

## ANEXO II – EJEMPLO DSI

# DISEÑO DEL SISTEMAS DE INFORMACIÓN “CAJEROS AUTOMÁTICOS”

Edición: 01  
Fecha: 01 de Abril de 2013

## CONTROL Y REGISTRO DE CAMBIO DEL DOCUMENTO

CONTROL	
Proyecto	CAJEROS AUTOMÁTICOS
Denominación	Diseño del Sistema de Información CAJEROS AUTOMÁTICOS
Fecha	01 de Abril de 2013
Edición	01
Grupo	Profesores
Autores	Fco Javier Gil Cumbreiras, Francisco A. Gómez Vela

REGISTRO DE CAMBIOS		
VERSIÓN	DESCRIPCIÓN DEL CAMBIO	FECHA DEL CAMBIO
V1	Versión Inicial	01 de Abril de 2013

1	DEFINICIÓN DEL SISTEMA .....	5
1.1	Arquitectura del Sistema .....	5
1.2	Requisitos No funcionales y Estándares, Normas y Restricciones del proyecto .....	7
1.3	Subsistemas de Diseño .....	7
1.4	Requisitos de Operación y seguridad .....	9
2	ARQUITECTURA DE SOPORTE.....	11
2.1	Subsistemas de Soporte .....	11
2.1.1	Subsistema SUB 03 - Módulo Correo .....	11
2.1.2	Subsistema SUB 04 - Módulo Peticiones.....	11
2.2	Mecanismos Diseño.....	12
3	MODELO FÍSICO DE DATOS.....	14
3.1	Diseño del Modelo Físico de Datos .....	14
3.2	Acceso a los Datos .....	17
4	DISEÑO DE CASOS DE USO .....	18
4.1	Subsistema de Análisis SUB01 - Operaciones del Cliente Cajero Automático .....	18
4.1.1	Diagrama de Casos de Uso .....	18
4.1.2	Casos de Uso Reales .....	19
4.1.3	Diagrama de Interacción entre Objetos.....	21
4.2	Subsistema de Análisis SUB02 - Operaciones Mantenimiento.....	23
5	DISEÑO DE CLASES .....	24
5.1	Subsistema de Diseño S1 .....	24
5.1.1	Modelo de Clases .....	24
5.1.2	Definición Clases .....	24
5.2	Subsistema de Diseño S2.....	25
6	DISEÑO DE INTERFACES .....	26
6.1	Subsistema de Diseño S1 .....	26
6.1.1	Navegación.....	26
6.1.2	Descripción de las interfaces .....	26
6.1.3	Descripción de los Informes.....	26
6.2	Subsistema de Diseño S2.....	27
7	ESPECIFICACIONES DE CONSTRUCCIÓN .....	28
7.1	Entorno de Construcción.....	28
7.2	Subsistemas de Construcción y Componentes.....	29
7.3	Elaboración de Especificaciones de Construcción.....	30
7.4	Elaboración de Especificaciones del Modelo Físico de Datos .....	30
8	CARGA INICIAL DE DATOS O MIGRACIÓN .....	32
8.1	Entorno de Carga Inicial o Migración .....	32
8.2	Procedimientos de Carga Inicial o Migración .....	32
9	PLAN DE PRUEBAS .....	34
9.1	Entornos de Pruebas .....	34
9.2	Definición de Niveles de Prueba .....	34
10	REQUISITOS DE IMPLANTACIÓN .....	36
10.1	Requisitos de Documentación .....	36
10.2	Requisitos de Implantación.....	37

# 1 DEFINICIÓN DEL SISTEMA

## 1.1 Arquitectura del Sistema

Recomendación: "Indicar las necesidades previstas de Almacenamiento, Procesamiento y Comunicaciones.

Diagrama de despliegue que defina la infraestructura técnica de los nodos y comunicaciones. Determinar la implementación de dichos elementos.

Si es posible, añadir una descripción de la configuración global de todos los elementos técnicos que participan en el sistema.

Si existen varios entornos, definirlo para cada entorno."

A continuación, en la Figura 1.1, se presenta la arquitectura técnica del sistema del cajero automático.

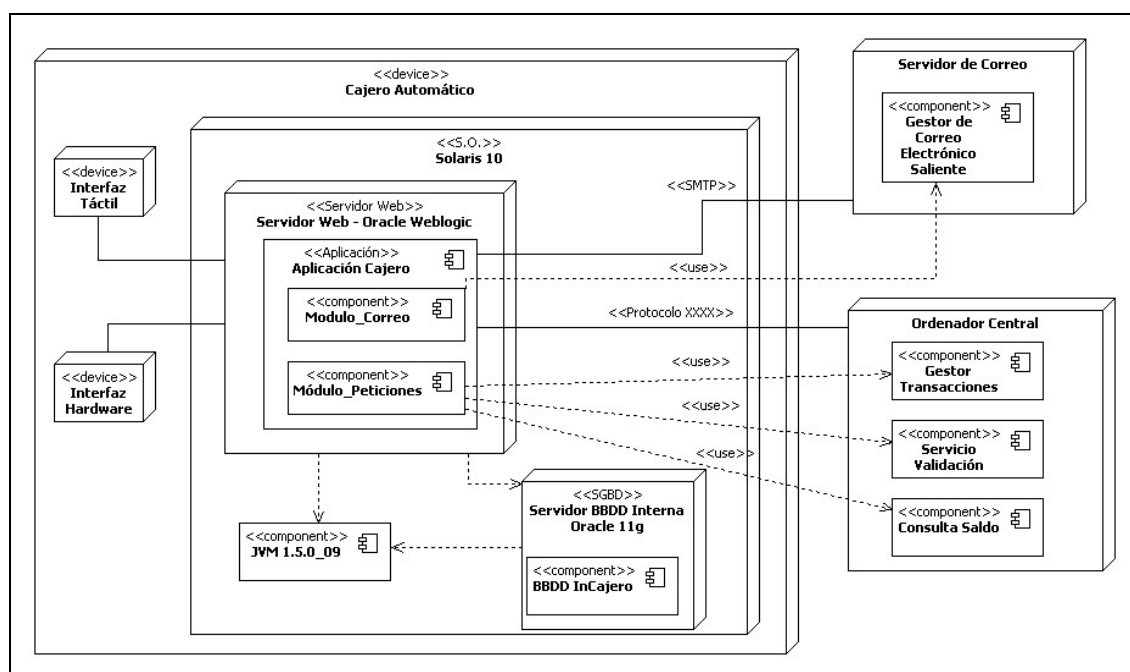


Figura 1.1: Despliegue del Sistema del Cajero Automático

Como puede verse, se trata de un sistema empujado en un cajero automático real, que dispone de dos interfaces de usuario análogos. Por un lado, haciendo uso de la interfaz táctil del cajero y por otro lado, haciendo uso del teclado que el cajero automático lleva incorporado.

Dicho cajero dispone de un sistema operativo Solaris 10, que tiene instalada una máquina virtual de java, cuya versión es la 1.5.0\_09.

El cajero automático contiene un Servidor de BBDD interno, que tiene la versión 11g. La base de datos en los cajeros automáticos recibe el nombre de *InCajero*. Esta base de datos, dispone de un espacio de 8 Gb, dividido en dos tablespaces, cada uno de 4 Gb, uno para los datos sobre transacciones diarias, TAB\_TRANS\_TG y otro para los datos de sincronización periódica, TAB\_SINC\_TG.

La aplicación que se ejecuta en el cajero automático, lo hace sobre un servidor Web, Oracle WebLogic.

El cajero automático dispone de 8 núcleos, cada uno con una capacidad de cálculo de NNNN Ghz. Esos núcleos se reparten por igual, asignándole 4 a al Servidor Web y 4 al Servidor de BBDD.

A toda la infraestructura interna del cajero, se le suman dos componentes, uno que gestiona los correos electrónicos y otro que opera las peticiones contra el ordenador central del banco, según puede verse en la Figura 1.1, y según se expuso en el documento de análisis ya entregado.

Se aplicará una arquitectura en tres capas, que serán la capa de Presentación, la capa de Negocio y la Capa de Datos. Esta arquitectura y las tecnologías asociadas quedan representadas en la Figura 1.2.

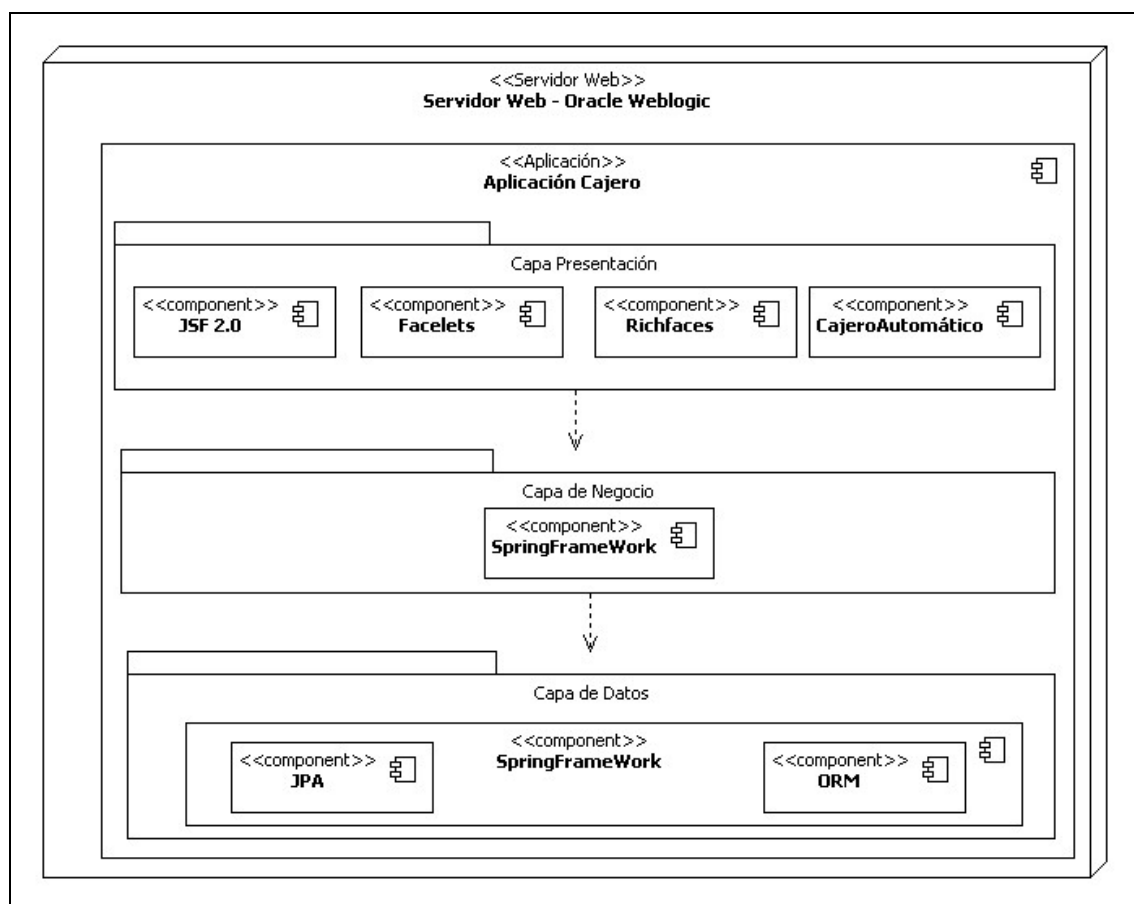


Figura 1.2: Arquitectura en tres Capas del Sistema del Cajero Automático

## 1.2 Requisitos No funcionales y Estándares, Normas y Restricciones del proyecto

Recomendación: *"Especificar los requisitos no funcionales que estén relacionados con la arquitectura."*

*Definir los estándares, normas y recomendaciones técnicas que aplicarán en el desarrollo del sistema."*

En todo el proceso de desarrollo, deberá atenderse a las recomendaciones de la metodología Métrica V3.

A la hora de desarrollar los interfaces de diseño, deben tenerse en cuenta las limitaciones de uso de la interfaz de teclado que ofrece el cajero automático. Sólo podrán diseñarse pantallas que respeten este interfaz, cuyas características quedan expuestas en el documento técnico "TEC-XXXX DDDDDDDDDDDDD" suministrado por el proveedor del hardware del cajero automático.

En el desarrollo del sistema del cajero automático, debe aplicarse el patrón arquitectónico de diseño Modelo-Vista-Controlador. Y debe aplicarse un patrón DAO, para la capa de acceso a datos.

El patrón arquitectónico MVC, podrá aplicarse haciendo uso del framework JSF. Y el patrón DAO, estará soportado por Spring JPA y Spring ORM.

## 1.3 Subsistemas de Diseño

Recomendación: *"Diagrama de paquetes que muestre los subsistemas de diseño y como se relacionan. Identificar si los subsistemas son de soporte o específicos"*

Siguiendo con la identificación de subsistemas realizada en la etapa de análisis, se proponen dos subsistemas, que agrupan funcionalidades comunes. Según se especificó, los subsistemas funcionales serían:

- SUB01 - Operaciones del Cliente Cajero Automático: Operaciones realizadas por los clientes del banco en el cajero.
- SUB02 - Operaciones Mantenimiento: Operaciones realizadas por el operador de cajero, empleado del banco.

Por otro lado, y como ya se ha visto, se propone aplicar un patrón arquitectónico MVC, lo cual supone una partición del sistema en tres grupos de elementos: Los elementos que forman parte del Modelo, los que forman parte de la Vista, y los que forman parte del Controlador.

En cualquier caso, estas dos particiones se superponen, porque los casos de uso ubicados en un subsistema funcional, tendrán su reflejo en las tres particiones del patrón MVC, tal y como puede verse en la Figura 1.3.

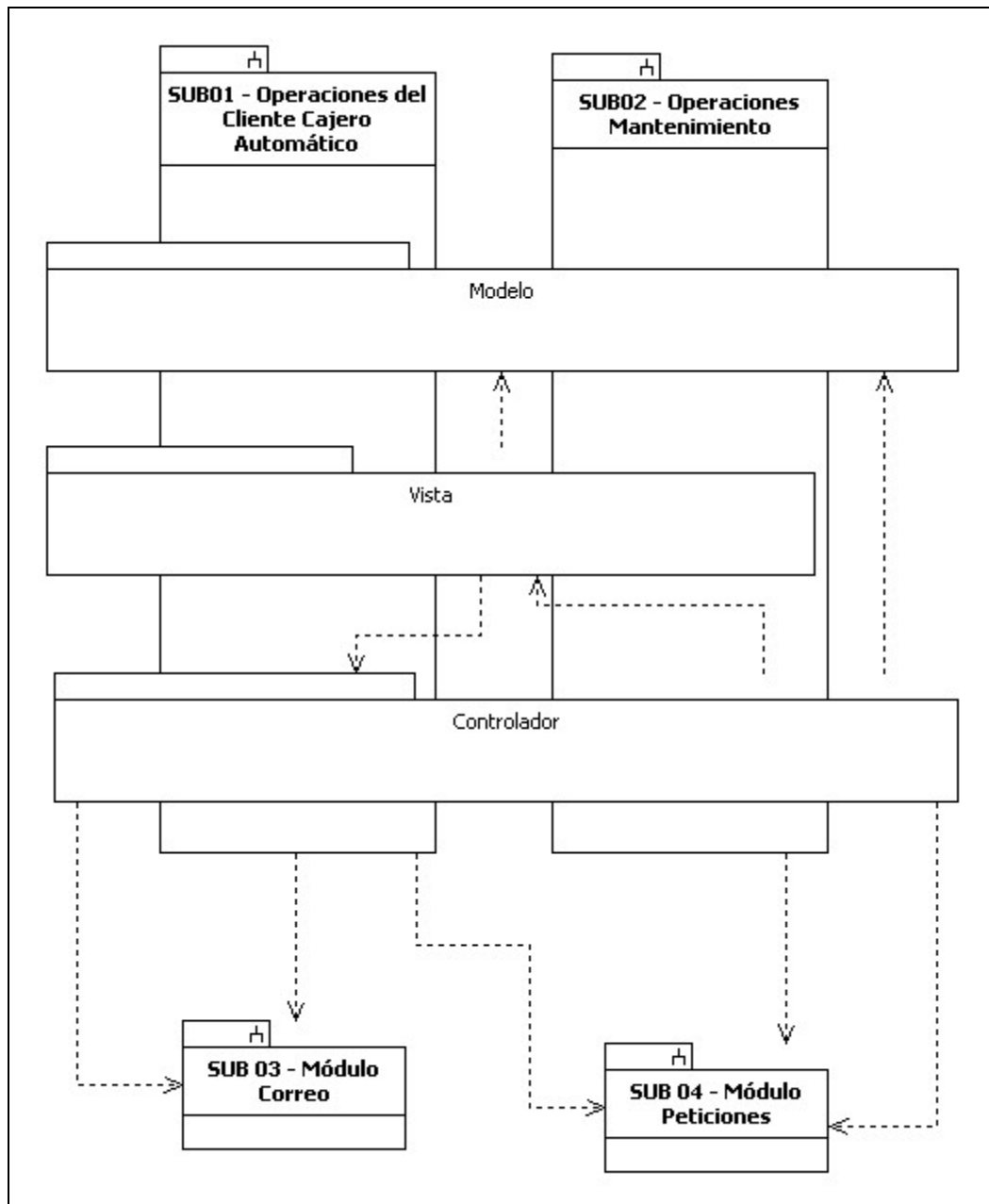


Figura 1.3: Diagrama de Paquetes de Diseño

De este modo, una funcionalidad del cajero automático tendrá elementos en la Vista, en el Modelo y en el Controlador.

Además, se identifican dos subsistemas de diseño adicionales, que representan las funcionalidades de conexión con sistemas externos, en concreto, se trata de:



SUB 03 - Módulo Correo: Interactúa con el sistema externo gestor de correo electrónico. SUB01 depende de este subsistema para enviar los correos electrónicos asociados a las transferencias de dinero, realizadas desde el cajero. Y en el controlador estarán ubicadas las llamadas a los servicios ofrecidos por este subsistema.

SUB 04 - Módulo Peticiones: Interactúa con el ordenador central del banco, mediando con él las peticiones realizadas por SUB01 y por SUB02, que serán realizadas desde el Controlador.

Por tanto, se identifican cuatro subsistemas de diseño, dos específicos y dos de soporte:

- SUB01 - Operaciones del Cliente Cajero Automático. Subsistema específico.
- SUB02 - Operaciones Mantenimiento. Subsistema específico.
- SUB 03 - Módulo Correo. Subsistema de soporte.
- SUB 04 - Módulo Peticiones. Subsistema de soporte.

Y además, se deben tener en cuenta, los paquetes de soporte, que implementan el patrón Modelo-Vista-Controlador.

## 1.4 Requisitos de Operación y seguridad

Recomendación: "*Requisitos no funcionales relacionados con la operación y seguridad.*"

Debe tenerse en cuenta los requisitos no funcionales identificados en la etapa de análisis, derivados de las restricciones de seguridad propias de un sistema como éste, y de los protocolos de operación interbancarios, que afectan a determinadas operaciones.

RNF-001	Protocolo de Operación InterBancaria
Versión	01
Autores	Grupo Profesores
Fuentes	Subdirector de Oficina Central Director de Oficina Periférica N°1 Director de Oficina Periférica N°2
Objetivos asociados	• Todos.
Descripción	El sistema deberá cumplir los protocolos de operación interbancaria, especificados en la norma NXXXXX.
Comentarios	--

<b>RNF-002</b>	<b>Comunicaciones cifradas</b>
<b>Versión</b>	01
<b>Autores</b>	Grupo Profesores
<b>Fuentes</b>	Responsable del Servicio de Tecnologías y comunicaciones
<b>Objetivos asociados</b>	<ul style="list-style-type: none"> <li>• Todos.</li> </ul>
<b>Descripción</b>	El sistema deberá comunicarse con el ordenador central del banco utilizando un cifrado, del tipo CXXXXXX, de la misma forma que ya operan otras comunicaciones dentro del propio banco, y para que sean compatibles.
<b>Comentarios</b>	--

<b>RNF-003</b>	<b>Alta disponibilidad</b>
<b>Versión</b>	01
<b>Autores</b>	Grupo Profesores
<b>Fuentes</b>	Responsable del Servicio de Tecnologías y comunicaciones
<b>Objetivos asociados</b>	<ul style="list-style-type: none"> <li>• Todos.</li> </ul>
<b>Descripción</b>	El sistema deberá prestar un servicio de alta disponibilidad, con sistema de reactivación autónoma, en caso de que exista algún tipo de fallo en la operativa del mismo
<b>Comentarios</b>	--

## 2 ARQUITECTURA DE SOPORTE

### 2.1 Subsistemas de Soporte

Recomendación: "Definir los módulos que forman parte de los subsistemas de soporte. Puede tratarse de frameworks de desarrollo"

Los subsistemas de soporte que se han identificado en el apartado 1.3 son:

- SUB 03 - Módulo Correo.
- SUB 04 - Módulo Peticiones.

A continuación se presenta el diseño detallado de cada subsistema.

#### 2.1.1 Subsistema SUB 03 - Módulo Correo

INCLUIR DIAGRAMA DE CLASES  
INCLUIR DIAGRAMA DE INTERACCIÓN (COLABORACIÓN O SECUENCIA), PARA LAS CLASES QUE TENGAN UNA FUNCIONALIDAD ESPECÍFICA.  
DEFINIR LOS SERVICIOS OFERTADOS POR EL SUBSISTEMA  
DEFINIR LAS INTERFACES QUE DEBEN RESPETARSE A LA HORA DE INTERACTUAR CON EL MÓDULO DE CORREO.  
DEFINIR LAS CLASES QUE FORMAN EL MÓDULO

#### 2.1.2 Subsistema SUB 04 - Módulo Peticiones

INCLUIR DIAGRAMA DE CLASES  
INCLUIR DIAGRAMA DE INTERACCIÓN (COLABORACIÓN O SECUENCIA), PARA LAS CLASES QUE TENGAN UNA FUNCIONALIDAD ESPECÍFICA.  
DEFINIR LOS SERVICIOS OFERTADOS POR EL SUBSISTEMA  
DEFINIR LAS INTERFACES QUE DEBEN RESPETARSE A LA HORA DE INTERACTUAR CON EL MÓDULO DE CORREO.  
DEFINIR LAS CLASES QUE FORMAN EL MÓDULO

## 2.2 Mecanismos Diseño

Recomendación: "Describir y diseñar los patrones o guías de diseño necesarios.  
Puede tratarse de frameworks de desarrollo"

Como ya se ha mencionado, se utiliza un patrón arquitectónico MVC, y un patrón DAO para el acceso a datos, todo esto dentro de una arquitectura en tres capas, Presentación, Negocio y Datos.

Para estructurar la aplicación, según estos patrones, y teniendo en cuenta la arquitectura tecnológica referida en el apartado 1.1, los mecanismos genéricos de diseño se pueden representar con el diagrama de clases, que se muestra a continuación, en la Figura 2.1:

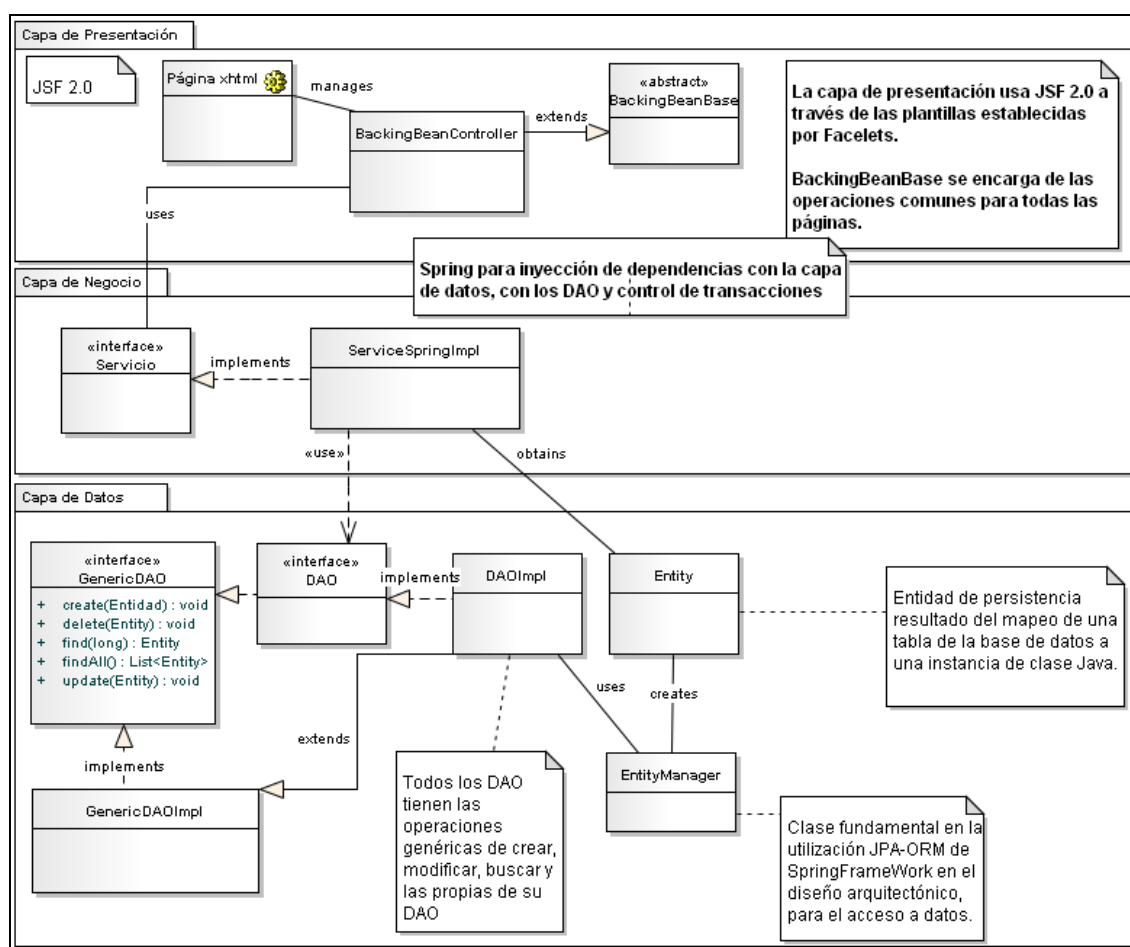


Figura 2.1: Diagrama de Clases que representa los mecanismos genéricos de diseño

Con esta arquitectura de clases, se aplican el patrón arquitectónico MVC, mediante JSF, apoyando el diseño de la página.xhtml con un backingbean de apoyo, que actúa como controlador. El modelo al que tiene acceso el patrón MVC se localiza en la interfaz Servicio, implementada por la clase ServiceSpringImpl. El Servicio, además de proporcionar acceso al modelo, opera en la capa de negocio, pues contiene las operaciones de negocio.

También, se aplica el patrón DAO, en la capa de datos. Se dispone de una interfaz DAO, y de su implementación, objetos típicos en este patrón, y la clase Entity representa a la entidad básica que se corresponderá con un registro de la base de datos, y que además es el TransferObject que se utiliza en el patrón DAO.

Por tanto, por cada objeto del Modelo de Dominio (nombre es el nombre del objeto), deben implementarse:

Tres clases de la capa de datos:

- nombreDAO: Interfaz DAO
- nombreDAOimpl: Implementación DAO
- nombre: Entidad

Y dos clases de la Capa de Negocio:

- nombreServicio: Interfaz Servicio
- nombreServicioSpringImpl: Implementación del Servicio con Spring, que contiene las reglas de negocio.

Y por cada pantalla de la interfaz del sistema, deben implementarse dos clases (teniendo en cuenta que la página se denomina *página*):

- Página xhtml: Módulo de Interfaz
- paginaBackingBean: Controlador del MVC

## 3 MODELO FÍSICO DE DATOS

### 3.1 Diseño del Modelo Físico de Datos

Recomendación: "Incluir Diagrama ERD o diagrama de Tablas.

Identificar las tablas de BBDD que se van a utilizar en el sistema, y para describir cada tabla proporcionando nombre, columnas, tipos de datos de las columnas, clave primaria, claves ajenas, índices y restricciones. Puede utilizarse la siguiente tabla formal:

Nombre	Nombre		
Descripción	descripción		
Atributos			
Campo	Tipo	Obligatorio	Descripción
nombre1	tipo	S/N	descripción
...	...	...	...
nombren	tipo	S/N	descripción
Clave primaria			
Nombre	Columnas	Secuencia	
nombre de la clave primaria	Lista de campos que componen la clave primaria	Secuencia de base de datos que genera la clave, si Aplica.	
Claves ajenas			
Nombre	Destino	Columnas	
nombre de la Clave Ajena 1	Nombre de la tabla a la que hace referencia	Campo o Campos que componen la clave ajena	
...	...	...	
nombre de la Clave Ajena N	Nombre de la tabla a la que hace referencia	Campo o Campos que componen la clave ajena	
Claves únicas			
Nombre	Columnas		
nombre Clave única 1	Campo o Campos que componen la clave única		
...	...		
nombre Clave única N	Campo o Campos que componen la clave única		
Restricciones			
Nombre	Columnas	Restricción	
Nombre de la restricción	nombre de los campos involucrados	Descripción, expresión lógica o pseudocódigo que define la restricción	
...	...	...	
Nombre de la restricción	nombre de los campos involucrados	Descripción, expresión lógica o pseudocódigo que define la restricción	

"

El modelo físico de datos se presenta en el diagrama entidad-relación que se muestra en la Figura 3.1.

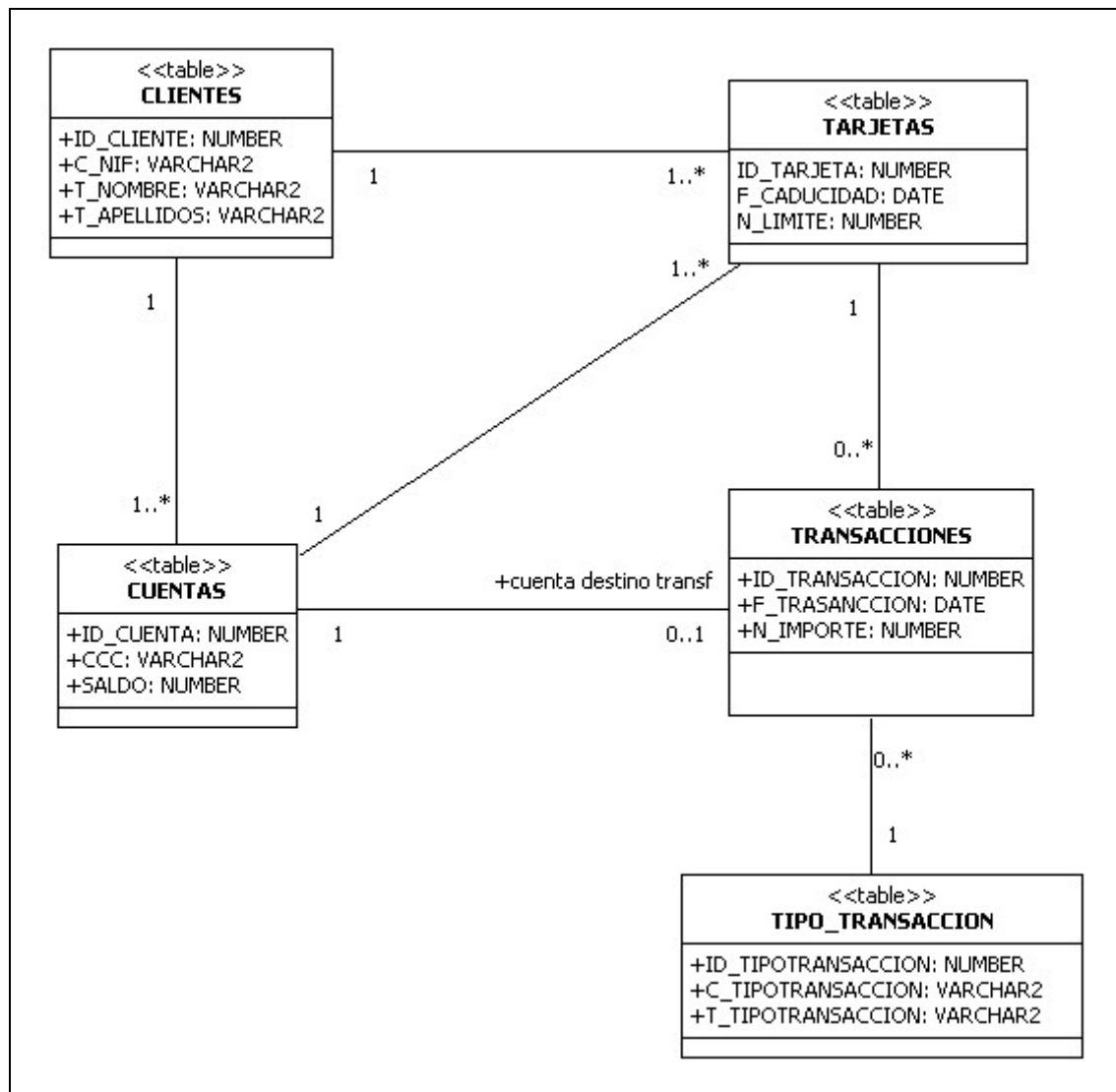


Figura 3.1: Modelo Físico de Datos – **OJO NO ESTÁ COMPLETO** -

A continuación se presenta el diseño detallado de cada tabla del modelo físico.

Nombre	CLIENTES		
Descripción	Tabla de clientes del banco		
Atributos			
Campo	Tipo	Obligatorio	Descripción
ID_CLIENTE	NUMBER(12)	S	Código interno del cliente
C_NIF	VARCHAR2(9)	S	NIF del cliente
T_NOMBRE	VARCHAR2(40)	S	Nombre del cliente
T_APELLIDOS	VARCHAR2(100)	S	Apellidos del Cliente
Clave primaria			
Nombre	Columnas	Secuencia	
CLIENTE PK	ID CLIENTE	SEQ CLIENTE	

Claves ajenas		
Nombre	Destino	Columnas
--	--	--
Claves únicas		
Nombre	Columnas	
NIF_UK	C_NIF	
Restricciones		
Nombre	Columnas	Restricción
--	--	--

...CUENTAS...

...TARJETAS...

...TIPO\_TRANSACCION...

Nombre		TRANSACCIONES	
Descripción		Tabla de transacciones del cajero automático	
Atributos			
Campo	Tipo	Obligatorio	Descripción
ID_TRANSACCION	NUMBER(12)	S	Código interno de la transacción
F_TRANSACCIÓN	DATE	S	Fecha de la transacción
N_IMPORTE	NUMBER(9,2)	S	Importe de la transacción
ID_TARJETA	NUMBER(12)	S	Clave ajena de la tarjeta que opera en la transacción
ID_CUENTA_DESTINO	NUMBER(12)	N	Clave ajena de la cuenta de destino de una transferencia. No siempre se rellenará este campo, sólo si el tipo de transacción es transferencia
ID_TIPOTRANSACCION	NUMBER(12)	S	Clave ajena del tipo de transacción
Clave primaria			
Nombre	Columnas		Secuencia
TRANSACCION_PK	ID_TRANSACCION		SEQ_TRANSACCION
Claves ajenas			
Nombre	Destino	Columnas	
TRANS_TARJETA_FK	TARJETAS.ID_TARJETA	ID_TARJETA	
TRANS_CUENTA_DESTINO_FK	CUENTAS.ID_CUENTA	ID_CUENTA_DESTINO	
TRANS_TIPOTRANS_FK	TIPO_TRANSACCION.ID_TIPOTRANSACCION	ID_TIPOTRANSACCION	
Claves únicas			
Nombre	Columnas		
--	--		
Restricciones			
Nombre	Columnas	Restricción	
CONST_CCC_DESTINO	ID_CUENTA_DESTINO	//El código 23 corresponde con el tipo de transacción Transferencia (ID_CUENTA_DESTINO IS NOT NULL AND ID_TIPOTRANSACCION = 23) OR (ID_CUENTA_DESTINO IS NULL AND ID_TIPOTRANSACCION <>23)	

...RESTO DE TABLAS...



### 3.2 Acceso a los Datos

Recomendación: "Determinar como la aplicación accede a los datos. Será necesario definir las cadenas de conexión, los DB link a otras BBDD, vistas materializadas, vistas materializadas hacia otras BBDD.

Y pueden incorporarse diagramas de despliegue, que clarifiquen cuales son los accesos a los datos."

En el sistema del cajero automático, el acceso a los datos se produce siempre desde las clases del modelo, del patrón MVC, a través del framework de Spring, y haciendo uso de los módulos JPA y ORM, que ocupan la capa de datos en la arquitectura propuesta en el apartado 1.1.

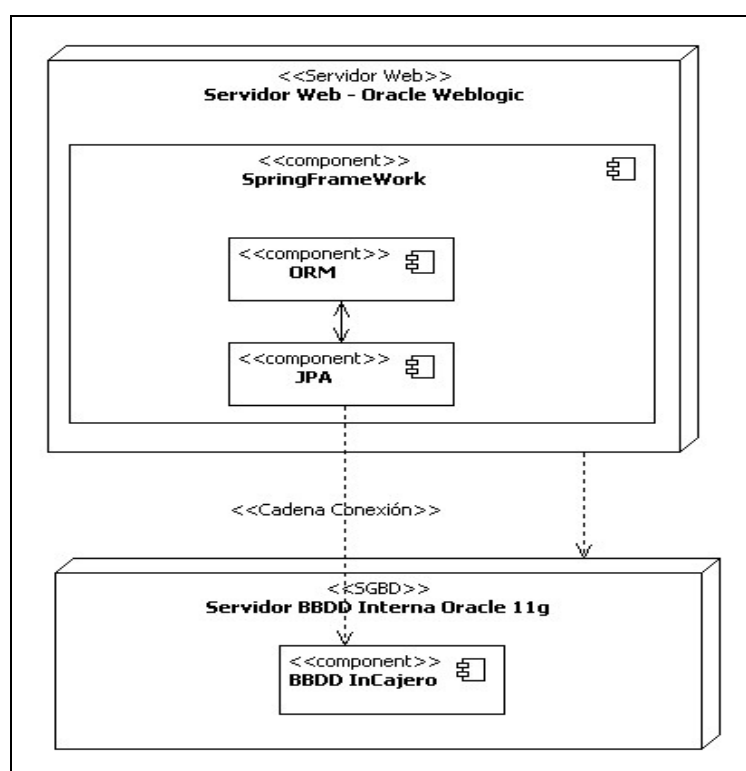


Figura 3.2: Camino de Acceso a los Datos

## 4 DISEÑO DE CASOS DE USO

### 4.1 Subsistema de Análisis SUB01 - Operaciones del Cliente Cajero Automático

#### 4.1.1 Diagrama de Casos de Uso

Recomendación: "Diagrama de Casos de Uso."

Después de revisar los casos de uso identificados en la fase de análisis, se concluye que la distribución de la funcionalidad en el sistema del cajero automático sigue siendo la misma, y por tanto, el diagrama de casos de uso correspondiente al subsistema SUB01 - Operaciones del Cliente Cajero Automático, mantiene los mismos casos de uso, y la misma interacción entre actores y casos de uso y entre casos de uso, que fue descrita en el análisis, y que ahora se representa en la Figura 4.1.

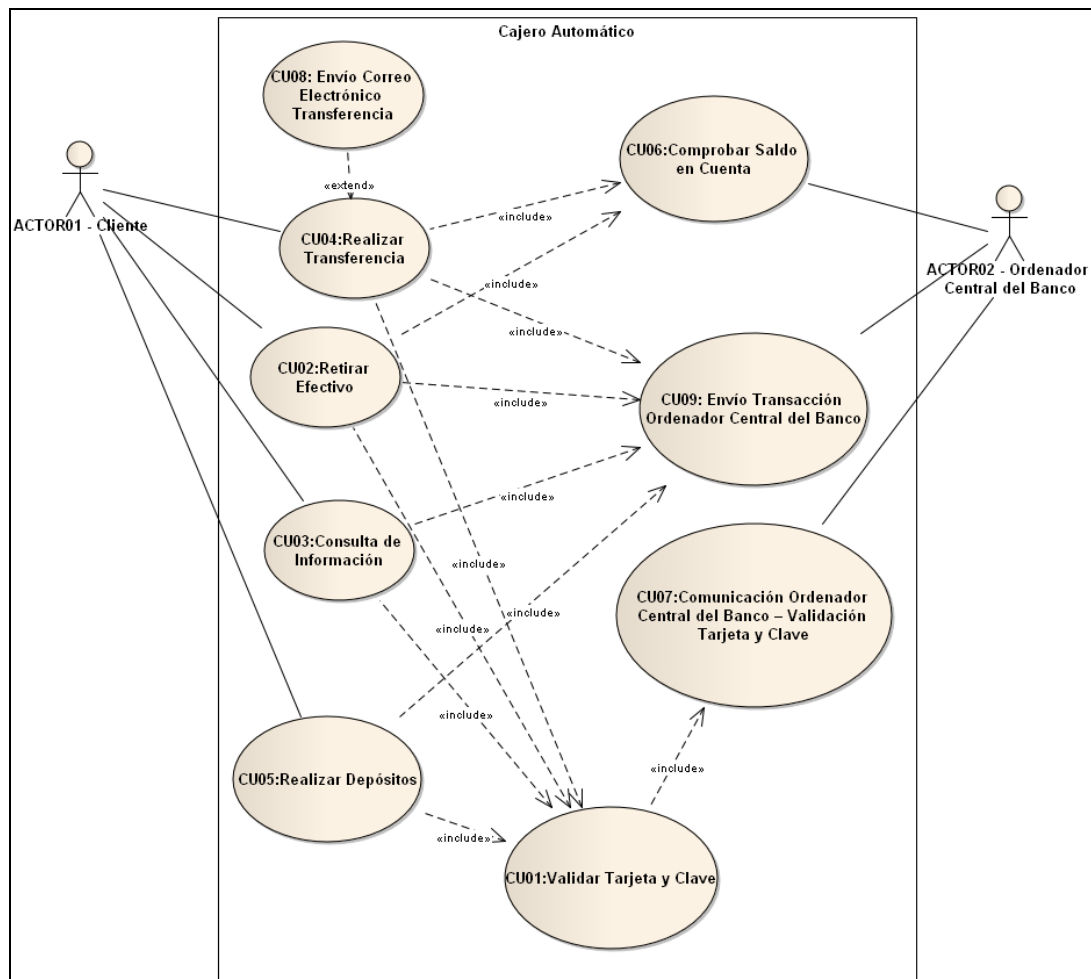


Figura 4.1: Casos de Uso del Subsistema SUB01

Los casos de uso CU06, CU07, CU08 y CU09 no pertenecen propiamente a este subsistema, pero son necesarios para su realización. El CU08 pertenece al subsistema SUB03 – Módulo de

Correo, identificado como subsistema de soporte en el apartado 1.3 de este documento. Los CU06, CU07 y CU09 pertenecen al subsistema SUB04 – Módulo de Peticiones, que también fue identificado como subsistema de soporte.

#### 4.1.2 Casos de Uso Reales

Recomendación: *"Incorporar los casos de uso reales, con los diagramas de robustez correspondientes"*

A continuación se incorporan los casos de uso reales, donde se detalla la interacción con el sistema y la funcionalidad soportada, **y se han reescrito en forma detallada**. Los casos de uso que se detallan son los siguientes:

- CU01 - Validar Tarjeta y Clave
- CU02 - Retirar Efectivo
- CU03 - Consulta de Información
- CU04 - Realizar Transferencia
- CU05 - Realizar Depósitos

Para cada uno de ellos, se añade el diagrama de robustez que apoya la consistencia del mismo, y que se ha utilizado para desarrollar los diagramas de secuencia de diseño del próximo apartado.

##### – CU01 - Validar Tarjeta y Clave:

CU-01	Validar Tarjeta y Clave	
Descripción	El sistema debe permitir a los clientes autenticarse en el sistema si la tarjeta pertenece a la red de cajeros y si la clave introducida es la correcta para dicha tarjeta.	
Actores	Cliente, Ordenador Central del Banco	
Precondición	La interfaz del cajero debe ser la identificada como IU0001 – Espera Cliente	
Flujo Normal	PASO	ACCIÓN
	1	El cliente del banco introduce la tarjeta en la ranura del interfaz del cajero destinada para tal fin
	2	El cajero automático realiza una validación de si la tarjeta pertenece a la red de cajeros
	3	El cajero automático solicita la clave desde la interfaz IU-0002
	4	El cliente proporciona la clave y pulsa el botón Aceptar
	5	El cajero automático inicia el C07:Comunicación Ordenador Central del Banco-Validación Tarjeta y Clave

	6	El cajero automático recibe la conformidad del Ordenador Central del Banco y avisa, mediante un mensaje en pantalla, al cliente que puede comenzar a realizar las operaciones que desee
	7	El cajero automático navega a la interfaz IU-0003, donde ofrece el menú de las operaciones disponibles.
Flujos Alternativos	PASO	ACCIÓN
	2	El cajero automático verifica que la tarjeta no pertenece a la red de cajeros, se cancela la operación y expulsa la tarjeta.
	3	El cajero automático navega a la interfaz IU-00012, y avisa al cliente de que la tarjeta no es válida
	4	El cliente pulsa el botón Aceptar en la IU-00012.
	5	El cajero automático navega a la interfaz IU-0001
	PASO	ACCIÓN
	6	El cajero automático recibe la no conformidad de la clave, y la clave se ha introducido menos de tres veces
	7	El cajero automático avisa al cliente de que la clave no es correcta y vuelve al paso 3 del escenario principal
Flujos Alternativos	PASO	ACCIÓN
	6	El cajero automático recibe la no conformidad de la clave, y la clave se ha introducido tres veces
	7	El cajero automático navega a la interfaz IU-00012, y avisa al cliente de que la tarjeta no es válida
	8	El cliente pulsa el botón Aceptar en la IU-00012.
	9	El cajero automático navega a la interfaz IU-0001
Postcondición	La tarjeta se valida junto con la clave correctamente y el cliente puede comenzar a operar en el cajero automático.	
Observaciones	--	

A continuación se presenta el diagrama de robustez con el que se ha detallado el caso de uso anterior. Puede comprobarse como se ha detectado una nueva clase, RedCajero, que será necesaria añadir al modelo de negocio, en el diagrama de clases de diseño.

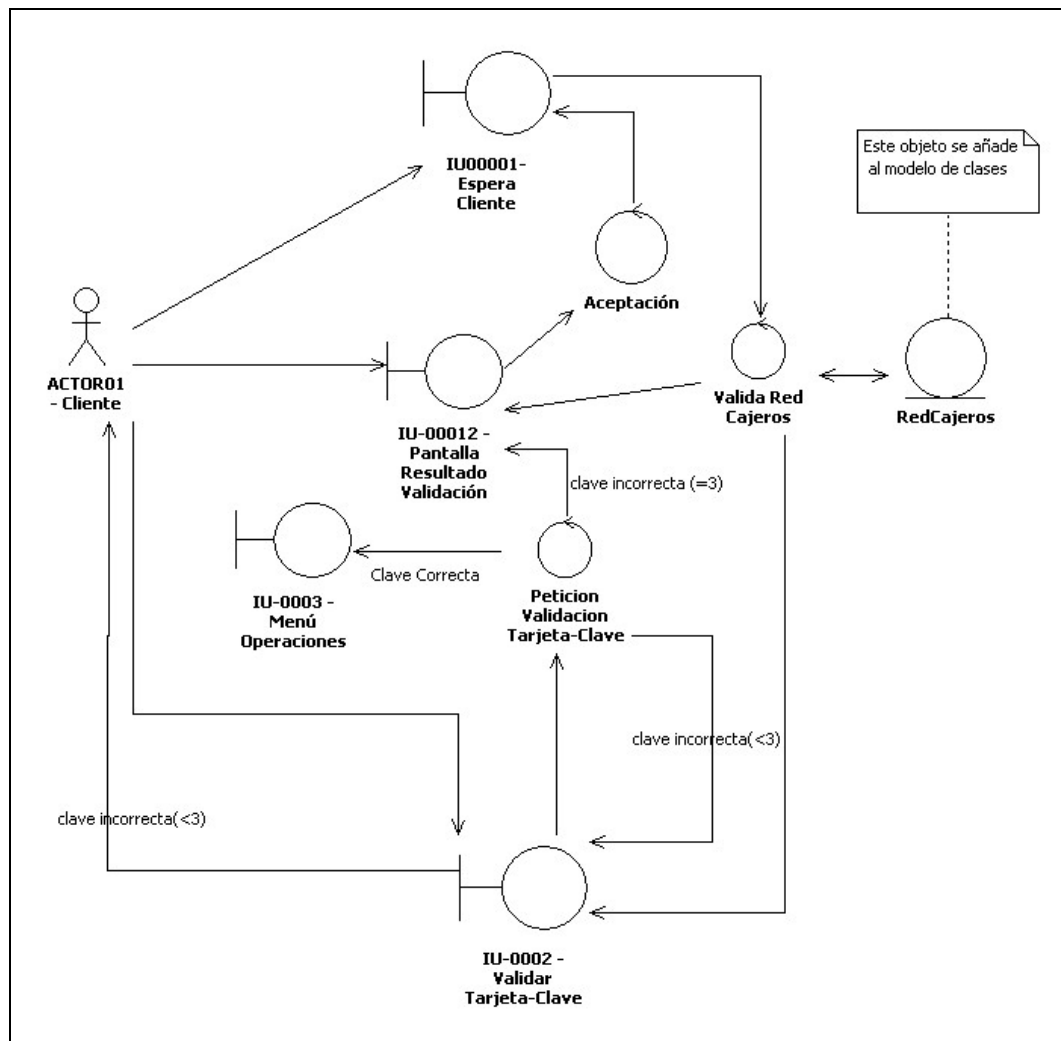


Figura 4.2: Diagrama de Robustez del Caso de Uso CU01 – Validar Tarjeta y Clave

...RESTO DE CASOS DE USO CON SUS DIAGRAMAS DE ROBUSTEZ...

### 4.1.3 Diagrama de Interacción entre Objetos

Recomendación: "Describir cómo interactúan las clases identificadas en los casos de uso del subsistema que se está modelando, utilizando los diagramas de interacción, preferentemente diagramas de secuencia, aunque pueden ser diagramas de colaboración."

A continuación, se incorpora un diagrama de interacción por cada caso de uso del sistema del cajero automático. Los que contemplen una interacción excesivamente compleja, serán divididos en varios diagramas, con el objetivo de aclarar el diseño.

– CU01 - Validar Tarjeta y Clave:

En el diagrama representado en la Figura 4.3, se muestra la interacción secuencial entre los distintos objetos, relacionados con el caso de uso CU01.

Debe recordarse que los Boundary IU-NNNN corresponden con páginas xhtml, según se ha explicado en el apartado 2.2. Pero también, se ha explicado que cada página xhtml tiene asignado una clase backbean, que hace las veces de controlador MVC, y por tanto, este backbean controla la navegación. Por tanto, los mensajes 4, 8 y resto, denominados “Navega” serán implementados por los backbean correspondientes. En este sentido, debe tenerse en cuenta que estos backbean serán:

- IU-0001 – Espera Cliente: esperaClientebackingBean.java
- IU-0002 – Validar Tarjeta-Clave: validarTarjetaClavebackingBean.java
- IU-0003 – Menú Operaciones: menuOperacionesbackingBean.java
- IU-00012 - ....

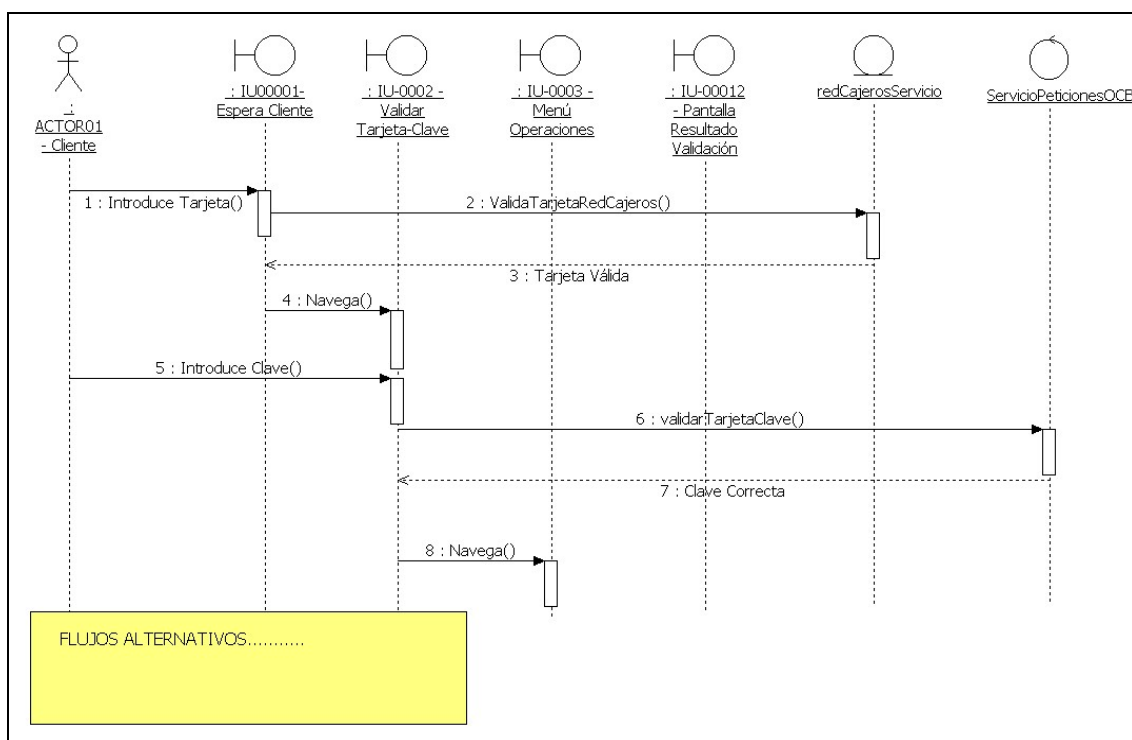


Figura 4.3: Diagrama de Secuencia para el Caso de Uso CU01 – Validar Tarjeta y Clave

Los mensajes 1 y 5 corresponden a interacciones del cliente con el cajero automático, en su mayor parte, realizadas mediante la interacción con la interfaz hardware del cajero automático.

Aunque en el diagrama de robustez, ha aparecido la entidad RedCajeros, en este diagrama colocamos la clase RedCajerosServicio, porque es la que implementa las reglas de negocio, según se ha explicado en el apartado 2.2. de este documento. Por tanto, la operación reflejada en el mensaje 2 debe ser implementada en dicha clase.

Tal y como se ha descrito en el apartado 2.1.2, el Módulo de Peticiones que trata con el ordenador central del banco, está accesible desde la clase ServicioPeticionesOCB, que ofrece la operación validarTarjetaClave, que aparece en el mensaje 6.

...SOBRE LOS FLUJOS ALTERNATIVOS...

#### 4.2 *Subsistema de Análisis SUB02 - Operaciones Mantenimiento*

...IGUAL QUE PARA SUB01...

## 5 DISEÑO DE CLASES

### 5.1 Subsistema de Diseño S1

#### 5.1.1 Modelo de Clases

Recomendación: "Diagrama de clases de diseño, con las clases que formen parte del subsistema, con las clases ya refinadas. Identificar clases abstractas, herencias, asociaciones, etc."

#### 5.1.2 Definición Clases

Recomendación: "Para cada clase del subsistema, definir atributos (nombre, tipo, restricciones, etc.), definir las operaciones (nombre, parámetros y visibilidad), y para las clases más complejas, definir un diagrama de transición de estados, para comprender la funcionalidad soportada por dichas clases.

Se puede utilizar la siguiente clase formal:

CL-NNNN	NOMBRE DESCRIPTIVO DE LA CLASE		
Versión	Nº de la Versión actual de la clase		
Autores	Nombre de los autores o identificación del grupo		
Descripción	Descripción de las responsabilidades de la clase		
Atributos	Nombre	Tipo	Descripción
	Atributo1		
	...		
	AtributoN		
Operaciones	Nombre	Descripción	
	Operacion1		
	...		
	OperacionN		
Comentarios	Comentarios adicionales a la especificación de la clase		

..



## 5.2 Subsistema de Diseño S2

...



## 6 DISEÑO DE INTERFACES

### 6.1 Subsistema de Diseño S1

#### 6.1.1 Navegación

Recomendación: "Definir la navegación definitiva entre ventanas, refinando la navegación entre módulos de interfaz, ya definida en el documento de ASI."

#### 6.1.2 Descripción de las interfaces

Recomendación: "Realizar un diseño técnico para cada ventana del sistema, concretando todos los detalles necesarios para su construcción."

Puede utilizarse una tabla formal parecida a la utilizada en análisis, en la que habría que ampliar los detalles, incorporando, parámetros de entrada y de salida, eventos generales de la pantalla o de los campos, validaciones que se realizan, etc.

IU-NNNN: Nombre					
Descripción	Descripción breve de las funciones del módulo				
Campos	Nombre	Tipo Datos	Editable/ Consulta	Oblig.	Descripción
	Campo				Descripción del campo
Botones/Enlaces	Nombre		Acción		
	Botón		Descripción de la acción que se lleva a cabo cuando se pulsa el botón		
	Enlace		Descripción de la acción que se lleva a cabo cuando se pulsa el enlace		

"

#### 6.1.3 Descripción de los Informes

Recomendación: "Realizar un diseño técnico para cada informe del sistema, concretando todos los detalles necesarios para su construcción."

*Puede utilizarse una tabla formal, parecida a la utilizada en el análisis, en la que habría que especificar cuestiones como parámetros de entrada y de salida.*

<b>IF- NNNN: Nombre</b>				
<b>Descripción</b>	Descripción del informe			
<b>Módulo de Interfaz</b>	IU-NNNN			
<b>Datos</b>	<b>Campo</b>	<b>Ordenación</b>	<b>Tipo Datos</b>	<b>Descripción</b>
	Campo	1, 2, ...y definir si es descendiente o ascendente		Describir brevemente qué representa el campo en el informe
<b>Resumen/Acumulado</b>	<b>Resumen</b>			<b>Campos del Resumen</b>
	Descripción Resumen o acumulado			Enumerar los campos que son agrupados

"

## 6.2 Subsistema de Diseño S2

...

## 7 ESPECIFICACIONES DE CONSTRUCCIÓN

### 7.1 Entorno de Construcción

Recomendación: "Describir el entorno tecnológico de construcción, incluyendo las herramientas utilizadas, las restricciones impuestas y demás requisitos no funcionales que tengan impacto sobre el entorno de construcción. Puede ser conveniente introducir un diagrama de despliegue."

A continuación se muestran las especificaciones del entorno de construcción, en cuestión de herramientas de desarrollo, servidores de BBDD y para herramientas de apoyo, y las versiones de cada componente utilizado. Esta información se representa en el siguiente esquema, mostrado en la Figura 7.1.

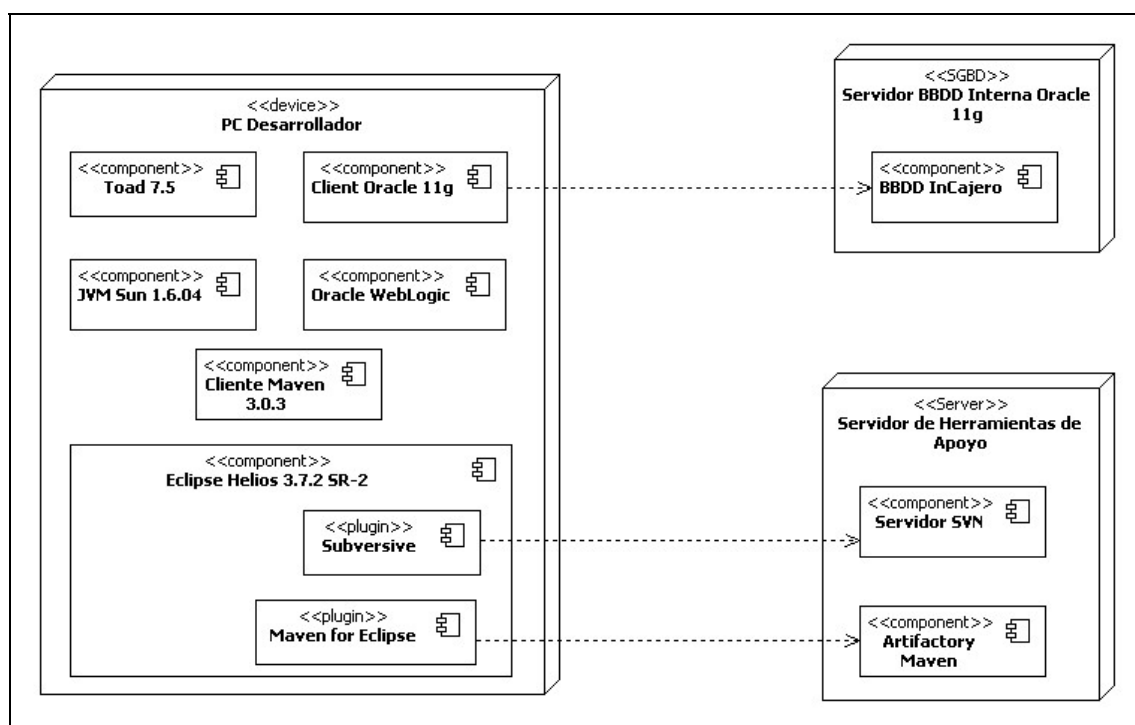


Figura 7.1: Diagrama de Despliegue del Entorno de Desarrollo

Puede verse como los equipos de los desarrolladores dispondrán de un cliente Oracle, que permitirá conectar con la BBDD, estando permitido el uso de la herramienta de acceso a la BBDD Toad 7.5. También estos equipos deben estar provistos de un servidor Oracle Weblogic, para poder desarrollar los distintos componentes del sistema, y probarlos en los propios equipos. También cada equipo de desarrollador, dispondrá de una instalación del cliente de maven, tal y como puede verse en la Figura 7.1. El IDE utilizado en este proyecto es el Eclipse Helios 3.7.2 SR-2. Además, el Eclipse contiene los dos plugins mostrados en el esquema anterior, y otros que se enumeran a continuación:

- Oracle Enterprise Pack for Eclipse 12.1.1.01
- JBoss Tools RichFaces 3.3.1.v2
- JAutodoc 1.10.0
- WebLogic Server 10.3.6
- Ant 1.7.1
- Oracle DataModeler 3.1.4 Patch
- ...**RESTO DE PLUGINS**...

## 7.2 Subsistemas de Construcción y Componentes

Recomendación: "Se utilizará un diagrama de paquetes donde se representen los paquetes de construcción del software, que podrían ser agrupaciones funcionales. Debe representarse la dependencia entre los paquetes.

Por cada paquete, debe especificarse los componentes que lo forman, y podría añadirse el diagrama de componentes asociado."

A continuación se presentan los paquetes de diseño, sus relaciones, y los componentes que lo forman. Debe tenerse en cuenta, que tanto SUB-01 como SUB-02 corresponden a agrupaciones funcionales con las que el usuario puede interactuar. Sin embargo, los subsistemas SUB03 y SUB04 son subsistemas de soporte, y realizan tareas específicas de conexión con sistemas externos al cajero. Estos paquetes, sus relaciones y sus componentes se presentan en la Figura 7.2.

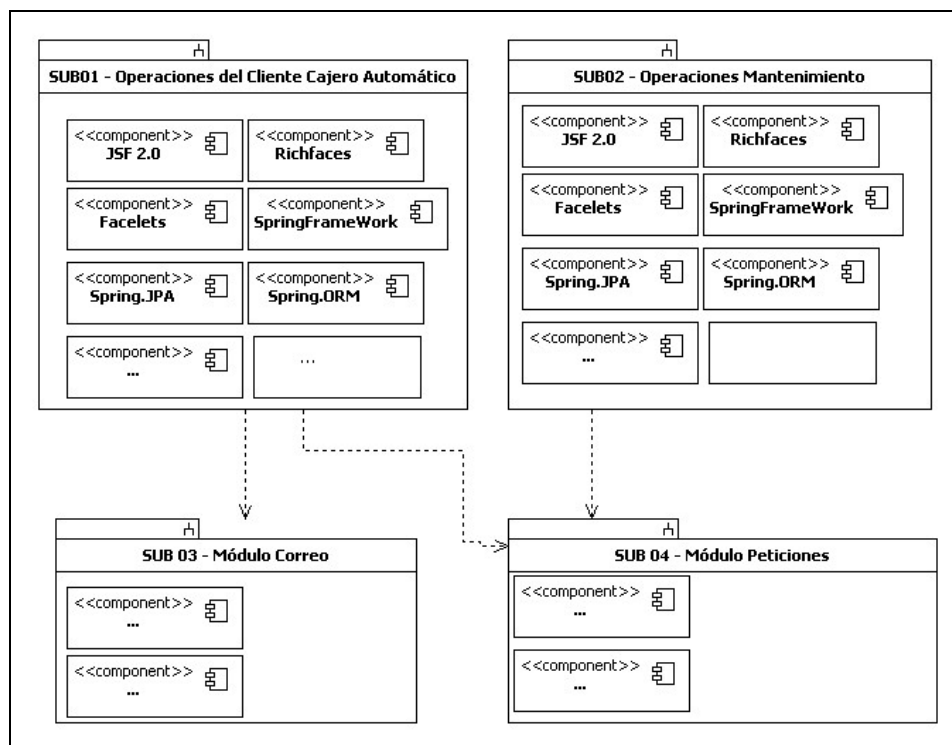


Figura 7.2: Diagrama de Paquetes de Diseño con sus componentes software asociados

Por ese motivo, SUB-01 y SUB02 mantienen una distribución de componentes similares, mientras que SUB03 y SUB04 no contiene componentes dedicados a la interfaz, y sin embargo, contiene componentes destinados a la conexión con sistemas externos al cajero.

A continuación, se presenta la utilidad de cada componente y la versión que se está aplicando:

- JSF 2.0: JSF: es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.
- RichFaces: Extensión de JSF, ...
- Facelets: Extensión de JSF, que pretende mejorar algunos aspectos.
- SpringFrameWork: Framework de desarrollo Spring.....
- ... RESTO DE COMPONENTES DEL DISEÑO DEL SISTEMA...

### 7.3 Elaboración de Especificaciones de Construcción

Recomendación: "Especificar qué hace falta para la construcción, compilación y generación de ejecutables o instalables, y como se debe proceder.

Puede incluirse un diagrama de componentes.

Debe desarrollarse una especificación detallada de cada componente. "

--NO APLICA--

No será necesario porque tengo descritos las características de navegación en los interfaces, tengo descritos los mecanismos genéricos de diseño, donde diré qué ficheros deben desarrollarse, tengo descritas las clases, sus interfaces, herencias, atributos y métodos, etc.

### 7.4 Elaboración de Especificaciones del Modelo Físico de Datos

Recomendación: "Describir como generar los scripts de BBDD, a partir del modelo físico de datos."

El modelo de datos físico descrito en el apartado 3.1 de este documento está diseñado en la herramienta CASE HHHHHHHHHH, que es capaz de generar los scripts de BBDD directamente a partir del propio modelo. A cada tabla de la base de datos debe añadirsele 6 campos, que sirven de campos de auditoría. Estos campos tienen la siguiente descripción:

Campo	Tipo	Obligatorio	Descripción
F_CREATE	DATE	S	Fecha de creación del registro

ID_CLICREATE	NUMBER(12)	N	Código del cliente que crea el registro.
ID_OPCREATE	NUMBER(12)	N	Código del operador de cajero que crea el registro
F_UPDATE	DATE	N	Fecha de actualización del registro
ID_CLIUPDATE	NUMBER(12)	N	Código del cliente que modifica el registro
ID_OPUPDATE	NUMBER(12)	N	Código del operador de cajero que actualiza el registro

Los campos ID\_CLICREATE y ID\_OPCREATE son excluyentes, de tal forma que registros que tengan relleno ID\_CLICREATE, no tendrán relleno ID\_OPCREATE, y al contrario pasa lo mismo. La misma regla aplica a los campos de auditoría de actualización ID\_CLIUPDATE y ID\_OPUPDATE.

Aunque los campos ID de auditoría tienen relación directa con las tablas CLIENTES y OPERADORES, no se establece la restricción correspondiente a la clave ajena, para facilitar las operaciones de mantenimiento contempladas en el subsistema SUB 04 - Módulo Peticiones.

Sin embargo, se delega la tarea de mantener la integridad referencial en estos campos, a la propia aplicación, y a unos triggers de auditoría que deberán construirse para cada tabla de la BBDD. Estos triggers son los siguientes y deben realizar las tareas descritas a continuación:

Trigger	Evento de Disparo	Descripción
CREATE_NOMBRETABLA	After Create	Debe rellenar el campo F_CREATE con el sysdate del sistema, y debe...
...	...	...
...	...	...
...	...	...

Donde NOMBRETABLA debe sustituirse por el nombre de la tabla a la que esté asociada el trigger que se está creando.

## 8 CARGA INICIAL DE DATOS O MIGRACIÓN

### 8.1 Entorno de Carga Inicial o Migración

Recomendación: *"Describir el entorno tecnológico de la carga inicial y/o migración, utilizando un diagrama de despliegue."*

No es necesario realizar una migración, puesto que se trata de un cajero automático nuevo, que no tiene datos de otros sistemas anteriores. Por otra parte, sí será necesario realizar una carga inicial de datos, que configure las tablas auxiliares del sistema, antes de comenzar a utilizarlo.

Esta carga inicial es una de las operaciones de mantenimiento que están soportadas por el subsistema SUB02 - Operaciones Mantenimiento. En concreto, esta actualización inicial, es necesaria realizarla periódicamente, y podrá realizarse haciendo uso de la funcionalidad que soporta el caso de uso CU-12: Sincroniza con Ordenador Central del Banco.

Por tanto, no será necesario, tener en cuenta ninguna infraestructura adicional para la puesta en marcha del sistema. Aunque como para el desarrollo, las pruebas unitarias y las pruebas de integración son necesarios los datos de las tablas auxiliares, será necesario desarrollar una serie de scripts de BBDD que carguen dichas tablas, en el orden correcto, y con los datos actuales.

En el siguiente apartado, se muestra el diseño del procedimiento de carga inicial, para estos scripts, que sólo deben usarse durante la fase de construcción del sistema.

### 8.2 Procedimientos de Carga Inicial o Migración

Recomendación: *"Definir el proceso de migración y/o carga inicial, los procedimientos de carga inicial y/o migración que participan en el proceso, y el orden o jerarquía de lanzamiento. Para lo cual se utilizará un diagrama de actividad."*

*También debe realizarse el diseño detallado de cada procedimiento que participa en la migración o carga inicial. Para esta definición detallada, también pueden utilizarse diagramas de actividad."*

Se describe el proceso de la carga inicial de las tablas auxiliares, que sólo debe ser utilizado durante la fase de desarrollo. El proceso utilizado y el orden que sigue el lanzamiento de los scripts de carga, es muy similar a los pasos que deben seguirse para la construcción del CU-12: Sincroniza con Ordenador Central del Banco, ya descrito anteriormente.

El proceso de carga se describe en el diagrama de actividad, mostrado a continuación, en la Figura 8.1.



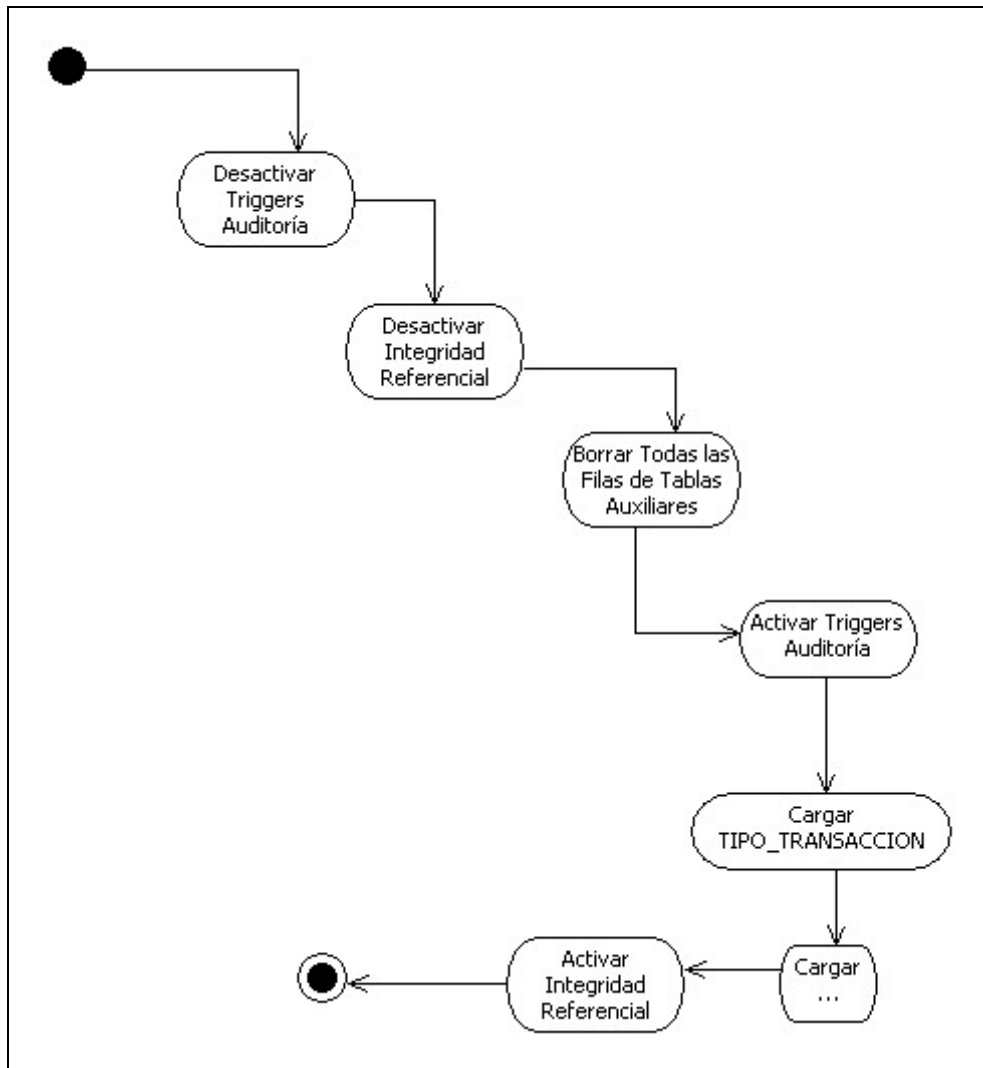


Figura 8.1: Diagrama de actividad que describe el proceso de carga inicial de las tablas auxiliares.

## 9 PLAN DE PRUEBAS

### 9.1 Entornos de Pruebas

Recomendación: *"Describir el entorno para llevar a cabo las pruebas del sistema, incluyendo restricciones operativas. Si se utilizan herramientas concretas de pruebas, especificarlas en este apartado. El origen de los datos de pruebas, y cualquier otra cuestión relevante del entorno de pruebas."*

*Para describir el entorno tecnológico, puede incluirse un diagrama de despliegue."*

En primer lugar, las pruebas unitarias serán llevadas a cabo por los desarrolladores, en sus propios equipos, haciendo uso de JUnit, por lo que, no será necesario ningún entorno adicional al descrito en el apartado 7.1.

Las pruebas de integración serán llevadas a cabo tanto por los desarrolladores como por un equipo del banco, destinado a tal fin. No será necesario utilizar un cajero automático real, puesto que la validación inicial será simulada. Como ya se expuso en el entregable de análisis, estas pruebas serán llevadas a cabo en un entorno replicado, con las características descritas en el apartado 1.2 de aquel documento, y que se exponen con más detalle en el apartado 1.1 de este documento. Como ya se ha expuesto, no se utilizará el hardware del cajero automático, y por otra parte, se dispondrá de una réplica del ordenador central del banco, para realizar pruebas, a la vez que es posible utilizar el mismo servicio de correo electrónico externo.

Por último, para llevar a cabo las pruebas de implantación y de aceptación se dispondrá de una interfaz física del cajero automático, donde las pruebas serán realizadas de forma muy cercana a la realidad. Por tanto, será necesario una réplica de un entorno tecnológico como el descrito en el apartado 1.1 de este documento y además será necesario un cajero automático como los que se implantarán en las distintas sucursales, que en este caso es *wincor-nixdorf - CINEO C2560 - Full function ATMs* (<http://www.wincor-nixdorf.com>).

Para llevar a cabo las distintas pruebas, será necesario realizar una extracción de un juego de datos, que no sea real, para que el equipo de desarrollo y el resto de personas que tengan acceso a dicha réplica no puedan obtener datos reales de los clientes del banco. Y en el caso de las cuentas de correo, se han creado unas genéricas a tal efecto.

### 9.2 Definición de Niveles de Prueba

Recomendación *"Definir niveles de prueba. Realizar pruebas de integración y de sistema con una carga de trabajo parecida a la de explotación. Realizar validaciones funcionales y no funcionales, procurando cubrir las excepciones."*

*Presentar los criterios que son necesarios cubrir para que se acepte cada prueba."*

Como ya se indicó en el entregable de análisis se han planteado cuatro niveles de pruebas: las pruebas unitarias, las pruebas de integración, las pruebas de implantación y las pruebas unitarias.

- **Pruebas Unitarias:** Serán llevadas a cabo por los desarrolladores, con JUnit, sobre los componentes: ServiciosDAO, Beans de apoyo a la interfaz, Controladores.....
- **Pruebas de Integración:** Serán llevadas a cabo, conjuntamente, por los desarrolladores y por un equipo formado por miembros designados por el banco para tal fin. Estas pruebas deben determinar si la operativa de cada subsistema cumple los requisitos asociados.
- **Pruebas de Implantación:** .....PRUEBAS DE IMPLANTACIÓN.....
- **Pruebas de Aceptación:** .....PRUEBAS DE ACEPTACIÓN.....



## 10 REQUISITOS DE IMPLANTACIÓN

### 10.1 Requisitos de Documentación

Recomendación: *"Deben especificarse los requisitos de documentación de usuario necesaria para operar con el nuevo sistema.*

*En ese sentido debería indicarse qué manuales son necesarios: de usuario, de explotación, etc., y las características de dichos documentos, como el tipo de formato, la estructura y el contenido, control de versiones, a quien van dirigidos, etc."*

Los requisitos de implantación relacionados con la documentación que debe ser generada son los que se muestran a continuación.

RI-001	Manual de Usuario de Operador de Cajero
Versión	01
Autores	Grupo Profesores
Fuentes	Usuarios Participantes
Descripción	<p>Debe desarrollarse un Manual de Usuario para los operadores de cajero, que contenga una descripción detallada de las operaciones de mantenimiento del cajero.</p> <p>Este documento debe respetar el formato indicado por la normativa interna del banco, y debe tener el siguiente índice:</p> <p>1. TTTTTTTT</p> <p>1.1 TTTTTTTT</p> <p>...</p>
Comentarios	--

RI-002	Ayuda Contextual Clientes Cajero
Versión	01
Autores	Grupo Profesores
Fuentes	Usuarios Participantes
Descripción	Debe desarrollarse una ayuda contextual, asociada a cada pantalla del sistema del cajero automático, que se integrará en dichas pantallas, mediante la utilidad HHHHHH. Dicha ayuda contextual debe facilitar la entrada de datos y la operativa general del sistema del cajero automático.
Comentarios	--

## 10.2 Requisitos de Implantación

Recomendación: "Deben especificarse necesidades de formación especiales, relacionadas con la operación, y sobre todo con la administración del sistema. También pueden existir requisitos relativos a la propia implantación del sistema en el entorno de operación, como son la infraestructura e instalación, pudiendo ser estos requisitos referentes software, hardware y comunicaciones."

RI-002	Formación Operadores de Cajero
Versión	01
Autores	Grupo Profesores
Fuentes	Usuarios Participantes
Descripción	Debe realizarse un plan de formación y llevarlo a cabo antes de poner en marcha los distintos cajeros. Este plan de formación debe ir orientado a la formación de los operadores de cajero y ... ...
Comentarios	--

..OTROS REQUISITOS DE IMPLANTACIÓN, RELACIONADOS CON SOFTWARE, HARDWARE Y COMUNICACIONES...