

DEPARTAMENTO
DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico 2

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

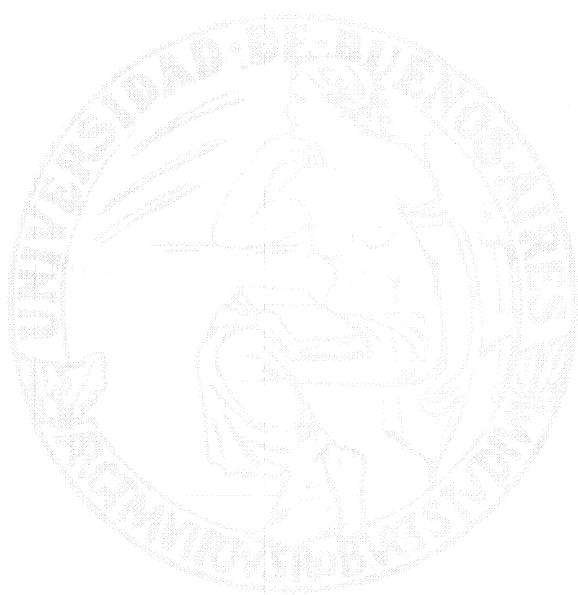
1^{er} cuatrimestre de 2015

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

REENTREGAR
CON MCYOLI PSM Y DA.
AGREGAR DIÁGRAMA DE CONTEXTO
DEL TPI
EXPLICAR LA INTERRELACIÓN ENTRE
DIÁGRAMAS.



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>

HAY UNA INTERSECCIÓN REAL CON EL SISTEMA. LO QUE SE PODEMOS VER A MEDIOS.

- USO DE EXTENDE EN AUTENTICARSE FUERA AL SISTEMA AL ALIGERAR PROBLEMAS EN EL RESTO DE LAS PERSONAS PARCIALES NO ESTAR AUTENTICADAS.
ES RECOMENDABLE USAR AUTENTICADO EN LA P.E.
- ERRORES VARIOS EN EL USO DE CO - EN INTERSECCIONES QUE SON COMpletamente FUERA DEL SISTEMA.
EJ: "COGRANDO CONTRAENTREGA".

DA + CO:

- SI NECESITAN PUEDES HACER MÁS DIAGRAMAS DE ACTIVIDAD.
NO HACE FALTA QUE SEA SOLO UNO.
- NUMERAR LOS CO Y REFERENCIARLOS EN EL DA (TRAZABILIDAD)

MCYOCL:

- VER CORRECCIONES EN EL DIAGRAMA.
- UNIFICAR SINTAXIS DE MERCOS
- CORREGIR LOS INVARIANTES.

Esta página fue dejada en blanco intencionalmente.

FSM = VER CORRECCIONES EN DIAGRAMA.

- HAY DEADLOCK!
- ME SUSPIRA VER MÁS INTERSECCIÓN ENTRE MÁQUINAS.

GENERAL:

- AGREGAR INFORMACIÓN (TEXTO/TABLAS) QUE CONECTE LOS DIAGRAMAS DANDOLE TRAZABILIDAD AL TP.
- EN LA REPORTECHA INCLUYAN EL TP1 CORREGIDO, ~~PERO~~
- PODRÍAN ENUMERAR LOS REDONDRIMIENTOS DEL TP1 Y MOSTRAR EN QUÉ DIAGRAMAS LOS COMPLEJOS

Contents

1	Introducción	2
2	Casos de Uso	4
3	Modelo conceptual	18
3.1	OCL	19
4	Máquinas de estados finitos	23
5	Diagrama de actividad	25
6	Conclusión	26

Parte I. Introducción

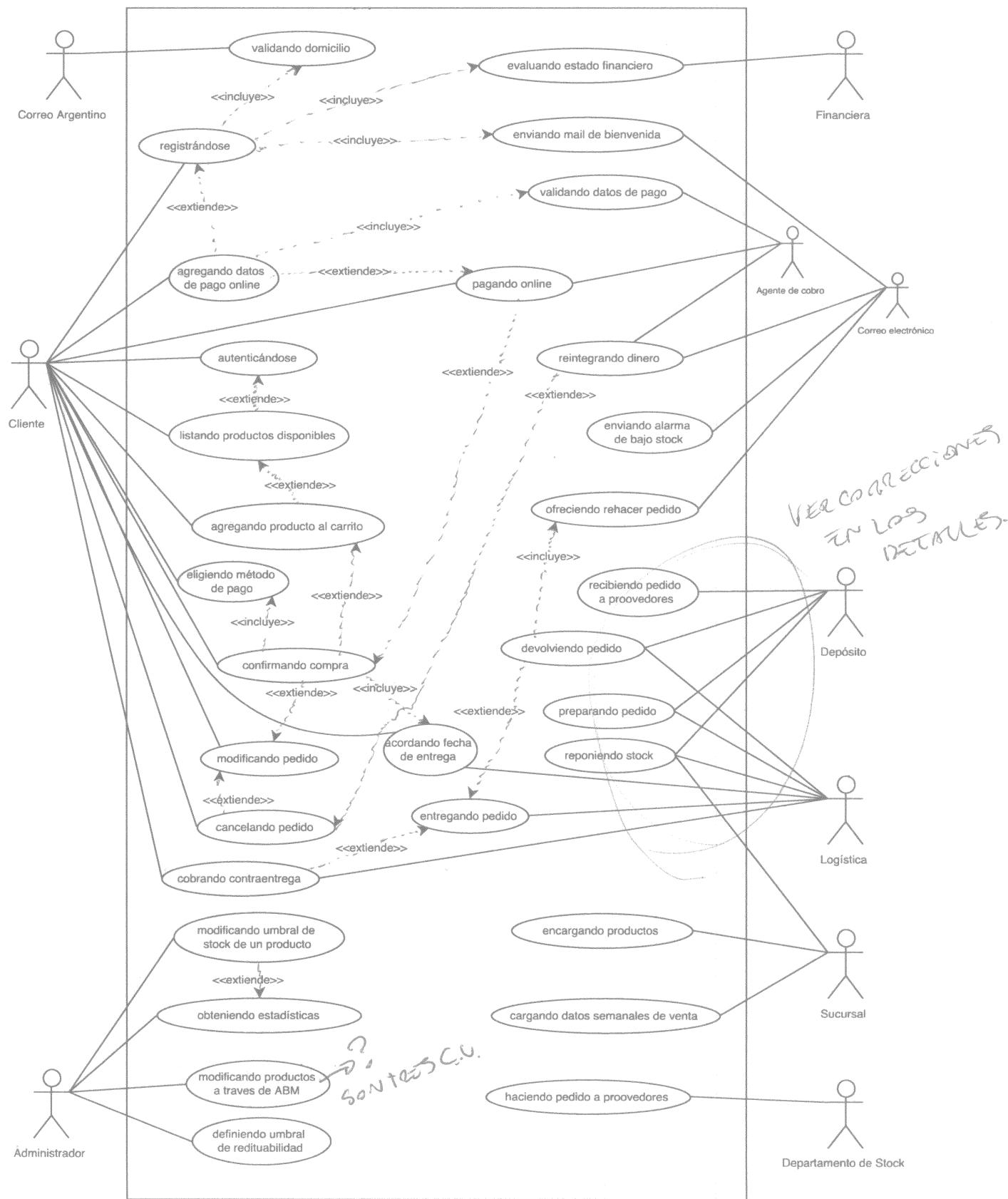
El presente Trabajo Práctico es una extensión del TP1: Mes%. Utilizando como base la temática y los modelos descriptos en el anterior, aplicamos diversas técnicas para documentar distintos aspectos del proyecto. Realizamos un Modelo Conceptual, abarcando todos los aspectos del sistema, para describir 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 II. Casos de Uso



Caso de Uso: Registrándose**Actors:** Cliente**Pre:** -**Post:** El cliente se encuentra registrado**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: evaluando estado financiero
6. El cliente ingresa su dirección
7. Incluye CU: validando domicilio
8. Si el cliente desea: es extendido por CU: Agregando datos de pago online
9. El cliente ingresa su mail
10. Incluye CU: enviando mail de bienvenida
11. El cliente ingresa al link de bienvenida
12. El sistema marca al usuario como validado
13. 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 e ir a 6.

- 11.1 Si luego de 10 días el usuario no ingresa al link de bienvenida, el registro se cancela, FIN CU.

este
y nos
nos

de este caso

de uso, serán
un C.U. aparte que puede hacerse en
otro momento. Por ej. Confirmación de Mail

Caso de uso 1: Registrándose

ESTO ES ALGO
INTERNO AL SISTEMA
QUE PODRÍAN
HACERLO MEDIADO
CON BA OFSM.

Caso de Uso: 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: Enviando mail de bienvenida**Actors:** Correo electrónico**Pre:** -**Post:** -**Curso Normal**

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

Curso Alternativo**Caso de uso 3: Enviando mail de bienvenida****Caso de Uso: 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 4: Autenticándose

*Si en el diagrama pones que extiende esto debe figurar en la descripción de consistencia.
Por eso les comentaba que su caso de uso quedaría raro, pues el paso 4*

Caso de Uso: Listando productos disponibles**Actors:** Cliente**Pre:** El cliente está autenticado**Post:** -**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. Si el cliente lo desea, se extiende con CU: Agrégando producto al carrito
6. FIN CU

Curso Alternativo

*Diálogos si el usuario desea extenderse a listado de productos disponibles.
Además de esta forma parecerá que si uno quisiera volver a listar products debiera volver a loguearse.*

Caso de uso 5: Listando productos disponibles

Caso de Uso: Agregando producto al carrito**Actors:** Cliente**Pre:** El cliente está autenticado**Post:** El carrito contiene al producto agregado**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. Si el cliente lo desea, es extendido por CU: confirmando compra
6. FIN CU

Curso Alternativo**Caso de uso 6: Agregando producto al carrito****Caso de Uso: Confirmando compra****Actors:** Cliente**Pre:** El cliente tiene un carrito armado**Post:** El cliente tiene un carrito reservado y confirmado**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: Calcular costo de envío → *NO ES UN CU DE SUGESTIÓN*
3. El sistema informa del costo total de la compra, incluyendo el envío.
4. Incluye: Acordando fecha de entrega
5. Incluye: Eligiendo método de pago → *NO PROCESAR UN CU EN SI MISMO
ES MÁS UN PAGO DE ESTE CU.*
6. Si el pago es online: Es extendido por: Pagando online.
7. El cliente confirma el pedido, 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 cliente no confirma el pedido luego de 10 minutos, los productos son reingresados a stock, y el carrito deja de estar reservado, vuelve a paso 1

El cliente podría no reservar el pedido luego de haber pagado online?

Caso de uso 7: Confirmando compra

Caso de Uso: Acordando fecha de entrega

Actors: Cliente, 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 8: Acordando fecha de entrega

Caso de Uso: Eligiendo método de pago

Actors: Cliente

Pre: El cliente tiene un carrito reservado

Post: El cliente eligió un método de pago permitido

Curso Normal

1. El sistema determina si el cliente tiene autorizado el pago contraentrega
2. El sistema presenta los métodos de pagos disponibles
3. El cliente elige el método de pago deseado
4. FIN CU

Curso Alternativo

Caso de uso 9: Eligiendo método de pago

Caso de Uso: Pagando online**Actors:** Cliente, agente de cobro**Pre:** El cliente eligió el método de pago online**Post:** -**Curso Normal**

1. Si el cliente lo desea, es extendido por CU: 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 envía un mail al cliente, informando la compra exitosa y adjuntando un comprobante de pago de la operación a través del sistema de correo electrónico.
8. 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

no figura
el agente
de pago
de ningún

como Actor .

→ Caso de uso 10: Pagando online
 El cliente está bien
 que el sistema informe al cliente del pago exitoso
 pero esperaría que en cualquier compra sucediera esto.

Caso de Uso: 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. FIN CU

Curso Alternativo

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

*El sistema
regresa
al punto
anterior*

Caso de uso 11: Agregando datos de pago online

Caso de Uso: Cancelando pedido

Actors: Cliente

Pre: El cliente tiene un pedido sin confirmar, o confirmado, pero sin armar en depósito.

Post: El pedido es cancelado.

Curso Normal

1. Si el pedido está sin confirmar, lo cancela, [FIN CU]
2. La reserva de productos se anula, y los mismos se reingresan a stock
3. Si fue pagado, es extendido por CU: Reintegrando dinero
4. FIN CU

Curso Alternativo

¿Qué sucede si el cliente no confirma? Porque va directo a FIN CU?

Caso de uso 12: Cancelando pedido

Caso de Uso: Modificando pedido

Actors: Cliente

Pre: El cliente tiene un pedido sin confirmar, o confirmado, pero sin armar en depósito.

Post: El pedido es modificado.

Curso Normal

1. El cliente quita o agrega los productos que desee, siempre y cuando estén en stock.
2. Si el pedido está sin confirmar, se registra la modificación, [FIN CU].
3. Se cancela el pedido anterior: es extendido por CU Cancelando Pedido.
4. Se genera un nuevo pedido con los datos modificados.
5. Es extendido por CU: Confirmando compra.
6. FIN CU

Curso Alternativo

Idem... Noticias no me quedan
que quisieron expresar.
No hay que reintegrar dinero
en caso que hayan sacado productos
del carrito?

Caso de uso 13: Modificando pedido

Caso de Uso: 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 contrasta los datos de pago del cliente contra el agente de cobro correspondiente a través de una API.
2. FIN CU

Curso Alternativo

1. El sistema envía los datos.
2. El agente de cobro envía un mensaje informando que los datos son correctos.

Caso de uso 14: Validando datos de pago

Caso de Uso: Reintegrandó dinero**Actors:** Agente de cobro, Correo electrónico**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 envía un mail al cliente, informando que el pedido fue anulado, a través del sistema de correo electrónico, adjuntando los comprobantes de pago
4. 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.

Caso de uso 15: Reintegrandó dinero**Caso de Uso: Validando domicilio****Actors:** Correo argentino**Pre:** El cliente ingresó los datos de su domicilio**Post:** Los datos de su domicilio fueron validados**Curso Normal**

1. El sistema verifica los datos de domicilio del cliente mediante la API del Correo Argentino
2. FIN CU

Curso Alternativo

- 2.1 Si los datos de domicilio no son correctos, se marcan como inválidos. FIN CU

Caso de uso 16: Validando domicilio

Caso de Uso: Preparando pedido

Actors: Depósito, Logística

Pre: El cliente tiene un pedido confirmado

Post: El cliente tiene un pedido preparado

Curso Normal

1. El sistema brinda el listado de productos a preparar, junto con el domicilio y la fecha y hora de entrega

2. Un operario del depósito marca el pedido como preparado y se encarga de empaquetar los productos

3. Depósito informa del pedido a logística, y se pacta el envío para la fecha correspondiente; los datos de la operación se cargan en el sistema

4. La información correspondiente al envío: remito, hoja de ruta, factura, etcétera, se empaqueta junto con el mismo.

5. FIN CU

Curso Alternativo

Logística no participa en la interacción con el sistema.
Eso podrísimos hacerlo con DA
fuera del sistema.

Caso de uso 17: Preparando pedido

Caso de Uso: Entregando pedido

Actors: Logística, Depósito ?

Pre: El cliente tiene un pedido preparado

Post: Se realiza un intento de entrega

Curso Normal

1. Logística retira el pedido.

2. Logística entrega el pedido al cliente.

3. Si el pago es contraentrega, es extendido por CU: Cobrando contraentrega

4. Logística registra la entrega al cliente satisfactoria

5. FIN CU.

Curso Alternativo

2.1 Si el cliente no recibe el pedido, es extendido por CU: Devolviendo pedido.

Caso de uso 18: Entregando pedido

Caso de Uso: Cobrando contraentrega

Actors: Logística, Cliente

Pre: El cliente eligió contraentrega como método de pago

Post: El pago del pedido fue efectuado

Curso Normal

1. El transportista presenta la factura de compra

2. El cliente entrega el dinero correspondiente

3. FIN CU

Curso Alternativo

X No es un caso de uso! Es una
interacción por
fuera del sistema.

Caso de uso 19: Cobrando contraentrega

Caso de Uso: Devolviendo pedido**Actors:** Logística, Depósito**Pre:** La entrega del pedido fue fallida**Post:** El pedido está anulado y los productos aprobados fueron reingresados a stock*Los Clientes NO PARTICIPA DEL C.U.***Curso Normal**

1. Logística devuelve stock al depósito
2. Encargados del depósito hacen un inventario de la mercadería malograda
3. La mercadería en buen estado es reingresada a stock
4. Depósito carga la falta del cliente y el costo generado a la empresa
5. El pedido es anulado.
6. Incluye: Ofreciendo rehacer pedido.
7. FIN CU

Curso Alternativo*→ FUERZA DEL SISTEMA***Caso de uso 20: Devolviendo pedido****Caso de Uso: Ofreciendo rehacer pedido****Actors:** Correo electrónico**Pre:** El cliente tiene un pedido anulado.**Post:** Se le ofrece al cliente rehacer el pedido.**Curso Normal**

1. El sistema genera un link hacia una orden de compra con el mismo carrito del pedido anulado
2. El sistema de correo electrónico envía un mail al cliente con el link y una invitación a rehacer el pedido
3. FIN CU

Curso Alternativo**Caso de uso 21: Ofreciendo rehacer pedido****Caso de Uso: Envío alarma de bajo stock****Actors:** Correo electrónico**Pre:** Existe al menos un producto con stock menor al límite estipulado**Post:** Se avisa de la falta de stock al Departamento de Stock**Curso Normal**

1. El sistema prepara una lista de todos los productos por debajo del límite estipulado y el stock necesario para re establecerlos
2. El sistema de correo electrónico envía un mail con los datos al Departamento de Stock
3. FIN CU

Curso Alternativo**Caso de uso 22: Envío alarma de bajo stock**

Caso de Uso: Haciendo pedido a proveedores

Actors: Departamento de stock

Pre: Una alarma de bajo stock fue activada

Post: Se realiza un nuevo pedido a un proveedor

Curso Normal

1. El Departamento de Stock determina al mejor proveedor y efectúa una compra
2. El Departamento de Stock carga el comprobante de pedido al sistema
3. FIN CU

Curso Alternativo

ME IMPRESIONA QUE ADENDESS DE LAS COMPROBANTES CARGARAN LA INFO DEL PEDIDO.

Caso de uso 23: Haciendo pedido a proveedores

Caso de Uso: Recibiendo pedido a proveedores

Actors: Depósito

Pre: -

Post: La mercadería es ingresada al depósito

Curso Normal

1. Empleados del depósito guardan la mercadería
2. Depósito registra el ingreso y el sistema actualiza el stock de los nuevos productos
3. FIN CU

Curso Alternativo

FUERA DEL SISTEMA...

Caso de uso 24: Recibiendo pedido a proveedores

Caso de Uso: Reponiendo stock

Actors: Depósito, Logística, Sucursal

Pre: Hay un pedido de reposición de la sucursal

Post: Los productos requeridos son ingresados a la sucursal

Curso Normal

1. Depósito reserva todo el stock necesario para completar el pedido, o el máximo posible en caso de que la reserva de stock sea menor a esta cantidad, informando la baja de los productos correspondientes.
2. Un empleado del depósito empaqueta los productos reservados y avisa a logística.
3. Depósito entrega el pedido a logística junto con una hoja de ruta
4. Logística transporta el envío hasta la sucursal
5. La sucursal repone los productos en las góndolas
6. FIN CU

Curso Alternativo

- 1.1 Si no se ha reservado el stock necesario para el pedido, se espera a que llegue una reposición y se vuelve a 1

HAY ALGUNA INTERACCION
INTERNA CON EL
SISTEMA?

REPOSI

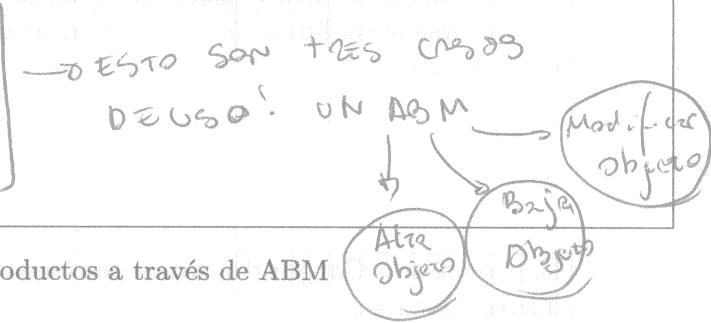
Caso de uso 25: Reponiendo stock

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

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

Curso Alternativo**Caso de uso 26: Definiendo umbral de redituabilidad****Caso de Uso: Modificando productos a través de ABM****Actors:** Administrador**Pre:** -**Post:** Los productos solicitados son modificados**Curso Normal**

1. El sistema muestra una interfaz para agregar, dar de baja, y modificar productos
2. El administrador ingresa la operación requerida
3. El sistema realiza los cambios necesarios y confirma la operación
4. FIN CU

Curso Alternativo**Caso de uso 27: Modificando productos a través de ABM****Caso de Uso: 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 límite de productos, es extendido por CU: Modificando umbral de stock de un producto.
5. FIN CU.

Curso Alternativo

→ NO NECESITA TENER NUEVOS
SENTIDO QUE ESTE
CUOSO DE USO SE TIENDE
DEL OTRO.

Caso de uso 28: Obteniendo estadísticas

Caso de Uso: Modificando umbral de stock de un producto

Actors: Administrador

Pre: El producto existe

Post: El producto tiene un nuevo umbral de stock

Curso Normal

1. El administrador informa del nuevo límite para el producto
2. FIN CU

Curso Alternativo

→ si es de un producto en particular one imagino que esto debería tener que ver con un listado de productos.

Caso de uso 29: Modificando umbral de stock de un producto

Caso de Uso: Encargando productos

Actors: Sucursal, Depósito

Pre: La sucursal está autenticada, desea encargar productos

Post: Los productos deseados fueron encargados

Curso Normal

1. La sucursal ingresa el listado de productos que desea encargar.
2. El sistema informa a depósito del pedido.
3. El sistema le confirma a la sucursal que el pedido fue encargado.
4. FIN CU

Curso Alternativo

→ NO ESTÁ DEFINIDO EL AUTENTICAR.

¿Cómo SELECCIONA EL PRODUCTO EL ADMIN?

→ ¿QUÉ SIGNIFICA ESTO?
NO PARECE QUE EL DEPÓSITO DEBIERA

Caso de uso 30: Encargando productos

Caso de Uso: 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

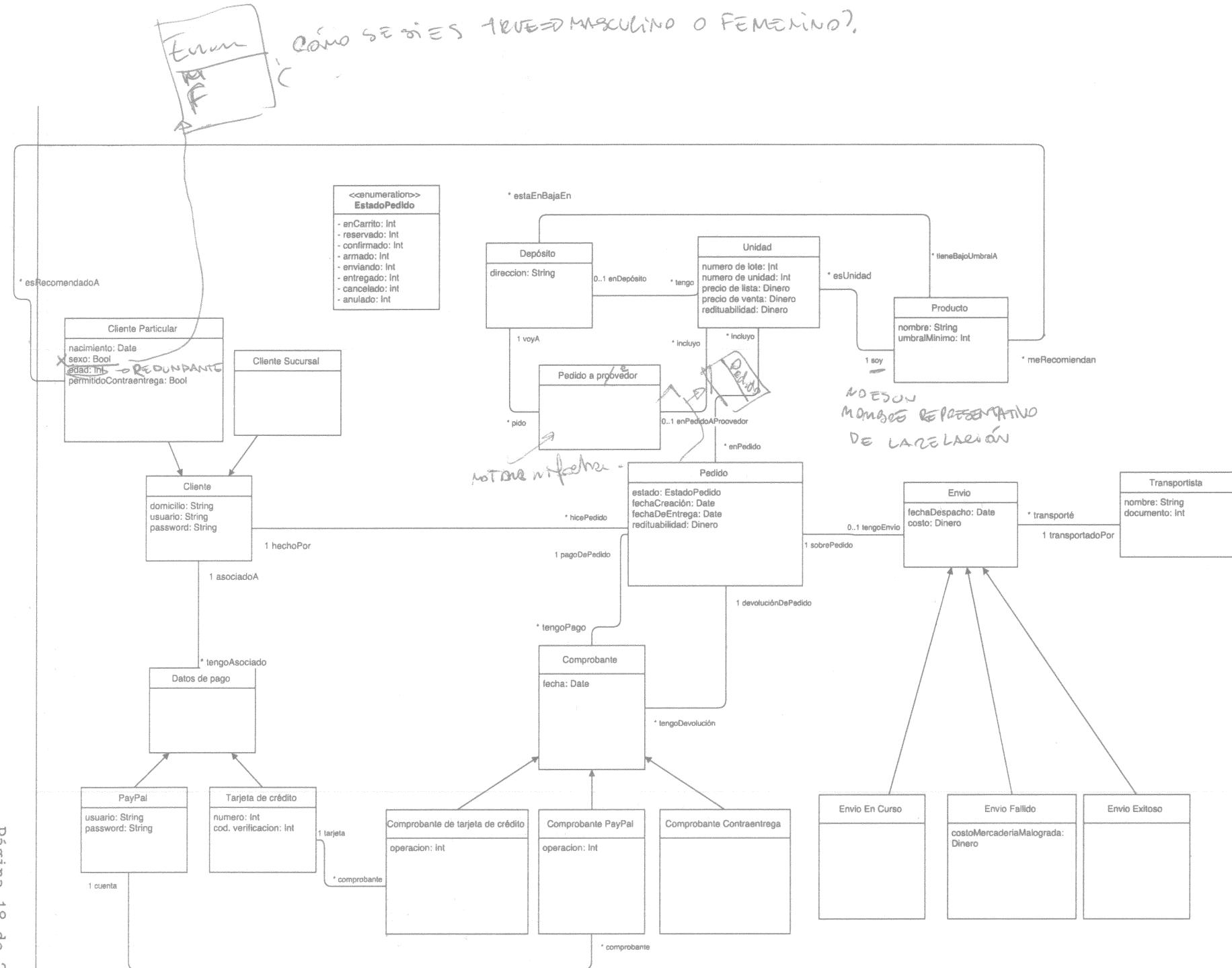
Curso Alternativo

PARTICIPAR DE ESTE C.U.
AL VECES SEAN DROGADOS.

Caso de uso 31: Cargando datos semanales de venta

Nombre	Apellido	Nota
Diego	Barrios	9.5
Facundo	Benegas	8.8
Agustín	Di Alessio	9.2
Leandro	Rodríguez	8.5
Patricia	Vanecek	9.0

Parte III. Modelo conceptual



3.1 OCL

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

Context: Unidad

Inv: self.redituabilidad = self.precioDeVenta - self.precioDeLista

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

Context: Pedido

```

ganancia()= if self.estadoPedido == Concretado
    then self.unidades->collect(redituabilidad)->sum()
else 0
endif

costo() if! self.envio->empty() & self.envio.size() == 1
    then self.envio.costo
else 0
endif

+ if self.envio.oclisTypeOf(EnvioFallido)
    then self.envio.costoMercaderiaRechazada
else 0
endif

Inv: self.redituabilidad = ganancia - costo

```

este atributo no existe

con la forma que cumple los

maestros.

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

Context: Cliente Particular

ESTRAS
QUE EL UMbral
redituabilidadCliente = self_pedidos->collect(redituabilidad)->sum()
se mire
Inv: permitidoContraentrega = redituabilidadCliente >
EN redituabilidad
umbralRedituabilidad → si es una constante desigualdad o hacen mención de
ACUMULADAS → se cumple la condición de que el cliente tiene una redituabilidad menor que el umbral
pero EN SI
4. Un cliente no puede tener dos pedidos vigentes.

Context: Cliente Particular

Inv: self_pedidos->filter(x | x.estado != entregado and x.estado != anulado and x.estado != cancelado).size() < 2

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

Context: Producto

Inv: self.estaNebajaEn->forall(d | d.tengo->filter(u | u.soy = self).size() < self.umbralMinimo)

Este No
implica que
si esté por
debajo del
umbral
pertenece a la colección. De hecho si está en la colección es valido para todo P.
El resultado es true.
Aunque la condición
no se cumpla

- Para cada pedido, fechaCreacion < fechaDespacho < fechaEntrega

Context: Envio

Inv: self.sobrePedido.fechaCreacion < self.fechaDespacho <
self.sobrePedido.fechaEntrega

o es un Duplicado

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

pedidoAProovedor

depósito

depósito y en pedido en estado enCarrito, anulado, cancelado

en pedido confirmado

Context: Unidad

Inv:

(self.enPedidoAProovedor->size() == 1 and self.enDeposito->size() == 0
and self.enPedido->size() == 0)
or
(self.enDeposito->size() == 1 and self.enPedidoAProovedor->size() == 0
and self.enPedido->size() == 0)
or
(self.enDeposito->size() == 1 and self.enPedidoAProovedor->size() == 0
and self.enPedido->size() > 0 and self.enPedido.estado in {enCarrito,
cancelado, anulado})
or
(self.enDeposito->size() == 0 and self.enPedidoAProovedor->size() == 0
and self.enPedido->filter(p | p.estado notIn {enCarrito, cancelado,
anulado}).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 algun cliente parecido a C

Context: Cliente Particular

edadParecida(c1, c2) = |c1.edad - c2.edad| < 4 *es un conjunto.*

productosComprados(c) = self.hicePedido.incluyo.soy

frecuenciaProducto(c, p) = productosComprados(c).filter(p1 | p1 == p).size()

frecuenciasProductosOrdenadas(c) =

productosComprados(c)→asSet()→sortedBy(p | frecuenciaProducto(c, p)).reverse()

diezMasComprados(c) = frecuenciasProductosOrdenadas(c)→subSequence(1, 10)

clienteParecido(c1, c2) = edadParecida(c1, c2) and

intersection(diezMasComprados(c1), diezMasComprados(c2)).size() >= 5

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

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

Context: Comprobante de tarjeta de credito

Inv: self.tarjeta.asociadoA == self.pagoDePedido.hechoPor

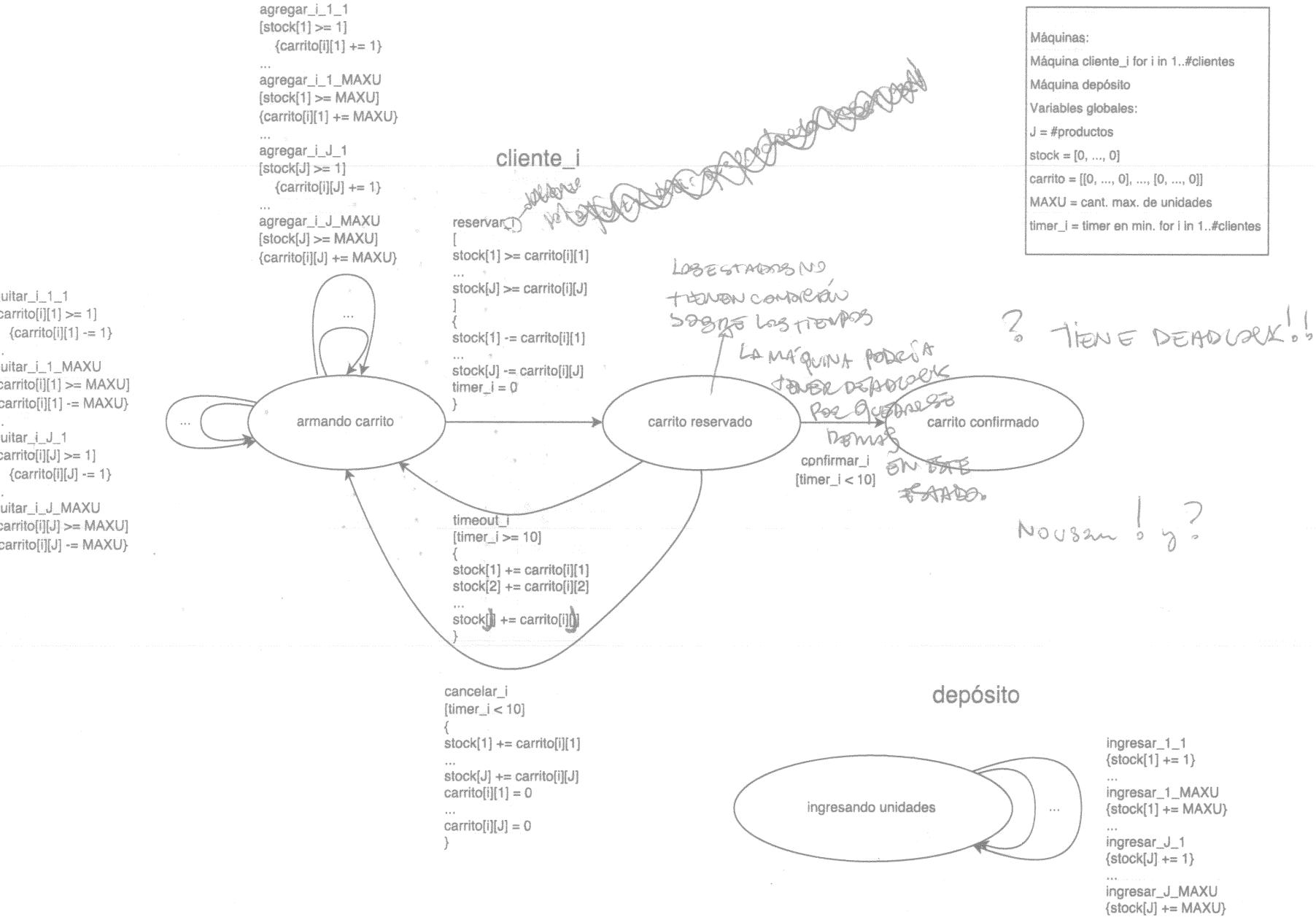
Context: Comprobante PayPal

Inv: self.cuenta.asociadoA == self.pagoDePedido.hechoPor

UNO PODRÍA USAR LA TARJETA DE UN FAMILIAR.

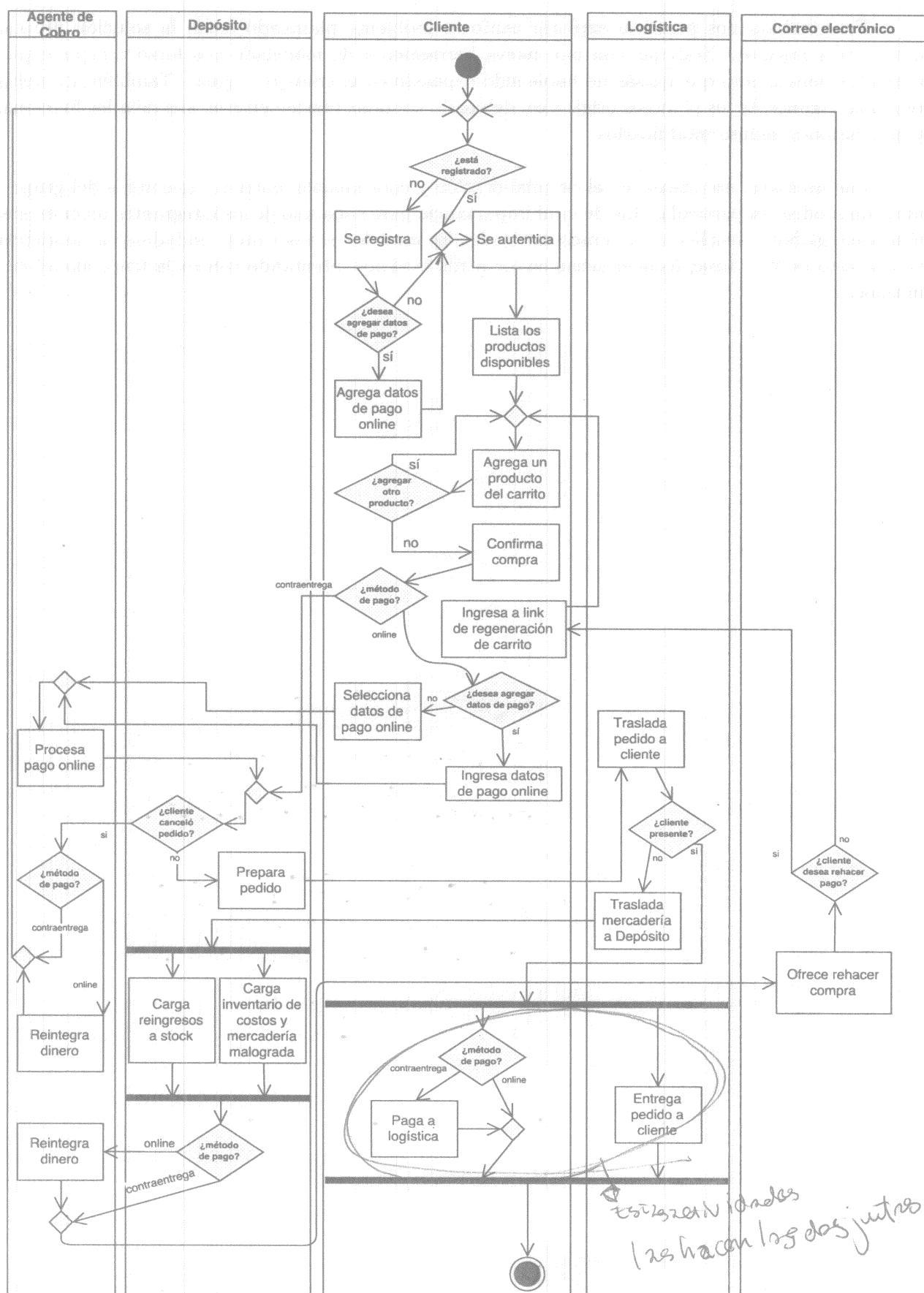
NO SE SI HA DE PEDIRLO.

Parte IV. Máquinas de estados finitos





Parte V. Diagrama de actividad



Parte VI. Conclusión

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.