



# DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

# Recuperatorio Trabajos Prácticos 1 y 2

Análisis preliminar del sistema de software para la cadena de supermercados Mes%

1<sup>er</sup> cuatrimestre de 2015

# Ingeniería De Software I

Tutor: *Ezequiel Castellano*

## Grupo 10

Integrante	LU	Correo electrónico
Barrios, Leandro Ezequiel	404/11	ezequiel.barrios@gmail.com
Benegas, Gonzalo Segundo	958/12	gsbenegas@gmail.com
Di Alessio, Adrian Alejandro	631/06	adrianalejandro86@hotmail.com
Rodriguez, Pedro	197/12	pedro3110.jim@gmail.com
Vanecek, Juan	169/10	juann.vanecek@gmail.com

**Facultad de Ciencias Exactas y Naturales**  
Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

<http://www.fcen.uba.ar>

Esta página fue dejada en blanco intencionalmente.

# Contents

<b>1</b>	<b>Introducción TP1</b>	<b>2</b>
1.1	Contexto . . . . .	2
<b>2</b>	<b>Presunciones / Aclaraciones adicionales</b>	<b>3</b>
<b>3</b>	<b>Vistas</b>	<b>3</b>
3.1	Diagrama de Contexto . . . . .	3
3.1.1	Agentes . . . . .	3
3.1.2	Listado de Fenómenos . . . . .	5
3.1.3	Diagrama de Contexto . . . . .	8
3.2	Modelo de Objetivos . . . . .	9
3.2.1	Listado de objetivos . . . . .	9
3.2.2	Diagramas de Objetivos . . . . .	13
3.3	Escenarios representativos de uso . . . . .	14
3.3.1	A: Registro online . . . . .	14
3.3.2	B: Registro presencial con sistema de reputación . . . . .	14
3.3.3	C: Pago online . . . . .	14
3.3.4	D: Pago contrareembolso . . . . .	15
3.3.5	E: Entrega correcta . . . . .	15
3.3.6	F: Ausente durante entrega con límite de entregas fallidas fijo . . . . .	16
3.3.7	G: Cancelación de pedido . . . . .	16
3.3.8	H: Modificación de pedido. . . . .	16
3.4	I: Preparar y cerrar pedido . . . . .	17
3.5	J: Alarma y reposición por bajo stock en depósitos . . . . .	17

<b>4</b>	<b>Discusión</b>	<b>17</b>
4.1	O-Refinamiento: Registro del cliente . . . . .	23
4.2	O-Refinamiento: Ranking del cliente . . . . .	25
<b>5</b>	<b>Conclusiones TP1</b>	<b>28</b>
<b>6</b>	<b>Introducción TP2</b>	<b>30</b>
<b>7</b>	<b>Casos de Uso</b>	<b>31</b>
<b>8</b>	<b>Modelo conceptual</b>	<b>45</b>
8.1	OCL . . . . .	46
<b>9</b>	<b>Máquinas de estados finitos</b>	<b>49</b>
<b>10</b>	<b>Diagrama de actividad</b>	<b>51</b>
<b>11</b>	<b>Conclusión TP2</b>	<b>52</b>

## Parte I. Introducción TP1

Mediante el presente trabajo, se nos pone frente a la tarea de realizar un **análisis de requerimientos** y una **propuesta de un sistema de software** para la cadena de supermercados MES%, a pedido del **CEO**<sup>1</sup> de la compañía. La propuesta debe ser planteada de tal forma que esta transmita claramente la manera en que será cumplido cierto conjunto de objetivos, junto con los mecanismos mediante los cuales el software se relacionará con los sistemas ya existentes, y su beneficio directo y potencial.

A continuación, se incluye un resumen coloquial de los requerimientos, que describe el contexto dentro del cual se solicitó la implementación del sistema.

### 1.1 Contexto

La cadena MES% ha venido creciendo rápidamente desde sus comienzos, debido a sus precios. Pero este crecimiento se ha detenido en los últimos seis meses. Se han identificado dos razones por las cuales se presume que las ventas han disminuído:

1. «descontento de los clientes debido al excesivo tiempo de espera en las colas de las cajas».
2. «ventas no concretadas por falta de mercadería».

Con el objetivo de eliminar esos problemas, el **CEO** propuso los siguientes puntos, que se utilizarán de base para el desarrollo de nuestra propuesta:

- las compras deben realizarse desde la web **www.mespociento.com**
- los usuarios deben estar registrados
- se registra un pedido, acordando con el sistema una fecha de entrega
- se cuenta con camionetas para las entregas.
- se planea incrementar la cantidad de depósitos, pero no la de locales
- en un pedido online no se puede tener problemas de stock
- desde cualquier local se debe poder solicitar la reposición de stock
- considerar qué sucede cuando el envío no logra ser entregado
- se debe poder modificar un pedido pendiente
- no se puede realizar un pedido nuevo, si existen pedidos pendientes
- el pago se puede hacer online o contrareembolso
- se debe almacenar información para futuras estadísticas
- se espera que el sistema escale y sea seguro

<sup>1</sup>El tutor de este trabajo práctico, en este caso, cumple este rol.

## Parte II. Presunciones / Aclaraciones adicionales

Se incluyen a continuación aquellas presunciones que se utilizaron como complemento a las indicaciones originales. Estas pueden ser, o bien consultadas al **CEO** en caso de considerarlas lo suficientemente relevantes, o asumidas/interpretadas por cuenta propia.

- En caso de que la cadena de supermercados decida inaugurar más **depósitos**, tanto la reposición de stock como los trasladados interdepósito estarán a cargo de la compañía, sin ningún tipo de intervención por parte del **sistema**. A efectos prácticos, entonces, se asumirá que existe un solo **depósito**, cuyo stock es la sumatoria del stock de todos los **depósitos**, y que todos los pedidos serán entregados desde el mismo.
- Una empresa de **logística** se encargará de todos los trasladados que se originen en cualquier **depósito**. El **sistema** deberá contar con una interfaz adecuada que actúe como canal de comunicación bilateral con dicha empresa. Este canal deberá ser apto tanto para realizar consultas y pedidos de forma online, a través de solicitudes directas al servidor de la compañía de **logística**, como para informar o ingresar datos de forma indirecta, a través de la impresión o la carga manual de: remitos de traslado, comprobantes de pago, facturas, hojas de ruta, etcétera.
- El **sistema** deberá llevar, en todo momento, un control activo (en tiempo real, en línea) del stock del **depósito**.
- Existe un **Departamento de Stock**, que se encarga de organizar los pedidos a las proveedoras cuando un **depósito** se queda sin stock. Todo esto se hace por fuera del sistema, las proveedoras no son Agentes de nuestro modelo de contexto. Asumimos que el **Departamento de Stock** se comunica con **logística** (la logística propia o la de la proveedora, cualquiera sea el caso) en cuanto para que lleve los pedidos hasta el **depósito** correspondiente.
- Cada pedido se cierra al momento en que el **depósito** informa que el mismo fue armado. Consecuentemente, se asume que estos no necesariamente permanecerán abiertos hasta el momento en que son retirados por **logística**.

## Parte III. Vistas

### 3.1 Diagrama de Contexto

#### 3.1.1 Agentes

1. Depósito
2. Cliente
3. Sucursal
4. Sistema
5. Logística

6. API de Logística
7. Departamento de Stock
8. Agente de Cobro
9. Administrador
10. Sistema de Correo Electrónico
11. Financiera(API)<sup>2</sup>
12. Correo Argentino(API)
13. Proveedor

---

<sup>2</sup>Ejemplo: VERAZ

### 3.1.2 Listado de Fenómenos

1. Configuración y uso por parte del Administrador
  - (a) El Administrador define el umbral de redituabilidad en el Sistema
  - (b) El Administrador consulta estadísticas en el Sistema
  - (c) El Administrador modifica producto a través de ABM
2. Registro presencial cliente
  - (a) El Cliente presenta la documentación en la Sucursal
  - (b) El Cliente presenta la factura asociada al domicilio en la Sucursal
  - (c) El Cliente presenta datos de tarjeta en la Sucursal
  - (d) La Sucursal presenta datos de tarjeta del cliente al Agente de Cobro
  - (e) La Sucursal carga datos del cliente en el Sistema
3. Registro online cliente
  - (a) El Cliente presenta datos de identificación personal y de domicilio al Sistema
  - (b) El Cliente presenta datos de pago al Sistema
  - (c) El Sistema valida domicilio con el Correo Argentino
  - (d) El Sistema evalúa estado financiero del cliente con la Financiera.
  - (e) El Sistema le indica al Correo Electrónico que envíe mail de confirmación al Cliente.

**FIX ME:** poner que sist. pide validacion de datos de pago a agente de cobro, y la vuelta

4. Armado carrito
  - (a) El Cliente ingresa credenciales en el Sistema
  - (b) El Sistema muestra stock disponible al Cliente
  - (c) El Sistema muestra recomendaciones al Cliente
  - (d) El Cliente selecciona mercadería en el Sistema
5. Confirmación compra
  - (a) Cliente confirma carrito al sistema
  - (b) Sistema ratifica al cliente stock disponible
  - (c) Sistema muestra mensaje de error de producto no disponible al cliente
  - (d) El sistema obtiene el costo del envío de la API de Logística
  - (e) El sistema informa el costo de la compra y el envío al cliente
  - (f) El sistema pregunta próximas fechas libres a la API de logística
  - (g) El sistema presenta las posibles fechas de entrega al cliente
  - (h) El cliente indica la fecha de entrega deseada al sistema

- (i) El sistema presenta los métodos de pago disponibles al cliente
- (j) El cliente elige el método de pago
- (k) El sistema muestra datos de pago asociados al cliente
- (l) El cliente indica dato de pago
- (m) El sistema redirige al cliente a la ventana del Agente de Cobro
- (n) El cliente autoriza la operación en la ventana del Agente de Cobro
- (o) El agente de cobro envía estado y comprobante al sistema
- (p) El sistema pide mail de comprobante de pago al Correo Electrónico
- (q) El sistema informa de error de pago al cliente
- (r) El sistema muestra error de timeout de carrito al cliente
- (s) El Sistema ofrece pago de multa al cliente
- (t) El cliente paga multa online (mediante agente de cobro)
- (u) El cliente paga multa en efectivo a logística

#### 6. Modificación de pedido

- (a) El Cliente modifica un pedido no cerrado en el Sistema
- (b) El Sistema le da un crédito al Cliente si este quita mercadería paga

#### 7. Preparación y entrega

- (a) El Cliente confirma el pedido al Sistema
- (b) El Sistema avisa al Depósito que prepare el pedido
- (c) El Depósito informa egreso de mercadería al Departamento de stock
- (d) El Depósito avisa al Sistema que el pedido está cerrado
- (e) El Depósito entrega mercadería a Logística
- (f) Si forma de pago es contraentrega:
  - i. Cliente paga el pedido a Logística
- (g) Logística entrega el pedido al Cliente

#### 8. Entrega fallida

- (a) El Cliente no recibe pedido a Logística
- (b) Si el pedido fue pagado:
  - i. El Sistema le envía una nota de crédito como reintegro del pedido al Agente de Cobro.
  - ii. Agente de Cobro genera el comprobante de reintegro y se la envía al Sistema.
  - iii. El Sistema envía el comprobante al Cliente.
- (c) Logística devuelve el stock al Depósito
- (d) El Sistema arma un e-mail con el link al pedido fallido y la propuesta para que el cliente lo vuelva a hacer, y se lo envía al Sistema de Correo
- (e) El Sistema de Correo electrónico envía mail al Cliente.

#### 9. Reposición stock sucursal

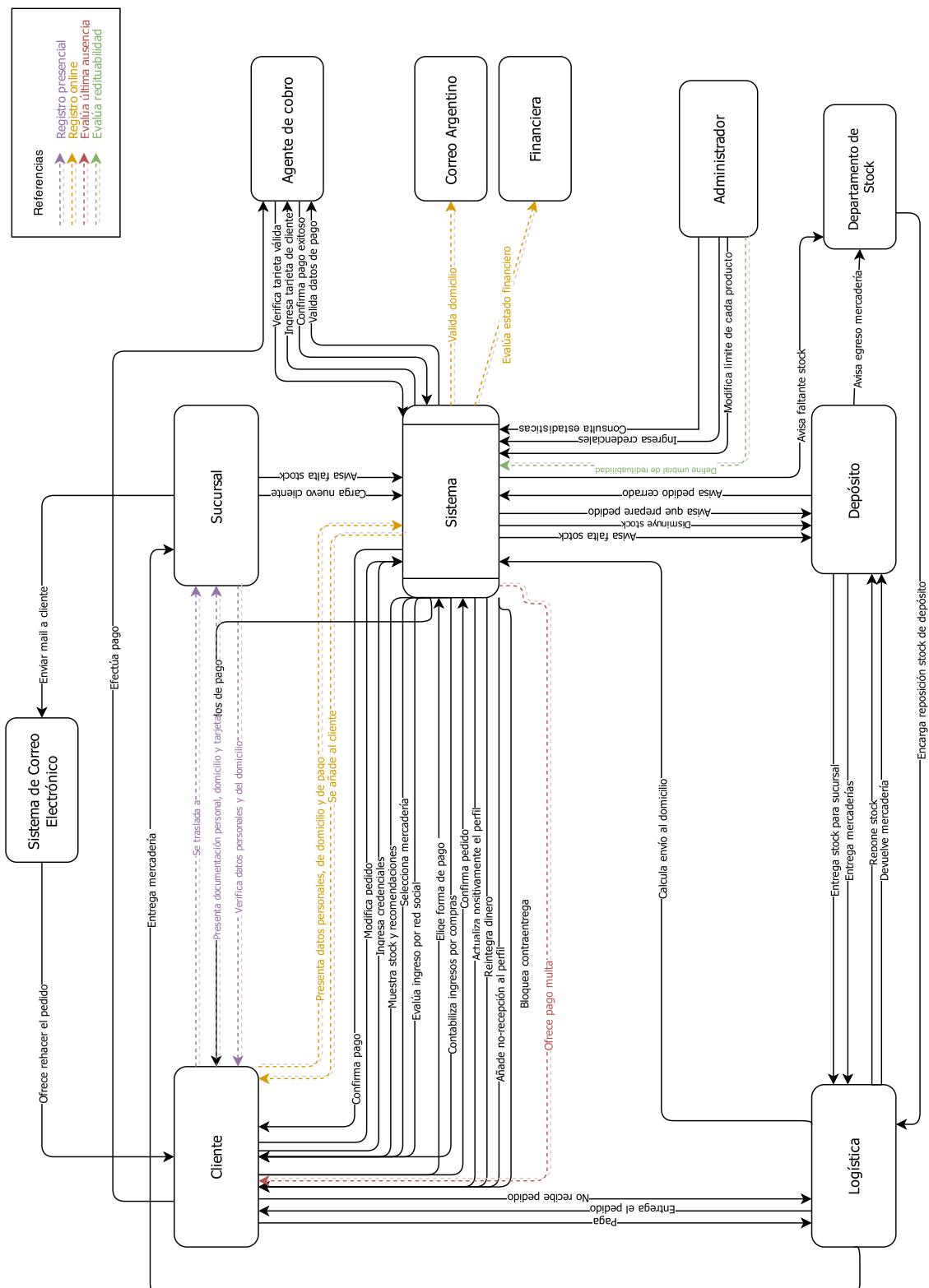
- (a) La Sucursal avisa cuando falta stock al Sistema
- (b) El Sistema avisa falta de stock en sucursal al Depósito

- (c) El Depósito arma pedido y entrega el stock a Logística
- (d) El Sistema disminuye stock en el Depósito
- (e) Logística entrega stock a la Sucursal

10. Reposición stock deposito

- (a) El Departamento de stock modifica los límite mínimos de stock estipulado para cada producto en el Sistema
- (b) El Sistema avisa que hay faltante de producto al Departamento de stock
- (c) El Departamento de stock encarga reposición de stock del depósito a Logística
- (d) Logística repone stock al Depósito

### 3.1.3 Diagrama de Contexto



## 3.2 Modelo de Objetivos

### 3.2.1 Listado de objetivos

#### 1 - Objetivo primario

El objetivo primario del sistema es revertir la disminución de ventas (1) que se viene produciendo en la cadena de supermercados hace meses. Para ello, se nos brinda la información de que las dos causas de las disminuciones son las largas colas en el supermercado y el agotamiento del stock (1.3), de donde se desprenden los objetivos (1.1) y (1.2).

#### 1.1 - Reducción de colas

Dentro del contexto de la reducción de colas, se tomó la presunción de que cada venta online estaría disminuyendo las colas en la sucursal (1.1.2), dado que de este modo ese cliente, que normalmente estaría haciendo cola en el supermercado, puede encargar los productos desde la comodidad de su casa. De este modo, se toma el objetivo de permitir que el usuario compre de forma online (1.1.1).

##### 1.1.1 - Permitir al usuario la compra online

Consideramos que para permitir que el cliente compre un pedido de forma online, deben ocurrir dos cosas, lograr que el cliente encargue el pedido (1.1.1.1)<sup>3</sup>, y lograr que el pedido sea cerrado y finalizado (1.1.1.2), que comienza con la confirmación del pedido por parte del cliente, y culmina con la entrega del pedido (o la cancelación/anulación, según el caso).

---

<sup>3</sup>lo cual incluye todo el proceso desde que el cliente se registra en el sitio, realizando la selección de productos, acordando una fecha de entrega y eligiendo una forma de pago

### 1.1.1.1 - Lograr[Cliente encarga pedido]

Para que un cliente encargue un pedido, es necesario que el mismo pueda identificarse en el sistema (1.1.1.1.1), que el pedido sea armado (1.1.1.1.2), que se pueda acordar una fecha de entrega (1.1.1.1.3), una forma de pago (1.1.1.1.4) y que si esta última es online, se le pueda cobrar.

#### 1.1.1.1.1 - Lograr[Cliente identificado]

Para que el cliente pueda ser identificado por el sistema, es necesario que se encuentre registrado (1.1.1.1.1.1), y que se autentique (1.1.1.1.1.2)

#### 1.1.1.1.1.1 - Lograr[Cliente registrado]

La forma en que un cliente se registra es uno de los o-refinamientos propuestos. Dividimos el registro entre online y presencial. El registro online implica que el cliente se conecte al sitio, cargando sus datos personales, y sus datos de pago, los cuales son verificados por el sistema al momento de la registración. El registro presencial, en cambio, requiere que el cliente se dirija a la sucursal con la documentación necesaria para verificar identidad, domicilio y datos de pago, los cuales son archivados y verificados por un empleado de la sucursal, y posteriormente cargados al sistema. Más pormenores sobre este refinamiento, junto con la justificación del puntaje asignado a los objetivos blandos, están explicados posteriormente.

#### 1.1.1.1.1.2 - Lograr[Cliente se autentica de forma segura]

La autenticación requiere que el cliente ingrese sus credenciales (1.1.1.1.1.2.1). Por otro lado, también es necesario proveer una interfaz que cuente con los mecanismos de seguridad adecuados (1.1.1.1.1.2.2), por ejemplo conexión HTTPS.

#### 1.1.1.1.2 - Lograr[Armado de pedido]

Para que el pedido sea armado, consideramos que es necesario que el cliente pueda ver el stock disponible para cada producto (1.1.1.1.2.1), y que logre seleccionar la mercadería adecuada (1.1.1.1.2.3). Para ello, planteamos el objetivo de mostrar recomendaciones personalizadas para cada cliente (1.1.1.1.2.2), que varían según los distintos datos relacionados a su usuario, como pueden ser, perfil de comprador, generado a través del historial de compras, compras de otros clientes, relacionados según escala socioeconómica, edad, domicilio, etc.

### 1.1.1.1.3 - Lograr[Acordar fecha de entrega]

En el transcurso del armado del pedido, es necesario que el cliente acuerde una fecha de entrega. Para ello, es necesario que el sistema le ofrezca un rango de fechas disponibles (1.1.1.3.2), las cuales serán calculadas e informadas por logística (1.1.1.3.1), y obtenidas en tiempo real, de algun modo, por ejemplo a través de algún tipo de interfaz entre nuestro sistema y el sistema de logística. En caso de no haberla, será necesario contar con una forma de calcularla. Luego de mostrarles las fechas, el cliente elegirá (1.1.1.3.3) la que más le convenga.

### 1.1.1.1.4 - Lograr[Acordar forma de pago]

Además de que el cliente seleccione la fecha de entrega, es necesario que acuerde con el sistema una forma de pago. Para ello, el sistema deberá informar al cliente cuales son las formas de pago que tiene disponibles (1.1.1.4.2), y el cliente deberá elegir entonces la que le resulte de su conveniencia (1.1.1.4.3). Hay un o-refinamiento aquí, sobre el objetivo de evaluar si se deberá permitir la contraentrega.

#### 1.1.1.1.4.1 - Lograr[Evaluar si permitir contraentrega]

Para evaluar si el cliente puede realizar un pedido por contrareembolso, proponemos dos opciones.

La primera de ellas es un límite de entregas fallidas (1.1.1.4.1.1), contenido dentro de la programación del sistema, e instanciado en 1 en nuestros ejemplos, es decir, se limita la posibilidad de que el cliente no esté presente en la última entrega. Llegado tal caso, se le bloqueará la modalidad contraentrega (1.1.1.4.1.1.1) y se ofrecerá el pago de una multa (1.1.1.4.1.1.2) para poder desbloquearla (1.1.1.4.1.1.3). El valor de la multa, además, no es fijo, sino que dependerá directamente del costo del envío al domicilio de ese cliente en particular (1.1.1.4.1.1.2.1.1), y el costo de la mercadería que no pudo ser reingresada a stock (1.1.1.4.1.1.2.1.2). Se le ofrecerán dos modalidades de pago, pagar en efectivo al momento de recibir el envío de un pedido distinto (1.1.1.4.1.1.2.2.1) o pagar online (1.1.1.4.1.1.2.2.2).

La segunda opción es un límite de pérdida de redituabilidad (1.1.1.4.1.2). Para ello, el Administrador deberá definir previamente un umbral de pérdida de redituabilidad (1.1.1.4.1.2.1), el cual el sistema deberá segurarse de que sea respetado en todo momento (1.1.1.4.1.2.2). En caso de que el cliente produzca costos de tal forma que la redituabilidad se encuentre por bajo del umbral, la contraentrega deberá bloquearse (1.1.1.4.1.2.2.3). Para ello, es necesario contabilizar las ganancias (1.1.1.4.1.2.2.1) y costos (1.1.1.4.1.2.2.2) generados por el cliente. La redituabilidad, entonces, es una relación ganancia/costo, la cual no deberá bajar más allá del umbral definido previamente.

Ambas opciones están detalladas más adelante, junto con la asignación de puntajes a los objetivos blandos.

#### 1.1.1.1.5 - Lograr[Si la forma de pago es online, cobrar]

Si el cliente elige el pago online, además, hay que cobrarle. Para ello, se cuenta con la participación del cliente, el cual deberá validarse a través del agente de cobro, para que el importe le sea deducido de su cuenta bancaria.

### 1.1.1.2 - Lograr[El pedido es finalizado y cerrado]

Una vez que un pedido está encargado, es necesario que el mismo sea cerrado. Aquí pueden ocurrir dos cosas, que el usuario cancele el pedido antes de que el mismo sea preparado (1.1.1.2.2), caso en el que el mismo se anula y se le devuelve el dinero, o que el usuario confirme el pedido (1.1.1.2.1). En este último caso, será necesario preparar el pedido (1.1.1.2.1.1) a la vez que el stock es restado del conteo del sistema (1.1.1.2.1.2), y luego lograr que el mismo sea entregado a logística (1.1.1.2.1.3). Además de todo esto, será necesario asegurar que el envío sea realizado (1.1.1.2.1.4).

#### 1.1.1.2.1.4 - Lograr[Realizar el envío]

Al momento de realizar el envío, pueden ocurrir dos situaciones, o que el cliente reciba el pedido (1.1.1.2.1.4.1), o que no lo reciba (1.1.1.2.1.4.2). Por simplicidad, se englobaron todas las situaciones en las que el cliente puede no recibir el pedido, en una situación “el cliente no está presente”. Es decir, también es este el caso cuando un cliente está presente pero por alguna razón se niega a recibir el pedido.

#### 1.1.1.2.1.4.2 - Lograr[Realizar el envío]

En caso de que el cliente no reciba el pedido, será necesario deshacer la operación, es decir, reintegrar el dinero de ser necesario (1.1.1.2.1.4.2.1), marcar el pedido como no recibido en el historial del cliente (1.1.1.2.1.4.2.2), y que logística devuelva el stock al depósito para que sea reingresado (1.1.1.2.1.4.2.3). Esto significa que Depósito realizará la evaluación de los productos (ya que algunos pueden haberse roto o perdido su calidad en el transcurso del envío), y reingresará a stock los que considera que se encuentran adecuados. Además, se le informará al cliente que su pedido no fue recibido mediante un email, y se le ofrecerá rehacerlo (1.1.1.2.1.4.2.4).

## 1.2 - Mantener[Stock disponible]

Para mantener el stock disponible, se deberá mantener el stock disponible tanto en las sucursales (1.2.1) como en los depósitos (1.2.2)

### 1.2.1 - Mantener[Stock disponible en sucursales]

Para mantener el stock disponible en las sucursales es necesario lograr realizar un conteo periódico del stock en las sucursales (1.2.1.1), y encargar una reposición por medio del sistema en cuanto el stock sea bajo (1.2.1.2). Por otro lado, será necesario lograr que el depósito sea informado en cuanto haya un pedido nuevo (1.2.1.3). Respecto a esto, la idea es que el depósito mantenga un sistema interno propio que cuente con acceso a la misma base de datos que el sistema web. También es necesario armar el pedido, y entregarlo a logística (1.2.1.4), de esto se encargaría el depósito. Finalmente, logística debe entregar la reposición de stock a la sucursal (1.2.1.5).

### 1.2.2 - Mantener[Stock disponible en depósitos]

Para poder mantener el stock en el depósito, es necesario contar con un límite mínimo de productos, de forma tal de asegurarse de que cada producto se encuentre siempre por encima de este límite (1.2.2.2). Como cada producto es distinto del otro, y tienen distintos ritmos de ventas, este límite no puede ser igual para todos los productos, ni puede estar contenido dentro del código del programa. Para determinar este límite, entonces, será necesario contar con una forma de modificar los productos y sus límites, según las estadísticas de ventas de cada uno (1.2.2.1).

#### 1.2.2.1 - Lograr[Modificar productos y sus límites según estadísticas]

Por un lado se ofrece la capacidad típica de agregar nuevos productos, modificar los existentes, o dar de baja otros (ABM, 1.2.2.1.1). Utilizando esta interfaz, el administrador deberá modificar los productos, estipulando el límite inferior de stock para cada uno de ellos (1.2.2.1.3). Para ayudarlo, el sistema le ofrece estadísticas elaboradas sobre el registro de cada transacción de los clientes. Se puede visualizar la información de manera interactiva y se pueden aplicar herramientas avanzadas de *machine learning* sobre los datos (1.2.2.1.2).

#### 1.2.2.2 - Mantener [Stock de cada producto por encima de límite estipulado por empresa]

El sistema monitorea constantemente el stock de cada producto, y cuando uno de estos desciende por debajo del límite estipulado, se inicia un proceso de reabastecimiento. Como el producto no está en suficiente cantidad en nuestros depósitos, es necesario recurrir al Departamento de Stock (1.2.2.2.1) que se encarga de conseguir el producto a través de los distintos proveedores, los cuales serán encomendados para su entrega a la logística (1.2.2.2.2). Luego se inicia un proceso logístico que culmina con el producto en el depósito (1.2.2.2.3), y el nuevo stock ingresado al sistema (1.2.2.2.4).

## 3.2.2 Diagramas de Objetivos

### 3.3 Escenarios representativos de uso

#### 3.3.1 A: Registro online

**Alice** no es cliente habitual de la cadena mesporciento, ya que le queda lejos de su casa como para ir caminando. Tras enterarse de la nueva posibilidad de comprar online, se conecta al sitio web [www.mesporciento.com](http://www.mesporciento.com) a través de su computadora, y registra un nuevo usuario “`alice_gatita93`”, ingresando para ello sus **datos personales**, tales como nombre, domicilio, teléfono, email, y una **contraseña segura**. El sitio verifica que los datos ingresados son correctos, y luego le envía un mail de confirmación con un vínculo en el que **Alice** presiona para validar su cuenta. **Alice** entonces reingresa al sitio, utilizando ahora su nuevo usuario.

#### 3.3.2 B: Registro presencial con sistema de reputación

**Bob** es cliente habitual de la cadena mesporciento, y aprovecha una de sus rutinarias compras para registrar su usuario en su sitio web. Para ello, se encargó previamente de juntar la documentación requerida para el registro: el documento de identidad, y una acreditación de domicilio, en particular lleva la última factura de luz. Luego de las compras, le pregunta a la cajera en dónde debe registrarse, y le indican que se dirija a la sección de informes.

En la sección de informes no hay nadie, por lo que **Bob** debe esperar unos 10 minutos hasta que aparezca la encargada. Esta le pide la documentación, y la verifica mientras le entrega a **Bob** un cuestionario de datos personales para que lo complete. Luego, **Bob** le entrega el cuestionario, y la empleada verifica que los datos del cuestionario y la documentación coincidan. Entonces, le saca fotocopias a la documentación y al cuestionario. Le entrega la fotocopia del cuestionario a **Bob**, mientras que la fotocopia de la documentación es archivada junto con el cuestionario en un sobre de papel madera, que a su vez es apilado junto a muchos otros sobres de aspecto similar. Finalmente, le informa a **Bob** que se le avisará por mail en cuanto el registro se encuentre completo, y allí mismo le brindarán las instrucciones para acceder al sitio.

Luego de dos semanas, **Bob** recibe un email de parte del remitente [felicitaciones@mesporciento.com](mailto:felicitaciones@mesporciento.com), y asunto «Bienvenido a una nueva forma de comprar», en el que se le informa que ya se encuentra habilitado su usuario “`marley_b420`”

#### 3.3.3 C: Pago online

**Charlie** se conecta a la web de mesporciento desde su tablet, con su usuario “`je_suis_moi`”, dispuesto a iniciar su compra de productos semanal, de forma online. Para ello, revisa el listado de productos, y agrega a su **carrito virtual** los que necesita. Una vez que el carrito contiene todos los productos que desea, presiona el botón ‘‘FINALIZAR COMPRA’’, el cual lo dirige a la pantalla de cierre de pedido. En esta pantalla, se le informa del costo total de la compra, y se le ofrecen opciones de fechas y horarios posibles de entrega, de entre las que **Charlie** elige el Miércoles de la semana que viene, por la mañana, ya que sabe que en ese horario va a estar en su casa.

Luego, en la siguiente pantalla, elige la opción de Pago Online, y el sitio le solicita que elija un método de pago de entre las distintas opciones disponibles. Ya que **Charlie** confía mucho en el sistema PayPal (como ejemplo de Agente de Cobro), lo elige, tras lo cual se abre una ventana externa que

redirige al sitio de PayPal, en que tras ingresar su usuario y su clave se le solicita confirmar el valor de la compra. Luego de que esta ventana se cierra, el sitio le muestra una animación muy jocosa de un tigre mirando un reloj, mientras debajo se puede leer la frase ‘‘Por favor, espere, estamos validando el pago...’’. Tras unos segundos, el tigre comienza a bailar, el mensaje se desvanece, y lo reemplaza un nuevo mensaje ‘‘Su pago se encuentra confirmado. Le hemos enviado un mail con la información de su pedido. Gracias por confiar en nosotros. En unos instantes, será redirigido a la página principal.’’.

### 3.3.4 D: Pago contrareembolso

**Dave** tiene un problema de adicción al casino. Normalmente, con la ayuda de sus amigos y familiares, lo controla sin mayores inconvenientes. Pero hace 1 semana tuvo un viaje laboral, y en el último día, libre para todos los empleados, no resistió la tentación de jugar una o dos tiradas de ruleta, con el efectivo que llevaba encima. Tuvo la mala suerte de que le fue relativamente bien, ganó ambas jugadas lo que le hizo cuadruplicar su efectivo. Envalentonado por su repentino y misterioso golpe de suerte, se dirigió a la casilla de venta de fichas, y gastó todo el dinero de su cuenta bancaria en fichas. También compró fichas con su tarjeta de crédito, en un pago, hasta alcanzar el límite. Compró en total 450 fichas, y volvió a dirigirse a la ruleta. Su plan inicial era realizar una paciente Martingala, pero un rayo cósmico atravesó su mente momentos antes de colocar la apuesta, y supo entonces que debía elegir el número 7. Claro, porque este era el séptimo día del viaje laboral, y había tenido mucha suerte, por lo que el siete era un buen número. Claramente, **Dave** perdió todo su dinero, y no solo eso, sino que se endeudó gravemente, saturando el límite de su tarjeta de crédito.

Al volver a su casa, le cuenta lo sucedido a su tío, pidiéndole que no le cuente a nadie, y este le presta dinero ‘‘hasta que logre salir de la situación’’. Como no tiene tiempo de ir al supermercado, ya que debe hacer horas extras para pagar sus deudas, aprovecha el sistema de compras online de mesporciento para encargar las provisiones de la semana durante la noche. Se autentica en el sitio con su usuario, ‘‘lucky\_guy\_00’’, elige los productos indispensables para el resto del mes, y pacta una fecha de entrega para el día siguiente. Al momento de elegir la opción de pago, advierte que no puede realizar un pago online, ya que la tarjeta se encuentra saturada, por lo que opta por elegir la opción de pago contrareembolso.

Al otro día, temprano, suena el timbre, y recibe el pedido, el cual paga en efectivo, y les deja una modesta propina a los muchachos para que carguen las bolsas hasta la cocina de su casa.

### 3.3.5 E: Entrega correcta

El Martes 13 de Abril **Erin** realizó un pedido de torta de cumpleaños, cotillón y un regalo grandioso para ser entregado el Martes siguiente, en conmemoración del 50-cumpleaños de su tía. Aprovecha la posibilidad para elegir que su pedido sea entregado en la casa de su tía, y no en su domicilio. Para ello, intenta modificar el domicilio de su usuario ‘‘ireland\_green’’, pero el sistema no se lo permite. Entonces, como **Erin** es muy inteligente, crea un nuevo usuario ‘‘tia\_50’’, especial para esta ocasión, el cual completa con los datos de su tía. Por comodidad, además, lo paga en línea, con una tarjeta de crédito (la suya, no la de su tía), ya que por costumbre familiar está prohibido hablar de dinero durante el cumpleaños de una tía, y quiere evitar malos momentos durante el episodio festivo.

Llegado el día, están todos festejando, ya con algunas copitas encima, cuando la tía grita a **Erin**: ‘‘¿y la torta? ¿y los juguetitos que me habías prometido?’’. En ese instante, justamente, suena el timbre, y resultan ser los empleados de mesporciento. **Erin** les abre la puerta y les indica dónde dejar la mercadería. Les dice que no puede darles propina por una costumbre familiar, tras lo cual los empleados regresan a su camión, apesadumbrados.

### 3.3.6 F: Ausente durante entrega con límite de entregas fallidas fijo

**Frank** es una persona muy olvidadiza. Tanto es así, que durante la mañana de hoy, fue hasta el banco a cobrar un cheque, para terminar dándose cuenta que no lo había llevado. No fue sino hasta la mañana del día siguiente, al leer su email, que se enteró que, durante su ausencia en el banco, había recibido una visita del supermercado Mes%, al cual había justamente encargado una compra el día anterior. Utilizando un vínculo provisto dentro del mismo mensaje, programó la visita para ese mismo día, al mediodía, y luego se ató un piolín en el dedo corazón para recordarlo. Entonces miró un poco de televisión, y tras terminar el programa, cuando se dispuso a cambiar de canal, se dio cuenta que su dedo, el del piolín obviamente, estaba totalmente ennegrecido y arrugado. Peor aún, a pesar de desatarlo, este había perdido la sensibilidad, y no recuperaba su rosadito color habitual. **Frank** se asustó tanto, que corrió raudo hasta la calle, y tomó un taxi hasta el hospital más cercano, sin advertir que estaba en pijama, y que este no dejaba nada a la imaginación. En el hospital, les explicó que, por razones que no podemos repetir, para él era muy importante este dedo, y que no podía perderlo. Entonces le dieron una bata para que pueda poner sobre el pijama y proteger la sensibilidad del resto de los pacientes, y le realizaron estrambóticos procedimientos médicos. Luego de un par de horas **Frank** pudo recuperar el funcionamiento habitual de su dedo. Cuando el médico le preguntó que por qué se le había puesto así el dedo, **Frank** le explicó que se había atado algo. Tras lo cual el médico hizo una obvia segunda pregunta, lo cual provocó algo similar a un click en algún recóndito lugar del cerebro de **Frank**, seguido de una catarata de imágenes mentales, la mayoría de ellas relacionadas al supermercado Mes%. Entonces, repentinamente, se levantó, corrió hasta la puerta del hospital, y tomó un taxi nuevamente hasta su casa. Al llegar, abrió su mail, para enterarse de que nuevamente había perdido la entrega. Se le informaba, además, de que su usuario “f\_estein” había perdido la posibilidad de realizar compras contrareembolso, hasta pagar una multa de \$100, lo cual ciertamente lo puso de muy mal humor.

### 3.3.7 G: Cancelación de pedido

**Gabriel** había hecho un pedido el lunes a la mañana, pagándolo con tarjeta de crédito. A la tarde descubre mejores precios en los chinos de la vuelta, y decide cancelar la compra para hacerla en ese lugar que le resulta más económico (haciendo las cuentas, llega a la conclusión de que con la diferencia podía comprar un videojuego para sus hijos). Entonces, vuelve rápido de los chinos a su casa, y entusiasmado chequea que aún pueda hacer la cancelación. Para ello ingresa al sitio [www.mespociento.com](http://www.mespociento.com) con su cuenta “gabi\_25x8”, poniendo su contraseña, y luego seleccionando el pedido vigente. Selecciona la opción de modificación de pedido y posteriormente, al comprobar que no está cerrado, lo cancela.

El sistema le confirma la cancelación y se le hace una nota de crédito para reintegrarle el dinero. Finalmente **Gabriel** vuelve contento a los chinos a hacer la compra y con lo que le sobra, compra el videojuego para sus hijos (¡aunque como padre responsable, primero lo juega él para enseñarle a ellos!).

### 3.3.8 H: Modificación de pedido.

En la casa de **Helena** van a festejar la navidad. Unas horas después de que hacer un pedido con la mercadería necesaria para preparar la cena, su esposo **Horacio** le comenta que habían sido invitados a la cena su vecina **Hermenegilda**, junto con sus hijos, debido a que el marido y los parientes de esta aún no habían regresado del exterior, ya que se encontraban en un viaje por asuntos laborales. Por

este motivo, **Helena** decide modificar el pedido que tenía hecho, incrementando la cantidad de productos comprados (como por ejemplo las gaseosas). Para ello, ingresa al sitio [www.mesporciento.com](http://www.mesporciento.com) validando los datos de su cuenta, y chequea que aún no haya sido cerrado dicho pedido. Al comprobarlo, selecciona la opción para modificarlo, y agrega la mercadería que desea. Luego selecciona contrareembolso como opción de pago, y sistema le confirma la modificación. Dado que el pedido había sido originalmente pagado de forma online, **Helena** deberá abonar la diferencia en efectivo en cuanto lo reciba.

### 3.4 I: Preparar y cerrar pedido

*Ismael*, un empleado del depósito, entra al sistema interno y revisa los pedidos pendientes. Selecciona el primero de la lista (están ordenados por prioridad) y lo prepara poniendo en una caja los productos encargados. Luego lo cierra en el sistema interno.

### 3.5 J: Alarma y reposición por bajo stock en depósitos

Luego de que una clienta comprara diez packs de gaseosas Coca-Cola, se agotó el stock de dicho producto, por lo que saltó la alarma del sistema. El sistema entonces le avisa al Departamento de Stock de forma automática. Un superior del Departamento de Stock recibe el aviso y hace los pedidos correspondientes a Logística, quien finalmente se encarga de reponer el stock.

FIX ME: arreglar

## Parte IV. Discusión

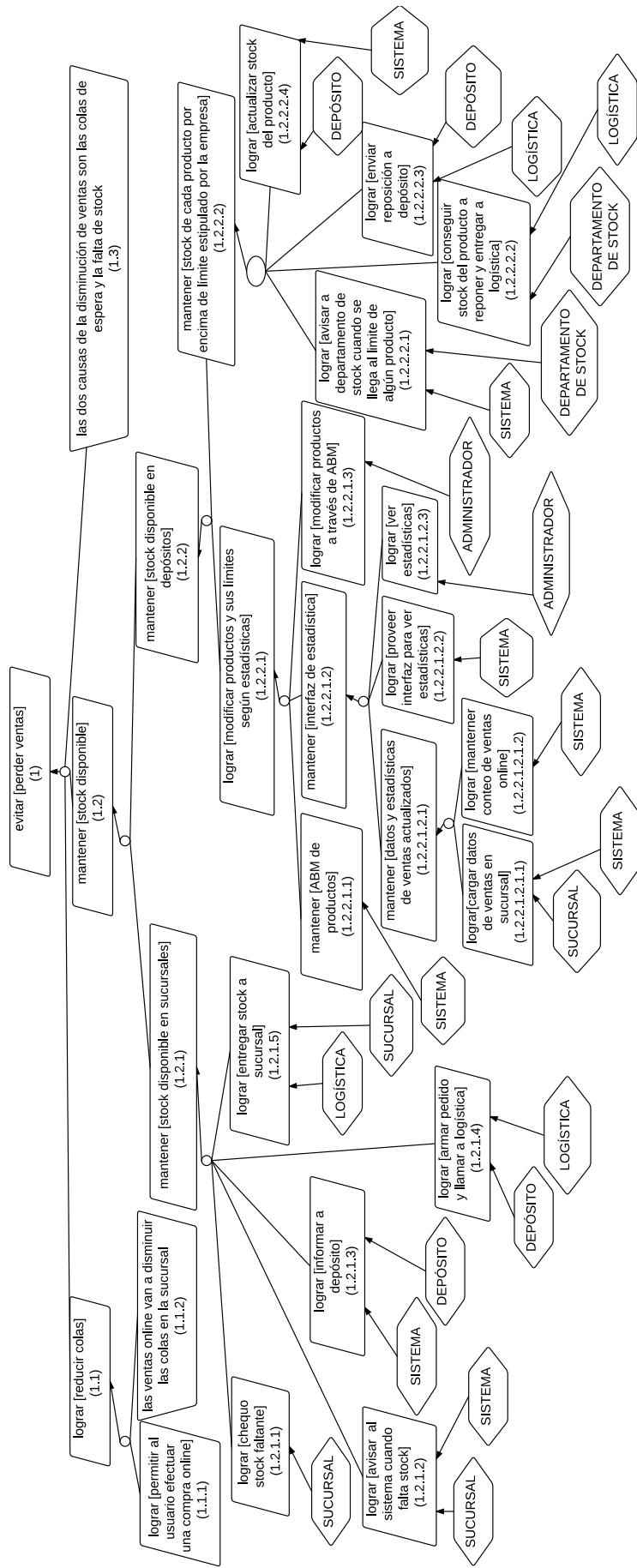


Figure 3.2.1: Objetivo: Evitar perder ventas

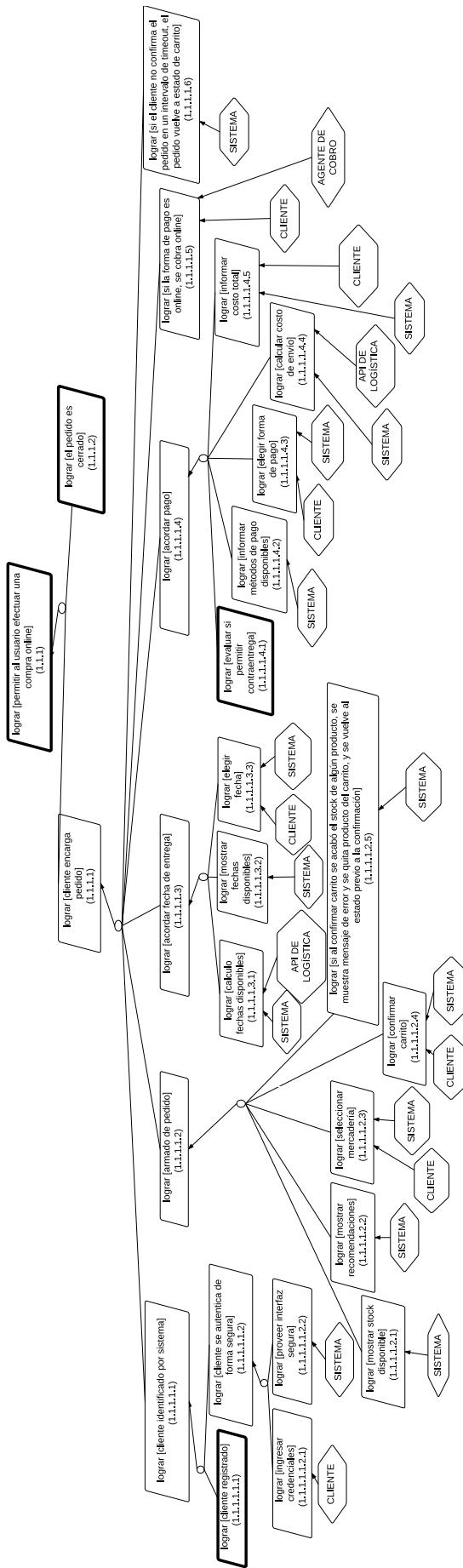


Figure 3.2.2: Objetivo: Lograr permitir al usuario efectuar una compra online

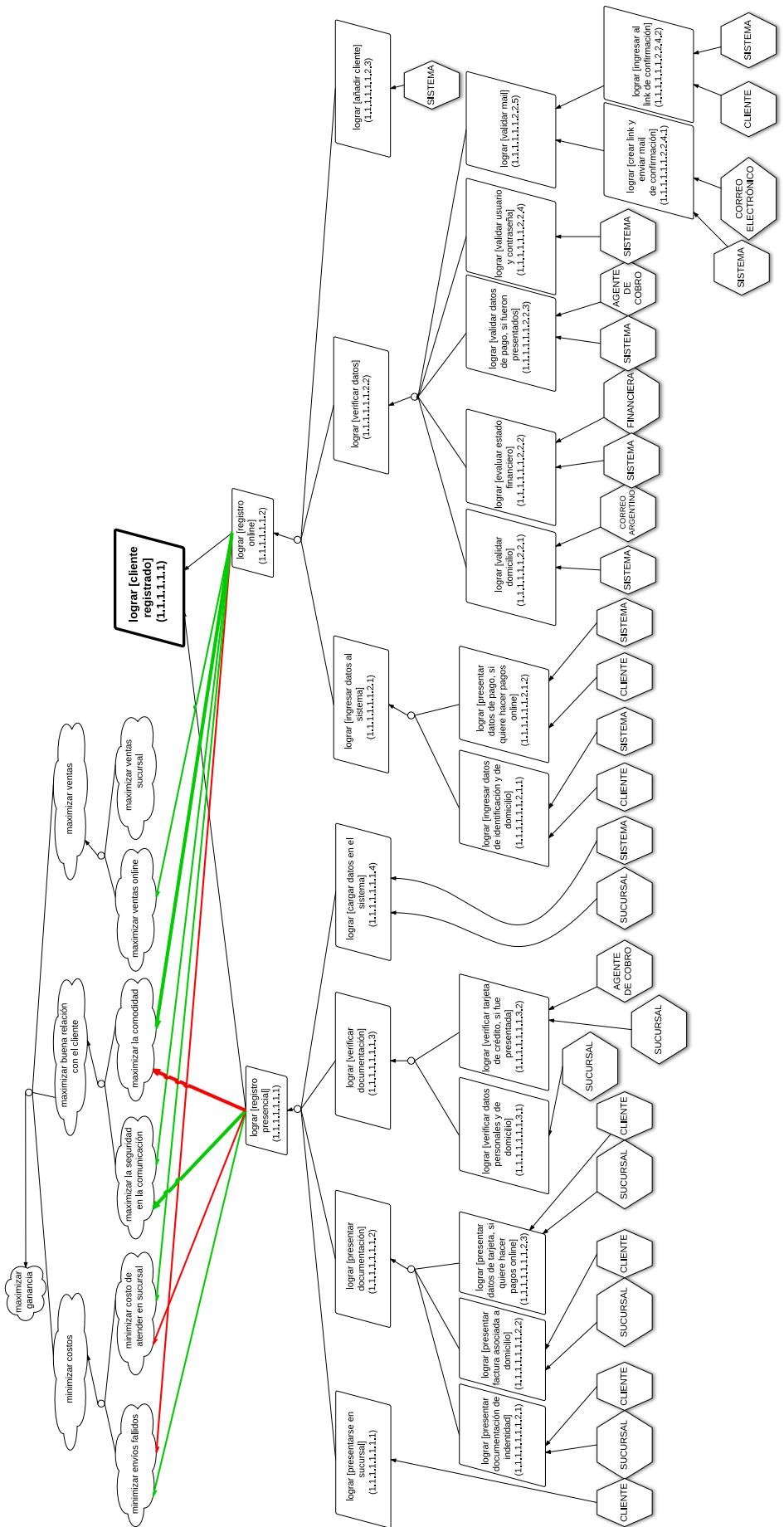


Figure 3.2.3: Objetivo: Lograr cliente registrado

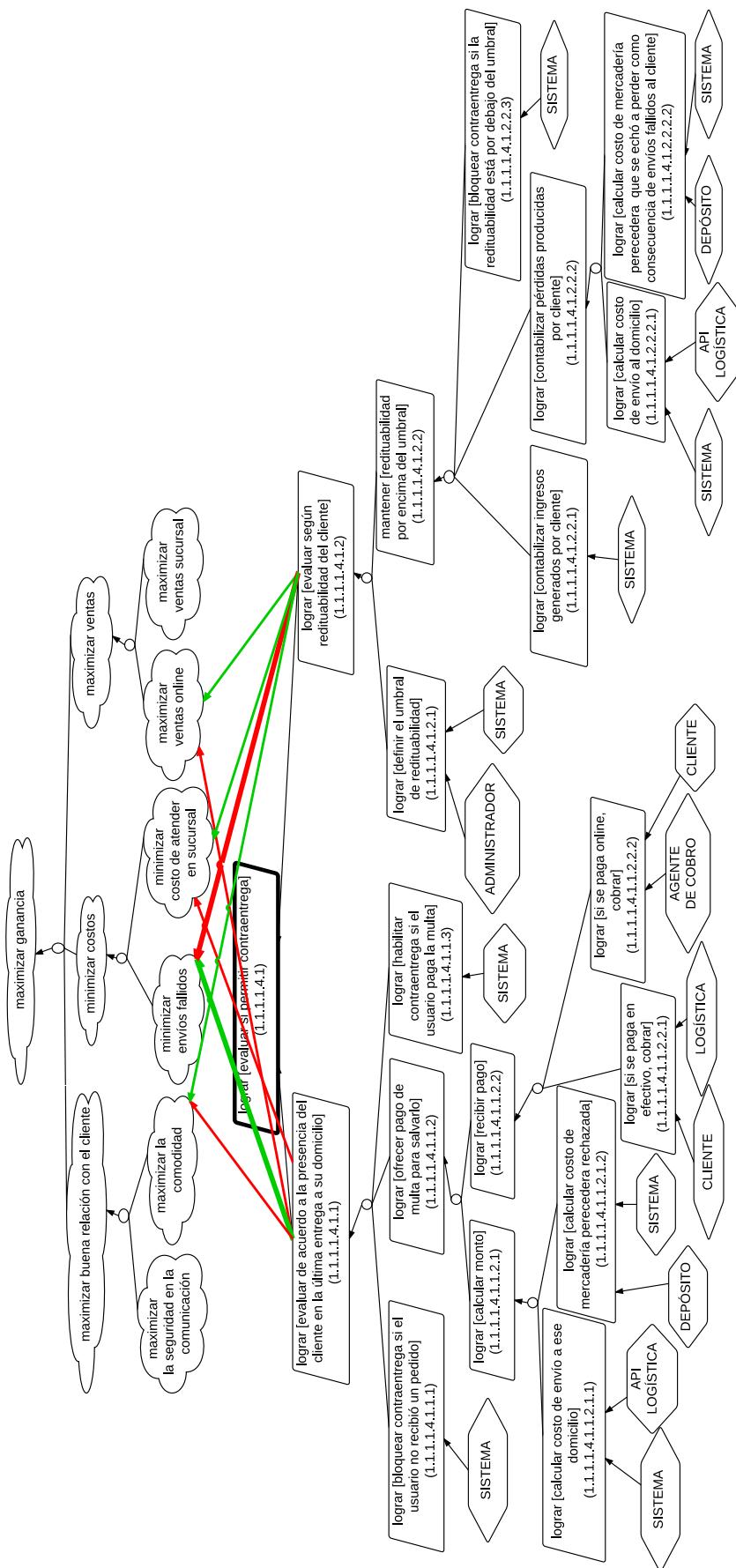


Figure 3.2.4: Objetivo: Lograr evaluar si permitir contraentrega

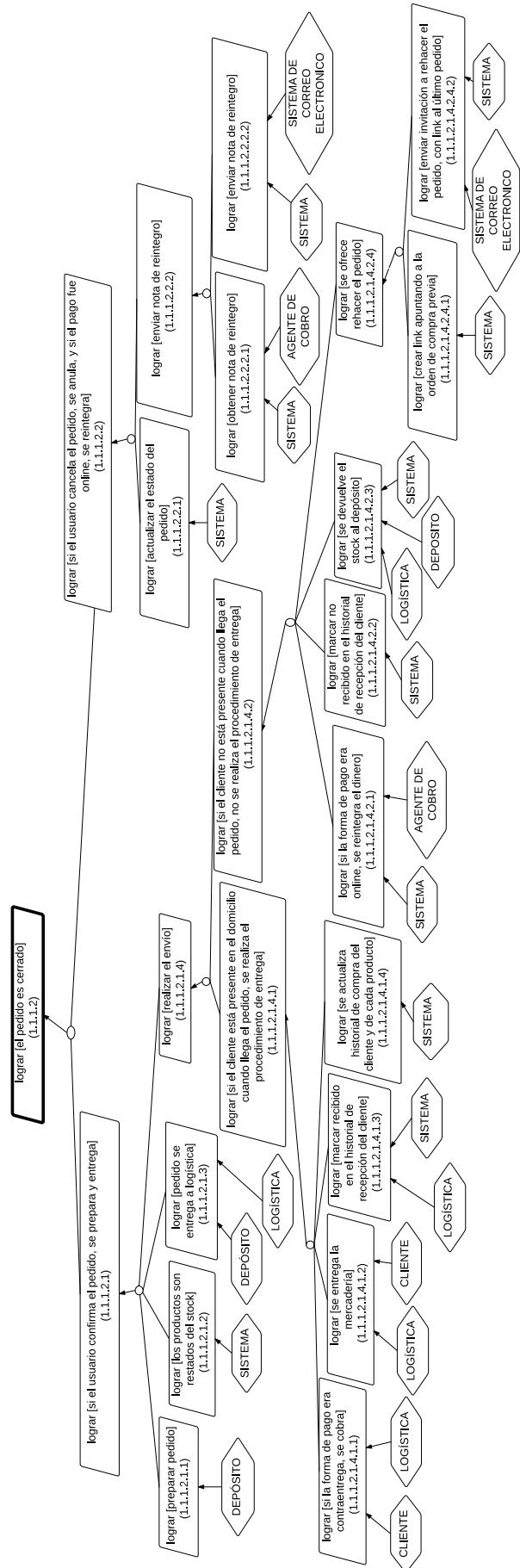


Figure 3.2.5: Objetivo: Lograr el pedido es cerrado

## 4.1 O-Refinamiento: Registro del cliente

Objetivos		En sucursal	Online
Maximizar ventas	Maximizar ventas online	O	+
	Maximizar ventas sucursal	O	O
Minimizar costos	Minimizar envíos fallidos	+	-
	Minimizar costo de atender en sucursal	-	+
Maximizar relación con cliente	Maximizar la seguridad en la comunicación	++	+
	Maximizar la comodidad	--	++

### Ventas

La opción de registro online es evidentemente más simple y accesible, lo que ocasionaría que más clientes opten por registrarse y comprar online, sobre todo aquellos clientes que no acostumbran a comprar normalmente en ese supermercado (ya que esto no impactaría tanto en los clientes habituales). De lo anterior, se deduce que la opción de registro online impactaría positivamente en el objetivo de maximizar las ventas online.

### Costos

Al obligar a los clientes a registrarse presencialmente en una sucursal, exigiéndoles la documentación correspondiente (por ejemplo, dni, un impuesto, una factura de servicio) que acredite su domicilio y su identidad, se estaría obteniendo una verificación confiable de los mismos, lo que repercute en una minimización de los envíos fallidos ocasionados por personas que presentan datos falsos, o sin ser esta su intención, cometan errores al escribir los datos. El registro online, por el contrario, aumenta la posibilidad de que sucedan esos inconvenientes, lo que impactaría negativamente en la minimización de los envíos fallidos.

### Relación con cliente

**Maximizar la seguridad en la comunicación:** Al registrar los datos del cliente en la sucursal, en persona, se presume que la transmisión de los datos del cliente es siempre más segura que si se lo hace online. De todos modos, según lo solicitado por el CEO, la comunicación se establecería por un canal seguro, así que el riesgo de robo de información por medio de una escucha de red se encuentra notablemente suprimido. Desde esa perspectiva, ambos canales de comunicación tienen un nivel de confiabilidad similar.

Por otro lado, mediante el canal de registro online, una persona podría llegar a realizar un registro fraudulento, suplantando la identidad de otra persona, escudada en el anonimato que ofrece internet, mientras que presencialmente es mucho más difícil realizar este tipo de prácticas, de lo que concluimos que, si bien ambos canales son seguros, la vía presencial resulta bastante más confiable.

**Maximizar la comodidad:** Obligar al cliente a realizar el registro presencialmente, suponiendo además que este deba recolectar y presentar toda la documentación pertinente, le puede llegar a resultar molesto e incómodo. En el peor caso el cliente tendría que juntar y fotocopiar toda la documentación exigida, dirigirse a una sucursal en horario laboral, no necesariamente cerca de su domicilio, esperar a ser atendido, completar algún tipo de formulario, entregar la documentación, y esperar a que le informen de algún modo que su usuario fue dado de alta. Mediante el registro online, en cambio, el cliente podrá ingresar al website desde la comodidad de su casa, en cualquier horario, e ingresar sus datos. Inmediatamente, entonces, el cliente tendría la posibilidad de realizar una compra.

## 4.2 O-Refinamiento: Ranking del cliente

Objetivos		Límite de ausencias consecutivas (instanciado en 1)	Puntaje
Maximizar ventas	Maximizar ventas online	-	+
	Maximizar ventas sucursal	0	0
Minimizar costos	Minimizar costos envíos fallidos	++	--
	Minimizar costo de atender en sucursal	-	+
Maximizar relación con cliente	Maximizar la seguridad en la comunicación	0	0
	Maximizar la comodidad	-	+

**Cliente Bueno / Cliente Malo:** Llamaremos cliente bueno a aquel que se asume que en el futuro va a producir ingresos mediante compras de algún tipo. En particular, cuando estemos hablando de compras online, un cliente bueno será aquel que en el futuro realizará compras online. Lo mismo sucede con las compras en sucursal. De igual modo, se define un cliente malo como aquel que, por cualquiera de las posibles razones, no va generar ingresos (o peor, va a generar costos) en el futuro. Obviamente, como se está hablando de eventos futuros, siempre que hablamos de “clientes buenos” y “clientes malos”, se estará hablando de la posibilidad de que un cliente sea bueno, o de que un cliente sea malo, y no de una certeza.

### Maximizar Ventas:

**Maximizar Ventas Online:** Al bloquearle al cliente la posibilidad de realizar ventas online contrareembolso, si bien es posible que algunos clientes buenos sigan comprando de forma online, mediante pago online, también existen las posibilidades de que otros clientes buenos opten por no realizar compras online, pasándose a la modalidad de compra en sucursal, o de que incluso dejen de comprar en esa cadena de supermercados, ya sea por inconveniencia, o por una reacción de enojo causada por el bloqueo.

Al realizar el bloqueo de ventas contrareembolso mediante un límite de entregas fallidas fijo (en este caso consideramos que se le permitiría solo una, pero el razonamiento es similar para un límite mayor), sería fácil que se de la posibilidad de que un cliente bueno (es decir, uno que en el futuro produciría ingresos por compras), deje de realizar compras online. Esto es porque al establecer un límite fijo, no se está teniendo en cuenta el comportamiento anterior del cliente, sino el simple hecho de que en un determinado momento falló un envío. El comportamiento anterior del cliente ayuda a, en mayor o menor medida, predecir cuál sería su comportamiento futuro. Para dar un ejemplo puntual, supongamos que falla la entrega a un cliente que realiza compras online de aproximadamente \$1000 cada día, y consecuentemente se le bloquea la posibilidad de realizar compras contrareembolso. Si esto

fue un error “de una sola vez”, y el cliente sigue realizando compras diarias de \$1000, y elige comprar en sucursal o directamente comprar en otra cadena de supermercados, se están perdiendo \$1000 pesos de compras online por día. Esta sería una situación no deseable.

A diferencia del ejemplo anterior, al decidir el bloqueo de las ventas contrareembolso a partir de la asignación de un puntaje que dependa de la reditabilidad de ese cliente, se está teniendo en cuenta el “historial” de ese cliente. Entonces, volviendo a la misma situación anterior, si un cliente realiza compras diarias, y en gran cantidad, y uno de esos días una de los envíos no es recibido, si bien es cierto que esto le estaría ocasionando costos a la empresa, y por lo tanto una pérdida de rentabilidad, se estaría aceptando una determinado umbral de variación en la rentabilidad, en pos de no perder la posibilidad de que un cliente bueno siga produciendo ingresos en el futuro.

**Maximizar Ventas Sucursal:** Se puede llegar a interpretar que al bloquearle la venta contrareembolso, un cliente podría optar por comprar en sucursal, y por lo tanto asumir que cualquiera de las metodologías aporta al objetivo de maximizar las ventas en sucursal. Más aún, siguiendo esta línea de pensamiento, el refinamiento que más frecuentemente produzca bloqueos de ventas contrareembolso sería el que más aporta a maximizar las ventas en sucursal.

Esta forma de pensar no nos parece correcta ya que, en todo caso, se podría considerar que el cliente está siendo coercionado, debido a que no tiene una voluntad real de realizar compras en la sucursal, o al menos no la tenía al momento en que se le hizo el bloqueo, sino que esta “decisión” se produce como consecuencia colateral del bloqueo, ya que en este caso al decidir realizar este bloqueo la empresa no lo está haciendo con el objetivo de estimular las ventas en sucursal, sino de evitar los costos que le producen las entregas fallidas.

Además, uno de los objetivos para los cuales fue solicitado el sistema de compras online, es evitar que las ventas que podrían realizarse de forma online, sean realizadas en sucursal, ya que eso genera inconvenientes en las colas y en el stock. Que un cliente deje de comprar online y compre en sucursal, entonces, está aportando a que se produzcan estos inconvenientes que se querían evitar en un principio. Estamos asumiendo que existen ciertos clientes que no tienen la posibilidad, o no quieren, realizar sus compras por internet. El objetivo de “maximizar las ventas en sucursal”, apunta realmente a que estos clientes puedan realizar sus compras sin inconvenientes. Desde este punto de vista, al producir una mayor demanda en las sucursales, se está atentando contra el objetivo.

Por los motivos expuestos, asignamos un puntaje neutral en ambos refinamientos, ya que si bien es cierto que en ambos casos se están “fomentando” las ventas en sucursal, lo cual podría ser considerado como positivo, también es cierto que esto no es una consecuencia directa, ni es el objetivo del refinamiento, y que este aumento en las ventas no necesariamente tiene un aporte significativo al objetivo de “maximizar las ventas”, sino que incluso puede jugarle en contra, generando una mayor demanda de stock y un abarrotamiento en las sucursales a causa de clientes que en realidad podrían estar comprando online. **Minimizar Costos: Minimizar envíos fallidos:** Es prácticamente imposible evitar los envíos fallidos, ya que son situaciones que no se pueden predecir.

Desde el punto de vista de las entregas contrareembolso, al cortarle la posibilidad de realizar pedidos contraentrega ante la primer entrega fallida, se está minimizando totalmente el riesgo de que se produzcan nuevas entregas fallidas. Ya que, si bien un cliente malo puede comprar pagando de forma online, y producir entregas fallidas, el costo de la entrega se le estaría debitando en cada compra, por lo que no se estaría produciendo un perjuicio para la empresa. Más aún, esto produciría un efecto en el comportamiento del cliente, ya que si este sabe de antemano que si no recibe una entrega perderá toda posibilidad de realizar nuevos pedidos contraentrega, va a ser mucho más cuidadoso. Este refinamiento ayuda, entonces, a minimizar los costos de entregas fallidas.

Si, en cambio, el bloqueo de pedidos contraentrega se realiza en base a un rating, existe la posibilidad de que un cliente, manteniendo un buen rating, produzca de todos modos muchas entregas fallidas. También se produciría un efecto negativo en el cliente, ya que al no percibir consecuencias directas puede quitarle importancia a “atender una entrega”, y no recibir entregas que en otro caso si recibiría (priorizando, quizás, otras actividades que le surjan).

Volviendo al mismo ejemplo utilizado anteriormente, supongamos que existe un cliente que realiza compras que producen ingresos diarios de \$1000, pero al menos una vez por semana falla alguna entrega. En la primera opción de refinamiento, luego de la primer entrega fallida, se le bloquearía la contraentrega, y por lo tanto todos los futuros costos de entregas fallidas serían eliminados. En la segunda opción, el cliente estaría produciendo costos de entregas fallidas de forma constante y, además, al notar que su irresponsabilidad no produce ningún tipo de consecuencias, no se esforzaría por subsanarla.

**Minimizar costos de atender en sucursal:** Asumiendo que ante una mayor cuota de clientes con contraentrega bloqueada, hay a su vez una mayor cuota de clientes que optan por comprar en sucursal, es evidente que la opción que más bloqueos a contraentrega produzca es la que más clientes estaría enviando a la sucursal.

Teniendo en cuenta que atender un cliente en la sucursal tiene costos inherentes, la opción que estaría minimizando los costos de atender en sucursal es la de bloquear la contraentrega ante la primer entrega fallida.

Del mismo modo, si se permite cierto umbral de entregas fallidas en base a la rentabilidad de ese cliente, el cliente no necesariamente se vería obligado a comprar en sucursal, y por lo tanto no produciría un gasto de atender en sucursal.

Además, existe otro costo, que es el de disponer de una persona que pueda cobrar la multa que el cliente debe pagar para subsanar el costo del envío no recibido, y que se le vuelva a permitir contraentrega. En el caso de los clientes que no tengan la posibilidad de realizar compras online, estos se dirigirían a la sucursal para pagar la multa. En cualquier caso el primer refinamiento, al producir más bloqueos, produciría más multas, y estaría aportando a este costo. En el caso del segundo refinamiento, al aceptar un umbral de pérdida de rentabilidad, el costo de las multas no necesariamente debe ser pagado por el cliente, sino hasta que se supere el umbral. En este caso se podría considerar, además, la posibilidad de que la multa sea pagada en efectivo, junto con la siguiente compra contraentrega.

### Relación con cliente:

**Maximizar la seguridad en la comunicación:** No parece haber ningún tipo de relación entre la forma en que se decide (o no) permitir la contraentrega, y la seguridad en la comunicación con el cliente. Asignamos puntajes neutros en ambos refinamientos.

**Maximizar la comodidad:** Asumiendo que la contraentrega es una comodidad que se le ofrece al cliente, ya que además de que se le brinda la posibilidad de comprar online, con todas las comodidades que esto acarrea, no necesita realizar el pago de forma online, lo que también puede considerarse una comodidad en caso de que el cliente no tenga forma de pagar online, o no desee hacerlo. Al bloquear la posibilidad de realizar contraentrega, entonces, se le está quitando esta comodidad. Dado el caso, el cliente tendrá que decidir si pagar online, o comprar en sucursal.

Si bien ambas opciones de refinamiento pueden desembocar en un bloqueo de la modalidad contraentrega, ya se explicaron las razones por las cuales el primer refinamiento tiene potencialmente una mayor tasa de bloqueos, por lo que va en contra de la comodidad del cliente. Además, considerando que pueden haber razones reales por las que el cliente pueda ausentarse durante el horario de entrega, un cliente que siempre tuvo una buena conducta, puede sentirse enojado de que se le bloquee la contraentrega, lo cual atenta contra la buena relación con el mismo. Por otro lado, al manejar un sistema de rating, si bien no se está aportando de forma directa a la comodidad del cliente, se logra mantener una comodidad que de otra forma el cliente perdería, por lo que consideramos que hay un pequeño aporte aquí.

## Parte V. Conclusiones TP1

El presente trabajo resultó una efectiva introducción a varios de los problemas que la **Ingeniería de Software** busca resolver.

El primer paso fue plantear los **fenómenos** que consideramos forman parte de la modificación a la cadena **Mes%**. Empezaron a surgir así los distintos conceptos que luego se transformarían en **agentes**, junto con las primeras **presunciones de dominio**.

Luego, nos abocamos al desarrollo de un **modelo de objetivos**. Durante el transcurso de este proceso fue necesario **refinar** nuestros datos sobre varios asuntos por lo que hubo una **interacción intensa** con el tutor, que hacia las veces de CEO de la cadena. El resultado de esta etapa de modelado quedó plasmado en un **diagrama de objetivos**.

Una vez que el diagrama de objetivos estuvo finalizado, esbozar el **diagrama de contexto** y exemplificar los potenciales **escenarios de uso** fue relativamente sencillo.

Algo que llamó la atención con respecto a trabajos prácticos de otras materias fue una necesidad abordar los problemas de forma grupal y tomar decisiones consensuadas. Todas las secciones del trabajo están conectadas y requirieron coordinación activa.

*Al hacer una evaluación sobre las dificultades que nos fueron surgiendo a lo largo del desarrollo de este TP, nos encontramos con muchos puntos que valen la pena destacar.*

Por un lado, la ya mencionada necesidad de mantener una comunicación fluida, permanente, y clara entre todos los integrantes, junto con la dificultad de coordinar las tareas, nos expuso a frecuentes confrontaciones y desacuerdos (necesarios), que enlentecieron el desarrollo del trabajo.

Un vicio constante fue pensar en la implementación al definir estos modelos más abstractos, limitando y forzando de forma prematura, quizás, el resultado.

La parte del mundo que estábamos describiendo nos resultó demasiado compleja como para ser descripta fielmente a través de los modelos con los que contábamos. Además de las limitaciones que cada uno de ellos mostró por separado, al integrarlos se sintió la falta de herramientas que nos permitieran modelar otros aspectos que, se presume, se explorarán más avanzada la materia.

Existió ciertamente una dificultad implementativa, relacionada directamente a la complejidad de los diagramas necesarios, frente a la sencillez de las herramientas utilizadas para realizarlos. Consideramos que esto es un detalle que vale la pena mencionar, ya que la diferencia entre los software de diagramas pagos (por ejemplo, Microsoft Visio) frente a aquellos gratuitos (por ejemplo, draw.io) es **muy significativa**, lo que nos sumó una dificultad innecesaria y que se encuentra más allá de la correcta aplicación de las técnicas aprendidas.

## Parte VI. Introducción TP2

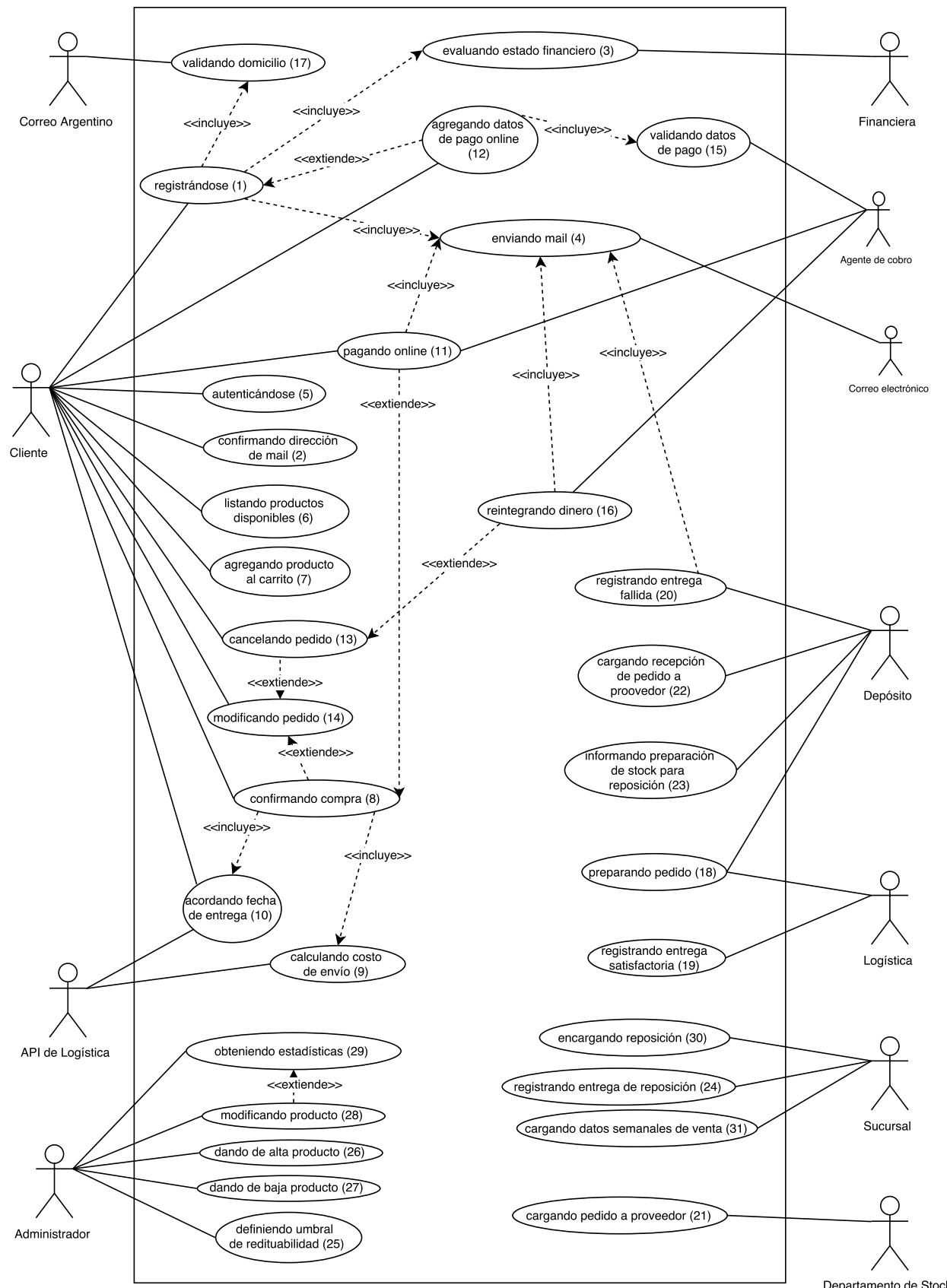
El presente Trabajo Práctico es una extensión del TP1: Mes%. Utilizando como base la temática y los modelos descriptos en el anterior, aplicamos diversas técnicas para documentar distintos aspectos del proyecto. Realizamos un Modelo Conceptual, abarcando todos los aspectos del sistema, para describir las entidades que intervienen en el mismo, sus atributos, y la semántica de sus relaciones, y un Modelo de Casos de Uso para describir las operaciones desde el punto de vista de los distintos agentes, representadas funcionalmente a través de un listado de acciones provistas, producidas o transmitidas por la máquina a través de la interfaz.

Para poder representar algunos aspectos más complejos decidimos utilizar otros modelos que nos permiten modelar los comportamientos de los agentes. En particular, usamos Diagramas de Actividad, para representar aquellas acciones o actividades que si bien no forman parte intrínseca de la máquina, influyen en la operatoria del sistema. Más concretamente, mediante la utilización de este modelo, pudimos caracterizar el flujo de las operaciones descriptas en Modelo de Casos de Uso, así como su interacción con agentes externos al sistema.

Finalmente, mediante el Modelo de Máquinas de Estado Finitas (FSM) pudimos representar las transiciones de estados que puede sufrir un agente, de acuerdo a los eventos que se disparan en el sistema. También nos sirvió para mostrar los fenómenos de sincronización, pudiendo darle simultaneidad arbitraria a los cambios de agentes partícipes de un mismo evento.

Tuvimos que tomar algunas decisiones sobre los o-refinamientos planteados en el Diagrama de Objetivos del Trabajo Práctico anterior. Con respecto a la forma de efectuar el registro, se optó por la modalidad online, ya que se trata de una metodología establecida hace tiempo en el mercado que ya es parte de la experiencia de usuario esperada por los clientes. También se decidió delegar la decisión de permitir contraentrega a un cliente a la evaluación de su nivel de redituabilidad, ya que se consideró este un mecanismo de control que no requiere participación activa de las partes implicadas, y que resulta eficaz para prevenir las pérdidas asociadas a envíos fallidos.

## Parte VII. Casos de Uso



Objetivo: 1.1.1 - Permitir al usuario la compra online

**Caso de Uso: 1) Registrándose**

**Actors:** Cliente

**Pre:** -

**Post:** Se ha enviado un mail de bienvenida, el usuario se encuentra pendiente de confirmar.

**Curso Normal**

1. El cliente ingresa su usuario y contraseña
2. El sistema valida que el usuario no exista.
3. El sistema valida que la contraseña sea segura.
4. El cliente ingresa sus datos personales
5. Incluye CU 3: evaluando estado financiero
6. El cliente ingresa su dirección
7. Incluye CU 17: validando domicilio
8. Si el cliente desea: es extendido por CU: Agregando datos de pago online
9. El cliente ingresa su mail
10. El sistema define el link y el contenido del mail de bienvenida
11. Incluye CU 4: enviando mail
12. FIN CU

**Curso Alternativo**

- 2.1 Si el usuario ya existe, mostrar mensaje e ir a 1.
- 3.1 Si la contraseña no es segura, informar al cliente e ir a 1
- 5.1 Si el cliente tiene deudas, denegarle el registro. FIN CU.
- 7.1 si el domicilio no es válido, mostrar mensaje de error e ir a 6.

Caso de uso 1: Registrándose

Este caso de uso responde al objetivo **1.1.1.1.1.1 - Lograr[Cliente registrado]**.

Los pasos 1, 4, 6 y 9 se corresponden con el objetivo **lograr [ingresar datos de identificación y de domicilio]** (1.1.1.1.1.2.1.1), y con el fenómeno **3a: el cliente presenta datos de identificación y de domicilio**.

Los pasos 2 y 3 se corresponden con el objetivo **lograr [validar usuario y contraseña]** (1.1.1.1.1.2.2.4).

El paso 5 responde al objetivo **lograr [evaluar estado financiero]** (1.1.1.1.1.2.2.2), y se corresponde con el fenómeno **3d: sistema evalúa estado financiero a través de financiera**.

El paso 7 responde al objetivo **lograr [validar domicilio]** (1.1.1.1.1.2.2.1), y al fenómeno **3c: el sistema valida domicilio con el correo argentino**

El paso 8 responde al objetivo **lograr [presentar datos de pago, si quiere hacer pagos online]** (1.1.1.1.1.2.1.2) y al evento **3b: cliente presenta datos de pago al sistema**.

Los pasos 10 y 11 responden al objetivo **lograr [validar mail]** (1.1.1.1.1.2.2.5) y al evento **3e: el sistema le indica al correo electrónico que envíe mail de confirmación al cliente**.

**Caso de Uso: 2) Confirmando dirección de mail****Actors:** Cliente**Pre:** Se ha enviado un mail de confirmación**Post:** El usuario se confirma**Curso Normal**

1. El cliente ingresa al link de confirmación
2. El sistema marca al usuario como validado
3. FIN CU

**Curso Alternativo**

- 2.1 Si el link no es válido, se le informa al usuario.
- FIN CU

Caso de uso 2: Confirmando dirección de mail

Este caso de uso se corresponde con el objetivo **lograr [validar mail]** (1.1.1.1.1.2.2.5), y con el evento al evento **3e: el sistema le indica al correo electrónico que envíe mail de confirmación al cliente.**

El paso 2 se corresponde con el objetivo **lograr [añadir cliente]** (1.1.1.1.1.2.3).

**Caso de Uso: 3) Evaluando estado financiero****Actors:** Financiera**Pre:****Post:****Curso Normal**

1. El sistema envía request a la API de estado financiero con el DNI del cliente
2. El sistema parsea la respuesta de la API

**Curso Alternativo**

Caso de uso 3: Evaluando estado financiero

Este caso de uso responde al objetivo **lograr [evaluar estado financiero]** (1.1.1.1.1.2.2.2), y con el fenómeno **3d: sistema evalúa estado financiero a través de financiera.**

**Caso de Uso: 4) Enviando mail****Actors:** Correo electrónico**Pre:** El sistema definió un mensaje para un cliente**Post:** Se ha delegado el envío de un correo electrónico**Curso Normal**

1. El sistema define el asunto, cuerpo y destinatario (cliente) del nuevo mensaje.
2. El sistema pide al servidor de correo electrónico enviar el mensaje.
3. El servidor de correo electrónico confirma el encolamiento del mensaje.

**Curso Alternativo**

Caso de uso 4: Enviando mail

Este caso de uso se corresponde con el fenómeno **8e: el sistema de correo electrónico envía mail al cliente.**

**Caso de Uso: 5) Autenticándose****Actors:** Cliente**Pre:****Post:** El cliente se encuentra autenticado**Curso Normal**

1. El cliente ingresa el usuario y la contraseña
2. El sistema verifica los datos ingresados por el cliente
3. El cliente es redirigido al portal de bienvenida
4. FIN CU

**Curso Alternativo**

- 3.1 Si los datos son inválidos, se muestra un mensaje de error
- 4.1 FIN CU

Caso de uso 5: Autenticándose

Este caso de uso se corresponde con el evento **4a) el cliente ingresa credenciales en el sistema**, y con el objetivo **1.1.1.1.2 - Lograr[Cliente se autentica de forma segura]**, y con sus subobjetivos.

**Caso de Uso: 6) Listando productos disponibles****Actors:** Cliente**Pre:** El cliente está autenticado**Post:** Se le envió al cliente un listado personalizado de productos**Curso Normal**

1. El cliente ingresa al listado de productos
2. El sistema obtiene los productos que están en stock
3. El sistema obtiene las recomendaciones para el usuario
4. El sistema muestra los productos y las recomendaciones que están en stock
5. FIN CU

**Curso Alternativo**

Caso de uso 6: Listando productos disponibles

Los pasos 1, 2 y 4 se corresponden con el objetivo **lograr [mostrar stock disponible] (1.1.1.2.1)** y con el fenómeno **4b: el sistema muestra stock disponible al cliente**.

Los pasos 3 y 4 se corresponden con el objetivo **lograr [mostrar recomendaciones] (1.1.1.2.2)** y con el fenómeno **4c: el sistema muestra recomendaciones al cliente**.

**Caso de Uso: 7) Agregando producto al carrito****Actors:** Cliente**Pre:** El cliente tiene el listado de productos**Post:** El producto es agregado al carrito**Curso Normal**

1. El cliente hace click sobre el producto
2. El sistema muestra un dropdown con la cantidad de unidades que están disponibles en ese momento
3. El cliente elige la cantidad
4. El sistema agrega el producto al carrito y calcula el monto total
5. FIN CU

**Curso Alternativo****Caso de uso 7: Agregando producto al carrito**

Este caso de uso se corresponde con el objetivo **lograr [seleccionar mercadería]** (1.1.1.1.2.3) y con el fenómeno **4d: el cliente selecciona mercadería en el sistema**. El paso 2 se corresponde con el objetivo **lograr [mostrar stock disponible]** (1.1.1.1.2.1).

**Caso de Uso: 8) Confirmando compra****Actors:** Cliente**Pre:** El cliente tiene un carrito armado**Post:** El cliente tiene una compra reservada y confirmada**Curso Normal**

1. El sistema ratifica la disponibilidad de stock para cada producto, y los reserva para el cliente; el carrito se encuentra reservado
2. Incluye CU 9: Calculando costo de envío
3. El sistema informa del costo total de la compra, incluyendo el envío.
4. Incluye CU 10: Acordando fecha de entrega
5. El sistema determina si el cliente tiene autorizado el pago contraentrega, y presenta los métodos de pagos disponibles
6. El cliente aprueba el costo y las fechas informadas, e indica el método de pago deseado
7. Si el pago es online: Es extendido por CU 11: Pagando online.
8. El pedido es confirmado
9. FIN CU

**Curso Alternativo**

- 1.1 Si algún producto ya no tiene disponibilidad, se resta del carrito y se le informa al usuario; vuelve al paso 1.
- 7.1 Si el pago es online: si el pago no puede ser efectuado, vuelve a paso 7.
- 8.1 Si el pedido no pudo ser confirmado luego de 10 minutos, los productos son reingresados a stock, y el carrito deja de estar reservado, vuelve a paso 1

**Caso de uso 8: Confirmando compra**

Este caso de uso responde al objetivo **lograr [confirmar carrito]** (1.1.1.1.2.4), y al fenómeno **5a: el cliente confirma el carrito al sistema**.

Los pasos 1 y 1.1, se corresponden con el objetivo **lograr [si al confirmar carrito se acabó el stock de algún producto, se muestra mensaje de error y se quita producto del carrito,**

y se vuelve al estado previo a la confirmación] (1.1.1.1.2.5) y con los fenómenos **5b: sistema ratifica al cliente stock disponible** y **5c: sistema muestra mensaje de error de producto no disponible al cliente**.

El paso 2 se corresponde con el fenómeno **5d: sistema obtiene el costo del envío de la API de Logística** y con el objetivo **lograr [calcular costo de envío]** (1.1.1.1.4.4).

El paso 3 se corresponde con el fenómeno **5e: sistema informa el costo de la compra y el envío al cliente**, y con el objetivo **lograr [informar costo total]** (1.1.1.1.4.5).

El paso 4 se corresponde con los fenómenos **5f, 5g y 5h**, y con el objetivo **1.1.1.1.3 - Lograr[Acordar fecha de entrega]** y sus subobjetivos.

El paso 5 se corresponde con el evento **5i: sistema presenta los métodos de pago disponibles al cliente** y con los objetivos **1.1.1.1.4.1 - Lograr[Evaluar si permitir contraentrega]** y **lograr [informar métodos de pago disponibles]** (1.1.1.1.4.2).

El paso 6 se corresponde con el evento **5j: el cliente elige el método de pago** y con el objetivo **lograr [elegir forma de pago]** (1.1.1.1.4.3).

Los fenómenos **5k, 5l, 5m, 5n, 5o y 5p** se corresponden con el paso 7, y con el objetivo **1.1.1.1.5 - Lograr[Si la forma de pago es online, cobrar]**.

El paso 7.1 se corresponde con el evento **5q: El sistema informa de error de pago al cliente** y está relacionado con el objetivo **1.1.1.1.5 - Lograr[Si la forma de pago es online, cobrar]**.

El paso 8.1 se corresponde con el evento **5r: sistema muestra error de timeout de carrito al cliente**, que se corresponde con **lograr [si el cliente no confirma el pedido en un intervalo de timeout, el pedido vuelve a estado de carrito]** (1.1.1.1.6).

#### Caso de Uso: 9) Calculando costo de envío

**Actors:** API de Logística

**Pre:** El domicilio fue previamente validado.

**Post:**

##### Curso Normal

1. El sistema le consulta a la API de logística por el costo de envío hacia el domicilio de un cliente
2. La API le devuelve al sistema el costo de envío asociado a ese domicilio.
3. FIN CU

##### Curso Alternativo

Caso de uso 9: Calculando costo de envío

**Caso de Uso: 10) Acordando fecha de entrega****Actors:** Cliente, API de Logística**Pre:** El cliente tiene un carrito reservado**Post:** El pedido tiene fecha tentativa de entrega**Curso Normal**

1. El sistema pregunta próximas fechas libres a la API de logística
2. El sistema presenta las posibles fechas al cliente
3. El cliente elige la fecha deseada
4. FIN CU

**Curso Alternativo**

Caso de uso 10: Acordando fecha de entrega

El paso 1 se corresponde con el evento **5f: sistema pregunta próximas fechas libres a la API de logística** El paso 2 se corresponde con el evento **5g: el sistema presenta las posibles fechas de entrega al cliente** El paso 3 se corresponde con el evento **5h: el cliente indica la fecha de entrega deseada al sistema**

**Caso de Uso: 11) Pagando online****Actors:** Cliente, Agente de Cobro**Pre:** El cliente eligió el método de pago online**Post:** El pago del cliente fue acreditado**Curso Normal**

1. Si el cliente lo desea, es extendido por CU 12: agregando datos de pago online
2. El sistema muestra datos de pago asociados al cliente
3. El cliente indica método y datos de pago
4. El sistema abre una ventana del agente de cobro con los datos de la transacción.
5. El cliente realiza la operación a través del agente de cobro, generando un token comprobante del pago.
6. El sistema recibe el comprobante de pago, y lo verifica contra el agente de pago.
7. El sistema prepara un mail para el cliente, adjuntando un comprobante de pago de la operación.
8. Incluye CU 4: enviando mail
9. FIN CU

**Curso Alternativo**

- 3.1 Si el cliente no posee datos de pago, ir a paso 1
- 5.1 Si el agente de cobro rechaza el pago, ir al paso 1.
- 6.1 Si el token de pago es inválido, informar al usuario, e ir al paso 1

Caso de uso 11: Pagando online

**Caso de Uso: 12) Agregando datos de pago online****Actors:** Cliente**Pre:** El cliente está autenticado**Post:** El cliente posee un nuevo dato de pago asociado a su cuenta**Curso Normal**

1. El cliente elige el método de pago online de entre las opciones disponibles
2. Según el método de pago elegido, el cliente ingresa los datos de autenticación solicitados.
3. Incluye: validando datos de pago
4. El sistema asocia los datos de pago a la cuenta del cliente
5. FIN CU

**Curso Alternativo**

- 4.1 Si los datos de pago son inválidos regresa a paso 1.

Caso de uso 12: Agregando datos de pago online

**Caso de Uso: 13) Cancelando pedido****Actors:** Cliente**Pre:** El cliente tiene un pedido sin armar en depósito.**Post:** El pedido es cancelado.**Curso Normal**

1. Si el carrito está confirmado, la reserva de productos se anula, y los mismos se reingresan a stock.
2. Si el pedido ya fue pagado de forma online, es extendido por CU 16: Reintegrando dinero
3. El pedido es cancelado.
4. FIN CU

**Curso Alternativo**

Caso de uso 13: Cancelando pedido

**Caso de Uso: 14) Modificando pedido****Actors:** Cliente**Pre:** El cliente tiene un pedido sin armar en depósito.**Post:** El pedido es cancelado, y un nuevo pedido con las modificaciones es creado.**Curso Normal**

1. El cliente quita o agrega los productos que desee, siempre y cuando haya disponibilidad de stock.
2. Si el carrito está sin confirmar, se registra la modificación
3. Si el carrito fue confirmado, se cancela el pedido anterior: es extendido por CU 13 Cancelando Pedido.
4. Si el carrito fue cancelado, se genera un nuevo pedido con los datos modificados.
5. Si un nuevo pedido fue generado, es extendido por CU 8: Confirmando compra.
6. FIN CU

**Curso Alternativo**

Caso de uso 14: Modificando pedido

**Caso de Uso: 15) Validando datos de pago****Actors:** Agente de Cobro**Pre:** El cliente ingresó los datos de pago**Post:** Los datos de pago fueron validados**Curso Normal**

1. El sistema envía los datos de pago del cliente al Agente de Cobro a través de una API.
2. El Agente de Cobro informa sobre la validez de los datos de pago.
3. FIN CU

**Curso Alternativo**

- 2.1 Si los datos de pago no son válidos, el sistema los marca como inválidos, FIN CU

Caso de uso 15: Validando datos de pago

**Caso de Uso: 16) Reintegrandos dinero****Actors:** Agente de cobro**Pre:** El cliente canceló un pedido**Post:** El dinero correspondiente al pedido fue reintegrado al cliente**Curso Normal**

1. El sistema contacta al agente de cobro, solicitando la anulación de las operaciones correspondientes al pago del pedido.
2. El agente de cobro anula las operaciones de pago solicitadas, y entrega un número de operación y un comprobante de anulación para cada una de ellas.
3. El sistema prepara un mensaje para el cliente, informando que el pedido fue anulado, adjuntando los comprobantes de devolución, y lo devuelve al cliente por pantalla.
4. Utilizando el mensaje anterior, incluye caso de uso 4: Enviando mail.
5. FIN CU

**Curso Alternativo**

- 3.1 Si el pago no puede ser anulado, se le informa de la situación al cliente, brindándole los números de operación correspondiente.

Caso de uso 16: Reintegrandos dinero

**Caso de Uso: 17) Validando domicilio****Actors:** API de Correo Argentino**Pre:** El cliente ingresó los datos de su domicilio**Post:** Los datos de domicilio del cliente fueron validados**Curso Normal**

1. El sistema envía los datos de domicilio del cliente mediante la API del Correo Argentino
2. La API del Correo Argentino responde con un mensaje informando la validez del domicilio enviado
3. FIN CU

**Curso Alternativo**

Caso de uso 17: Validando domicilio

Este caso de uso responde al objetivo **lograr [validar domicilio]** (1.1.1.1.1.2.2.1) y al fenómeno **3c: el sistema valida domicilio con el correo argentino**.

**Caso de Uso: 18) Marcando pedido como preparado****Actors:** Depósito**Pre:** El depósito terminó de preparar un pedido**Post:** El pedido es marcado como preparado**Curso Normal**

1. Un operario del depósito carga en el sistema que terminó de preparar el pedido
2. El sistema marca el pedido como preparado
3. FIN CU

**Curso Alternativo**

Caso de uso 18: Marcando pedido como preparado

**Caso de Uso: 19) Registrando entrega satisfactoria****Actors:** Logística**Pre:** Se ha realizado una entrega satisfactoria**Post:** La entrega fue registrada en el sistema**Curso Normal**

1. Logística registra la entrega al cliente satisfactoria
2. Si el pago fue contraentrega, logística carga los datos de recepción del dinero
3. FIN CU

**Curso Alternativo**

Caso de uso 19: Registrando entrega satisfactoria

**Caso de Uso: 20) Registrado entrega fallida****Actors:** Depósito**Pre:** La entrega del pedido fue fallida**Post:** El pedido está anulado y los productos aprobados fueron reingresados a stock**Curso Normal**

1. La mercadería en buen estado es reingresada a stock
2. Depósito carga la falta del cliente y el costo generado a la empresa
3. El pedido es anulado.
4. El sistema genera un mensaje conteniendo una invitación a rehacer el pedido, y un link hacia una orden de compra con el mismo carrito del pedido anulado
5. Incluye CU 4: Enviando mail.
6. FIN CU

**Curso Alternativo**

Caso de uso 20: Registrado entrega fallida

**Caso de Uso: 21) Cargando pedido a proveedor****Actors:** Departamento de Stock**Pre:** Se hizo un pedido a un proveedor**Post:** El pedido está cargado**Curso Normal**

1. El Departamento de Stock carga la información del pedido y su comprobante al sistema.
2. FIN CU

**Curso Alternativo**

Caso de uso 21: Cargando pedido a proveedor

**Caso de Uso: 22) Cargando recepción de pedido a proveedor****Actors:** Depósito**Pre:** Llegó un pedido al depósito**Post:** Los nuevos productos son ingresados a stock**Curso Normal**

1. Depósito registra el ingreso.
2. El sistema actualiza el stock de los nuevos productos
3. FIN CU

**Curso Alternativo**

Caso de uso 22: Cargando recepción de pedido a proveedor

**Caso de Uso: 23) Informando preparación de stock para reposición****Actors:** Depósito**Pre:** Hay un pedido de reposición de la sucursal**Post:** Los productos requeridos son restados del stock**Curso Normal**

1. Depósito informa al sistema que el pedido de reposición se encuentra en preparación.
2. El sistema resta del stock las unidades correspondientes al pedido.
3. El sistema le brinda una respuesta al depósito, ofreciendo la descarga de una planilla que contiene un sumario de las unidades que deberán ser enviadas a la sucursal, junto con la información interna que permita agilizar el proceso de preparación (localización dentro del depósito, números de empaque, etcétera).
4. El Depósito informa la correcta preparación del pedido.
5. El sistema marca el pedido preparado, y pone a disposición del Depósito una hoja de ruta / remito de traslado, para su impresión y futura utilización.
6. FIN CU

**Curso Alternativo**

Caso de uso 23: Informando preparación de stock para reposición

**Caso de Uso: 24) Registrando entrega de reposición****Actors:** Sucursal**Pre:** Un envío de reposición a sucursal fue entregado correctamente**Post:** La llegada del envío es registrada**Curso Normal****Curso Alternativo**

1. La sucursal ingresa al sistema, y marca el pedido de reposición como entregado.
2. FIN CU.

Caso de uso 24: Registrando entrega de reposición

**Caso de Uso: 25) Definiendo umbral de redituabilidad****Actors:** Administrador**Pre:** -**Post:** Se redefine el umbral de redituabilidad**Curso Normal****Curso Alternativo**

1. El administrador ingresa el nuevo umbral de redituabilidad
2. El sistema guarda el nuevo umbral
3. FIN CU.

Caso de uso 25: Definiendo umbral de redituabilidad

**Caso de Uso: 26) Dando de alta producto****Actors:** Administrador**Pre:** -**Post:** El producto solicitado es dado de alta**Curso Normal****Curso Alternativo**

1. El administrador ingresa al ABM provisto por el sistema
2. El administrador ingresa el producto
3. El sistema da de alta el producto y comunica la operación
4. FIN CU.

Caso de uso 26: Dando de alta producto

**Caso de Uso: 27) Dando de baja producto****Actors:** Administrador**Pre:** -**Post:** El producto solicitado es dado de baja**Curso Normal**

1. El administrador ingresa al ABM provisto por el sistema
2. El administrador ingresa el producto
3. El sistema da de baja el producto y comunica la operación
4. FIN CU.

**Curso Alternativo**

Caso de uso 27: Dando de baja producto

**Caso de Uso: 28) Modificando producto****Actors:** Administrador**Pre:** -**Post:** El producto solicitado es modificado**Curso Normal**

1. El administrador ingresa al ABM provisto por el sistema
2. El administrador ingresa el producto y la modificación
3. El sistema realiza los cambios solicitados y confirma la operación
4. FIN CU.

**Curso Alternativo**

Caso de uso 28: Modificando producto

**Caso de Uso: 29) Obteniendo estadísticas****Actors:** Administrador**Pre:** -**Post:** Se envían las estadísticas al administrador**Curso Normal**

1. El administrador solicita las estadísticas a través del sistema
2. El sistema genera las estadísticas de venta de cada producto y compras de cada usuario, y las prepara para su adecuada visualización
3. El administrador descarga las estadísticas a través del sistema
4. Si administrador desea modificar algún producto, es extendido por CU 28: Modificando producto.
5. FIN CU.

**Curso Alternativo**

Caso de uso 29: Obteniendo estadísticas

**Caso de Uso: 30) Encargando reposición****Actors:** Sucursal**Pre:** La sucursal se conecta desde una terminal especial pre-autenticada.**Post:** Los productos deseados fueron encargados**Curso Normal**

1. La sucursal ingresa el listado de productos que desea encargar.
2. El sistema guarda el pedido como pendiente de preparación.
3. El sistema le confirma a la sucursal que el pedido fue encargado.
4. FIN CU.

**Curso Alternativo**

Caso de uso 30: Encargando reposición

**Caso de Uso: 31) Cargando datos semanales de venta****Actors:** Sucursal**Pre:** La sucursal tiene datos de venta para informar**Post:** El sistema contiene datos de venta actualizados**Curso Normal**

1. La sucursal carga los datos de venta de la semana a través de una interfaz sencilla, por ejemplo una planilla de cálculos.
2. El sistema procesa los datos de venta, y los integra a su base de estadísticas
3. FIN CU.

**Curso Alternativo**

Caso de uso 31: Cargando datos semanales de venta

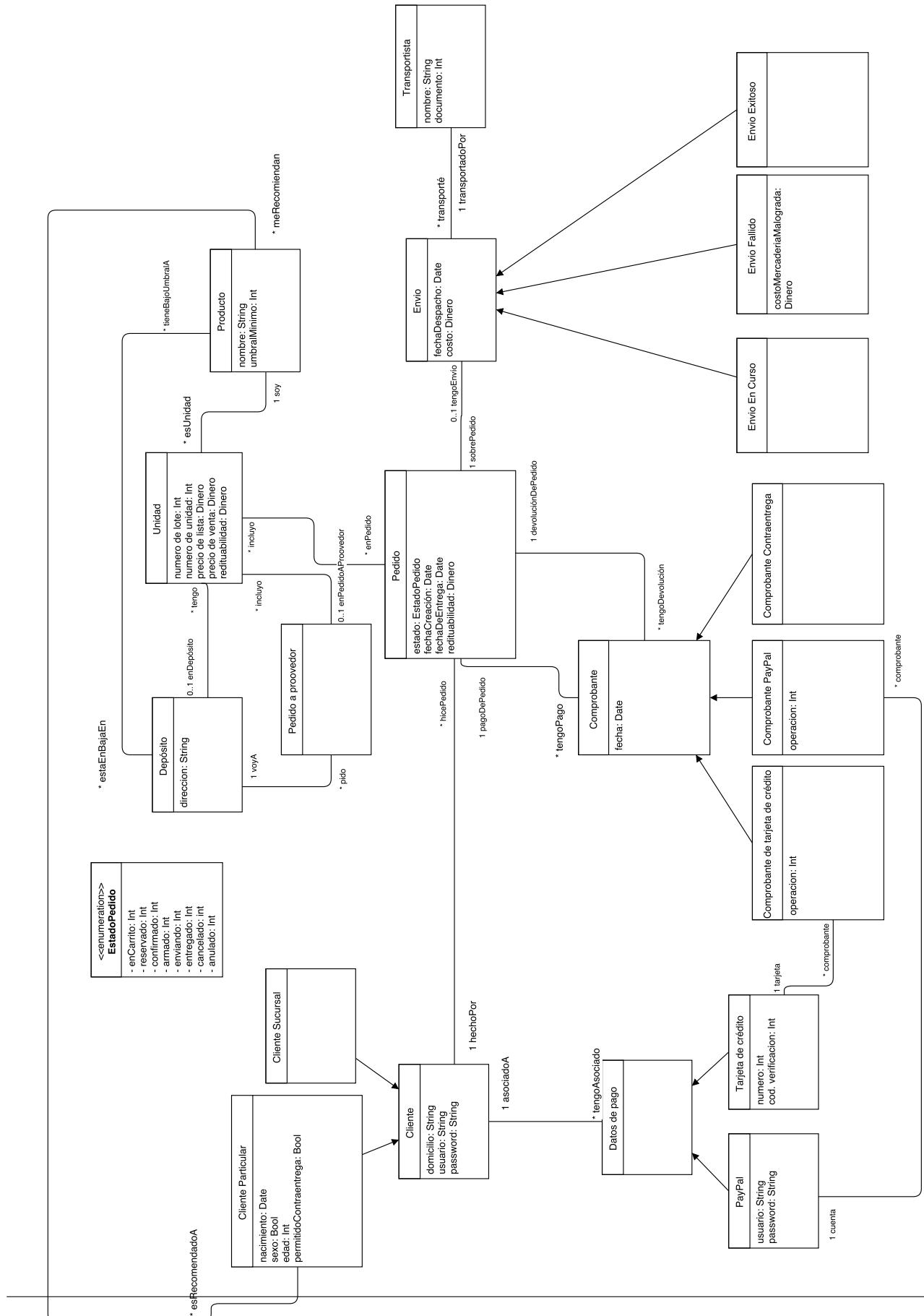
**Caso de Uso: 32) Consultando pedido pendiente****Actors:** Depósito**Pre:** El depósito se conecta desde una terminal especial pre-autenticada.**Post:** El sistema entrega la información del pedido consultado**Curso Normal**

1. Un empleado del depósito ingresa al sistema.
2. El sistema muestra la lista de pedidos pendientes.
3. El empleado selecciona un pedido y el sistema muestra el detalle.
4. FIN CU

**Curso Alternativo**

Caso de uso 32: Consultando pedido pendiente

## Parte VIII. Modelo conceptual



## 8.1 OCL

1. La redituabilidad de una unidad es su precio de venta menos su precio de lista.

```
Context: Unidad
Inv: self.redituabilidad = self.precioDeVenta - self.precioDeLista
```

2. La redituabilidad de un pedido es la ganancia menos el costo.

```
Context: Pedido
```

```
ganancia(p) = if p.estadoPedido == Concretado
    then p.unidades->collect(redituabilidad)->sum()
    else 0
    endif
costo(p) = if p.envio->size() == 1
    then p.envio.costo
    else 0
    endif
+ if p.envio->size() == 1 and p.envio.ocliIsTypeOf(EnvioFallido)
    then p.envio.costoMercaderiaMalograda
    else 0
    endif
Inv: self.redituabilidad = ganancia(self) - costo(self)
```

3. Se le permite la contraentrega si y solo si la redituabilidad del cliente es mayor al umbral de redituabilidad.

```
Context: Cliente Particular
```

```
redituabilidadCliente(c) = c.pedidos->collect(redituabilidad)->sum()
Inv: self.permitidoContraentrega = redituabilidadCliente(self) > -
GLOBAL_VARIABLE(umbralRedituabilidad)
```

4. Un cliente no puede tener dos pedidos vigentes.

```
Context: Cliente Particular
Inv: self.pedidos->filter(x | {entregado, anulado, -
cancelado})->excludes(x.estado)).size() < 2
```

5. Un Producto está bajo en stock en un Depósito si y sólo sí la cantidad de unidades de dicho producto en el Depósito está por debajo del umbral mínimo de dicho producto.

```
Context: Producto
Inv: Deposito.allInstances()->forall(deposito |
    (self.estaEnBaja->filter(deposito2 | deposito = deposito2).size() -
    = 1) =
    (deposito.tengo->filter(unidad | unidad.soy = self).size() < -
    self.umbralMinimo)
)
```

6. Para cada pedido,  $fechaCreacion < fechaDespacho < fechaEntrega$

```
Context: Envio
Inv: self.sobrePedido.fechaCreacion < self.fechaDespacho < -
      self.sobrePedido.fechaEntrega
```

7. Una unidad está en solo uno de los siguientes lugares:

pedidoAProveedor

depósito

depósito y en pedido en estado enCarrito, anulado, cancelado

en pedido confirmado

```
Context: Unidad
Inv:
(self.enPedidoAProveedor->notEmpty() and self.enDeposito->isEmpty() -
 and self.enPedido->isEmpty())
or
(self.enDeposito->notEmpty() and self.enPedidoAProveedor->isEmpty() -
 and self.enPedido->isEmpty())
or
(self.enDeposito->notEmpty() and self.enPedidoAProveedor->isEmpty() -
 and self.enPedido->notEmpty() and {enCarrito, cancelado, -
 anulado}->includes(self.enPedido.estado))
or
(self.enDeposito->isEmpty() and self.enPedidoAProveedor->isEmpty() and -
 self.enPedido->filter(p | {enCarrito, cancelado, -
 anulado}->excludes(p.estado)).size() == 1)
```

8. Un cliente C tiene un producto recomendado P si y solo si:

C nunca compro P y

P pertenece a la lista de 10 más comprados de algún cliente parecido a C

```
Context: Cliente Particular
edadParecida(cliente1, cliente2) = |cliente1.edad - cliente2.edad| < 4

productosComprados(cliente) = cliente.hicePedido.incluyo.soy

cantUnidadesCompradas(producto, cliente) = -
productosComprados(cliente)->filter(p | p = producto)->size()

cantUnidadesCompradasOrdenadas(cliente) = -
productosComprados(cliente)->asSet()->sortedBy(p | -
cantUnidadesCompradas(p, cliente))->reverse()

diezMasComprados(cliente) = -
cantUnidadesCompradasOrdenadas(cliente)->subSequence(1, 10)
```

```

clienteParecido(cliente1, cliente2) = edadParecida(cliente1, -
cliente2) and intersection(diezMasComprados(cliente1), -
diezMasComprados(c2))->size() >= 5

Inv: self.meRecomiendan->forAll(producto | -
productosComprados(self)->excludes(producto)
and ClienteParticular.allInstances()->exists(c | -
clienteParecido(self, c) and diezMasComprados(c)->includes(p))
)

```

9. Comprobantes deben tener asociados datos de pago registrados por el cliente.

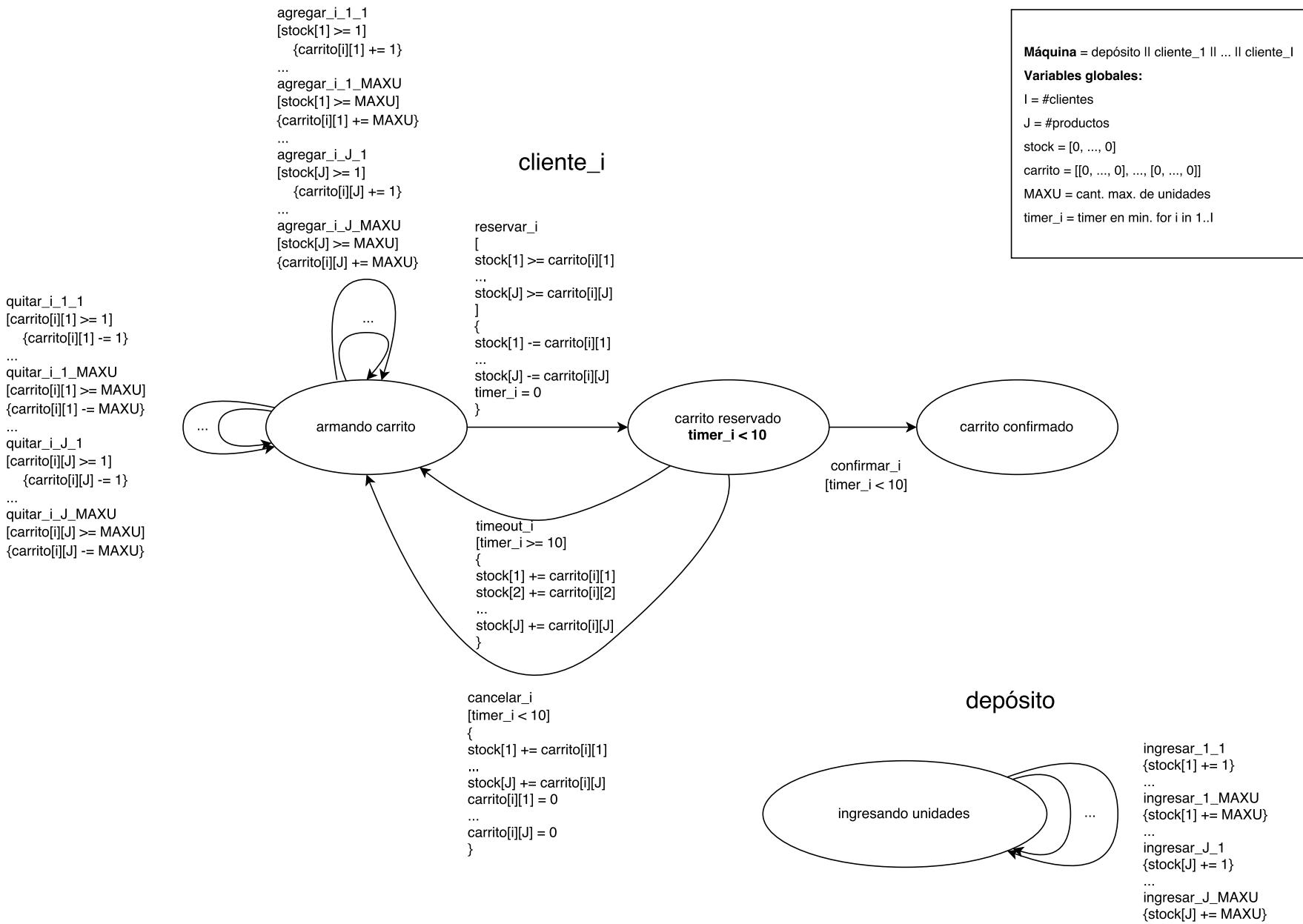
Observación: la tarjeta puede tener como titular a un familiar, pero quien la registra es el cliente.

```

Context: Comprobante de tarjeta de credito
Inv: self.tarjeta.asociadoA = self.pagoDePedido.hechoPor
Context: Comprobante PayPal
Inv: self.cuenta.asociadoA = self.pagoDePedido.hechoPor

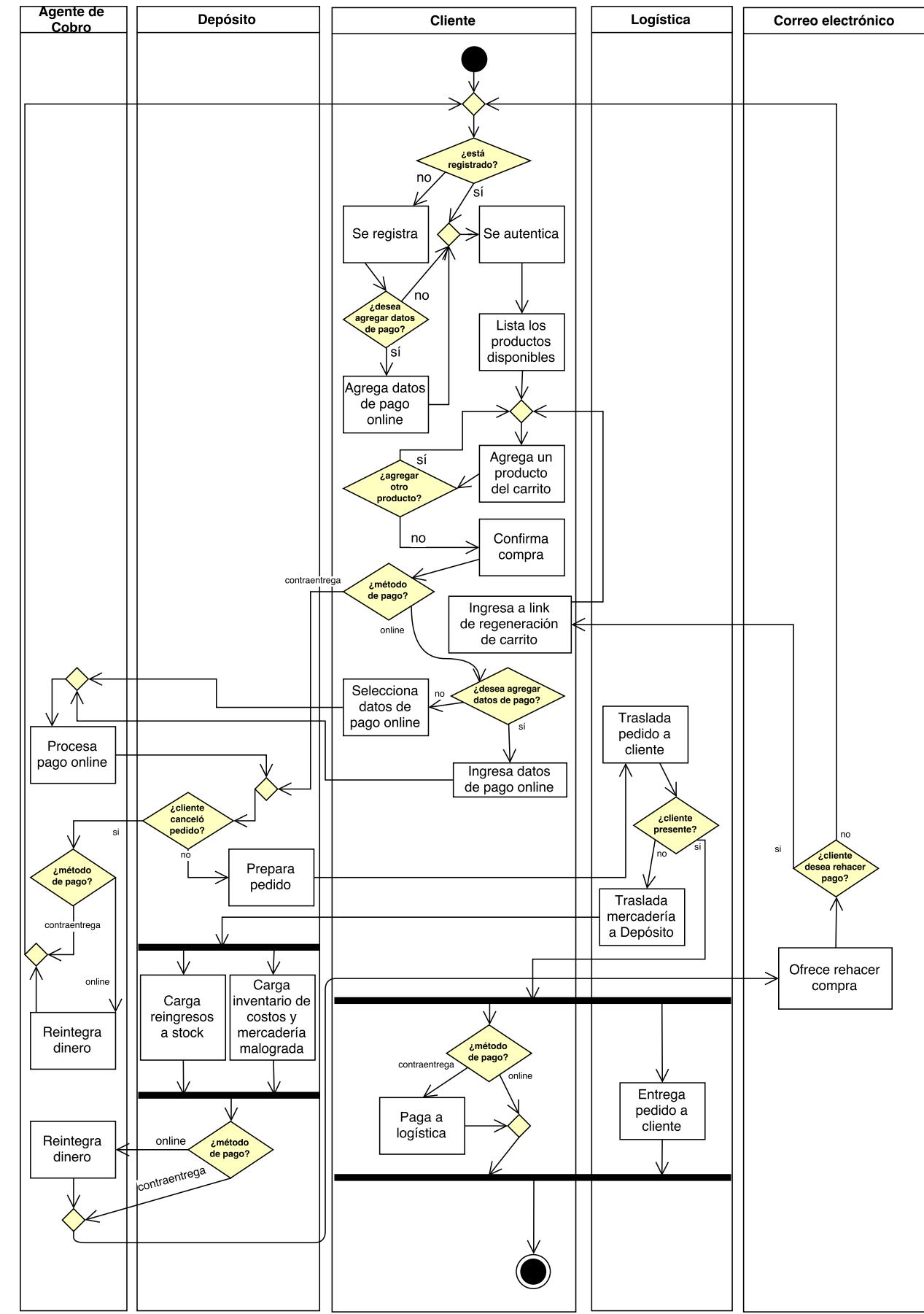
```

## Parte IX. Máquinas de estados finitos





## Parte X. Diagrama de actividad



## Parte XI. Conclusión TP2

Este trabajo nos permitió explorar tanto el problema planteado como la solución propuesta desde otros ángulos. Trabajar con una nueva herramienta de modelado nos forzó a dejar explícito cierto funcionamiento que quizás no había sido expuesto en el trabajo original. También nos obligó a repensar algunos de los planteos originales, debiendo ahondar fuertemente en sus detalles funcionales, y en ocasiones incluso modificarlos.

Fue necesario un intenso nivel de colaboración y coordinación entre los miembros del grupo. Al idear un modelo en particular, fue de vital importancia para cada uno de los integrantes tener en mente el modelo global —implícito, integrado por todos los modelos— y ser muy cuidadosos al modificarlo, ya que esto podía causar incoherencias en los otros modelos, atentando contra la trazabilidad de los mismos.