**Registro Histórico de Cambios en el Documento**

**Tabla 1.** Versiones del documento

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Código** | **Fecha** | **Elaboró** | **Revisó** | **Descripción Cambio** | **Versión** |
| **C-00** | **28/04/2021** | ***Líder de equipo:*** *Sebastián Orozco* |  | **Especificación inicial del documento** | **1.0** |
|  |  |  |  |  |  |

**Especificación y Análisis de Requisitos**

En el presente documento se presentan especificaciones de diferentes aspectos relacionados con el desarrollo del proyecto ***Restaurantes Saboreo***, como lo son: La identificación del problema que requiere una solución de software, el equipo de Desarrollo de software encargado de él junto a los respectivos roles y responsabilidades que tomará cada integrante del equipo, se definen los diferentes canales, momentos de reunión y reglas del juego del equipo para el desarrollo del proyecto; se presenta una descripción general de los posibles usuarios que harán uso del sistema así como sus roles y/o responsabilidades a la hora de usar la aplicación, se realiza un análisis preliminar de problemas específicos del proyecto y posibles soluciones a estos, se identifican los objetivos del negocio y las diferentes funcionalidades del software, se describen características de calidad, el control de la misma y reglas del negocio, se presenta diagrama de contexto del sistema así como la estrategia para el desarrollo, plan de iteraciones en conjunción con los diferentes casos de uso.

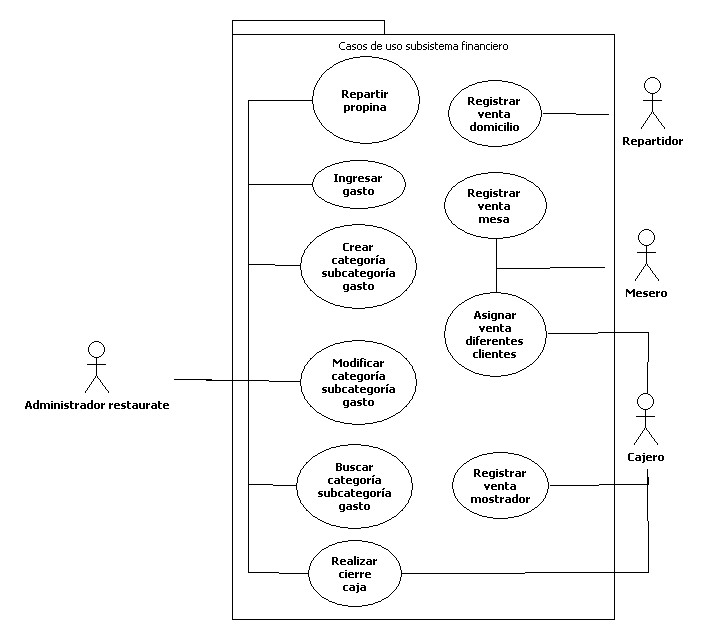
# Modelo de Casos de Uso

**Tabla 2.** Actores

|  |  |
| --- | --- |
| **Actor** | **Descripción** |
| Cajero. | Persona que maneja la Caja, manipulación del dinero. |
| Mesero | Persona que se encarga de la atención al cliente. |
| Administrador | Persona encargada de la gestión general del sistema |
| Chef | Persona encargada de todo lo relacionado a la cocina |

**Tabla 3.** Casos de uso subsistema financiero

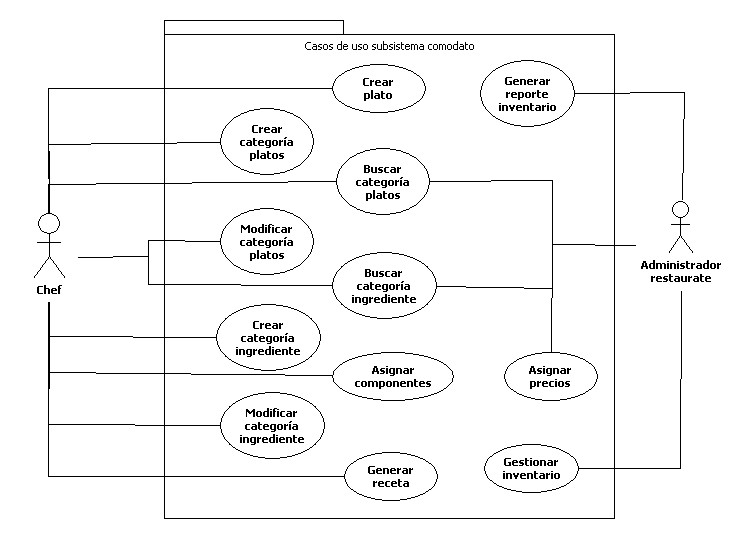
|  |  |  |
| --- | --- | --- |
| **Subsistema financiero** | | |
| **ID** | **Caso de uso** | **Descripción** |
| **SFCU01** | Repartir propina | Permitir la repartición de las propinas equitativamente entre los meseros de turno |
| **SFCU02** | Ingresar gastos de Restaurante | Este caso de uso permite ingresar los gastos realizados por el restaurante de acuerdo a categoría |
| **SFCU03** | Crear categorías y subcategorías de gastos | Este caso de uso permite la creación de las categorías y subcategorías de los gastos del restaurante |
| **SFCU04** | Modificar categorías y subcategorías de gastos | Este caso de uso permite la modificación de las categorías y subcategorías de los gastos del restaurante |
| **SFCU05** | Buscar categorías y subcategorías de gastos | Este caso de uso permite |
| **SFCU06** | Realizar cierre de caja | Permitir realizar el arqueo de caja viendo un reporte detallado de ingresos y egresos en un rango de fechas. |
| **SFCU07** | Asignar ventas a diferentes clientes | Permite asignar las ventas de un cliente que no se encuentre en ninguna mesa. |
| **SFCU08** | Registrar venta mostrador | Permite el registro de las ventas realizadas en mostrador |
| **SFCU09** | Registrar venta mesa | Permite el registro de las ventas realizadas en mesa |
| **SFCU10** | Registrar venta domicilio | Permite el registro de las ventas a domicilio |



**Imagen 1.** Diagrama de casos de uso subsistema financiero

**Tabla 4.** Casos de uso subsistema comodato

|  |  |  |
| --- | --- | --- |
| **Subsistema comodato** | | |
| **ID** | **Caso de uso** | **Descripción** |
| SCCU01 | Crear categorías de platos | El sistema debe permitir la creación de categorías de platos |
| SCCU02 | Modificar categorías de platos | El sistema debe permitir la modificación de categorías de platos |
| SCCU03 | Buscar categorías de platos | El sistema debe permitir la búsqueda de categorías de platos |
| SCCU04 | Crear categorías de ingredientes | El sistema debe permitir la creación de categorías de ingredientes |
| SCCU05 | Modificar categorías de ingredientes | El sistema debe permitir la modificación de categorías de ingredientes |
| SCCU06 | Buscar categorías de ingredientes | El sistema debe permitir la búsqueda de categorías de ingredientes |
| SCCU07 | Asignar componentes | El sistema debe permitir asignar componentes a una receta |
| SCCU08 | Asignar precios | El sistema debe permitir asignar precio a una receta |
| SCCU09 | Generar receta | El sistema debe permitir generar una receta |
| SCCU10 | Crear plato | El sistema debe permitir crear un plato |
| SCCU11 | Gestionar inventario | El sistema debe permitir realizar la gestión del inventario |
| SCCU12 | Generar reporte inventario | El sistema debe permitir obtener un informe del inventario |



**Imagen 2.** Diagrama de casos de uso subsistema comodato

# Características de Calidad / Requisitos NO Funcionales

**Tabla 5.** Características de calidad

|  |  |  |
| --- | --- | --- |
| **Código** | **Característica** | **Descripción** |
| PRY-CC-001 | Desempeño | El sistema debe permitir conocer información en tiempo real. |
| PRY-CC-002 | Escalabilidad | El sistema debe permitir su crecimiento y adaptarse a nuevas funcionalidades. |
| PRY-CC-003 | Seguridad | El sistema debe proteger los datos y lógica frente a accesos no autorizados. |
| PRY-CC-004 | Disponibilidad | El sistema deberá permanecer en estado activo ocultando las fallas que se presenten en el servicio. |

# Restricciones

Tabla 6. Restricciones del proyecto

|  |  |
| --- | --- |
| **Código** | **Descripción** |
| PRY-RE-001 | El sistema en la parte Backend deberá realizarse en el framework dotNet |
| PRY-RE-002 | El sistema en la parte Frontend deberá realizarse en el framework angular |

# Primera iteración

En esta primera iteración se busca sentar las bases para el desarrollo del proyecto, así como desarrollar la mayoría de funcionalidades del subsistema financiero del restaurante. En este sentido se realizarán los casos de uso que, a consideración propia, demanden mayor de trabajo, principalmente del subsistema en cuestión.

## Objetivos de Negocio

1. Administrar y categorizar de manera eficiente y concreta los gastos de los restaurantes para llevar registros de estos.
2. Tener un sistema de administración y contabilidad de las categorías de ventas (mesa, domicilio y mostrador) de los restaurantes.
3. Administrar y cerrar de manera eficiente la caja y la facturación de los restaurantes.
4. Asignación de las propinas dentro de los restaurantes.

## Características de Calidad

* Desempeño
* Escalabilidad

## Requisitos de usuario

En la primera iteración se realizarán los siguientes casos de usos:

1. Repartir propina.
2. Crear categorías y subcategorías de gastos.
3. Modificar categorías y subcategorías de gastos.
4. Asignación de ventas a clientes.
5. Registrar ventas por mostrador.
6. Registrar ventas en mesa.

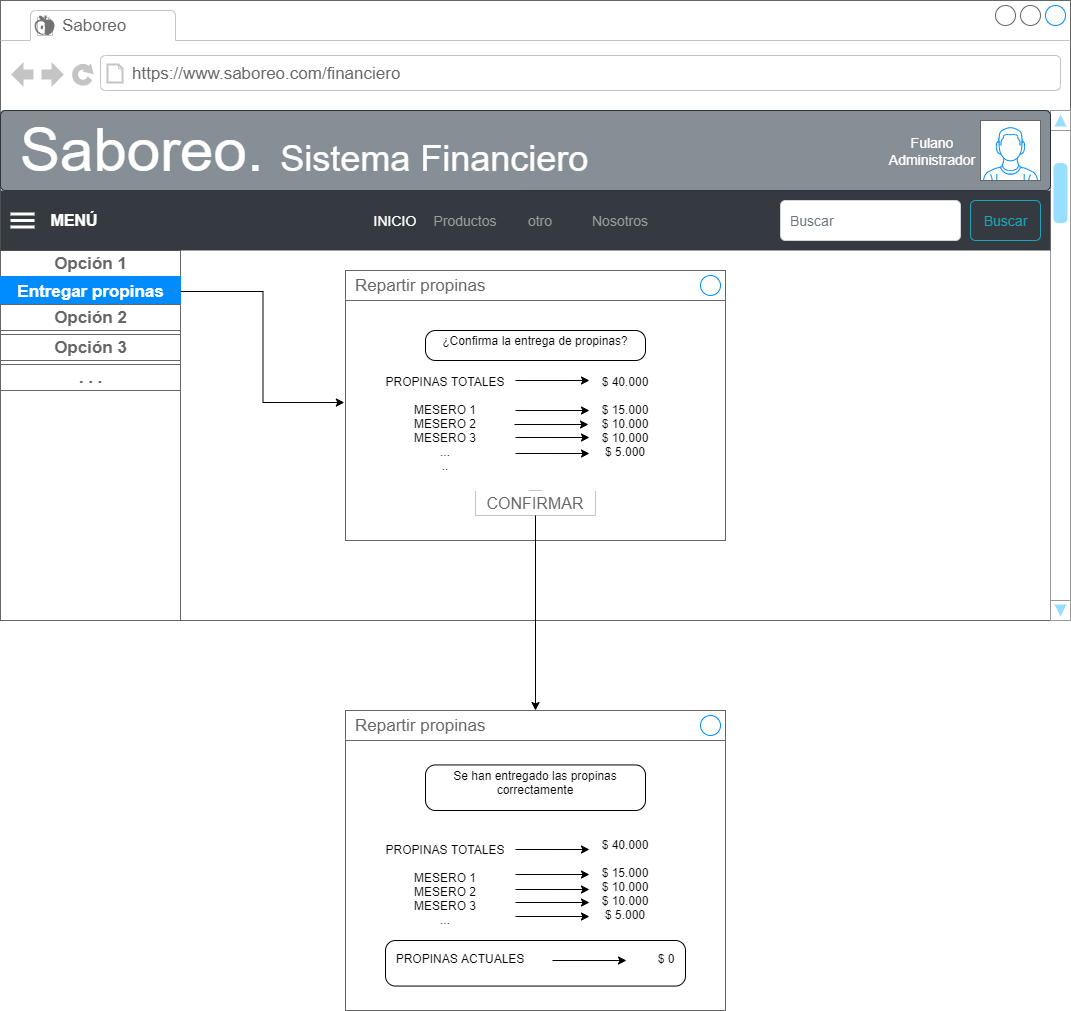
**CASO DE USO SFCU01-REPARTIR PROPINA**

**Especificación de caso de uso**

**Tabla 7.** Especificación caso de uso SFCU01\_RepartirPropina

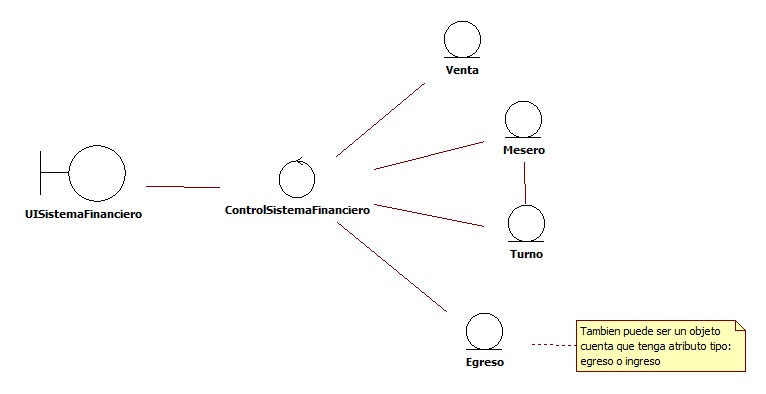
|  |  |
| --- | --- |
| **ID** | **SFCU01** |
| **Nombre** | Repartir propinas |
| **Descripción** | Permite acumular las diferentes propinas (en un periodo de tiempo: Turno) y que estas sean repartidas de forma equitativa entre los meseros de acuerdo a los turnos realizados |
| **Actores** | Administrador Restaurante |
| **Precondiciones** | -El actor ya está debidamente autenticado en el sistema  -Hay información registrada de al menos un mesero  -Se registra cuanto es la propina de cada venta |
| **Postcondiciones** | -Se registra el egreso por la cantidad equivalente a las propinas  -Se asigna una propina para cada mesero |
|  | |
| **Flujo normal de eventos:** | |
| **1.** El actor selecciona la opción "Repartir propinas". **2.** El sistema valida que hayan propinas en las cuentas de las ventas de la jornada  **3.** El sistema define que meseros trabajaron durante la jornada.  **4.** El sistema obtiene las horas totales trabajadas y las horas que trabajo cada mesero.  **5.** El sistema calcula el porcentaje de trabajo de cada mesero con respecto a las horas trabajadas totales por los meseros.  **6.** El sistema calcula las propinas correspondientes a cada mesero de acuerdo a su porcentaje de trabajo en la jornada.  **7.** El sistema registra el egreso de las cuentas del restaurante.  **8.** El sistema muestra un mensaje donde se muestre las propinas de cada mesero y la opción de confirmar la entrega de propinas. | |
| **Flujos alternos y excepciones** | |
| **A-** No hay propinas:  En el paso 2 del flujo normal:  **a.** El sistema tiene registrado que no hay propinas, es decir, valor de cero.  **b.** El sistema muestra que no hay propinas. | |
| **B-** No se confirma la entrega de propinas:  En el paso 7 del flujo normal:  **a.** No se confirma la entrega de propinas por parte del actor.  **b.** Se muestra un mensaje de cancelación del proceso.  **c.** Termina caso de uso | |

**Mockup**

****

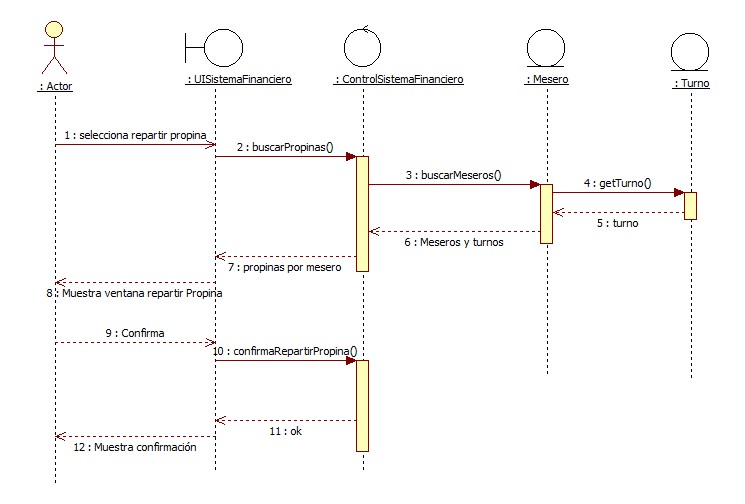
**Imagen 3.** Mockup SFCU01\_RepartirPropinas

**Modelo de clases del análisis**

****

**Imagen 4.** Modelo de clases del análisis SFCU01\_RepartirPropinas

**Diagrama de secuencia**

****

**Imagen 5.** Diagrama de secuencia SFCU01\_RepartirPropinas

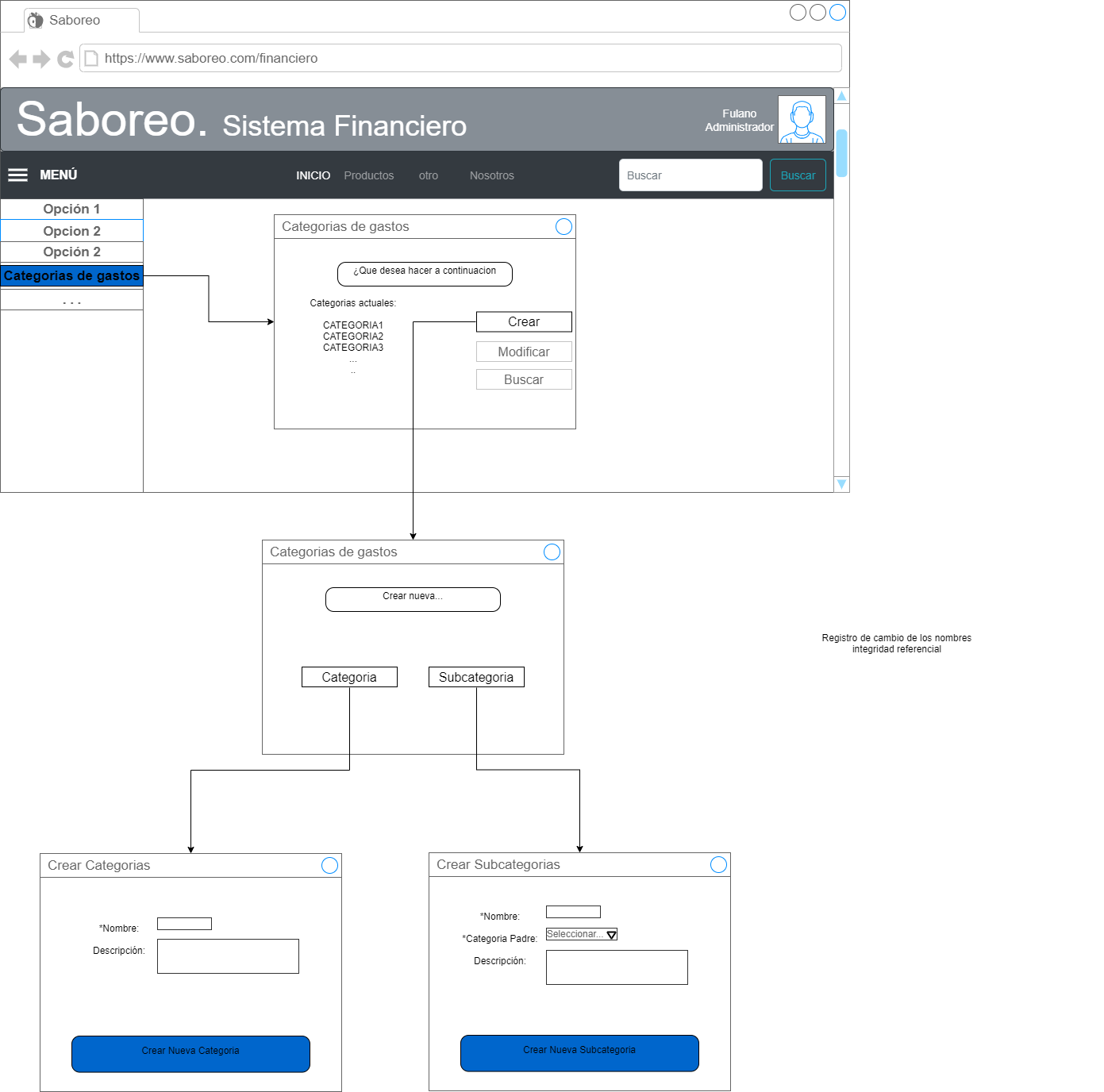
**CASO DE USO SFCU03-CREAR CATEGORÍA Y SUBCATEGORÍA DE GASTO**

**Especificación de caso de uso**

**Tabla 8.** Especificación caso de uso SFCU03\_CrearCatSubcatGasto

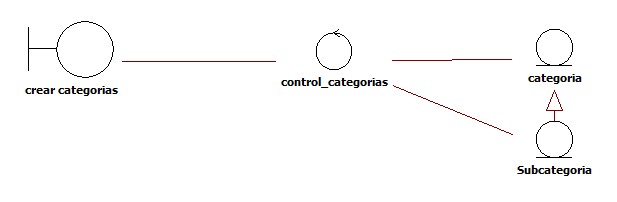
|  |  |
| --- | --- |
| **ID** | **SFCU03** |
| **Nombre** | Crear categorías y subcategorías de gastos |
| **Descripción** | Este caso de uso permite la creación de las categorías y subcategorías de los gastos del restaurante |
| **Actores** | Administrador(es) restaurante |
| **Precondiciones** | • El actor está autenticado en el sistema |
| **Postcondiciones** | Caso exitoso:  • Se crea una nueva categoría o subcategoría de gastos |
|  | |
| **Flujo normal de eventos** | |
| Crear nueva categoría:   1. El actor selecciona la opción CREAR 2. El sistema despliega un formulario con dos botones para crear nueva categoria   o nueva subcategoría   1. El actor selecciona el botón CATEGORÍA 2. El sistema despliega un formulario con los datos necesarios para una nueva categoría 3. El usuario completa los campos requeridos y selecciona el botón crear nueva categoria 4. El sistema alerta que se ha creado una nueva categoría | |
| **Flujos alternos**  Crear nueva subcategoría  • En el paso 3 del flujo normal   1. El actor selecciona el botón SUBCATEGORÍA 2. El sistema despliega un formulario con los datos necesarios para una nueva subcategoría 3. El usuario completa los campos requeridos y selecciona el botón crear nueva subcategoría 4. El sistema verifica la validez de los datos y crea una nueva subcategoría 5. El sistema alerta que se ha creado una nueva subcategoría | |
| **Excepciones**  Los datos proporcionados son inválidos/insuficientes:   * El sistema mostrara un mensaje indicando que hay campos inválidos o vacíos   La categoría ya existe:   * El sistema retorna una excepción indicando que ya existe la categoría a crear | |

**Mockup**

****

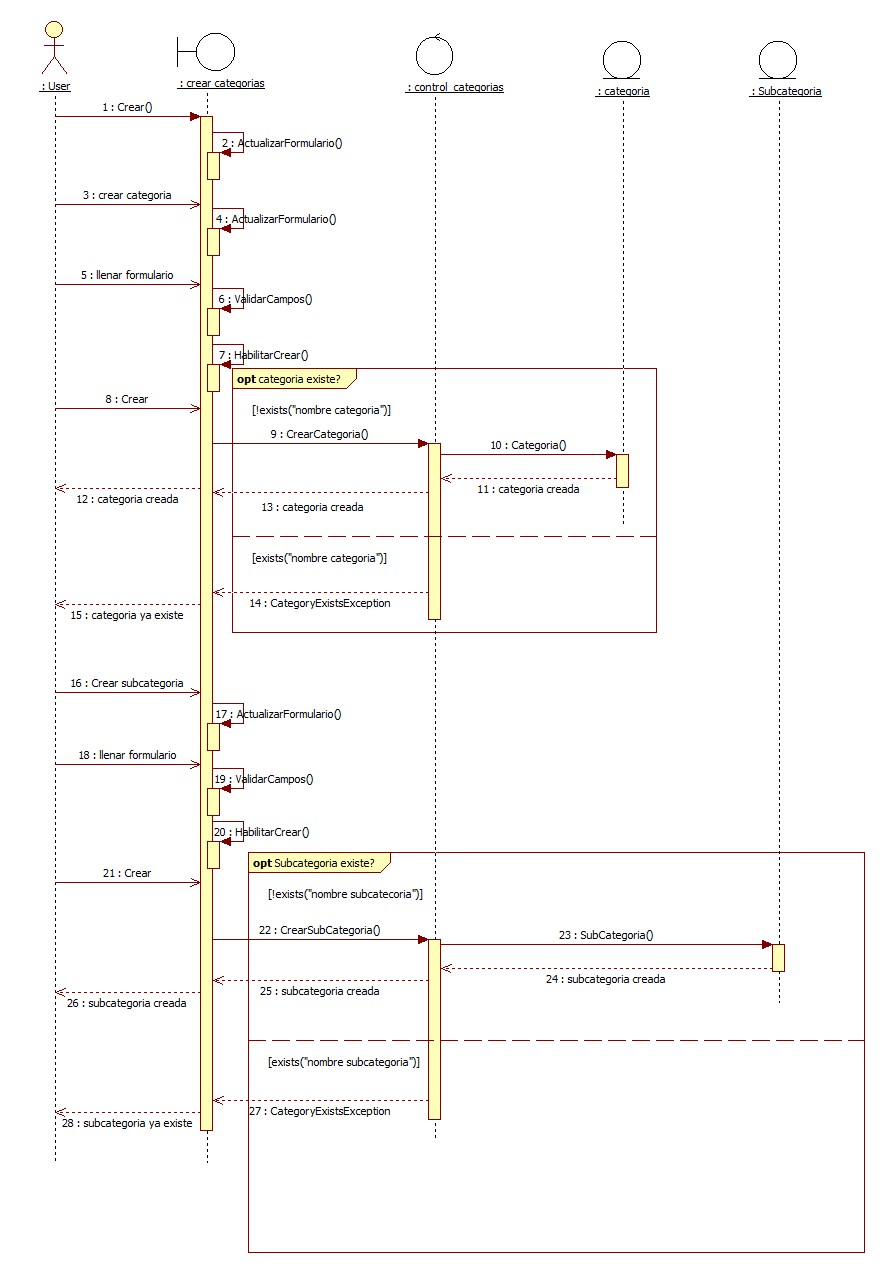
**Imagen 6.** Mockup SFCU03\_CrearCatSubcatGasto

**Modelo de clases del análisis**

****

**Imagen 7.** Modelo de clases del análisis SFCU03\_CrearCatSubcatGasto

**Diagrama de secuencia**

****

**Imagen 8.** Diagrama de clases SFCU03\_CrearCatSubcatGasto

**CASO DE USO SFCU04-MODIFICAR CATEGORÍA Y SUBCATEGORÍA DE GASTO**

**Especificación de caso de uso**

**Tabla 9.** Especificación caso de uso SFCU04\_ModificarCatSubcatGasto

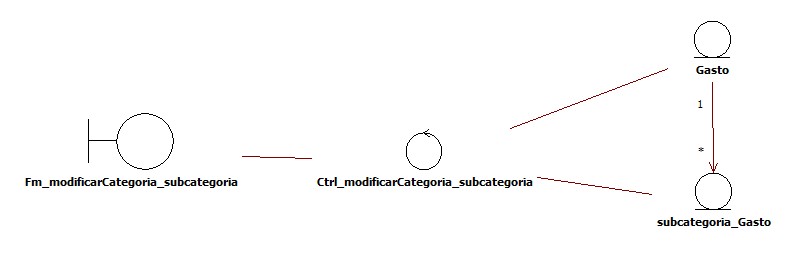
|  |  |
| --- | --- |
| **ID** | **SFCU04** |
| **Nombre** | Modificar categorías y subcategorías de gastos |
| **Descripción** | Este caso de uso permite la modificación de las categorías y subcategorías de los gastos del restaurante |
| **Actores** | Administrador(es) restaurante |
| **Precondiciones** | * El actor está autenticado en el sistema * Debe haber al menos una categoría o subcategoría registrada |
| **Postcondiciones** | Caso exitoso:  • Se modifica una categoría o subcategoría de gastos |
|  | |
| **Flujo normal de eventos** | |
| Modificar una categoría o subcategoría:     1. El actor selecciona la opción MODIFICAR 2. El sistema despliega una lista de categorías y subcategorías correspondiente a la categoría padre. 3. El actor selecciona una CATEGORIA o SUBCATEGORIA. 4. El sistema despliega un formulario con los campos habilitados para modificar una categoría o subcategoría. 5. El usuario modifica los campos deseados. 6. El sistema habilita el botón de guardar cambios. 7. El actor presiona el botón de guardar cambios. 8. El sistema alerta que se ha modificado una nueva categoría o subcategoría. | |
| **Flujos alternos** | |
| **Excepciones**  Los datos proporcionados son inválidos/insuficientes:   * En el paso 5 de flujo normal el sistema mostrará un mensaje indicando que hay campos inválidos o vacíos.   No existen categorías o subcategorías creadas:   * En el paso 1 del flujo normal el sistema mostrará un mensaje indicando que no existen categorías y subcategorías creadas. | |

**Mockup**

****

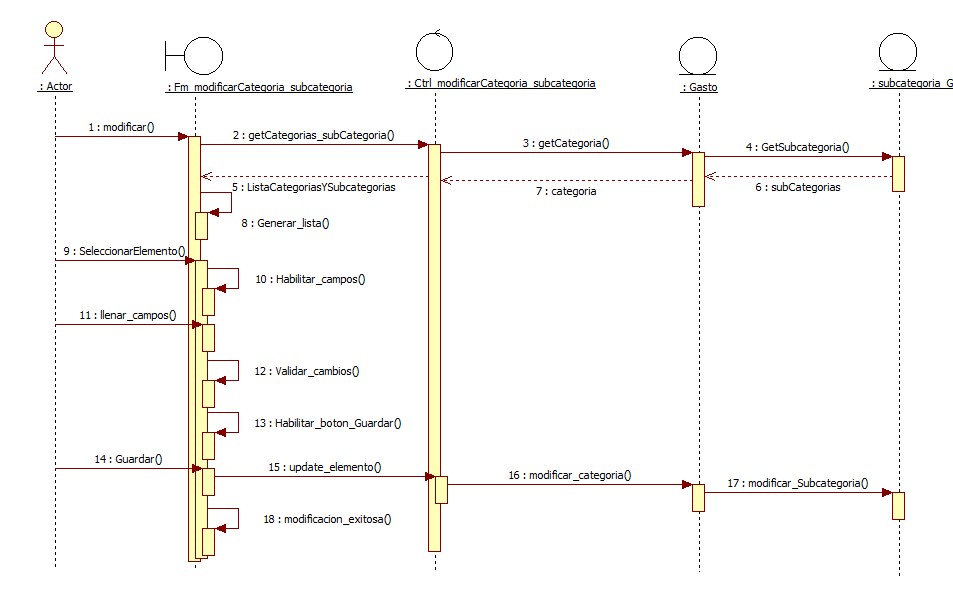
**Imagen 9.** Mockup Tabla 9. Especificación caso de uso SFCU04\_ModificarCatSubcatGasto

**Modelo de clases del análisis**

****

**Imagen 10.** Modelo de clases del análisis SFCU04\_ModificarCatSubcatGasto

**Diagrama de secuencia**

****

**Imagen 11.** Diagrama de secuencia SFCU04\_ModificarCatSubcatGasto

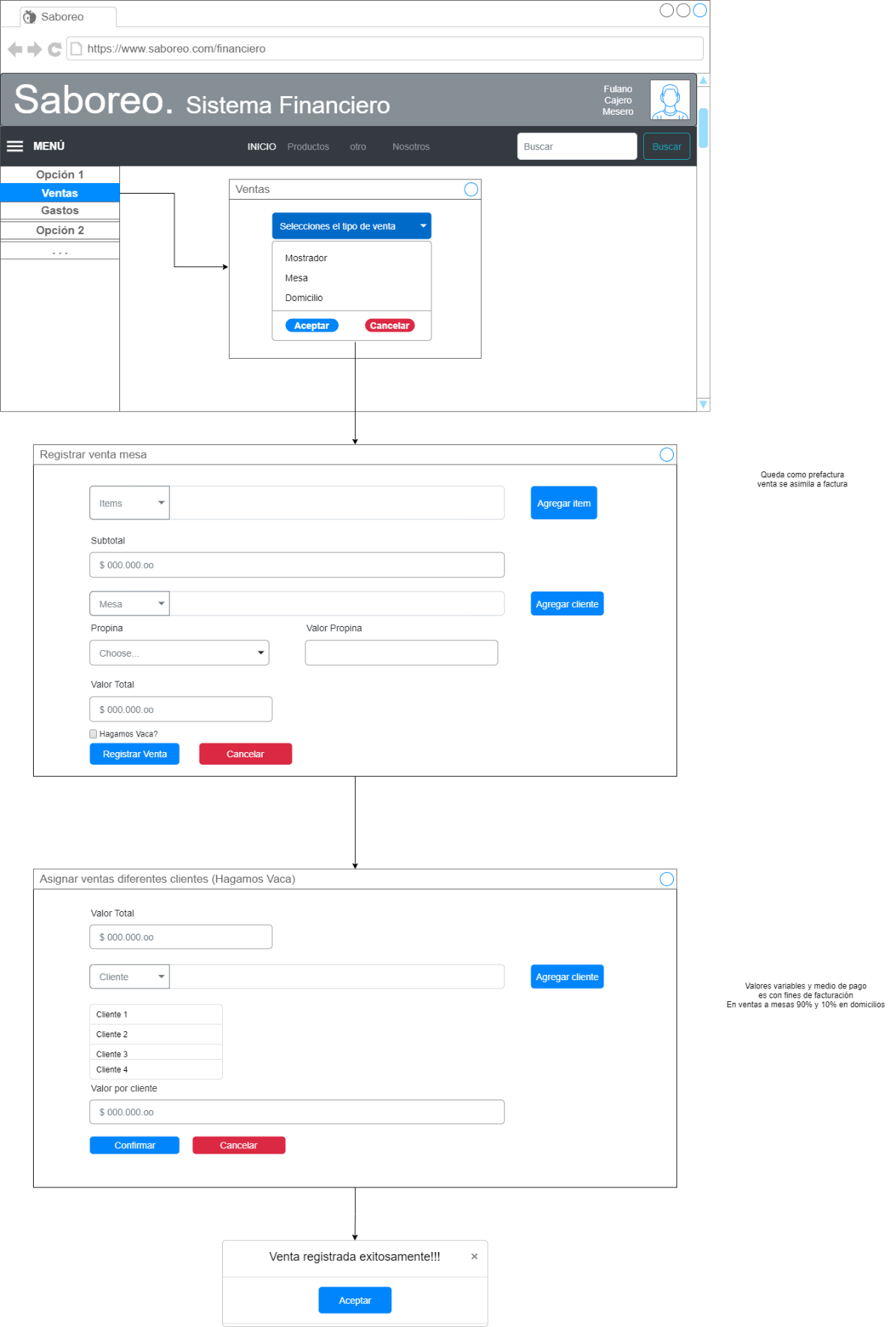
**CASO DE USO SFCU07-ASIGNAR VENTA A DIFERENTES CLIENTES**

**Especificación de caso de uso**

**Tabla 10.** Especificación de caso de uso SFCU07\_AsignarVentaDiferCliente

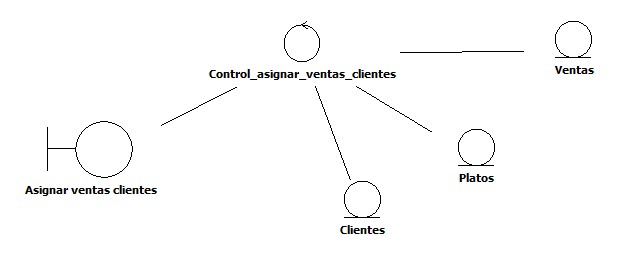
|  |  |
| --- | --- |
| **ID** | **SFCU07** |
| **Nombre** | Asignar ventas a clientes |
| **Descripción** | Permite asignar las ventas de una mesa a clientes distintos para el pago individual (Hagamos vaca) |
| **Actores** | Cajero - Mesero |
| **Precondiciones** | * Que el actor se encuentre autenticado en el sistema. * Que exista una venta registrada. * Que los clientes se encuentren registrados. |
| **Postcondiciones** | * La venta en mesa quedará dividida por clientes. |
|  | |
| **Flujo normal de eventos** | |
| 1. El actor selecciona la opción de ventas. 2. El sistema despliega una ventana emergente que contiene la información del tipo de ventas que se encuentran en el sistema. 3. El actor debe seleccionar el tipo de venta -Mesa-. 4. El sistema despliega una ventana con la información correspondiente al registro de venta en mesa. 5. El actor deberá ingresar los ítems correspondientes del pedido, seleccionándolos de una lista desplegable con buscador. 6. El actor deberá indicar si el cliente acepta pagar la propia y el valor de la misma. 7. El actor selecciona la opción “Hagamos vaca y registra la venta” 8. El sistema despliega una ventana con la información necesaria para asignar la venta en diferentes partes. 9. El actor deberá agregar los clientes de la lista desplegable con buscador. 10. El actor deberá asignar el valor por cada cliente. 11. El actor registra la venta. 12. El sistema despliega un modal de confirmación de la venta. | |
| **Flujos alternos y Excepciones** | |
| 1. El cliente no se encuentra registrado   En el paso 9 del flujo normal si el cliente no se encuentra registrado.   * 1. El sistema muestra un mensaje indicando que el cliente no se encuentra registrado.   2. El actor deberá registrar el cliente. | |

**Mockup**

****

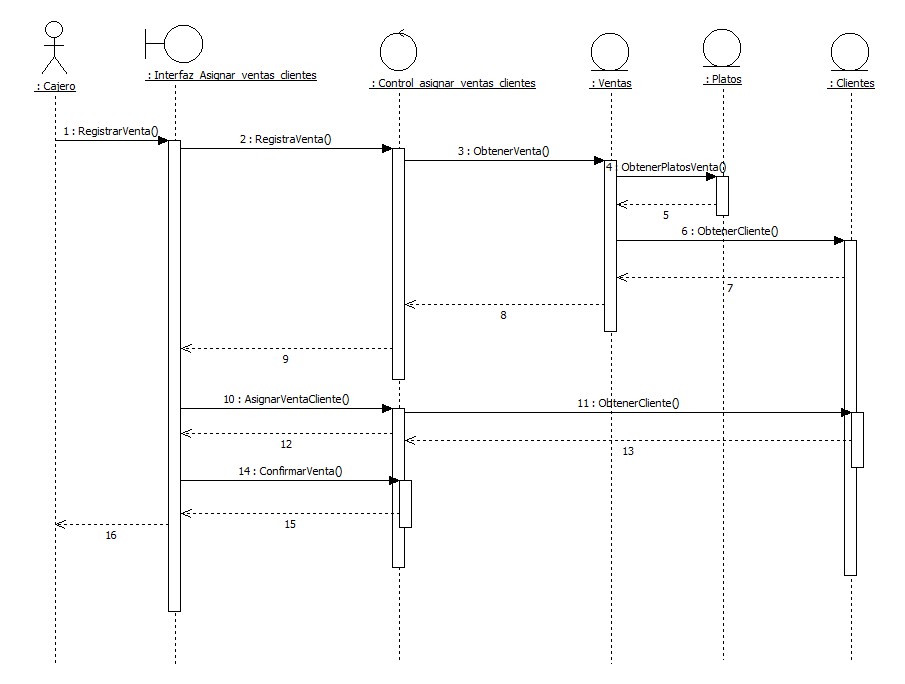
**Imagen 12.** Mockup SFCU07\_AsignarVentaDiferCliente

**Modelo de clases del análisis**

****

**Imagen 13.** Modelo de clases del análisis SFCU07\_AsignarVentaDiferCliente

**Diagrama de secuencia**

****

**Imagen 14.** Diagrama de secuencia SFCU07\_AsignarVentaDiferCliente

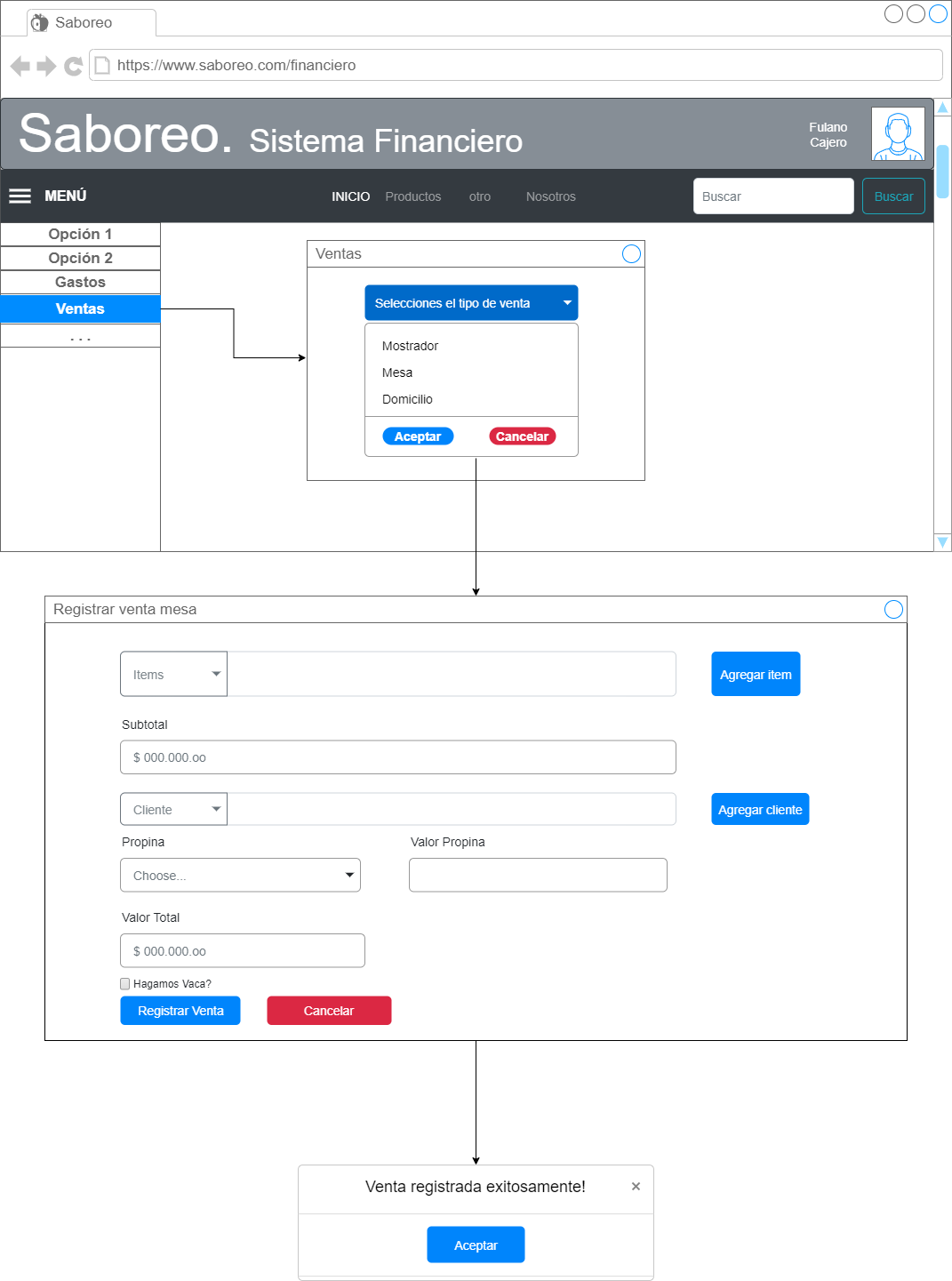
**CASO DE USO SFCU08-REGISTRAR VENTA POR MOSTRADOR**

**Especificación de caso de uso**

**Tabla 11**.Especificación caso de uso SFCU08\_AsignarVerntaMostrador

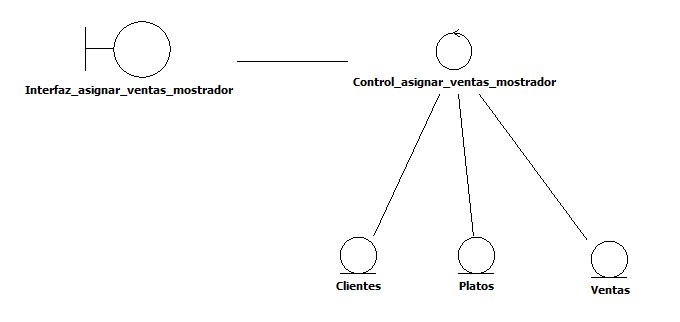
|  |  |
| --- | --- |
| **ID** | **SFCU08** |
| **Nombre** | Asignar ventas por mostrador |
| **Descripción** | Permite asignar las ventas de un cliente que no se encuentre en ninguna mesa |
| **Actores** | Cajero |
| **Precondiciones** | * Que el actor se encuentre autenticado en el sistema. * Que haya por lo menos un plato para adicionar a la venta |
| **Postcondiciones** | * Se registra la venta del cliente |
|  | |
| **Flujo normal de eventos** | |
| 1. El actor selecciona la opción de ventas. 2. El sistema despliega una ventana emergente que contiene la información del tipo de ventas que se encuentran en el sistema. 3. El actor debe seleccionar el tipo de venta -Mostrador-. 4. El sistema despliega una ventana con la información correspondiente al registro de venta en mostrador. 5. El actor deberá ingresar los ítems correspondientes del pedido, seleccionandolos de una lista desplegable con buscador. 6. El actor deberá indicar si el cliente acepta pagar la propia y el valor de la misma. 7. el actor ingresa el documento del cliente 8. El actor registra la venta. 9. El sistema despliega una ventana con la confirmación de la venta. 10. El sistema regresa al estado 1 del flujo normal de eventos | |
| **Flujos alternos y Excepciones** | |
| 1. El cliente no se encuentra registrado   En el paso 6 del flujo normal si el cliente no se encuentra registrado.   * 1. El sistema muestra un mensaje indicando que el cliente no se encuentra registrado.   2. retorna al flujo normal en el paso 8 | |

**Mockup**

****

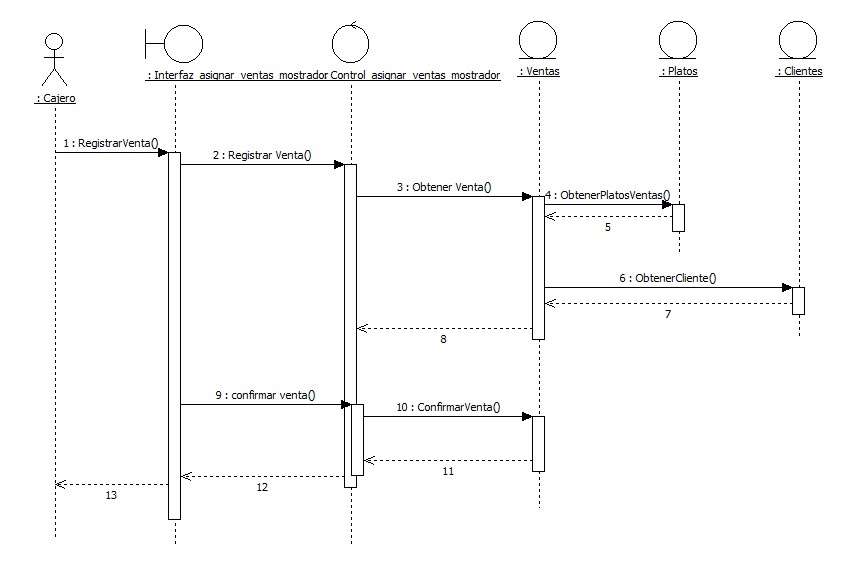
**Imagen 15.** Mockup SFCU08\_AsignarVerntaMostrador

**Modelo de clases del análisis**

****

**Imagen 16.** Modelo del análisis SFCU08\_AsignarVerntaMostrador

**Diagrama de secuencia**

****

**Imagen 17.** Diagrama de secuencia SFCU08\_AsignarVerntaMostrador

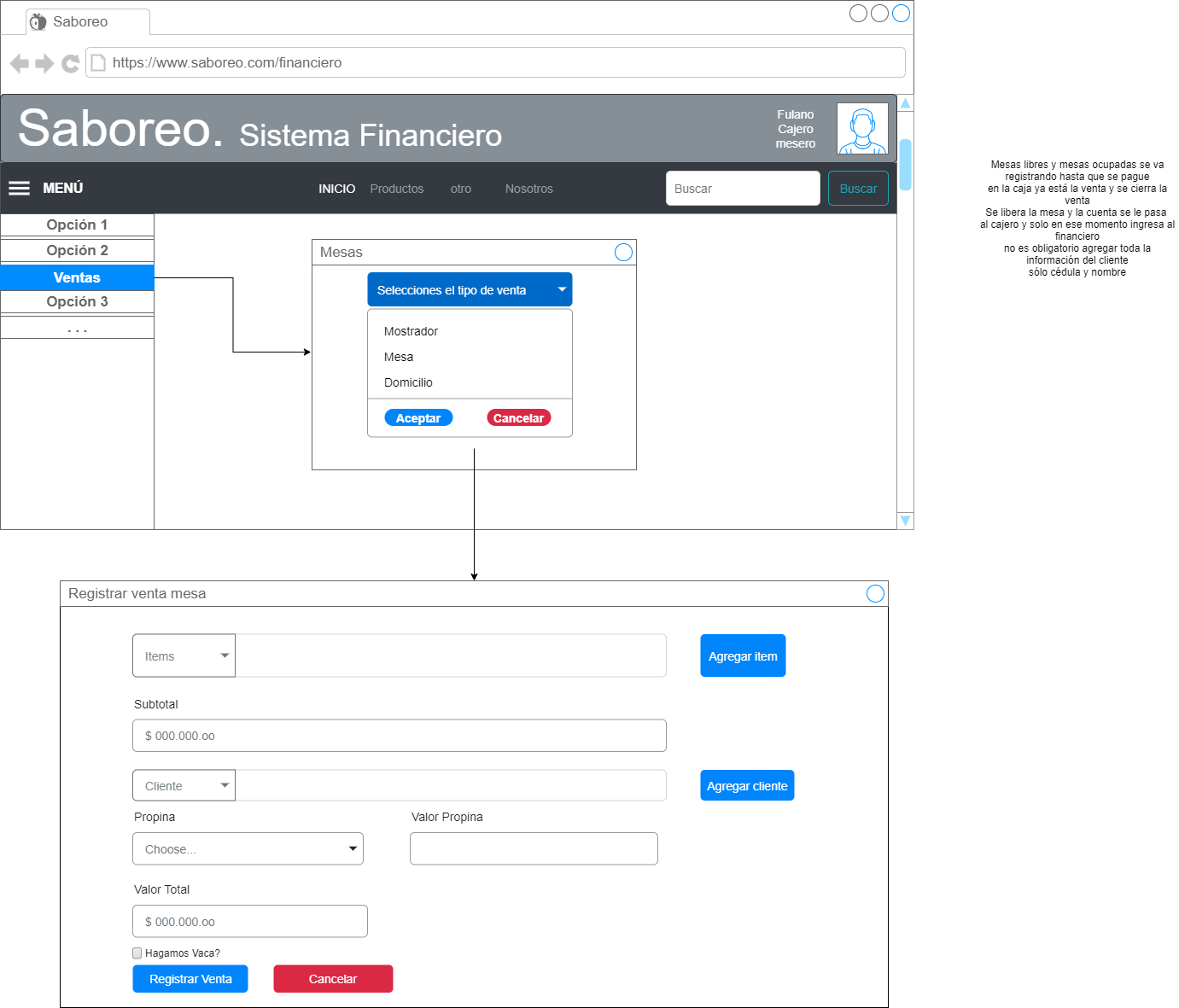
**CASO DE USO SFCU09-REGISTRAR VENTA EN MESA**

**Especificación de caso de uso**

**Tabla 12.** Especificación caso de uso SFCU09\_RegistrarVentaMesa

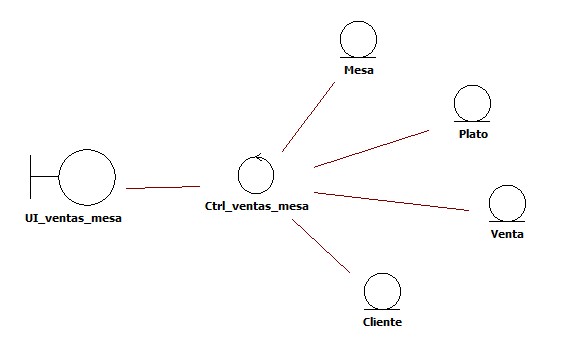
|  |  |
| --- | --- |
| **ID** | **SFCU09** |
| **Nombre** | Registrar ventas en mesa. |
| **Descripción** | Permitir registrar ventas realizadas en una mesa del restaurante. |
| **Actores** | Administrador - Mesero |
| **Precondiciones** | * Que el actor esté autenticado en el sistema * Que el actor tengo acceso a las mesas * Que existan productos en el sistema para la venta. * Que exista al menos una mesa en el sistema. |
| **Postcondiciones** | * Se crea un pedido asociado a la mesa. |
|  | |
| **Flujo normal de eventos** | |
| 1. El actor selecciona la opción pedidos. 2. El sistema abre la vista donde se muestran las mesas existentes en el sistema. 3. El actor selecciona la mesa a la que se le registrará la venta. 4. El sistema despliega una modal donde se mostrarán los productos disponibles para la venta con su respectivo precio. 5. Se habilita habilita un formulario para agregar información de usuario (opcional). 6. El actor selecciona los productos y la cantidad de cada uno que desea pedir el usuario. 7. sistema valida los datos del pedido y del usuario. 8. El sistema arroja un mensaje de creación de pedido exitosa. | |
| **Flujos Alternos y Excepciones** | |
| 1. Pedido existente   En el paso 4 del flujo normal si ya existe un pedido asociado a la mesa que no se ha cerrado   1. Se muestran los productos ya añadidos al pedido 2. Se muestra la información del usuario ya registrada 3. Se regresa al paso 5 del flujo normal 4. Error de venta   En el paso 7 del flujo normal, en caso de que haya un error con los productos del pedido seleccionados   * 1. Se arroja un mensaje indicando al usuario el error y que intente nuevamente   2. Se regresa al paso 4 del flujo normal  1. Error de datos de usuario   En el paso 7 del flujo normal, en caso de que hay un dato de usuario erróneo   * 1. Se arroja un mensaje indicando el error   2. Se regresa al paso 5 del flujo normal. | |

**Mockup**

****

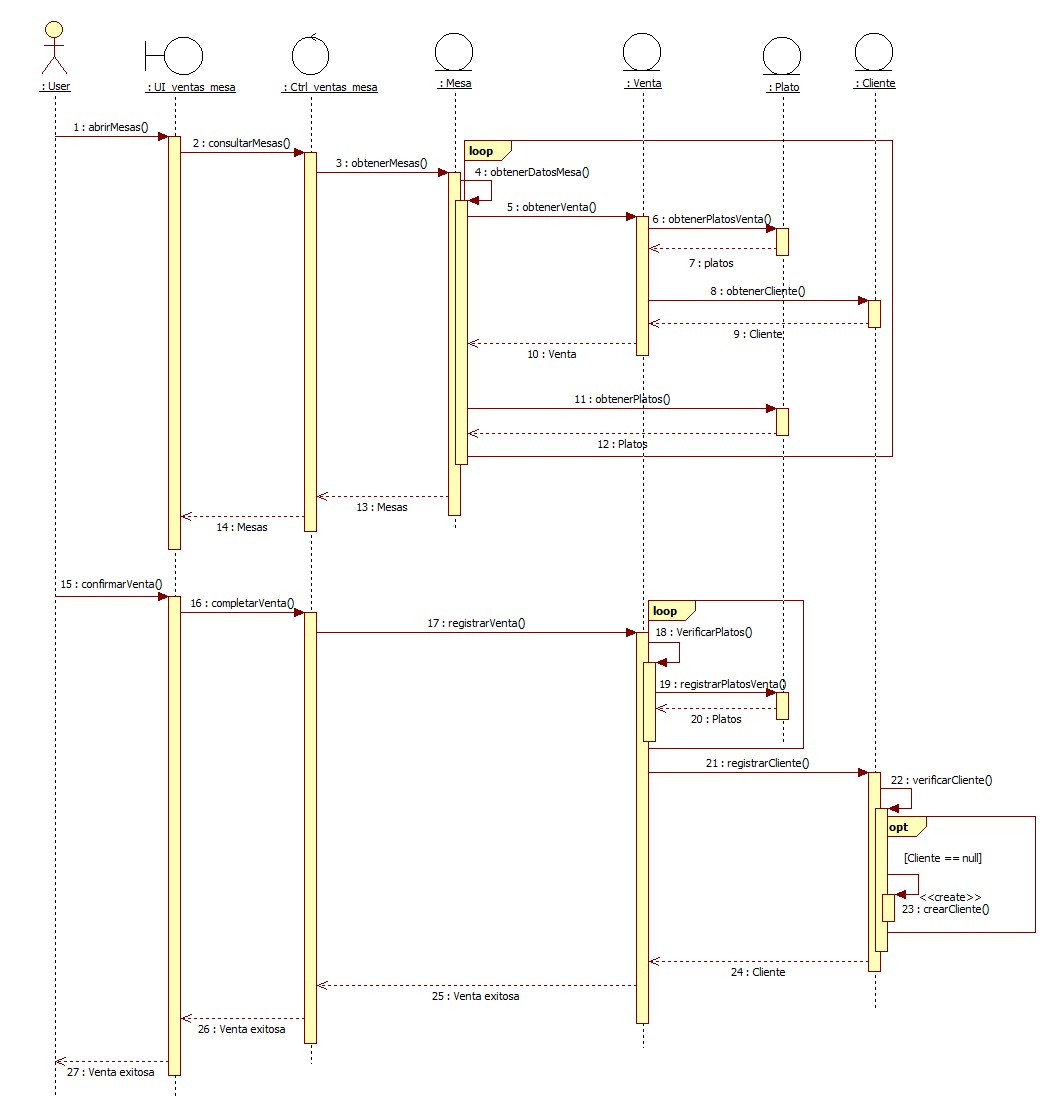
**Imagen 18.** Mockup SFCU09\_RegistrarVentaMesa

**Modelo de clases del análisis**

****

**Imagen 19.** Modelo de clases de análisis SFCU09\_RegistrarVentaMesa

**Diagrama de secuencia**

****

**Imagen 20.** Diagrama de secuencia SFCU09\_RegistrarVentaMesa