2015). Building Microservices: Designing Fine-Grained Systems. O'Reilly Media.

Fielding, R. T., & Taylor, R. N. (2002). Principled design of the modern Web architecture. ACM Transactions on Internet Technology (TOIT), 2(2), 115-150. doi:10.1145/514183.514185

Masood, F., & Dhamodharan, K. (2019). Learning Node.js Development: Learn the fundamentals of Node.js, and deploy and test Node.js applications on the web. Packt Publishing.