



COLEGIO DE POSTGRADUADOS

INSTITUCIÓN DE ENSEÑANZA E INVESTIGACIÓN EN CIENCIAS AGRÍCOLAS

CAMPUS MONTECILLO

POSTGRADO DE SOCIOECONOMÍA, ESTADÍSTICA E INFORMÁTICA
CÓMPUTO APLICADO

EL EFECTO DEL CONTAGIO EN SISTEMAS DE INFORMACIÓN
POR MEDIO DE TARJETAS INTELIGENTES

JAVIER HERNÁNDEZ RUIZ

T E S I S

PRESENTADA COMO REQUISITO PARCIAL

PARA OBTENER EL GRADO DE

MAESTRO EN CIENCIAS

MONTECILLO, TEXCOCO, EDO. DE MÉXICO

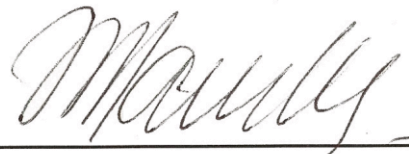
2010

La presente tesis, titulada: “**EL EFECTO DEL CONTAGIO EN SISTEMAS DE INFORMACIÓN POR MEDIO DE TARJETAS INTELIGENTES**”, realizada por el alumno: **Javier Hernández Ruiz**, bajo la dirección del Consejo Particular indicado, ha sido aprobada por el mismo y aceptada como requisito parcial para obtener el grado de:

**MAESTRO EN CIENCIAS
SOCIOECONOMÍA, ESTADÍSTICA E INFORMÁTICA
CÓMPUTO APLICADO**

CONSEJO PARTICULAR

CONSEJERO:



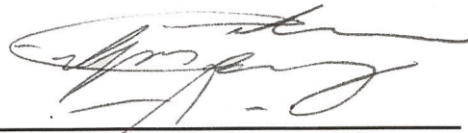
Dr. Juan R. Bauer Mengelberg

ASESOR:



Dr. Paulino Pérez Rodríguez

ASESOR:



M.C. Edgar Ramírez Galeano

Montecillo, Texcoco, Edo. de México, Noviembre 2010.

DEDICATORIAS

A mis padres, el **Sr. Javier Hernández Morales** y la **Sra. Ma. Yolanda Ruiz Cisneros**, por todo el apoyo y sabiduría que he recibido de ellos a lo largo de toda mi vida.

A mis hermanos, **Alex Hernández Ruiz** y **Estefanía Hernández Ruiz**, por todo el apoyo moral que me han dado para poder terminar este ciclo de mi vida.

A mis tíos, el **Sr. Usiel Joaquín Hernández Morales** y la **Sra. Carlota Tostado Magaña**, por todo el apoyo y las muestras de solidaridad que me han dado.

Con mucho cariño.....

Javier H.R.

AGRADECIMIENTOS

A DIOS

Por estar conmigo siempre y haberme permitido terminar satisfactoriamente una etapa mas de mi vida.

A MIS PADRES

Quiero agradecerles inmensamente por todo el apoyo que me han dado siempre y por haberme guiado sabiamente a lo largo de mi desarrollo.

A MI CONSEJERO

Dr. Juan Ricardo Bauer Mengelberg, por toda la paciencia que me ha tenido, los consejos y sugerencias que me ha brindado, pero sobre todo por el apoyo constante que he recibido de él desde mi ingreso al postgrado.

Al Dr. David Hebert Del Valle Paniagua, por el apoyo que me ha dado, en especial en la búsqueda de literatura para el desarrollo de ésta tesis.

A TODOS LOS PROFESORES DEL PROGRAMA

Por haberme ayudado con su conocimiento en esta etapa de mi formación profesional.

AL CONSEJO NACIONAL DE CIENCIA Y TECNOLOGÍA (CONACYT)

Por el apoyo económico que me otorgó para poder llevar a cabo mis estudios de Maestría.

AL COLEGIO DE POSTGRADUADOS

En especial al Programa de Cómputo Aplicado, por todas las enseñanzas y las experiencias obtenidas y por la amistad recibida de todos los que en él laboran.

A MIS COMPAÑEROS DEL COLEGIO DE POSTGRADUADOS

Por todos los momentos que compartimos a lo largo de esta etapa.

CONTENIDO

	Página
ÍNDICE DE FIGURAS	vii
ÍNDICE DE TABLAS	viii
RESUMEN	ix
ABSTRACT	x
CAPÍTULO 1. INTRODUCCIÓN	1
CAPÍTULO 2. MARCO TEÓRICO Y METODOLOGÍA	3
2.1. TARJETAS INTELIGENTES Y LECTORAS DE TARJETAS	3
2.2. SISTEMAS DE INFORMACIÓN	8
2.2.1. MÉTODO DE DESARROLLO	8
2.2.2. SISTEMAS VERSUS PAQUETES	13
2.3. BASES DE DATOS RELACIONALES	14
2.3.1. CONCEPTOS FUNDAMENTALES	14
CAPÍTULO 3. EL EFECTO CONTAGIO MEDIANTE TARJETAS INTELIGENTES	19
CAPÍTULO 4. EL BAZAR COMO EJEMPLO DEL EFECTO CONTAGIO	22
4.1. SITUACIONES PARA LAS CUALES SE ELABORÓ EL PAQUETE	22
CAPÍTULO 5. FASE I DEL PAQUETE BAZAR	25
5.1. ACTIVIDADES DE LA FASE I	25
5.2. DESCRIPCIÓN GENERAL DEL PAQUETE QUE SE ELABORARÁ	26
CAPÍTULO 6. FASE II	28
6.1. ACTIVIDADES DE LA FASE II	28
6.2. REQUISITOS	29
6.3. ACTORES	30
6.3.1. ADMINISTRADORES	30
6.3.2. CAJEROS	31
6.3.3. PUESTEROS	31
6.3.4. VISITANTES	31

6.4. DETERMINACIÓN DE LOS “USUARIOS” DEL SISTEMA	32
6.5. OBJETIVO FUNDAMENTAL DEL BAZAR (PARA QUÉ SIRVE).....	32
6.6. ENTIDAD (DE DATOS) FUNDAMENTAL.....	32
6.7. DONDE ESTARÁN LOS DATOS.....	32
6.8. LISTA DE FUNCIONES “PRELIMINAR”	32
6.8.1. MODELO CONCEPTUAL.....	34
6.8.2. RELACIONES ENTRE LAS ENTIDADES.....	35
6.9. PROTECCIÓN DE DATOS (SEGURIDAD)	35
6.9.1. USO NO AUTORIZADO DE FUNCIONES.....	36
6.9.2. ALTERACIÓN DE DATOS SIN EL USO DE PROGRAMAS DEL SISTEMA	38
6.9.3. PÉRDIDA O DESTRUCCIÓN DE DATOS	38
6.9.4. CLONACIÓN DE TARJETAS	38
6.9.5. USO DE TARJETAS Y BASES DE DATOS DE UN EVENTO (INSTANCIA) EN OTRO EVENTO	39
6.9.6. EN LA RECOLECCIÓN DE DATOS, ALGUIEN “BORRA” REGISTROS DE LOS ARCHIVOS DE TRANSPORTE.....	40
6.10. LA CONTRIBUCIÓN DEL EFECTO CONTAGIO	40
6.11. SELECCIÓN DE TARJETAS Y LECTORAS PARA EL BAZAR	42
6.12. PROCEDIMIENTOS Y CONTROLES	46
6.13. INTERFACES USUARIO-SISTEMA.....	48
6.14. ASPECTOS RELACIONADOS CON EL USO DEL PAQUETE	50
6.15. ESPECIFICACIONES DE PROGRAMACIÓN	51
6.16. NIVEL DE DEPURACIÓN	51
6.17. MATERIALES NECESARIOS.....	51
6.18. LA LISTA DE FUNCIONES DEL SISTEMA	52
CAPÍTULO 7. FASE III.....	56
7.1. DESCRIPCIÓN DE LA FASE III	56
7.2. DISEÑO DE LA BASE DE DATOS	57
7.2.1. CRITERIOS APLICADOS PARA EL DISEÑO.....	57
7.2.2. LA BASE DE DATOS BAZARBASE	58
7.2.3. DESCRIPCIÓN DE LOS CAMPOS DE LA BAZARBASE	58
7.3. DEFINICIÓN DE LOS PROGRAMAS	62
7.3.1. MÓDULO INICIO DE SESIÓN	62
7.3.2. MÓDULO FUNCIONES-GENERALES	63

7.3.3. MÓDULO PUESTEROS.....	65
7.3.4. MÓDULO CAJEROS.....	67
7.3.5. MÓDULO ADMINISTRADORES.....	68
7.3.6. MÓDULO CONTAGIOS	68
7.3.7. MÓDULO TODO-LECTORAS-Y-TARJETAS	69
7.3.8. MÓDULO CONCENTRACIÓN DE DATOS.....	69
7.3.9. MÓDULO CONSULTAS-DIVERSAS.....	70
7.4. LA BASE DE DATOS DE LA PRIMERA VERSIÓN DEL PAQUETE	70
CAPÍTULO 8: FASE IV	71
8.1. LOS PROGRAMAS	71
8.1.1. RUTINA DE CONEXIÓN CON LA LECTORA.....	71
8.1.2. RUTINA DE CONEXIÓN CON LA TARJETA.....	72
8.1.3. RUTINA DE DESCONEXIÓN DE LA TARJETA	73
8.1.4. RUTINA DE DESCONEXIÓN DE LA LECTORA	73
8.2. DOCUMENTACIÓN.....	73
8.2.1. LA LISTA DE FORMAS DEL PRYBAZAR	73
8.3. INSTRUCTIVOS DE OPERACIÓN	74
8.4. INSTALACIÓN PARA PROBAR EL SISTEMA	74
8.5. DETALLES DE LOS PROCEDIMIENTOS DE EMERGENCIA Y EXCEPCIÓN.....	75
8.6. DATOS PARA LA DEMOSTRACIÓN DEL PAQUETE	75
CAPÍTULO 9. FASE V	76
CONCLUSIONES	80
REFERENCIAS	81
ANEXOS	84

ÍNDICE DE FIGURAS

		Página
Figura 1	Tarjeta Inteligente seccionada	5
Figura 2	Tarjeta Hibrida, muestra como se encuentran los 2 chips	6
Figura 3	Ejemplo de Tarjeta Superinteligente	7
Figura 4	Lector de TI de contacto	7
Figura 5	Lector de TI de contacto con reconocimiento biométrico	8
Figura 6	Lector de TI de contacto tipo PIN-PAD	8
Figura 7	Diagrama que ejemplifica el efecto del contagio mediante las TI	20
Figura 8	Tarjeta elegida para usarse en el sistema BAZAR	45
Figura 9	Lectora elegida para usarse en el sistema BAZAR	46
Figura 10	Modelo de datos conceptual	52
Figura 11	La base de datos BazarBase	58
Figura 12	Forma de Ventas	66
Figura 13	Forma de Recargas/Retiros/Devoluciones	67

ÍNDICE DE TABLAS

	Página
Tabla 1 Usuarios	59
Tabla 2 Operaciones del Administrador	59
Tabla 3 Cajas	60
Tabla 4 Caja-Cajero	60
Tabla 5 Operaciones de Cajero	60
Tabla 6 Puestos	61
Tabla 7 Puesto_Puestero	61
Tabla 8 Productos_del_Puesto	61
Tabla 9 Puesto_Ventas	61
Tabla 10 Módulos de los Programas	62

EL EFECTO DEL CONTAGIO EN SISTEMAS DE INFORMACIÓN POR MEDIO DE TARJETAS INTELIGENTES

Javier Hernández Ruiz
Colegio de Postgraduados, 2010

RESUMEN

Las tarjetas inteligentes se utilizan cada vez más en algunos sistemas de información. En especial se aprovechan las posibilidades que ofrecen para procesos de autenticación del titular, y para autorizaciones para ciertas actividades o accesos a áreas restringidas. Su uso como medio de pago es cada vez más difundido, aunque en algunos casos no se trata de tarjetas PVC sino de otros dispositivos en los cuales se inserta un chip. Finalmente, se usan como una base de datos móvil: los datos que se graban pueden ser leídos o actualizados en otras computadoras; probablemente la aplicación más común sean los sistemas de salud. Se concibió otro uso de las tarjetas: transferir información entre computadoras. Lo que se ha llamado efecto de contagio consiste en que una tarjeta, al ser leída por una computadora, la infecta con ciertos datos grabados en la tarjeta, y del mismo modo, la computadora a su vez infecta con otros datos a la tarjeta, es decir, le graba datos que desea enviar a otra computadora. Se diseñó un paquete para uso en bazares para ilustrar el concepto de contagio. La función principal será permitir a los visitantes pagar con su tarjeta inteligente, y los administradores del evento posteriormente pagan a los que ofrecieron productos o servicios a los visitantes. Se describe el paquete, con énfasis en los aspectos de seguridad y confiabilidad, especialmente los proporcionados precisamente por la inclusión del contagio como parte del mismo. Se describe el proceso de desarrollo del paquete y el producto final obtenido.

Palabras Clave: Bases de datos distribuidas, Computación Móvil, Seguridad, Integridad y Protección, Sistemas, Tarjetas Inteligentes.

ABSTRACT

The use of smart cards in information systems has become widespread. Though they are especially used in processes involving authentication or authorization, an ever increasing number of applications use them as payment cards, though in these sometimes other devices are used to imbed a chip, not only PVC cards. Many systems make use of them as mobile databases: data is recorded on a card, and thus is available in other computers. Health systems are amongst the main users of this feature. An additional use of smart cards was formulated: they could be used to transfer information from one computer to others. Thus the contagion effect was conceived: a card is infected by every computer it visits (the computer will record data on its memory) and vice-versa, the card will infect computers with some of the data it transports. This in many instances can replace a network as far as having data available in several computers, amongst other purposes, as a means of backup or replication of data. A software product called BAZAR was defined and developed using this concept, especially to strengthen security and reliability issues. Visitors to a bazar will pay vendors with their smart card, and vendors will collect their amounts due from a central computer, which will concentrate all the sales made. The product is described, as well as the activities performed during its development.

Keywords: Distributed databases, Mobile Computing, Security, Integrity and Protection, Smartcards, Systems.

CAPÍTULO 1. INTRODUCCIÓN

Se formuló un proyecto de investigación relacionado con el uso de tarjetas inteligentes (TI) en Sistemas de Información (SI). La primera etapa consistió en hacer una relación de los tipos de sistemas que usan tarjetas inteligentes, y en especial, cuáles de las características que ofrecen son aprovechadas por cada uno de los SI. Se limitó el estudio a aquellos SI que utilizaban la memoria de la TI para almacenar datos, y determinar el uso de estos datos, es decir, para qué usaban los sistemas esa información. Esto a su vez permitió clasificar a los SI en los que usan la memoria EEPROM (regrabable) para efectos de autorización y los que la usan como datos para operaciones futuras. El estudio de estos últimos sistemas demostró que cada vez más se usan las TI para sistemas de salud pública, en los que se actualizan los datos médicos del paciente para que, en visitas futuras al mismo médico u otro, estén disponibles y puedan, a su vez, ser actualizados. En otras palabras, se usa la TI como una base de datos móvil. La característica que permite este uso es evidente: los datos están donde está el paciente, puesto que lleva la tarjeta consigo.

Esto sugirió una pregunta: ¿no se podría aprovechar la TI como dispositivo móvil, pero ahora no solo para poner a disposición los datos donde esté la TI, sino para aprovechar el hecho de que se usa en diversas computadoras? Puesto que la respuesta fue afirmativa, se formuló un concepto: se usaría una TI para transportar (copiar) datos de una computadora a otra. La limitante del impacto de esta posibilidad sobre el sistema para el que se usa es que no se sabe a cuáles computadoras les llegan estos datos originados en otras computadoras. De modo que el concepto, aunque aplicable en cualquier sistema, puede ser insuficiente o aún inútil para otros.

La etapa final del ante-proyecto consistió en encontrar un sistema (o varios) que pudieran aprovechar esta capacidad de la TI como dispositivo de transporte. De una vez, se bautizó esta posibilidad como el “efecto contagio”: la TI se infecta con los datos de una computadora que “la lee”, y a su vez infecta a todas las que visita en el futuro.

El sistema seleccionado para el proyecto de investigación es el BAZAR. Se describe como un paquete que servirá a bazares para que los visitantes del mismo efectúen sus pagos en los puestos con una TI como medio de pago. En el bazar habrá computadoras dotadas de una lectora de tarjetas, pero no necesariamente estarán conectadas entre sí de algún modo, por ejemplo, por una red. El efecto contagio se aprovecha para darle seguridad y confiabilidad al sistema, especialmente para evitar pérdidas de información producto de descomposturas de dispositivos de hardware, pero también para prevenir o detectar ciertos tipos de acciones no deseadas que pudieran afectar negativamente a algún actor del sistema.

La descripción de las actividades realizadas y los resultados obtenidos se dividieron en capítulos. Tras una relación de los conceptos y conocimientos utilizados, el marco teórico, en particular los aspectos de las tarjetas inteligentes en cuanto a sus características físicas y lógicas (su funcionalidad), que constituyen el Capítulo 2, se describe la metodología utilizada para el desarrollo del paquete BAZAR, que fue el de las 5 fases de un sistema. Se dedica el Capítulo 3 al concepto central relacionado con el uso de las TI, el contagio. En el capítulo 4 se presenta una descripción general del paquete BAZAR, y se exponen los motivos que llevaron al diseño de las dos versiones del mismo. Los capítulos del 5 al 9 describen a detalle las 5 fases del desarrollo del sistema, y las conclusiones incluyen algunos usos adicionales – además de los bazares – del paquete.

CAPÍTULO 2. MARCO TEÓRICO Y METODOLOGÍA

Introducción

Para el proyecto de investigación que se describe en esta tesis, se usaron diferentes conceptos, conocimientos y en general, aspectos teóricos y prácticos de varias componentes de un sistema y de los elementos de todo tipo que se contemplaron para su inclusión o aprovechamiento en el proyecto actual.

Los elementos descritos, así como el nivel de detalle de sus descripciones, se seleccionaron en base al criterio: si no lo usó el proyecto, no se incluye. Por lo tanto, se puede considerar que este capítulo provee el marco teórico de la investigación, y la metodología utilizada para el proyecto mismo.

2.1. TARJETAS INTELIGENTES Y LECTORAS DE TARJETAS

La historia de las tarjetas inteligentes

El surgimiento de las tarjetas plásticas comenzó en USA alrededor de los años 50's. El bajo costo del PVC y la durabilidad del material hicieron que su uso fuera cada vez mas frecuente, reemplazando así a las tarjetas de cartón que se usaban anteriormente.

Las primeras tarjetas plásticas usadas como medio de pago fueron implementadas por "Diners Club" en 1950. Inicialmente fueron pensadas para cierta clase de personas, por lo que también sirvieron como símbolo de estatus, permitiendo a su titular pagar con su "nombre y prestigio" en lugar de efectivo. Cabe mencionar que solo restaurantes y hoteles selectos aceptaban estas tarjetas, por lo que llegaron a ser conocidas como tarjetas de viajes y entretenimiento (travel and entertainment cards).

Posteriormente, con la entrada de Visa y Mastercard al campo del dinero plástico en forma de tarjeta de crédito, el uso de las tarjetas plásticas se hizo cada vez más generalizado.

Con el tiempo se le fueron agregando mejoras a las tarjetas, principalmente como medidas de seguridad, como por ejemplo el imprimir en relieve el nombre del portador, imprimir también un código único de tarjeta, etc. Pero el principal

avance consistió en la incrustación de una banda magnética en la tarjeta, lo que permitió el almacenamiento de datos digitales en la tarjeta, legibles solo con el uso de un dispositivo especializado.

Sin embargo, a medida que la tecnología fue avanzando, estas tarjetas comenzaron a tener ciertas debilidades, encabezadas por el hecho que cualquier persona con acceso al equipo adecuado podría leer, borrar y grabar datos en la tarjeta sin restricción alguna.

En los años 70 hubo otro gran avance en cuanto a las tarjetas: fueron inventadas y patentadas las tarjetas inteligentes. Aun ahora existen algunas discusiones sobre quien fue el inventor original; Juergen Dethloff de Alemania, Ronald Moreno de Francia o Kunitaka Arimura de Japón.

El primer uso masivo de las tarjetas inteligentes fue en Francia en 1983 para el pago telefónico. En 1984 se realizó en Alemania un proyecto piloto también sobre pago telefónico, pero en este proyecto se evaluaron esencialmente 3 tipos de tarjetas: las tarjetas de banda magnética, las tarjetas holográficas (almacenamiento óptico) y las tarjetas inteligentes. En esta situación en particular, las tarjetas inteligentes casi de inmediato demostraron satisfacer todas las expectativas, en especial respecto a la fiabilidad y protección contra la manipulación, además de que prometían un alto grado de flexibilidad, por lo que de inmediato surgieron otras aplicaciones.

En 1996 Visa, Mastercard y Europay publicaron un estándar de interoperabilidad para el pago con tarjetas inteligentes; para 1997 había varios millones de tarjetas circulando para pago telefónico en Francia y Alemania. En el 2000 el estándar llamado EMV (Europay Mastercard Visa) fue revisado. Dicho estándar se ha introducido gradualmente a nivel mundial y ha ido reemplazando poco a poco a las tarjetas de banda magnética.

El desarrollo de las tarjetas inteligentes, combinado con la informática electrónica, ha resultado en nuevas posibilidades para dar solución a varios de los problemas antes mencionados. Actualmente hay muchas aplicaciones que se basan en el uso de tarjetas inteligentes, para lo cual también se han desarrollado

diversos tipos de tarjetas con diferentes tecnologías y atributos, de modo que los sistemas pueden seleccionar la tecnología apropiada.

Tras mencionar estos aspectos relacionados con el surgimiento de las tarjetas, veamos de qué se trata. Básicamente son tarjetas plásticas, al igual que las de banda magnética, pero en lugar de la banda tienen un chip incrustado con una mayor capacidad de almacenamiento (figura 1).



Fig. 1 Tarjeta Inteligente seccionada

De acuerdo a sus características, las tarjetas pueden ser clasificadas siguiendo varios criterios tales como capacidades, tamaño, etc. Sin embargo, la clasificación más común es la de tarjetas de contacto (es necesario insertar la tarjeta en una lectora) y tarjetas sin contacto (basta con acercar la tarjeta a cierta distancia de la lectora) cada una regida por diferentes estándares ISO. Estos no se detallan puesto que no fueron utilizados en la investigación que se describe.

Hay otra distinción entre tarjetas inteligentes de acuerdo a las funciones que ofrecen. Se describen a continuación los principales tipos de tarjetas que se ofrecen para diversas aplicaciones.

Tarjeta de memoria: son tarjetas que sirven únicamente como contenedores, disponen de un sistema operativo limitado con comandos básicos de lectura y escritura. Estas pueden ser tarjetas de memoria libre o memoria protegida; estas últimas tienen áreas de memoria protegida. Los diferentes modelos o tipos de tarjetas de memoria difieren entre sí exclusivamente por el tamaño de la memoria EEPROM (Electrically Erasable Programmable Read Only Memory) (Zoreda y Otón, 1994).

Tarjetas microprocesadas: contienen una estructura análoga a la del ordenador (procesador, memoria ram y rom), y tienen un sistema operativo en el que se incrustan, desde la fabricación, una o más aplicaciones que contienen una serie de comandos que pueden ser invocados. Los datos almacenados en estas tarjetas no pueden ser accesados de manera directa. El procesador toma el control de los datos y dependiendo de las condiciones (password, encriptación, o id externo) permite el acceso a los datos (Zoreda y Otón, 1994).

Tarjetas con coprocesador o tarjetas criptográficas: contienen un procesador criptográfico (criptoprocesador) que implementa dos algoritmos estándar de llave privada (DES y AES) y un algoritmo estándar de llave pública (ECC) (Eslami et al., 2006). Estas tarjetas usan cifrados y firmas digitales, por lo cual son usadas para almacenar de forma segura un certificado digital u otro tipo de información que requiere protección contra alteraciones o uso indebido.

Tarjetas Java: tienen la capacidad para ejecutar mini aplicaciones Java. El procesador de estas tarjetas contiene una pequeña máquina virtual de Java a la que se le pueden cargar dinámicamente aplicaciones desarrolladas específicamente para ese entorno (Wikipedia, 2010).

Tarjetas híbridas: como se muestra en la figura 2, combinan las dos tecnologías (tienen un chip de contacto y otro sin contacto). Se comunican con el sistema usando los métodos propios de cada tecnología. Cabe mencionar que estos chips pueden tener un microprocesador o ser simplemente chips de memoria (Hansmann et al. 2002).

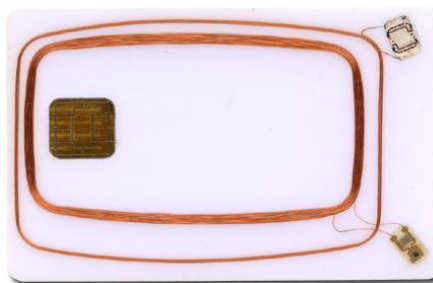


Fig. 2 Tarjeta Híbrida, muestra como se encuentran los dos chips

Tarjetas duales: son casi iguales a las híbridas, excepto que contienen un único chip que desarrolla ambas funciones (Wikipedia, 2010).

Tarjetas superinteligentes (figura 3): estas cumplen con las mismas funciones que las tarjetas microprocesadas, pero están equipadas con un pequeño teclado, un display y una batería. Estas tarjetas tienen un sistema autosuficiente para la ejecución de aplicaciones sencillas (Zoreda y Otón, 1994).



Fig. 3 Ejemplo de Tarjeta Superinteligente

Lectoras de tarjetas inteligentes

Las lectoras se pueden dividir de acuerdo al modo en el que leen y procesan las tarjetas, así como su conexión con una computadora.

Lectoras de contacto: como su nombre lo dice, funcionan insertando ya sea la tarjeta completa o bien solamente el chip de la tarjeta en la ranura de la lectora (figura 4).



Fig. 4 Lector de TI de contacto

Lectores sin contacto: pueden leer y grabar las tarjetas sin necesidad de que estas estén en contacto directo con el chip de la tarjeta; la distancia para realizar estas operaciones varía según el modelo de la lectora.

Lectores con reconocimiento biométrico: además de leer la tarjeta, tienen algún tipo de autenticación biométrica para validar la tarjeta al usuario. Se ofrecen con o sin contacto (figura 5).



Fig. 5 Lector de TI de contacto con reconocimiento biométrico

Lectores pin-pad: cuentan con un teclado y un display para poder realizar operaciones en una tarjeta. Pueden o no estar conectados a una PC. Este tipo de lectores son los que se usan comúnmente como terminales bancarias (figura 6).



Fig. 6 Lector de TI de contacto tipo PIN-PAD

Lectores de saldo (balance readers): son lectores inalámbricos que leen una sección específica de una tarjeta para poder desplegar el saldo de dicha tarjeta.

2.2. SISTEMAS DE INFORMACIÓN

2.2.1. MÉTODO DE DESARROLLO

Hay diversos métodos que se utilizan para la definición de un sistema. Mención especial merecen el Método estructurado y el Orientado a Objetos. Sin embargo, en muchas ocasiones se combinan ambos. En la investigación que se describe, se utilizó el método de las 5 fases de Bauer Mengelberg¹. En este

¹ El material relacionado con el método de análisis y diseño de sistemas fue tomado de apuntes de los alumnos de los cursos de Sistemas de Información, pero en la versión que se presenta aquí, editados por el director de esta tesis.

método, las actividades se agrupan de modo que los resultados de cada fase constituyen la base para la siguiente fase. En cada aplicación del método, es decir, el diseño de un sistema, se decide – de antemano – si las fases son finales o no, donde esto indica que se pueden hacer cambios a una fase anterior o no está disponible esta alternativa. En rigor, se postula que no se puede; en la práctica, en pocas ocasiones se logrará eliminar cambios a una decisión tomada previamente. El éxito del uso del método depende de dos factores: la aplicación rigurosa del método, es decir, que todos los integrantes de los equipos de trabajo *sepan* que su trabajo será final, y por lo tanto serán culpables de las consecuencias que resulten de sus omisiones o errores, y la flexibilidad de no hacer un disparate (cuando se sabe que se trata de tal) porque alguien, con anterioridad, cometió un error.

Solo se presentan aquí detalles de actividades o rubros cuando difieren de las otras teorías, o cuando se aplicará algún método especial para lograr un propósito.

Fase I: Se describe el sistema en términos muy generales.

Debe estar claro desde el principio si se trata de un sistema “a medida” o un paquete. En el primer caso, se aclara si se trata de un sistema nuevo o de uno que reemplaza a uno anterior, y en esta última situación, si se reemplaza un sistema especial o se trata de integrar varios sistemas existentes en uno solo, con la posible adición de componentes nuevas. Todo el resto del proceso de definición del sistema difiere considerablemente para los 4 casos.

Se determinan los actores del sistema (stakeholders) y se determinan el papel que desempeñan en cuanto a lo que necesitan y proporcionan. Cualquier método es aplicable para esta etapa: se puede comenzar por los datos y la relación entre los mismos, por las funciones principales o por los actores.

A continuación se enfocan los asuntos de viabilidad del proyecto (la construcción del sistema). Intervienen dos considerandos fundamentales: los recursos que demandará su elaboración e implementación y el impacto que tendrá sobre distintos aspectos de la empresa, en especial el que se refiere a la

operación del negocio o de las actividades, y el grado de cambio organizacional que implicará su uso. Esto resulta en una determinación de:

- Los recursos humanos, dinero, equipo (hardware, periféricos y comunicaciones), instalaciones físicas y software.
- El calendario aproximado de elaboración e instalación del sistema.
- La naturaleza del proyecto, especialmente en cuanto a la relación del equipo de trabajo encargado de elaborar el sistema y la empresa.
- La determinación de riesgos de todo tipo, especialmente la probabilidad de un fracaso, pero también el tipo de impactos que tendrían ciertas fallas sobre un área de la empresa o sus actores.

Con estos elementos, se toma la decisión: se hace o no se hace el sistema.

Observación importante

Hay una excepción notable a este último elemento de esta fase. En el caso de un paquete, y aun en algunas aplicaciones, se podrá optar por hacer el sistema en 2 o más “versiones”, de las cuales la primera se puede interpretar como un prototipo exhaustivo (es un sistema auténtico, no simulado) o como un “globo sonda”. El propósito es elaborar un sistema con menos recursos para determinar si presenta faltantes, deficiencias u otros aspectos, entre los cuales sobresalen los procedimientos. Puede ser que un procedimiento diseñado no funcione por las causas que fueren, entre las que destacan la no aceptación por algunos de los involucrados o alguna deficiencia en cuanto a control.

Cabe agregar que este tipo de situación – la elaboración de una versión que se podría llamar de prueba - no es aceptada como válida por muchos teóricos de sistemas. Sin embargo, en muchas situaciones, no hay alternativa. Esto es especialmente cierto en paquetes, donde muchas veces conviene decidir hacer dos versiones aun antes de iniciar el proceso de diseño del paquete.

Fase II: Diseño conceptual del sistema.

El resultado de esta fase es la descripción de la arquitectura del sistema, y el detalle de los elementos que se describen a continuación.

Se determinan a detalle los requisitos del sistema, utilizando las técnicas que resulten aplicables (entrevistas, cuestionarios, reuniones de trabajo, estudio de un sistema anterior o de las especificaciones de cada área de la organización involucradas, etc.) Esto resulta en una lista de actores (agrupados en cuanto a lo que necesitan y usan del sistema).

Se analizan las características de las personas que serán actores del sistema, para determinar si existen diferencias entre personas que tienen los mismos roles en el sistema. Un ejemplo trivial: algunos solo hablan francés. Estos análisis a su vez pueden modificar requisitos o agregarle algunos al sistema.

Decisión fundamental: se selecciona un objetivo fundamental o principal del sistema. Esto se hace de modo que todos los demás objetivos se puedan interpretar como de apoyo al objetivo fundamental. Una vez seleccionado este objetivo, se determina una entidad de datos principal, alrededor de la cual se elaborará el modelo conceptual.

Se definen:

- Un modelo de datos conceptual
- La lista de funciones del sistema
- Las excepciones contempladas y excluidas de todo tipo
- El rol de los actores del sistema
- Procedimientos y controles
- Interfaces, especialmente formas de ingreso de datos, consultas e informes
- Considerándose referentes a la confidencialidad y seguridad de los datos. Aquí se especifica el nivel de protección de los datos en todos sus sentidos, y se definen a detalle los procesos, controles y dispositivos para obtener los grados de seguridad necesarios.
- Un calendario de actividades, preferiblemente entregado como control de proyecto
- Se elabora un proyecto de instalación del sistema y se designa a los integrantes del equipo humano que realizará las tareas resultantes
- Se validan las componentes con las autoridades de la empresa encargadas del proyecto

- Se confirma o nombra otro líder del proyecto.

Fase III: Diseño físico del sistema.

En esta Fase se diseñan las bases de datos, archivos y programas, a nivel detallado. También se elaboran las especificaciones técnicas de equipo, infraestructura física y de comunicaciones y los proveedores de los materiales y servicios que se usarán.

- Se deciden detalles científicos, técnicos, algoritmos específicos, dispositivos de software y hardware para la protección de los datos y respaldos.
- Se procede a la contratación del personal para la siguiente fase.
- Se diseña la papelería u otros elementos que usará el sistema.
- Se elaboran controles de proyecto por tipo de actividad, y se nombra un encargado de cada subproyecto.
- Se asignan las actividades a los responsables de efectuarlas.

Fase IV: Implementación (programación, instalaciones de equipo y comunicaciones, pruebas)

- Se elaboran, prueban y depuran los programas.
- Se documentan los programas y procesos.
- Se elaboran instructivos de uso de los programas.
- Se instalan y prueban las componentes físicas de todo tipo.
- Se elaboran los procedimientos de emergencia.
- Se preparan datos para la demostración del sistema.

Fase V: Instalación del sistema

- Se instalan las componentes de todo tipo. En el caso de contratistas, se supervisa y confirma la instalación correcta de los productos y servicios.
- Se hacen demostraciones de los procesos a los actores involucrados.
- Se elabora el material de capacitación.
- Se capacita a los usuarios del sistema, incluyendo las situaciones de excepción o emergencia.

- Se instalan los procedimientos de seguridad y confidencialidad.
- Se dan de alta los usuarios del sistema.
- Se supervisa el inicio del uso del sistema.
- Se procede a la firma de finalización del proyecto.

2.2.2. SISTEMAS VERSUS PAQUETES

El término “paquete” se usa en este trabajo para designar un programa o sistema que no tiene un uso específico para el cual fue diseñado y elaborado. De ese modo, contempla el uso para diversas situaciones. Se llamará “instancia del paquete” a una aplicación (situación) que haga uso de las funciones proporcionadas por el paquete (software, hardware o una combinación de los mismos). Una aplicación de un paquete puede ser parcial, es decir, no usa todo el potencial del paquete. EXCEL es un paquete. El programa que registra el tráfico de telefonía de un local, hecho *ad hoc* para ese uso específico, no lo es (aunque naturalmente puede usar otros paquetes.)

A pesar de que es aplicable el método de las 5 fases, en el caso de un paquete difieren muchas de las actividades que integran su desarrollo. Se puede resumir la diferencia de este modo: aquí no se determinan los requisitos, sino se formulan. Esto a su vez implica la especificación de los actores, pero en especial surge una actividad nueva: ¿el paquete servirá para algo? En otras palabras ¿habrá instancias (reales) del paquete?

Las actividades que se realizan, especialmente en las fases I y II, son las mismas, pero el modo en el que se hacen es diferente en el caso de un paquete. En particular, la descripción de los actores ahora se refiere a los potenciales usuarios de una instancia.

Para la definición del paquete producto de la investigación que se describe, se usará el método de las 5 fases en su versión “paquetes”. La descripción del paquete se presentará precisamente de acuerdo a estas fases. En cada una de ellas, se describirán las actividades específicas de la fase, además de los resultados de estas actividades.

2.3. BASES DE DATOS RELACIONALES

2.3.1. CONCEPTOS FUNDAMENTALES

No se describen aquí las bases de datos, excepto para mencionar que un manejador de bases de datos relacionales (RDBMS) ofrece las funciones necesarias para crear y usar una base de datos.

El esquema de una base de datos

Base de Datos

 Tablas

 Campos

 Índices

 Relaciones

 Procedimientos almacenados

 Disparadores, funciones

 Vistas

Componentes de protección

 Palabras claves

 Autorizaciones

 Protección física de los datos

Funciones que proporciona el RDBMS

DDL: el lenguaje de definición de datos con el que se crea el esquema de una base, que consiste precisamente en la especificación de cada uno de los elementos de la base.

SQL: permite actualizar y consultar los datos contenidos en una base de datos.

UTILERIAS para diversas funciones de apoyo: respaldo, copiado, depuración, compresión, cambios a los datos consecuencia de alguna modificación.

Las bases de datos relacionales han dominado el mercado de bases de datos en las últimas décadas. Ello se debe especialmente al grado de

estandarización de los manejadores, que deben conformar con el estándar ANSI SQL/92. Esto a su vez ofrece la oportunidad de cambiar de proveedor de bases de datos sin esfuerzo.

El hecho de que la base sea relacional también tiene muchas ventajas. Como consultas, consistencia y confiabilidad.

Diseño de una base de datos

Cuando se usa una base de datos para una situación específica, se crea la base de datos en base a un esquema que se especifica para ella. Este proceso es lo que se conoce como diseño de una base de datos relacional. Para aprovechar al máximo las características de las bases relacionales, se han formulado 5 criterios de normalización, mismos que ayudan a obtener especialmente la consistencia de los datos, pero también para evitar repeticiones superfluas.

Criterios de normalización

Esencialmente hay cinco criterios de normalización (existen otros criterios derivados de estos) para una base de datos, los cuales consisten en:

Primer forma normal (1FN): una relación R se encuentra en 1FN sí y sólo sí todos los dominios simples subyacentes contiene solo valores atómicos. Es decir que el cruce de una fila con una columna solo tiene un dato (no existen grupos repetitivos).

Segunda forma normal (2FN): una relación R esta en 2FN sí y sólo sí está en 1FN y todos los atributos no clave, dependen funcionalmente de forma completa de la clave primaria.

Tercera forma normal (3FN): una relación R esta en 3FN sí y sólo sí está en 2FN y todos los atributos no clave dependen de manera no transitiva de la clave primaria.

Cuarta forma normal (4FN): una relación R esta en 4FN sí y sólo sí está en 3FN y no existen dependencias multivaluadas.

Quinta forma normal (5FN): una relación R esta en 5FN sí y sólo sí está en 4FN y toda dependencia de reunión R es una consecuencia de las claves candidatas.

En otras palabras para lograr tener una base relacional hasta la quinta forma normal debemos hacer lo siguiente:

1FN:

- Eliminar los grupos repetitivos de las tablas individuales
- Crear una tabla separada por cada grupo de datos relacionados
- Identificar cada grupo de datos relacionados con una clave primaria

2FN:

- Crear tablas separadas para aquellos grupos de datos que se aplican a varios registros
- Relacionar estas tablas mediante una clave externa

3FN:

- Eliminar aquellos campos que no dependan de la clave

4FN:

- En las relaciones varios a varios, entidades independientes no pueden ser almacenadas en la misma tabla

5FN:

- La tabla original debe poder ser reconstruida a partir de las tablas resultantes de la división

Situaciones en las que se pueden violar algunas normas

En general, se dice que para un sistema de transacciones (operaciones registradas en forma individual) se recomienda el uso de una base de datos normalizada. Sin embargo, para otros usos de las bases, por ejemplo para bodegas de datos, esto puede resultar en tiempos de respuesta más prolongados, por lo que se incluyen en el diseño elementos que no cumplen algunos de los criterios de normalización. Dos ejemplos pueden ilustrar este concepto.

Si una base de datos contiene elementos que se suman en algunos usos, por ejemplo se suman, una base normalizada no contendría un campo que

reflejara tal suma. Sin embargo, en un almacén de datos que necesite tanto la suma como el desglose, sería conveniente incluir la suma como campo para no calcularla en cada instancia de uso.

Otra situación común es la de reemplazar un relación por la repetición de un dato. En lugar de interpretar un código en cada uso, se le agrega a la tabla en cuestión la descripción o especificación del significado del código.

Sin embargo, en algunos casos hay otro tipo de violación que resulta sumamente atractivo por cuestiones de desempeño o de espacio en disco. En particular, el desempeño de una base de datos depende en forma muy considerable del número de elementos (registros) de sus tablas. En muchos casos, conviene simular la inclusión de un arreglo de elementos de la misma naturaleza en una misma tabla. Por ejemplo, en lugar de crear un registro para la venta total mensual de un producto, se incluyen 12 campos que reflejan los meses. Como se verá, en el paquete diseñado en el proyecto que se describe, precisamente se hizo uso de esta posibilidad.

Algunos manejadores RDBMS

La clasificación más importante en la actualidad es la de propietarios o libres. Sin embargo, cada vez aparecen más variantes de bases de datos relacionales que tienen diferencias en cuanto a la forma en la que se almacenan los datos y se implementan índices de las tablas. Puesto que no se contemplaron RDBMS excepto los tradicionales (row databases), no se describen aquí las bases verticales (los campos de una tabla se almacenan por columna) y otras arquitecturas que flexibilizan los conceptos, como son las híbridas (pueden contener objetos como campos de una tabla).

Se contempló el uso de un manejador tipo freeware (MySQL) pero se descartó porque complica notablemente la instalación de una instancia de un paquete, en comparación con ACCESS.

Como se verá, como decisión de diseño del paquete se seleccionó un manejador (ACCESS) que, si bien no es gratuito, permite utilizar las bases de datos elaboradas para él sin licencia adicional, excepto la del sistema operativo.

Esto no es cierto en cuanto al diseño e implementación de la base, para la cual se necesita una licencia.

Finalmente, cabe una observación. La selección de un RDBMS se debe basar en los siguientes factores: facilidad de diseño e implementación (hay pocas diferencias entre ellos); facilidad de instalación (actividades necesarias para su uso en una nueva instancia); costos asociados. Naturalmente el nivel de soporte técnico y seguridad de los datos también deben intervenir en la selección del manejador indicado para un sistema o paquete.

CAPÍTULO 3. EL EFECTO CONTAGIO MEDIANTE TARJETAS INTELIGENTES

Como se mencionó anteriormente, el uso de tarjetas inteligentes en sistemas de información aprovecha especialmente tres servicios que podría proporcionar dicho uso:

- **AUTENTICACIÓN:** es uno de los usos más difundidos de tarjetas inteligentes en sistemas de información.
- **ACCESO A INSTALACIONES:** cada vez más se usan las TI para este tipo de control. Una persona solo puede traspasar un control si tiene permiso para ello.
- **MEDIO DE PAGO:** la tarjeta se usa como monedero electrónico.

A pesar de que en otros países, en especial los europeos y asiáticos, pero en cada vez mayor medida los sudamericanos, comenzaron a usar las tarjetas como dispositivo de almacenamiento de información relacionada con el titular de una TI, esto no sucedió en la misma medida en América del Norte. En los últimos años, esta situación se compensó con el uso de chips en un número cada vez mayor de aplicaciones en los EEUU de Norte América, y también en México. Aplicaciones para sistemas de salud y medicina en general, tarjetas de lealtad de negocios y algunos sistemas de cuentas por pagar (de los titulares de las tarjetas) han proliferado y tienen cada vez mayor demanda e impacto. Designaremos a este tipo de aplicaciones de las TI como base de datos del titular. Su utilización permite que, en cada uso de la TI en alguna computadora (equipada con el hardware y software correspondiente, incluyendo la aplicación específica para la cual se usa la TI) se puedan usar los datos grabados en la misma, y actualizarlos con lo que sucede en el nuevo uso de la misma.

Bauer Mengelberg y González Espinosa, con la participación posterior del autor de esta tesis, concibieron un uso adicional de una TI en un sistema de información. Partiendo del uso de la TI como base de datos del titular, se le agregó una función: la TI para transmisión de datos entre computadoras. Al entrar en contacto con diversas computadoras, la TI puede servir para copiar información de una a otra de ellas. Eso condujo a la especificación de un concepto.

Una TI puede “contagiar” a una computadora con datos de otra en la cual se utilizó la tarjeta. Por la evidente analogía con una situación de salud, o quizá específicamente en epidemiología, se adoptó la terminología de esa especialidad. Una TI se infecta con datos de la computadora en la cual se usa, y a su vez infecta a ésta con datos que adquirió en otras computadoras (figura 7).

El uso de este efecto de contagio puede ser considerado superfluo en muchas situaciones, especialmente en dos tipos de sistemas: las computadoras no necesitan datos de otras, o las computadoras están conectadas de algún modo que permita la transferencia de datos entre ellas, como puede ser una red u otro mecanismo, por ejemplo, si en un sistema distribuido de computadoras no conectadas por red se incluye un procedimiento para la transferencia periódica de datos de una computadora a otras. Este tipo de argumentos – que el contagio sería superfluo – es esgrimido por cuantos usuarios potenciales de este tipo de efecto son consultados acerca de su opinión sobre el concepto mismo.

Sin embargo, hay situaciones en las cuales dicho efecto puede ser determinante: los actores y usuarios potenciales tendrán sospechas o temores que sólo se podrán disminuir o eliminar con dicho efecto. El sistema que se describe en este trabajo es precisamente de ese tipo.

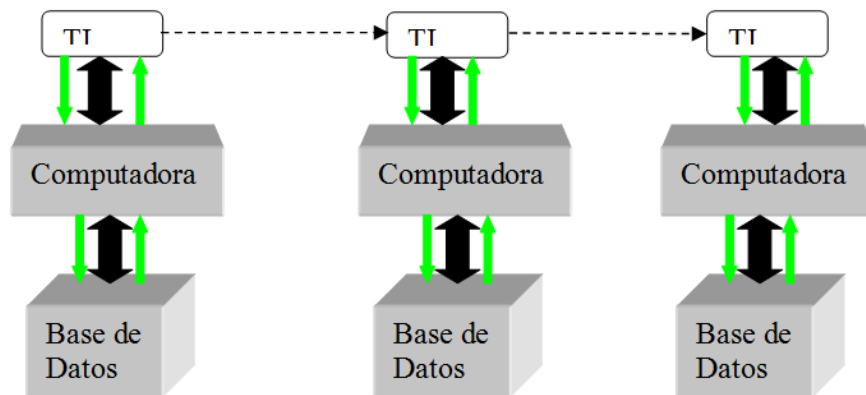


Fig. 7 Diagrama que ejemplifica el efecto del contagio mediante las TI

Diversos actores del sistema se negarían a usarlo porque imaginan situaciones en las cuales los podría perjudicar en algún sentido. Las personas se han acostumbrado a las tarjetas inteligentes como medio de pago, aun si no

confían totalmente en ellas. Pero hay diversos temores que subsisten en otras aplicaciones. Por ejemplo, si la tarjeta funciona como comprobante de algo (típicamente un pago, pero podría tratarse de otras operaciones) surge la sospecha de que alguien aducirá que no se realizó la operación registrada. El efecto de contagio permite disminuir estos temores en forma considerable, como se describirá en el sistema BAZAR en el resto de esta tesis.

CAPÍTULO 4. EL BAZAR COMO EJEMPLO DEL EFECTO CONTAGIO

4.1. SITUACIONES PARA LAS CUALES SE ELABORÓ EL PAQUETE

Se concibió un bazar, definido como venta de diversos artículos para beneficiar obras de caridad, alguna causa u organización (Dictionray.com, 2010).

Los visitantes del bazar adquirirán productos o contribuirán de otro modo con fondos a la causa o meta para la cual se organizó el bazar. Se decidió conceptualizar esta situación de este modo. Hay diversos “puestos”, y los encargados de los mismos serán “puesteros”, que pueden ser vendedores, proveedores de algún servicio o realizar otra actividad. Por ejemplo, además de comida y bebida en ciertos puestos, se ofrecen besos por un importe, tomarse la foto con alguien famoso, participar en alguna actividad (baile, competencia), asistir a un show o demostración, etc. La característica común es que el visitante paga y eso le da derecho a lo que se ofrece. Se designarán a todas las contribuciones como “ventas” o “compras”, independientemente de su naturaleza.

El dinero obtenido de este modo se concentra en un ente que denominaremos la administración del bazar. Las formas tradicionales de pago en estos bazares se pueden dividir en dos grupos: el visitante adquiere ciertos vales o boletos y los entrega como pago en los puestos, o paga en efectivo en cada uno de ellos. En la mayoría de ese tipo de eventos se adopta alguna modalidad del primer tipo, para evitar circunstancias que se imagina fácilmente el lector.

Posteriormente, la administración paga a los puesteros los importes convenidos, que pueden o no depender de lo que “vendieron”. En general, esto se realiza contra la presentación de los vales obtenidos de los visitantes.

El uso de tales vales tiene varios atributos que lo hacen poco confiable. Sin mencionar la posibilidad de la falsificación de los vales, puesto que se trata de un evento que en general es de beneficencia o consecución de fondos para alguna causa, el manejo de los vales, tanto en la emisión de los mismos y su posterior uso para el pago a los puesteros, hacen que no solo sea laborioso, sino poco confiable.

Estas circunstancias hacen evidente una alternativa: aprovechar alguna tecnología para obviar algunas de las actividades y dificultades que presentan los otros modos de pago. El uso de tarjetas de crédito podría ser una alternativa, pero no se contempló por diversas circunstancias. Las tarjetas de pago, en cambio, ofrecen muchas ventajas, mismas que, como en cualquier proceso o actividad, tienen un costo. Naturalmente, la adopción de una estrategia siempre incluirá una comparación de estos costos con los beneficios obtenidos.

El uso de computadoras ofrece una alternativa cada vez más atractiva y factible, esto último por el descenso constante de los costos asociados, empezando por las computadoras, pero incluyendo los periféricos y especialmente, la posibilidad de usar algún tipo de red.

Tras analizar algunos aspectos de los bazares, se decidió formular una alternativa nueva: usar tarjetas inteligentes como medio de pago, pero aprovecharlas para darle más solidez al sistema, especialmente en el sentido de contemplar excepciones que pudieran producir rechazo por parte de los organizadores de un evento. Por lo tanto, se definió el sistema que se describe, denominado BAZAR, con las características que se incluyen como requisitos y procedimientos del mismo.

Como se anunció, la descripción se presentará de acuerdo al método con el que se elaboró el paquete: el de las 5 fases, cada una de las cuales se incluyó en un capítulo por separado, precisamente para enfatizar este modo de diseño e implementación de sistemas.

Las dos versiones del paquete BAZAR

Como sucede en muchas ocasiones, conviene planificar el desarrollo de un paquete en dos etapas. Se elabora una versión inicial, misma que se prueba y, preferiblemente, se usa en una instancia específica. Con las observaciones y posibles situaciones no contempladas que surgen de ese uso inicial, se elabora una nueva versión que, ahora sí, se espera sea la definitiva. En el caso que nos ocupa, el paquete BAZAR, no se pudo usar en un evento por varias limitantes, entre ellas que no se pudo conseguir uno para tal efecto. Los costos asociados

influyeron en ello, puesto que inevitablemente nosotros, los integrantes del equipo de diseño del paquete, hubiéramos tenido que sufragar los gastos asociados.

Por lo tanto, se hizo una simulación de un evento, eso sí, con un número limitado de puestos, cada uno representado por una computadora desconectada – para dicho efecto – de las restantes. A pesar de que la situación no fue real, arrojó varios aspectos contemplados sólo en forma parcial en la primera versión, y algunas necesidades – o funcionalidad conveniente – que se incluyeron en la segunda versión.

Se terminaron las 5 fases para la primera versión, puesto que ello era una meta *sine qua non* del proyecto de investigación para obtener el grado académico para el que se elaboró esta tesis. Por el contrario, se llegó hasta la fase 3 en algunas actividades de la versión 2, pero no se completaron todas las de esa fase. Para efectos de la descripción del sistema en esta tesis, el director de la misma tomó una decisión: se describiría la versión 2, pero para algunos elementos – especialmente los programas – se mostrarían los de la versión original, puesto que no se terminaron los correspondientes de la otra versión. Se describirán los cambios, especialmente en cuanto al modelo de datos, para ilustrar precisamente ese aspecto de la elaboración de paquetes.

CAPÍTULO 5. FASE I DEL PAQUETE BAZAR

La decisión de elaborar o no el sistema

5.1. ACTIVIDADES DE LA FASE I

Comienza este capítulo con una enumeración de los objetivos de la fase I, expresados como siempre en sus resultados: lo que le entrega a la fase II.

1. Se describe el paquete en términos muy generales.
2. Se determinan los actores del sistema (stakeholders) y el papel que desempeñan en cuanto a lo que necesitan y proporcionan.
3. Viabilidad:
 - a. Recursos: humanos, dinero, equipo (hardware, periféricos y comunicaciones), instalaciones físicas y software.
 - b. Impacto que tendría su uso por un bazar. En el caso de un paquete, el costo que tendría para los organizadores de un evento, y las ventajas que obtendrían por usarlo.
 - c. Mercado potencial (producto del inciso anterior).
4. El calendario aproximado de elaboración del paquete. En el BAZAR, no se tomó en cuenta este elemento.
5. La determinación de riesgos de todo tipo: el tipo de los impactos de alguna falla sobre los actores (futuros usuarios de una instancia del paquete).

Resultado esperado de la Fase I

- Decisión: se elaborarán dos versiones del paquete.
- Líder: se designa líder del proyecto al titular de esta tesis.

Descripción general para proporcionarlo a la Fase II.

EVENTO; un bazar, con puestos que ofrecen productos u otro tipo de bienes o servicios; hay puestos, y cada uno tiene uno o más encargados de atender a los visitantes y recibir los importes que correspondan, Los visitantes tendrán tarjetas inteligentes con los que realizarán los pagos. Podrán recargar la tarjeta o retirar fondos de la misma cuando lo deseen. Se compensa a los puesteros de acuerdo

al servicio prestado (ventas, número de servicios, etc.) Los cajeros entregan el dinero a los administradores. Los puestos y cajeros tendrán computadoras y una lectora de TI. Las computadoras no estarán conectadas entre si, es decir, el sistema debe funcionar sin tales conexiones.

Característica fundamental agregada a la definición: los procesos de todo tipo deben tener un alto grado de confiabilidad y credibilidad por parte de los que pudieran usar el paquete.

Elemento tecnológico incluido como parte de la decisión: las tarjetas proporcionarán un efecto contagio.

5.2. DESCRIPCIÓN GENERAL DEL PAQUETE QUE SE ELABORARÁ

EQUIPO DE TRABAJO. Líder, Javier Hernández Ruiz. El equipo lo integrarán el mismo líder y el Dr. Bauer, con el apoyo del personal especializado que necesiten o consigan.

ACTORES (mencionados en singular, pero naturalmente esto no refleja que haya una sola persona de cualquiera de los tipos de actores).

- **ADMINISTRADOR:** Organizador del bazar
- **CAJERO:** el que recibe y entrega dinero a los visitantes
- **PUESTERO:** el que atiende un puesto y recibe el pago de visitantes
- **VISITANTE:** el que adquiere productos de los puestos.

ELEMENTOS DE HARDWARE

Una instancia del paquete necesitará computadoras sencillas. Especificación: deberán tener una versión del sistema operativo WINDOWS de la 95 en adelante. Para cada computadora, se necesita una lectora de tarjetas cuyas características se determinarán como parte del diseño del sistema.

Aunque sería conveniente tenerlas, no serán imprescindibles las UPS para cada una de las computadoras.

ELEMENTOS DE SOFTWARE

- Se usará exclusivamente el ACCESS como manejador de bases de datos.

- Se programará el paquete en Visual Basic 6.0.
- Se usarán los “drivers” necesarios para los equipos periféricos, en especial, las lectoras de tarjetas.
- No se utilizarán otros productos que impliquen el pago de licencias por uso de SW.

Funcionalidad general que deberá proporcionar

Emitir TI a los visitantes. Admitir carga y descarga de efectivo de una TI.
Registrar ventas en los puestos. Concentrar los fondos en computadoras de los administradores. Pagar a los puesteros los importes calculados a partir de sus ventas. Elaborar informes de los ingresos resultantes del evento.

A pesar de que ya hubo definiciones más detalladas de algunos puntos, como se trató de un equipo de trabajo que realizaría todas las fases, no se entregan estos detalles para la Fase II.

CAPÍTULO 6. FASE II

Diseño conceptual del sistema

6.1. ACTIVIDADES DE LA FASE II

El resultado de esta fase es la descripción de la arquitectura del sistema, y el detalle de los siguientes elementos:

REQUISITOS: se determinan a detalle los requisitos del sistema, utilizando las técnicas que resulten aplicables (entrevistas, cuestionarios, reuniones de trabajo, estudio de un sistema anterior o de las especificaciones de cada área de la organización involucradas, etc.). Esto resulta en una lista de actores (agrupados en cuanto a lo que necesitan y usan del sistema).

ACTORES; se analizan las características de las personas que serán actores del sistema, para determinar si existen diferencias entre personas que tienen los mismos roles en el sistema. Estos análisis a su vez pueden modificar requisitos o agregarle algunos adicionales al sistema.

OBJETIVO FUNDAMENTAL: se selecciona un objetivo fundamental o principal del sistema. Esto se hace de modo que todos los demás objetivos se puedan interpretar como de apoyo al objetivo fundamental.

ENTIDAD FUNDAMENTAL: se determina una entidad de datos principal, alrededor de la cual se elaborará el modelo conceptual.

ARQUITECTURA: computadoras, dispositivos, conexiones, suministros periódicos y de emergencia, arquitectura del software (qué y cómo se ofrecerá).

HW QUE USARÁ EL SISTEMA: computadoras, periféricos, servidores, alambrado o dispositivos para uso inalámbrico, unidades de respaldo de energía y de reemplazo de componentes que fallan, dispositivos externos para almacenamiento de datos y otros específicos para el paquete.

MODO DE USO DEL PAQUETE (instanciación): quien lo ofrece, opera e instala; cómo se describe el sistema a usuarios potenciales (instancias del paquete); pasos de una instancia y requisitos para llevarla a cabo; aspectos de costo y cobranza.

Los resultados específicos que se entregan a la fase III son:

- Un modelo de datos conceptual
- La lista de funciones del sistema
- Las excepciones contempladas y excluidas de todo tipo
- El rol de los actores del sistema
- Procedimientos y controles
- Interfaces, especialmente formas de ingreso de datos, consultas, informes
- Considerándolos referentes a la confidencialidad y seguridad de los datos. Aquí se especifica el nivel de protección de los datos en todos sus sentidos, y se definen los procesos, controles y dispositivos a detalle.
- Un calendario de actividades, preferiblemente entregado como control de proyecto.
- Se elabora un proyecto de instalación del sistema y se designan a los integrantes del equipo humano que realizará las tareas resultantes.
- Se validan las componentes con las autoridades de la empresa encargadas del proyecto.
- Se confirma o nombra otro líder del proyecto.

6.2. REQUISITOS

En base a la información recabada en la fase anterior y los objetivos específicos, se determinaron los requisitos contemplados en el diseño del paquete.

Se usó un método de determinación estructurado, aunque admitió adiciones a los requisitos de niveles superiores. De ese modo, se enlistaron los requisitos determinados en una fase inicial de este proceso, y se les agregaron detalles a medida que surgían de sugerencias, especialmente resultantes de otros requisitos.

- Datos: los datos que necesitan los diversos actores, y las fuentes de los mismos.
- Tarjetas emitidas; Saldo de una TI; carga y recarga de TI (sus importes); ventas de un puesto; importes recibidos por cajeros; importes recibidos por la

administración; importes entregados (pagos) a los puesteros; transferencia de dinero entre cajeros y administradores.

- Datos de apoyo: control de acceso a funciones del sistema: catálogos de productos y precios.

SEGURIDAD Y CONFIDENCIALIDAD

- Se determinó que los datos no serían confidenciales.
- Todos los datos del sistema estarán protegidos por una cifra de auditoría, calculada en base a los mismos para detectar alteraciones no autorizadas.
- Se incluirán medidas para evitar la clonación de tarjetas.
- Se incluirán medidas para evitar el uso de programas y TI de un evento en otro evento.
- Se incluirán dispositivos que limiten a los usuarios del sistema (en una instancia) en cuanto a lo que pueden hacer con los programas.

6.3. ACTORES

Observación; se definieron en la FASE I (en este caso). Se analizaron sus contribuciones y demandas al paquete resultante.

6.3.1. ADMINISTRADORES

6.3.1.1. PROPORCIONAN

- Definen a los usuarios; les asignan roles (modulo AC). Estos usuarios incluyen a todos los actores que usarán programas para introducir información o usarla en forma directa (no vía un tercero o un material producido por el sistema).
- Elaboran catálogos de puestos, cajas, cajeros-cajas, productos-precios y puestos-productos.
- Reciben dinero de cajeros.
- Entregan fondos (para cambiar billetes grandes) a cajeros.
- Pagan a puesteros.

6.3.1.2. NECESITAN

- Corte de caja (del administrador).
- Importe de las ventas en puestos.

6.3.2. CAJEROS

6.3.2.1. PROPORCIONAN

- Retiros y depósitos en tarjeta.
- Reciben fondos del administrador.
- Entregan fondos al administrador.

6.3.2.2. NECESITAN

- Comprobantes de dinero entregado.
- Ayuda para elaborar sus cortes de caja.

6.3.3. PUESTEROS

6.3.3.1. PROPORCIONAN

- Ventas.
- Optativamente, pueden agregar productos y precios a sus puestos (usan la función del administrador).

6.3.3.2. NECESITAN

- Información sobre sus ventas.
- Control sobre pagos recibidos de administradores.

6.3.4. VISITANTES

Los visitantes no usan el sistema (no tienen funciones autorizadas). Sin embargo, se incluirá una función adicional, para computadoras que no están destinadas a otro uso: al insertar una tarjeta de VISITANTE; proporcionan el saldo de la tarjeta y, si el visitante lo solicita, el desglose de las ventas.

6.4. DETERMINACIÓN DE LOS “USUARIOS” DEL SISTEMA

Tras este análisis de los actores, se determinó que los administradores, cajeros y puesteros son los “usuarios” del sistema. A estos se les agrega otro tipo de actor (Técnico): es el que realiza funciones complementarias, especialmente la instalación de una versión en las computadoras, conecta las lectoras, etc. Una función de los “técnicos” (aunque no sean tales) es la recolección de datos de las computadoras para incluirlas en la base de datos central (esto se aclara más adelante). Sin embargo, para efectos de sus roles, son administradores con limitantes en lo que pueden hacer.

6.5. OBJETIVO FUNDAMENTAL DEL BAZAR (PARA QUÉ SIRVE)

El método de diseño adoptado exige identificar un OBJETIVO FUNDAMENTAL del sistema. En el BAZAR es: que le lleguen todos los importes de las ventas al administrador.

A partir de esta función fundamental del sistema, se definen y especifican todas las restantes. Por ejemplo, para que le lleguen, debe saber lo que se vendió, lo que le entregaron los cajeros y poder comparar ambos importes.

6.6. ENTIDAD (DE DATOS) FUNDAMENTAL

A partir del objetivo fundamental, se especifica una ENTIDAD (de datos) FUNDAMENTAL: en este caso es VENTAS (puesto-visitante.) Hay otra entidad que compite con ella (efectivo recibido de cajeros) pero se la considera secundaria.

6.7. DONDE ESTARÁN LOS DATOS

Cada computadora tendrá una versión de la base de datos. Habrá una base central en la que se concentrarán todos los datos.

6.8. LISTA DE FUNCIONES “PRELIMINAR” PRINCIPALES

- Puestero
 - Ofrecer menú, aceptar pedido, validar saldo
 - Separar pedidos de a 6 productos (si es necesario)
 - Grabar en TI y Base de datos
- Carga-y-descarga de TI de visitantes
 - emitir TI de visitantes
 - Registrar depósitos (cargas) y retiros de la TI
 - consulta a saldos de TI
- Cajero entrega dinero al administrador
 - registro de la operación en base del administrador
 - registro de la cantidad entregada (para su corte de caja)
- Administrador
 - prepara (crea) una instancia
 - introduce un usuario “inicial”
 - entrega fondos al cajero
 - recibe dinero del cajero
 - paga a un puestero
- El contagio (descrito más abajo)
 - infectar computadora con datos de la TI
 - operaciones de cajero
 - ventas
 - última venta y última operación de cajero del titular de la TI
 - infectar TI con datos de la computadora
 - operaciones de cajero
 - ventas.

SECUNDARIAS (mantenimiento, es decir, A-B-C)

- Catálogos
- Usuarios
 - actualizar usuarios
 - cambio de roles
 - cambios de passwords propios

- cambios passwords de otros
- Puestos
- Cajas
- Puesto-puestero
- Puesto-producto
- Caja-cajero

6.8.1. MODELO CONCEPTUAL

LAS ENTIDADES IDENTIFICADAS Y LAS RELACIONES ENTRE LAS MISMAS

- TARJETAS DE VISITANTE: número interno, id-tarjeta, nombre visitante, saldo inicial, compras, operaciones de efectivo (carga y descarga), los datos de contagio (se definen en Fase III)
- TARJETAS DE USUARIOS DEL SISTEMA: número interno, id-tarjeta, nombre del usuario, roles, palabra clave (deformada). Sólo para puesteros, los importes recibidos de la administración (con fecha y hora)
- VENTAS (visitante, puesto, puestero, importe, detalle productos)
- Usuarios
- Cajas
- Puestos
- Productos (con precio)
- Puesto-producto (con precio)
- Caja-cajeros
- Operaciones de cajero
 - Cajero recibe dinero
 - De visitante
 - De administrador
 - Cajero entrega dinero
 - A visitante
 - A administrador
- Operaciones de administrador
 - Administrador recibe dinero de cajero
 - Administrador entrega dinero

- A cajero
 - A puestero
- Entidades para contagio
- Ventas
 - Todas las operaciones de cajero.

Observación: las constantes del sistema y de cada instancia se almacenan en una entidad por separado.

Las funciones y roles del sistema de control de acceso (ver más abajo) son constantes del paquete, y por ende, se decidió almacenarlas como constantes de los programas. Sin embargo, de cambiar la decisión, se agregaría una entidad al modelo de datos definitivo (será decisión técnica): Funciones y roles para el control de acceso a las funciones del BAZAR.

6.8.2. RELACIONES ENTRE LAS ENTIDADES

Usuarios: independiente.

Puestos y puesteros: muchos a muchos.

Caja y cajero: muchos a muchos.

Puestos y Productos: muchos a muchos.

Caja y operaciones de caja: uno a muchos.

Administrador – operaciones de administrador: uno a muchos.

Puesto – ventas uno a muchos.

Puestero a ventas uno a muchos.

6.9. PROTECCIÓN DE DATOS (SEGURIDAD)

A pesar de que la descripción del modo en que se contagiarán los datos de una computadora a otra debería estar en este lugar, este procedimiento se ha indicado en la FASE III.

Se contemplaron estos aspectos:

- Uso no autorizado de funciones
 - Alteración de datos sin el uso de programas del sistema (incluye agregar o eliminar registros)
 - Pérdida o destrucción de datos
 - Clonación de tarjetas
 - Uso de tarjetas y bases de datos de un evento (instancia) en otro evento
- A continuación, se especifican los dispositivos incluidos para evitar cada uno de ellos.

6.9.1. USO NO AUTORIZADO DE FUNCIONES

El BAZAR incluye un módulo de control de acceso tipo RBAC (role based access control).

Se elaboró una lista de funciones que se protegerán (solo las podrán invocar los usuarios autorizados). La lista se elaboró juntando funciones que no necesitaban una separación, es decir, no habría usuarios que pudieran usar una y no la otra.

Cada usuario tiene un rol como ACTOR del sistema (una persona – usuario del sistema – puede fungir como más de un tipo de actor).

- Rol como administrador (0=ninguno)
- Rol como cajero (0=ninguno)
- Rol como puestero (0=ninguno)

Las Funciones para AC y los roles se detallan a continuación. Observe que las funciones se numeran por grupo (en qué roles se ven reflejadas).

Administrador

Funciones:

- 1 Preparar una instancia (un bazar)
- 2 Emitir tarjetas de usuario (en blanco)
- 3 Operaciones de administrador
- 4 Consultas de administrador (operaciones y balance)

- 5 Cambiar palabra clave de otros usuarios
- 6 Operaciones técnicas (concentración de datos y otras)

Roles: 1-(111111); 2-(000001); 3-(010000); 4-(001100).

Las posiciones corresponden al número de función.

Cajero

Funciones:

- 1 Emitir TI de visitante con saldo inicial
- 2 Operaciones de dinero en cajas
- 3 consulta a operaciones de cajero y saldos de dinero

Roles 1-(111); 2-(100); 3-(0.0.1)

Puestero

Funciones:

- 1 Operaciones de puestero (ventas y cortes de caja)
- 2 Consulta de puestero
- 3 Dar de alta un producto o modificarlo

Roles: 1-(111); 2-(011); 3-(001)

Observación sobre el cambio de versión

El uso de las tarjetas de usuario no estaba contemplado en la primera versión. Proporcionarían su palabra clave al inicio de cada sesión. Sin embargo, un estudio del sistema y de cómo funciona “en vivo” nos mostró que era mejor dotar a cada actor de una tarjeta de usuario que, primordialmente, le serviría para autenticarse al inicio de la sesión. Posteriormente se incluyó otro uso en las de los puesteros: se grabarían en ella los importes ya cobrados, es decir, lo que ya les había pagado por los administradores. Esto se hizo para reemplazar los recibos que se planearon inicialmente para que los administradores pudieran demostrar

los pagos efectuados. El sistema no permitirá que le paguen a un puestero sin que presente su tarjeta, en la que están grabados los importes mencionados.

6.9.2. ALTERACIÓN DE DATOS SIN EL USO DE PROGRAMAS DEL SISTEMA

Se protegerán los datos que lo requieran, por cualquier motivo, contra alteraciones “ilegales”, donde este término indica que se efectuaron sin el uso de los programas del sistema. Cada vez que se efectúe una operación en el BAZAR, el programa deberá calcular y grabar, como parte de los datos que reflejan la operación, una “cifra de auditoría”. Es un entero de 4 bytes que se obtiene aplicando un algoritmo a cada carácter de una cadena que se forma concatenando los elementos de datos (campos) involucrados. El modo de generar estas cifras se delegó a la Fase III.

Cada vez que el sistema use una tarjeta o un registro de alguna tabla, validará que la cifra de auditoría corresponda a los datos grabados. Para ello, aplicará el mismo algoritmo usado para calcular la cifra y la comparará con la que contiene el campo “cifra de auditoría” que forma parte de los datos de todas las operaciones (y algunas otras cosas como palabras clave, identidad de tarjeta, saldos, etc.).

6.9.3. PÉRDIDA O DESTRUCCIÓN DE DATOS

El efecto contagio se incluye precisamente para prevenir estas pérdidas, De hecho, constituye un modo de respaldo de cualquier operación. Además, habrá respaldos periódicos de la base central, y la recolección de datos – que se hace según lo indican los organizadores del bazar – hará que esta base central constituya un respaldo de todas las demás computadoras.

6.9.4. CLONACIÓN DE TARJETAS

La clonación de tarjetas puede resultar atractiva a individuos que desean aprovecharse del sistema para obtener beneficios. El dispositivo fundamental que usa el BAZAR para prevenir esta posibilidad es el uso del número grabado en la

ROM (Read Only Memory) de la tarjeta. Para modificar dicho número, se necesita un programa de utilería que no se entregará a nadie: lo tendrán solamente los que ofrecen el servicio de BAZAR a sus clientes (los bazares).

Para tarjetas que no ofrecen una memoria protegida – casi no hay tales tarjetas, pero están contempladas de todos modos – se usa otro método, que puede ser usado para complementar el uso de la ROM. Se usan tarjetas impresas, lo que significa que se les imprimió algo con una impresora especial, Este proceso no es tan caro como era hace algunos años, puesto que se ofrecen impresoras más baratas (en algunos casos, con menos funciones) y la tinta también bajó de precio. Se pueden comprar tarjetas impresas, o se pudieran imprimir para cada instancia, es decir, para un bazar en particular.

Para proteger contra clonaciones, se elabora un diseño para un bazar en particular, y se intenta complicarlo de modo que no sea fácil de copiar. Se lo mantiene secreto hasta el día del evento, y se cuida que no “desaparezcan” tarjetas preimpresas. En lugar de este método, o además de usarlo, se pueden usar etiquetas con formas y diseños *ad hoc*, preferiblemente firmadas en forma autógrafa y en tinta muy distinguible, a las que se les puede añadir el número de tarjeta y el nombre del visitante cuando se le emita su TI. Esto tendrá la ventaja de evitar que le entreguen una tarjeta de otro visitante cuando se junten varios en un puesto o caja.

6.9.5. USO DE TARJETAS Y BASES DE DATOS DE UN EVENTO (INSTANCIA) EN OTRO EVENTO

Los usuarios de un evento podrían estar tentados de usar sus tarjetas y programas para otras instancias del BAZAR. La protección principal, pero no única, es la inclusión de un parámetro específico para cada instancia. Se genera como parte de la instanciación del paquete, y se usa como componente del algoritmo de cálculo de las cifras de auditoría de todo tipo de una instancia. Los detalles de la generación del parámetro se presentan en la fase III, puesto que se delegó a los informáticos definir el proceso adecuado.

Se encargó a los técnicos hacer que esta protección sea lo más sólida posible, pero sin agregar dispositivos de hardware o actividades adicionales por parte de los participantes en un evento (bazar).

6.9.6. EN LA RECOLECCIÓN DE DATOS, ALGUIEN “BORRA” REGISTROS DE LOS ARCHIVOS DE TRANSPORTE

Recordemos que, puesto que la misma operación se graba en varias computadoras (transferida mediante el contagio) la eliminación de registros sólo tendría efecto si se hace en los archivos de concentración. Para evitar este tipo de cosas, el programa que crea los archivos en los procesos de extracción de datos de una computadora, para incluirlos en la base de datos central, tiene cifras de control: se anota el número de “operaciones” (registros de cada tipo) y el total del importe, y se protegen estos números con una cifra de auditoría. Cuando se usa el archivo para el proceso de carga a la base de datos, se validan las cifras de control.

6.10. LA CONTRIBUCIÓN DEL EFECTO CONTAGIO

Se formularon las “dudas” que pudieran tener los actores, mismas que se detallan a continuación. Antes, vale la pena señalar que en cualquier sistema, si alguien hace algo “malo”, perjudica a otros. Es lo que se llama un fenómeno de suma cero. Por lo tanto, para analizar las actividades que pudieran beneficiar a alguien, mejor se hace lo contrario: se investiga a quién perjudican.

El administrador tiene “miedo” que se produzcan acciones o eventos que resulten en:

- Que no les llegue *todo* el dinero recaudado
- Que no les acrediten un pago que efectuaron a un tercero
- Que les digan que les entregaron más de lo que sucedió en la realidad
- Que alguien haga compras sin haber pagado por ellas de antemano (es decir, tener un saldo “valido” en su TI, como sucedería con una tarjeta clonada).

Del mismo modo, el cajero teme que

- No le acrediten un pago que efectuó a un administrador o a un visitante
- Le digan que le dieron más de lo que cobró en realidad.

El puestero, en cambio, teme que

- No le reconozcan todas sus ventas
- No le paguen todas sus ventas
- Le digan que ya le pagaron cuando no es cierto.

El visitante, por su parte, desconfía de

- Hace un depósito (recarga) y no lo registra el sistema
- Otros hagan compras o retiros y se los debiten a él
- Se rompe su tarjeta (ya no se puede leer)
- Pierde su tarjeta. Este temor es justificado, pero será SU culpa, y no la del sistema o culpa de un tercero, donde excluimos (pero contemplamos parcialmente) la posibilidad de un robo por la naturaleza del evento.

Estos temores se formulan de varios modos. ¿Qué pasa si se rompe la computadora, o si alguien me hace trampa o si me clonan mi tarjeta? Pero también hay otros tipos de actos que se sospechan que pueden ocurrir: ¡me inventaron un depósito (le aumentaron el importe)! o ¡eliminaron uno de mis pagos (o le redujeron el importe)!

Si se pretende que alguien use un sistema de cualquier tipo, hay que confrontar estos temores o sospechas. Precisamente muchos sistemas no usan dispositivos como TI por estos temores: la gente no les tiene confianza a los sistemas, y están convencidos de que cualquiera les puede hacer trampa o aprovecharse de ellos vía sistema. En términos usuales: ¡papelito habla!

Los argumentos que se pueden proporcionar a los actores de un sistema deben ser concretos, no generalidades que sólo prometen algo vago. Se deben analizar las causas que pueden resultar en situaciones de las descritas como temores de los actores del sistema. Para el BAZAR, estas causas son:

- Se elimina ilegalmente un registro de la base de datos
- Se agrega ilegalmente un registro a la base de datos
- Se alteran los datos de un registro

- Se elimina un “recibo” de una TI
- Se agrega un “depósito” a una TI
- Se clona una tarjeta de visitante con un saldo “inválido”
- Se clona una tarjeta de usuario del sistema
- Se alteran los roles asignados a un usuario
- Se consigue violar la protección ofrecida por las palabra claves
- Se modifica el titular de una tarjeta
- No se pueden leer los datos en una computadora
- No se pueden leer los datos en una TI.

Algunos de los dispositivos de seguridad con los que cuenta el BAZAR ya se han descrito anteriormente. El uso del efecto contagio proporciona protección especialmente contra la pérdida o destrucción de datos. Recordemos que las computadoras no estarán conectadas entre sí, de modo que, si no se hacen respaldos constantes (de hecho, con un proceso redundante en un dispositivo independiente de la computadora misma) la destrucción de un disco o un malfuncionamiento podría causar la desaparición de un dato importante para algún actor o visitante. El contagio tiene un efecto inmediato: cada operación realizada con una tarjeta se registra en la siguiente computadora con la que establece contacto (además de otros datos que se contagian).

Las bases de datos estarán protegidas por los dispositivos que ofrece el manejador de base de datos, en este caso, el ACCESS. Las palabras clave se modifican para cada instancia del paquete como parte del proceso de instalación. Los detalles de estas protecciones son funciones de la Fase III.

6.11. SELECCIÓN DE TARJETAS Y LECTORAS PARA EL BAZAR

Para determinar la tarjeta inteligente y la lectora de tarjetas que se usarían para el BAZAR, es decir, que se ofrecerían a las instancias, se tomó en cuenta lo siguiente:

- Las especificaciones mínimas resultantes del uso de las tarjetas
- El volumen de datos que se registraría en ellas, y el costo.

Para ello se analizaron algunos proveedores de tarjetas, como Gemplus uno de los principales proveedores de tarjetas desde sus inicios (actualmente Gemalto por la fusión en 2006 de Gemplus y Axalto) y SIEMENS que es uno de los grandes productores de chips para las tarjetas.

Actualmente existen muchas compañías que ofrecen distintos tipos de tarjetas. ACS por ejemplo ofrece las tarjetas ACOS en diferentes modelos, con una velocidad de transferencia de 223,000 bps, también tiene el modelo ACOS10 criptográfico con una velocidad de 115,000bps (ACS, 2010). ATMEL inicialmente vendía entre otros modelos de tarjetas de memoria, la AT24C64, con 8 kbytes de memoria EEPROM, que actualmente fue reemplazada por la AT24C64C con características similares (ATMEL, 2010). Precisamente esta última fue la que se decidió usar para el sistema bazar, como se verá mas adelante.

Las lectoras

Como se vio en el Capitulo 2, hay muchos modelos diferentes, los cuales difieren principalmente en la forma de leer las tarjetas y de procesar los datos, todo esto se puede resumir en lectoras stand-alone y PC-linked (ACS, 2010). Esto significa que algunos modelos están conectados a una computadora, ya sea de modo local o remoto, mientras que otras realizan las funciones como parte del dispositivo. Hay diferentes formas de conectar una lectora a una PC por ejemplo por medio de un puerto USB o bien por medio de un puerto IDE dentro una computadora e incluso hay lectoras que están incrustadas en un teclado. Cabe mencionar que las rutinas de programación para comunicarse con la lectora varían dependiendo del modelo elegido.

Selección de los dispositivos para el sistema bazar

El sistema bazar utiliza las tarjetas para almacenar las transacciones del propietario, pero también otras como las transacciones para propósitos del contagio. Por lo tanto se necesita de una cantidad de memoria considerable para la lectura y escritura. Además las transacciones deben estar protegidas de alguna

forma, para prevenir o detectar cambios. Sin embargo, los datos no son confidenciales, así que no es necesario encriptar los datos.

Tarjetas inteligentes u otros dispositivos de memoria

Se consideraron para la realización del sistema otros dispositivos tales como las memorias USB por su mayor capacidad de memoria. Se tomaron en cuenta dos aspectos principales que hicieron de la tarjeta el dispositivo elegido:

- Aunque el puerto USB esta disponible en la gran mayoría de las computadoras modernas, un lector especial debe estar presente para poder procesar los datos de una tarjeta, por ejemplo al intentar clonar una tarjeta o manipularla de alguna otra forma.
- Los lectores de tarjetas pueden ser utilizados miles de veces, mientras que la constante inserción de dispositivos en un puerto USB puede causar un mal funcionamiento del puerto a largo plazo, especialmente debido a la inserción de forma errónea.
- En caso de que una lectora se averíe es mucho más rápido reemplazarla que arreglar o cambiar el puerto USB.

Otro aspecto importante es que un dispositivo USB puede ser infectado por virus y demás software malicioso, cosa que no sucede con una tarjeta por dos razones: el virus tendría que saber como funciona el protocolo de comunicación con la tarjeta y la otra razón es que aun sabiendo eso, el virus tendría que enviar la información byte por byte a la tarjeta.

El tipo de tarjeta utilizada

Dado que el sistema no aprovecharía en su totalidad las capacidades de procesamiento de los chips de la tarjetas con procesador, en la versión actual del sistema se utilizaron solamente las tarjetas de memoria, aunque la posibilidad de utilizar el procesador de la tarjeta inteligente para garantizar la seguridad de los datos es atractivo, se considero innecesario ya que el sistema BAZAR incluye rutinas que proporcionan la seguridad necesaria, además, el uso de las tarjetas

inteligentes limitaría el mercado del sistema debido a que este sería mas caro. Naturalmente el sistema es fácilmente adaptable en caso de que se decidiera utilizar este tipo de tarjetas. En cuanto al tamaño de memoria se refiere para almacenar los datos se considero que 8 kbytes son suficientes para el sistema, por tanto la tarjeta de memoria que se seleccionó fue la AT24C64 con 8 kBytes (figura 8).

Si se decidiera utilizar una tarjeta con mayor cantidad de memoria, no sería necesario hacer cambios al sistema, por el contrario si se decidiera utilizar una tarjeta con menor cantidad de memoria el sistema si requerirá de cambios, pero estos serian cambios menores.



Fig. 8 Tarjeta elegida para usarse en el sistema BAZAR

El lector de tarjetas para el bazar

Para el BAZAR el lector tipo stand-alone fue descartado debido a que este tipo de lectoras es mas susceptible al robo y al mal funcionamiento, además de que los costos serian mayores. Los equipos no necesitan ser sofisticados o tener una gran capacidad de memoria o disco para el sistema, por lo tanto pueden ser usados modelos de equipos relativamente baratos. El lector ACR38 (PC-Linked) (ACS, 2010) fue el modelo elegido (figura 9) y se programaron las rutinas para el sistema BAZAR. Este lector soporta casi cualquier tipo de tarjetas de memoria SLE, además también es compatible con varios modelos de tarjetas inteligentes. También se tomaron en cuenta los costos, ya que este lector es mas barato que otros lectores que pueden ofrecer alguna funcionalidad adicional, pero para el BAZAR no son necesarias y no representaría ninguna ventaja.

Debido a que tanto la tarjeta seleccionada como la lectora seleccionada son de contacto, las rutinas fueron desarrolladas para funcionar con esta tecnología, sin embargo si se decidiera cambiar a tecnología sin contacto, solamente habría que cambiar las rutinas de comunicación del sistema con la lectora y algunas rutinas de grabación.



Fig. 9 Lectora elegida para usarse en el sistema BAZAR

6.12. PROCEDIMIENTOS Y CONTROLES

Durante todo el proceso de análisis y diseño del sistema, se investigan, formulan y critican procedimientos y controles que se incorporarán al sistema. Sin embargo, en este punto de la Fase II, es necesario precisarlos a detalle. Por lo tanto, se detallan los aspectos del sistema que requieren algún control o vigilancia, u otro tipo de procedimiento. Naturalmente se incluye la emisión de comprobantes, permisos impresos cuando se requieran, y cualquier cosa que requiera “normas” o “acuerdos”, pero sujetos a la obligación por parte de los actores del sistema a adherirse a ellos.

Para el BAZAR, precisamente porque se trata de un paquete, sólo se pueden formular reglas generales y recomendaciones.

Numeración de las operaciones: las ventas y operaciones de cajeros o de administradores tienen un número único que se les asigna en forma consecutiva.

Cada computadora (base de datos) tendrá una “serie”, que podría ser 100.000 a 199,999 para una computadora, 200.000 a 299,999 en otra, etc. Los rangos se determinan para cada evento. No se pueden usar letras puesto que los campos que reflejan los números de operación son enteros. Observe que esto no sucede por puestero o cajero, sino por la computadora usada. Cuando un

puestero usa la computadora de otro puesto, los números de sus ventas se asignarán con la regla de aquel puesto.

No se admitirán devoluciones de compras realizadas. El sistema no lo contempla.

Control (físico) de tarjetas: habrá un encargado de entregar las tarjetas (en blanco) a los que las necesiten, que necesitan un permiso expreso del administrador para que se las entreguen. Firmarán de recibido un lote de tarjetas (supongamos que 50, pero el número se fija en cada instancia) y, si las tarjetas tienen un número (ya sea interno o externo) se anota el rango de números o una lista de cada una de ellas, para el caso en que la numeración no sea corrida. En ese momento cesa la función del paquete en cuanto a control físico de las tarjetas – es responsabilidad del personal que operará el sistema.

Control de lectoras: se hace mediante un acuerdo entre el proveedor del servicio y los organizadores. Por ejemplo, se estipula un pago por cada lectora no devuelta al proveedor. El mismo tipo de arreglo se hace para las computadoras.

Comprobantes: el organizador puede optar por que para las operaciones entre cajeros y administradores, y entre administradores y puesteros, en particular, la recepción de dinero, invariablemente se reflejen en un recibo “en papel”, firmado por el que recibe el dinero. El sistema contempla la emisión de tales comprobantes, aunque no serán fiscales, es decir, sólo tendrán efecto dentro del organismo o empresa que organiza el bazar. Observación: no se elaboraron los programas correspondientes para la versión 1. Esta actividad se pospuso para la primera instancia real, es decir, si alguien decidiera usar el sistema.

Acciones resultantes de detección de intentos de violación a los dispositivos de seguridad: los organizadores del evento podrán indicar cuáles de las acciones se deberán tomar:

- Si el intento se produjo por un dato que estaba en la computadora
 - activar una sirena
 - deshabilitar el sistema BAZAR en esa computadora.
- Si se debió a un dato alterado en la tarjeta

- Se marca la tarjeta como “inservible” para el sistema BAZAR. Se podrá leer la tarjeta, pero no servirá para ninguna función del paquete
- El que está operando la computadora retiene la tarjeta (si fue donde se detectó el intento)
- El titular de la tarjeta será entrevistado para determinar el origen del intento de violación de datos.

El sistema BAZAR contiene un programa especial que reactiva el sistema en una computadora y/o la lectora en una computadora. Sin este programa, no será posible restablecer el servicio (excepto con acciones drásticas, tales como un reformato del disco o mediante la detección de los dispositivos que evitan el uso de BAZAR).

Independientemente de las acciones seleccionadas, BAZAR grabará en un archivo “propio” los datos contenidos en el registro alterado, o los de la tarjeta si la violación se produjo en los datos propios de ella.

Reclamaciones por parte de visitantes: inevitablemente habrá visitantes que reclamen que su saldo es menor al que debería ser. En ese caso, se le explicará a la persona cómo el sistema protege su saldo, y se le invitará a acudir al día siguiente (o después del “corte” final, cuando ya la base central tiene todos los datos de todas las computadoras) para que se le indique exactamente la composición del saldo que muestra la tarjeta.

6.13. INTERFACES USUARIO-SISTEMA

Generalidades: las formas se diseñaron con las siguientes características y objetivos:

- En lo posible, no debe cambiar de “forma” para una misma función
- Deberán minimizar la posibilidad de cometer un error, típicamente invocar una función equivocada
- Deben ser auto-explicadas; en caso de que haya una función confusa, se le agregará una “ayuda”

- Debe ser posible en cualquier función regresar al menú principal o detener la sesión (finalizarla)
- Los mensajes de error serán claros pero no exagerados (no explicar lo evidente)
- Los elementos gráficos solo se incluirán si contribuyen a la operación del sistema
- Cuando sea posible, se ofrecerán las funciones de modo que el usuario pueda operarlas con o sin el uso de su “mouse”, y con o sin su teclado
- El BAZAR en esta versión no ofrece imágenes de los productos.

OBSERVACIÓN: en esta sección se incluyen “normalmente” las formas diseñadas con una explicación de la funcionalidad que se pretende, y un “pre-instructivo” de operación para la forma. En esta tesis, por indicaciones específicas del director de la misma, no se muestran todas las formas. Algunas de ellas se mostrarán en la Fase III.

Forma VENTAS

- Se invoca desde el menú.
- Deberá mostrar los productos ofrecidos en una lista. Se diseñó la forma para un máximo de 7 productos. Si el puesto vende otros, se invocarán de a lotes de 7 productos (siguientes, anteriores).
- Al insertar una TI de visitante, aparece el nombre del titular y el saldo de la TI.
- El puestero indica “otro producto” uno tras otro. Se le asigna una cantidad 1.
- Se incrementa o disminuye la cantidad con los botones “-“ y “+”.
- Se puede quitar un producto de la lista.
- Cuando cambia la lista de productos (siguientes, anteriores) permanece el “pedido”.
- Cada vez que se cambia un pedido (agrega, quita producto o cambia cantidad) el sistema muestra el total de la compra.
- Cuando el total de la compra excede el saldo, aparecerá un cartel y no se agrega el cambio (el último) a la lista (el pedido.)

- La confirmación es definitiva.
- El puestero podrá cambiar su modo teclado/mouse cuando lo desee, sin salirse de la forma.
- Esta forma no ofrece ayudas.

Forma OPERACIONES DE CAJERO y de ADMINISTRADOR

NO se detallan aquí.

Formas de apoyo

Usuarios, productos-precios, cajas-cajeros, puesto-puestero, puesto-producto: las diseñarán en Fase III.

Consultas: las diseñarán en Fase III. Deben estar hechas con la regla general: mostrar primero lo general, y permitir invocar los detalles. Por ejemplo, la consulta al SALDO de una TI debe mostrar sólo EL NOMBRE y EL SALDO (actual) y los totales COMPRAS, DEPÓSITOS (Cargas), RETIROS.

Con una acción del usuario, mostrará los detalles de cada uno de estos conceptos.

6.14. ASPECTOS RELACIONADOS CON EL USO DEL PAQUETE

El paquete se usa para una instancia, que será un bazar. Hay un proveedor del servicio de sistemas, es decir, los organizadores del evento no serán los encargados de conseguir ni instalar computadoras, lectoras, tarjetas inteligentes. El proveedor de servicio proporciona todos estos elementos (aunque cabe la posibilidad de que use dispositivos que tenga el “cliente”).

Como parte de la instanciación del BAZAR para un evento, se modificarán las palabras claves necesarias para usar la base de datos del paquete.

En la fase III se describirán las actividades técnicas que se realizarán para una instanciación del paquete.

6.15. ESPECIFICACIONES DE PROGRAMACIÓN

1. Los programas deberán ser fáciles de usar (para que no requieran mucha capacitación).
2. La navegación del sistema deberá ser mínima. Cuando inicia una sesión, el usuario selecciona una función (principal) y se queda en ella hasta que decida cambiarla. Las formas especificadas reflejan esta característica,
3. Los programas deben estar modulados de modo que faciliten correcciones o cambios.
4. Se deberán indicar las excepciones o errores en forma clara, preferiblemente superpuestas al error cometido. Se debe evitar el uso de “cajas de mensaje”.
5. Se deben incluir las rutinas que permitan usar los programas con/sin teclado o mouse.

6.16. NIVEL DE DEPURACIÓN

Se decidió que el criterio de aprobación de la fase 4 consistiría en la realización de un evento, preferiblemente real (un bazar en el que prestaríamos el servicio), pero si no fuera posible, un evento simulado. Las funciones fundamentales deberían funcionar sin excepción alguna. Las de apoyo se probarían, y se podrían modificar (arreglar) posteriormente. Esta estrategia sólo fue posible porque se trata de un paquete: sería inadmisibles en un sistema a medida.

Para la simulación de un evento, se prepararían datos de prueba, con ciertos mínimos: por lo menos 5 puestos, 5 computadoras, que funcionarían también como cajas (se inicia una sesión con otra tarjeta de usuario.)

6.17. MATERIALES NECESARIOS

Para las fases restantes del desarrollo del sistema, se necesitarán:

- El driver de la lectora
- El programa de utilería para actualizar la memoria ROM de las tarjetas
- Lectoras de tarjetas
- 10 tarjetas de memoria

- Acceso a 4 computadoras distintas para las pruebas, con el sistema BAZAR instalado.

RESULTADO (datos para la FASE III)

El modelo de datos conceptual (figura 10). Los campos se detallaron previamente.

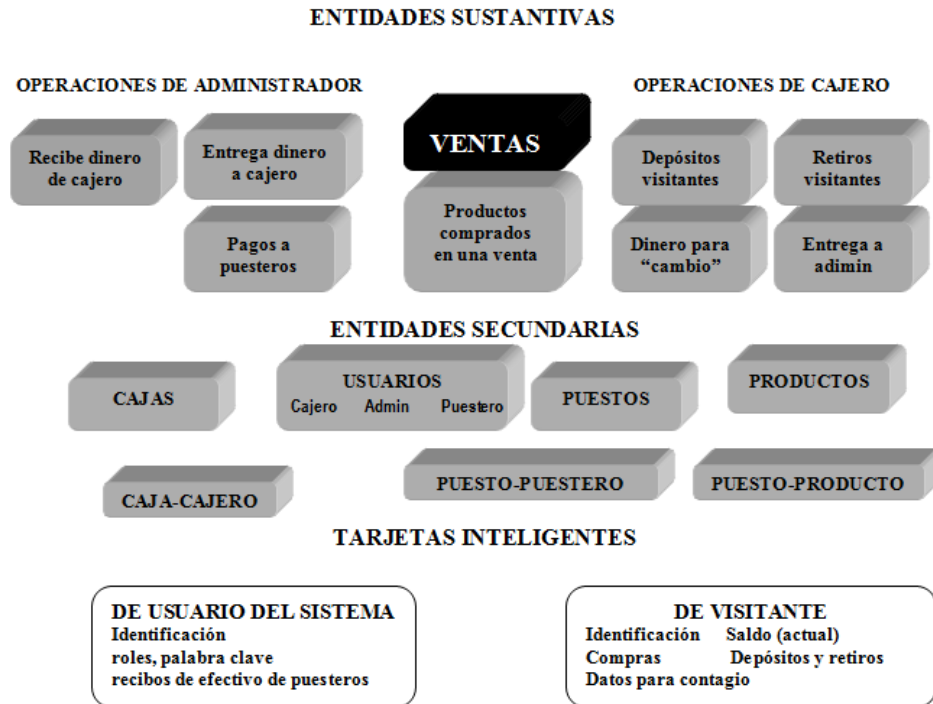


Fig. 10 Modelo de datos conceptual

6.18. LA LISTA DE FUNCIONES DEL SISTEMA

Funciones principales de Caja

- Formatea tarjeta: se encarga de darle el formato adecuado a cada tarjeta y asigna un número específico a la tarjeta, así como el nombre del cliente.
- Recarga Saldo: lee el saldo de la tarjeta y le suma el saldo que el cliente desee recargar y actualiza el saldo en la tarjeta y graba la operación en la base de datos con su cifra de auditoría.
- Retira Saldo: lee el saldo de la tarjeta, y le resta a ese saldo la cantidad que el cliente desea retirar, siempre y cuando la cantidad deseada sea menor o igual a la que está en la tarjeta; de ser así realiza la operación y la graba,

junto con el nuevo saldo, en la tarjeta; también graba la operación en la base de datos.

- Recoge Tarjeta: lee el saldo de la tarjeta, Si tiene saldo, registra un retiro por ese monto. Graba la operación en la TI y en la base de datos.
- Corte de Caja: lee todas las operaciones que ha realizado el cajero, retiros y depósitos de administración, emisiones, retiros, depósitos y recolección de tarjetas y calcula la cantidad que el cajero debe tener en ese momento.

Funciones principales de Puesteros

- Ofrece sus productos a un visitante: muestra los primeros 7 productos; puede ofrecer los 7 anteriores o siguientes; el visitante le indica qué desea, cuántos de cada producto; no acepta operaciones que excedan el saldo de la TI del visitante.
- Guarda venta: esta función se encarga de guardar una venta hecha por un puestero en la base de datos. Busca el número de la venta que corresponde y se lo asigna automáticamente y calcula la cifra de auditoría. Graba el importe total y el detalle de los productos en TI y en base de datos.
- Corte del puestero: verifica todas las ventas que ha hecho el puestero y calcula la cantidad que administración le debe al puestero en ese momento. Permite usar los pagos recibidos (de la TI del puestero) para determinar su saldo (lo que todavía no le pagaron).

Funciones principales de Administrador:

- Emite tarjeta de usuario (cajeros, puesteros y administradores): Le emite una tarjeta de usuario. Le asigna los roles de control de acceso.
- Actualización del catálogo de usuarios del evento. Funciones incluyen asignación de los roles del usuario como Administrados, como Cajero y como Puestero.
- Alta o Edición a un Producto: verifica el número del puestero y le permite dar de alta nuevos productos o editar lo ya existentes. Cambio de precio:

evidente (Esta función la puede usar un puestero, pero eso no cambia la función).

- Indica puestos a puesteros: en qué puestos puede vender.
- Alta a una Caja: da de alta una caja. Asigna un número de caja.
- Indica cajas que puede usar un cajero.
- Alta a un Puesto: da de alta un puesto.
- Recibe dinero de un cajero.
- Entrega dinero a un cajero.
- Pagos a puesteros: determina el máximo (lo que le deben). Puestero indica el importe deseado y se le entrega el dinero y registra la operación en la TI del puestero y en la base de datos.
- Recuperación de Datos: se encarga de leer de todas las bases los datos que se deseen leer, ya sean de un puesto o de una caja, para poder recuperarlas ya sean las ventas, operaciones de caja, o bien recuperar el saldo de una tarjeta dañada.

Funciones principales de Cliente

- Revisa Compras: la única función que tiene disponible el cliente. De la tarjeta lee todas las compras que ha realizado y las carga en pantalla para ver lo que ha comprado. Lo mismo sucede con las cargas y descargas (Depósitos y retiros) de efectivo a su tarjeta. También le da su saldo actual.

Funciones del módulo de Control de Acceso

- Implementación en los programas: una función que autoriza el uso de una rutina por un usuario, el tipo de rutina y, en algunos casos, el dato al que se refiere.

Funciones de apoyo:

- Consultas a depósitos y retiros administrativos
- Consulta de ventas
- Consulta de operaciones de caja

- Consulta de productos
- Cálculo y validación de la cifra de auditoría
- Conexión de la lectora de tarjetas
- Lectura y grabación de datos de la tarjeta

Las interfaces USUARIO-SISTEMA

No se reproducen las formas en este inciso. Como se anunció previamente, se muestran en la Fase III, que las usa para definir los programas que las procesan.

CAPÍTULO 7. FASE III

Diseño físico del sistema

7.1. DESCRIPCIÓN DE LA FASE III

En base a la información recibida del diseño conceptual del sistema, se procede a definir todos los aspectos técnicos e informáticos para la implementación. Estas actividades incluyen las que se enlistan, pero no excluyen a otras que se pueden presentar para casos particulares.

Diseño de las bases de datos definitivas, en base al modelo conceptual.

Se define la modulación del sistema (en cuanto a agrupación de rutinas y funciones) y se elaboran las formas que usarán los programas.

Definición detallada de los programas, usando las formas definidas como parte de las interfaces diseñadas. Para cada forma se especifican a detalles los disparadores que no sean totalmente evidentes.

La definición debe incluir todas las situaciones de error y excepción que se puedan presentar. Si el programador tuviera alguna duda sobre la definición o algún detalle del programa que le toca elaborar, acudirá con el líder de la Fase III y este le aclarará sus dudas. Observe que el nivel de detalle de la definición es función del conocimiento del sistema por parte del programador. Muchas veces este es parte del equipo de trabajo y ni siquiera necesita detalles.

También incluye el detalle de la componente de control de acceso a las funciones y los aspectos de seguridad, en este caso, el cálculo de cifras de auditoría. Para estas últimas, se proporcionan a los programadores las rutinas ya elaboradas en Visual Basic.

En particular, el sistema adquiere la palabra clave (única) para el uso de la base de datos en ACCESS con una rutina especial, que lee un archivo que está en el mismo directorio que el programa ejecutable.

Se elaboran controles de proyecto por tipo de actividad, y se nombra un encargado de cada subproyecto. Se asignan las actividades a los responsables de efectuarlas.

Se especifica a cada responsable de una actividad cómo debe entregarla al líder del proyecto. Esto incluye el nivel de documentación de sus respectivas tareas.

Mención especial merecen las actividades relacionadas con el uso de lectoras y tarjetas. Para no obligar a los programadores a estudiar los aspectos técnicos, se les entregarán las rutinas de conexión, lectura y grabación de tarjetas. Puesto que en el BAZAR le llegan los modelos de las tarjetas y lectoras, no hay actividades de selección de hardware o dispositivos. Como parte de la entrega de la fase anterior, se recibió el “driver” para la lectora seleccionada.

7.2. DISEÑO DE LA BASE DE DATOS

7.2.1. CRITERIOS APLICADOS PARA EL DISEÑO

Se deciden criterios generales, en especial, si se pretende que la base cumpla con las formas normales, y en caso contrario, cuáles de ellas se pueden violar y el motivo por el cual conviene hacerlo.

En el BAZAR, se decidió violar las formas normales en dos aspectos.

- Hay campos en la tabla de USUARIOS que están relacionados con tablas diferentes, pero se reemplazaron estas relaciones por rutinas de programación. Concretamente, un usuario puede ser administrador, cajero y/o puestero. El campo id-user de este modo está relacionado con diversas tablas.
- En la tabla SALES (ventas) se incluyeron 6 detalles de producto, en lugar de separar la información (uno a varios) en dos tablas.
- Se juntaron en una tabla única todas las operaciones de cajero, a pesar de que se pueden referir algunos campos a otros de dos tablas (visitantes y administradores).
- Lo mismo sucedió con la de operaciones de administradores, pero en este caso las tablas con las cuales se relacionaban eran puesteros y cajeros.

7.2.2. LA BASE DE DATOS BAZARBASE

El diseño definitivo (adoptado) se muestra en idioma inglés, puesto que así fue creada la base de datos. Sin embargo, se han traducido los nombres de las tablas (no así los campos.) No se incluyeron relaciones, mismas que, como ya se dijo, se reemplazaron por rutinas en los programas. La base no tiene procedimientos almacenados, funciones o disparadores, y no se usan vistas. El nombre de la base de datos es BazarBase (figura 11). La palabra clave default – que se usó en el desarrollo – es “bazarv2”.

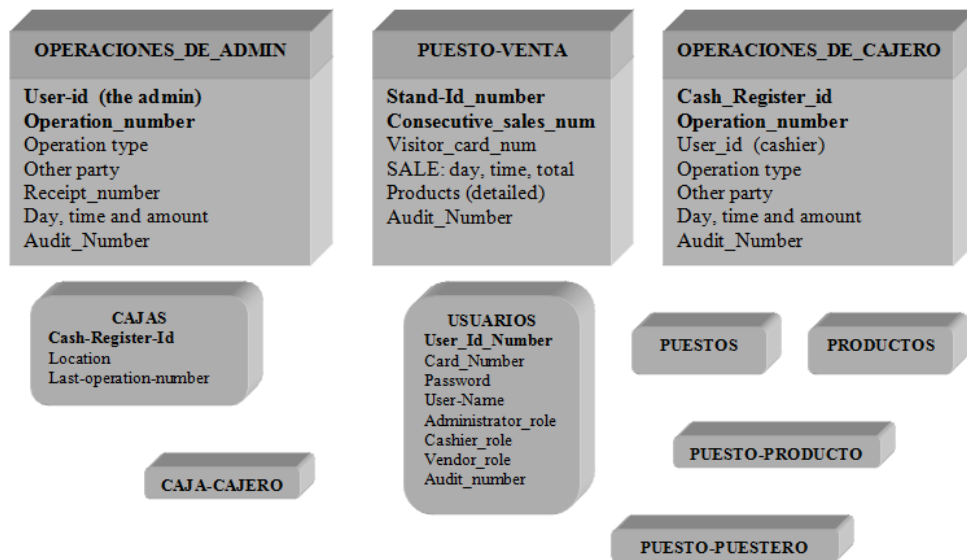


Fig. 11 La base de datos BazarBase

Observación: Los campos en **negritas** representan las claves primarias de las tablas, además nótese que a las tablas más importantes están descritas con mayor detalle en la figura.

7.2.3. DESCRIPCIÓN DE LOS CAMPOS DE LA BAZARBASE

Las tablas se han agrupado por el actor principal que las ocupa y se presentan de ese modo.

7.2.3.1. TABLAS DE ADMINISTRADOR

USUARIOS: para cada usuario de la instancia hay un registro con los campos mostrados en la tabla 1. Todo usuario tendrá un rol para cada una de las tres categorías en que están divididas; este indicará que funciones del sistema puede usar. Tenga en cuenta que un mismo usuario (persona) puede tener mas de un cargo (por ejemplo un puestero puede también ser un administrados).

OPERACIONES_DE_ADMINISTRADOR: todas las operaciones involucradas con la recepción y entrega de dinero se guardan en esta tabla (tabla 2). El tipo de operación esta dado por el siguiente código: 1.- un cajero entrega dinero en efectivo, 2.- dinero obtenido de una cuenta bancaria o de algún tercero, 11.- que un cajero retire dinero, 12.- pago a los vendedores, 13.- deposito de efectivo en una cuenta bancaria o similar, 14.- pago de los gastos. Las cantidades siempre serán grabadas como una cantidad positiva, teniendo en cuenta que las operaciones con un código mayor a diez disminuyen el saldo.

TABLA 1
CAMPOS DE LA TABLA DE USUARIOS

User-Id-number PK	Unique number assigned
Card_number	Necessary to initiate sessions
Password	Number computed from the actual pwd
User-name	
Administrator_rol	Access Control Roles
Vendor_rol	
Cashier_rol	
Audit-number	Computed from other fields to protect against alterations

TABLA 2
CAMPOS DE LA TABLA OPERACIONES DEL ADMINISTRADOR

Operation_number PK	A consecutive number
User-Id-number	The user that updates
Operation_type	Coded
Other_party	User number of Vendor or cashier
Amount	
Day	Number of day of the event
Time	Hour and minutes (operation)
Receipt number	Only if necessary
Audit-number	Computed from other fields

7.2.3.2. TABLAS DEL CAJERO

CAJAS: esta tabla contiene el catalogo de cajas habilitadas
Cash_Register_id es número asignado a la caja, Location: una descripción de donde está físicamente esa caja, Last-operation-number: para cada operación de

esta caja, se sumará 1 a este campo y ese es el número que se asignará a “operation-number” (tabla 3).

CAJA_CAJERO: esta tabla contiene el catalogo de cajas que pueden ser usadas por los cajeros (tabla 4).

OPERACIONES_DE_CAJERO: los campos de esta tabla se muestran en la tabla 5. Los tipos de operaciones del cajero están codificados acorde a los siguientes valores: 1.- emitir una tarjeta a un visitante (puede tener un saldo inicial), 2.- Recolección de una tarjeta, 3.- Deposito de saldo a una tarjeta de un visitante, 4.- el visitante retira dinero de su tarjeta, 5.- el cajero recibe dinero de un administrador, 6.- el cajero da dinero a un administrador, 7.- un puestero recibe dinero de un cajero (este ultimo solo esta planeado, normalmente el puestero recibe el dinero de un administrador).

TABLA 3
CAMPOS DE LA TABLA CAJAS

Cash_Register_id PK	Assigned to each cash-register
Location	Location of the cash-register
Last-operation-number	

TABLA 4
CAMPOS DE LA TABLA CAJA-CAJERO

Cash_register-id PK	Assigned to each register
User-Id-number PK	User-id of the cashier
Audit-number	Computed from other fields

TABLA 5
CAMPOS DE LA TABLA OPERACIONES DE CAJERO

Cash_register-id PK	Assigned to each register
Operation_number PK	A consecutive number
User-Id-number	User that updates
Operation_type	Coded
Other_party	Vendor card number or User-id
Amount	
Day	Number of day of the event
Time	Hour and minutes (operation)
Audit-number	Computed from other fields

Nótese que el campo “other_party” (tabla 5) contendrá el numero de la tarjeta o el id de otro usuario involucrado.

La base de datos contiene otra tabla llamada “CONTAGIO_DE_OPERACIONES_CAJERO” la cual contiene la misma estructura que la de OPERACIONES_DE_CAJERO, esta tabla será contagiada por los registros que contengan las tarjetas de los visitantes.

7.2.3.3. TABLAS DE PUESTEROS

PUESTOS: cada puesto debe estar registrado en esta tabla (tabla 6).

PUESTO_PUESTERO: en esta tabla están los registros de en que puestos pueden vender los puesteros (tabla 7).

TABLA 6
CAMPOS DE LA TABLA PUESTOS

Stand-id-number PK	Assigned to each stand
Responsible	User number of "owner"
Computer_id	The computer installed
IP-number	If connected to a network
Last_sales_number	(assigned consecutively)

TABLA 7
CAMPOS DE LA TABLA PUESTO_PUESTERO

Stand-id-number PK	Assigned to each stand
Vendor-id PK	User number of vendor
Audit_number	

PRODUCTOS_DEL_PUESTO: contiene los productos que ofrece cada puesto (tabla 8). Esta tabla provee de la posibilidad de checar inventarios en caso de que el puestero decida usarlo. Sin embargo no esta recomendado para el bazar debido a que hay muchas situaciones de excepción que podrían causar información errónea, por ejemplo si el vendedor hace una venta en la computadora de otro puesto.

PUESTO_VENTAS: cada venta hecha por el puesto se graba en esta tabla. Si la venta sobrepasa los 6 productos, esta se divide en dos o más registros de 6 productos. Hay un numero consecutivo asignado a cada registro y la combinación de este numero y el id del puesto son usados como la llave primaria (tabla 9).

La base de datos contiene otra tabla llamada "CONTAGIO_DE_VENTAS", la cual es contagiada por la tarjeta del visitante, esta tabla contiene exactamente los mismos campos que PUESTO_VENTAS, por lo tanto no tiene sentido mostrarla.

TABLA 8
CAMPOS DE LA TABLA PRODUCTOS_DEL_PUESTO

Stand-id-number PK	Assigned to each stand
Product-id PK	
Product-name	
Unit-price	
Uses_inventory	Yes o no
Total-provisions	Arrivals of product (In units)
Total_sales	Sales to visitors (In units)
Other-reductions	Breakage, spilling, etc.
Audit_number	

TABLA 9
CAMPOS DE LA TABLA PUESTO_VENTAS

Stand-id-number PK	Assigned to each stand
Consecutive_sales_num PK	Assigned to the sale
Visitor_card_number	(of the purchase)
Day_of_sale	
Time-of_sales	Hour and minutes
Product-id-1	Product code
Product-id-2 to 6	Another 5 similar fields
Quantity_1	How many units (first product)
Quantity_2 to 6	Another 5 similar fields
Total_amount	
Audit_number	

7.3. DEFINICIÓN DE LOS PROGRAMAS

Se elaborará un Proyecto (VB) único para todas las funciones: PryBAZAR. El programa ejecutable se llamará simplemente BAZAR.

La descripción de la definición de los programas en esta tesis siguió los siguientes lineamientos.

- Se presentan en módulos
- El nivel de descripción no será el mismo para programas individuales; se incluyen o no especificaciones y detalles de acuerdo a su necesidad (para comprender la función) y en general, sólo se ejemplifican los elementos que se incluyen en una definición proporcionada al programador con algunos programas.

Los módulos en los que se dividieron los programas se muestran a continuación en la tabla 10 (la numeración no refleja orden ni importancia, solo sirve de índice):

TABLA 10
MÓDULOS DE LOS PROGRAMAS

#	NOMBRE	DESCRIPCIÓN GENERAL
1	Inicio de sesión	Comienzo de sesión de un usuario
2	Funciones generales	Rutinas generales
3	Puesteros	Operaciones de puestero
4	Cajeros	Operaciones de cajero
5	Aministradores	Operaciones de administrador
6	Contagio	Todo lo de contagio
7	Lectoras y tarjetas	Rutinas lectora y tarjetas
8	Concentración de datos	Recolección y concentración
9	Consultas diversas	

7.3.1. MÓDULO INICIO DE SESIÓN

1. El usuario debe introducir su tarjeta de usuario antes de invocar la ejecución.
2. Inicia el programa. Conecta la lectora y lee la tarjeta.
3. Abre la TABLA Datos_de_la_instancia
 - a. extrae el parámetro único generado en la instanciación

- b. (esta rutina es secreta, y se le agregará al programa final) Mientras tanto, se “puentea”
- c. Toma de la tabla el día de inicio del evento.
- 4. Valida la cifra de auditoría de los datos del usuario
Si no corresponde, se termina el programa.
- 5. Accesa la tabla de USUARIOS con user-id-num.
Si no está, se emite una alarma (muchos “beeps” seguidos): alguien usa una tarjeta que encontró (u otro tipo de apropiación).
- 6. Si está activa la opción de usar el password grabado en la tarjeta, lo valida con lo que hay en la tabla; de lo contrario, pide la palabra clave y la valida.
Si no corresponde, lo invita a proporcionarla nuevamente. Tras 5 intentos, termina el programa y se hace sonar la “alarma”.
- 7. Se cargan sus roles a memoria.
- 8. Si el usuario tiene funciones como 2 o más actores, le envía un menú para que indique qué desea hacer. De lo contrario, le ofrece la forma VENTAS, Operaciones de Cajero u Operaciones de Administrador. Se selecciona una de las 3 opciones, que corresponden a lo que desea hacer en esta sesión. El programa distribuye la acción en 3 módulos. *No se muestra el menú en esta tesis, puesto que agregaría nada a la explicación.*

7.3.2. MÓDULO FUNCIONES-GENERALES

FUNCTION llogo_TI_DE_VISITANTE () as boolean

‘Regresa false si hubo violación

Se valida la tarjeta del visitante: si no es válida, Invocar rutina-violacion_tarjeta.

SUB rutina-violacion_tarjeta

‘Las acciones no se muestran en la tesis

Efectúa contagio_ de_TI_A BASE_ y_ de_ BASE_A_TI

Adicionalmente, se copia la ultima VENTA de la TI (es una compra del visitante actual) a la base de contagios de ventas y lo mismo se hace con la ultima operación de cajero registrada en la TI.

SUB arreglo_datos_a_auditar (0 to 29) as string
cuantos_hay as integer
cadena_a_proteger = FUNCTION Armado_de_la_cadena_a_auditar (el-orden
as integer) as string
'Usa el arreglo público arreglo_datos_a_auditar

Cálculo y validación de la cifra de auditoría.

'Le llega la operación (C-O-V, la-cadena, 3 parámetros y el valor anterior de la cifra – para comparar en caso de Verificación

Function Compute-or-Verify-audit (C-or-v as string, the_string as string, parm1
as integer, parm2 as integer, parm3 as integer, previous_value as integer)
as integer

' c-or-v: "C" = compute, "V" = verify

' parm 3 is optional (some uses will not need it)

' previous value furnished when c-or-v = "V"

FACTOR = f (UNIQ-PARAM, parm1, parm2, parm3)

INCR = g (UNIQ-PARAM, parm1 or parm2)

*' Only one is used. The choice is given by the parity or imparity of a digit of the
UNIQ-PARAM.*

AUD = 0

for each character of THE-STRING

 AUD = AUD + ASCII (character) * Factor

 Factor = Factor + Incr

if C-or-v = "C" then

 Return AUD

else if AUD = previous_value then

 Return 0 *' audit number checks*

else Return -1 *' indicator of WRONG AUDIT*

end function

SUB calcula_day_and_time (fecha as date)

‘ Actualiza campos públicos DIA_OPERACION y HORA_OPERACION

‘ Ya tiene el día de inicio del evento en memoria

SUB calcula_el_orden_de_armado (parámetros)

Se concatenan los elementos a proteger (del arreglo) para formar una cadena de caracteres. El orden en el que se concatenan cambia: hay 6 patrones de ordenamiento. Para cada invocación, se determina un entero P, $0 < P < 7$ a partir del PARM 1:

$P = (\text{PARM1 Modulo } 6) + 1$. Se anexan los campos a proteger a la cadena en el orden dictado por el patrón resultante.

7.3.3. MÓDULO PUESTEROS

Este módulo tiene solo dos funciones principales

- Ofrece y confirma una venta
- Consulta a Ventas (incluye un corte aproximado de ventas de un puestero)

SUB Ofrece-y-confirma-una-venta

La primera vez, carga los productos del puesto

Limpia los campos visitante y lista de compras de la forma FVENTAS y mostrarla (modal). Inicializa campos de trabajo.

Cuando un visitante inserta su tarjeta

Se invoca ES_VALIDA = llego_TI_DE_VISITANTE

Poner nombre del visitante y el saldo de la tarjeta en la forma.

El visitante indicará productos, uno tras otro. Se agregan a la lista de productos con cantidad 1.

Los botones sirven como indica el diagrama (para aumentar o disminuir la cantidad).

Se recalcula el total cada vez que sucede un cambio a la lista (productos o

cantidades). Se compara con el saldo. Si no alcanza, se informa al visitante y se elimina el último cambio.

Siempre se muestra el importe de la compra hasta ese momento.

CONFIRMA

Calcula cifra de auditoría

Graba operación en la TI y afecta el saldo (resta el total de la venta)

Graba operación en la base de datos (tabla VENTAS)

Limpia la forma; invita al cliente a otra operación; si no acepta, extrae la tarjeta y LISTO.

Este modulo usa la forma que se muestra en la figura 12.

Aquí aparece el catalogo de productos de un puesto.

Lee los datos de una tarjeta

Detalles de la venta

LEER SALDO (L)

Datos del Cliente
Cliente: Alberto Ramirez Saldo: \$ 1000

PRODUCTOS (F3)
Numero del Producto 7

1.- Pastel
2.- Gelatina
3.- Flan
4.- Tartaleta
5.- Pay de Durazno
6.- Café
7.- Malteada

Esta Venta

Producto	Precio	Cantidad	Totales
Pastel	20	2	40
Malteada	15	1	15

QUITAR DE LA VENTA (Q) - + Total de la venta 55

ANTERIOR (A) SIGUIENTE (S) CONFIRMAR VENTA (V) MENU (M)

Estos botones se usan para navegar entre las páginas de los productos. En caso de existir solo una pagina los botones se deshabilitan y desaparecen

Para modificar el pedido:
Quitar: elimina el producto del pedido.
“-” resta uno a la cantidad
“+” suma uno a la cantidad

Fig. 12 Forma de Ventas

No se documentan en la tesis las otras funciones de este módulo.

7.3.4. MÓDULO CAJEROS

Prepara la forma frecargas_retiros

Cuando un visitante inserta su tarjeta

Se invoca ES_VALIDA = lIlego_TI_DE_VISITANTE

Poner nombre del visitante y el saldo de la tarjeta en la forma.

El cajero indica el tipo de operación (Recarga, Retiro, Recolección) a realizar

Si es un recarga o retiro, se abre un cuadro con una calculadora como se muestra en la figura 13. El cajero indica la cantidad a retirar/recargar.

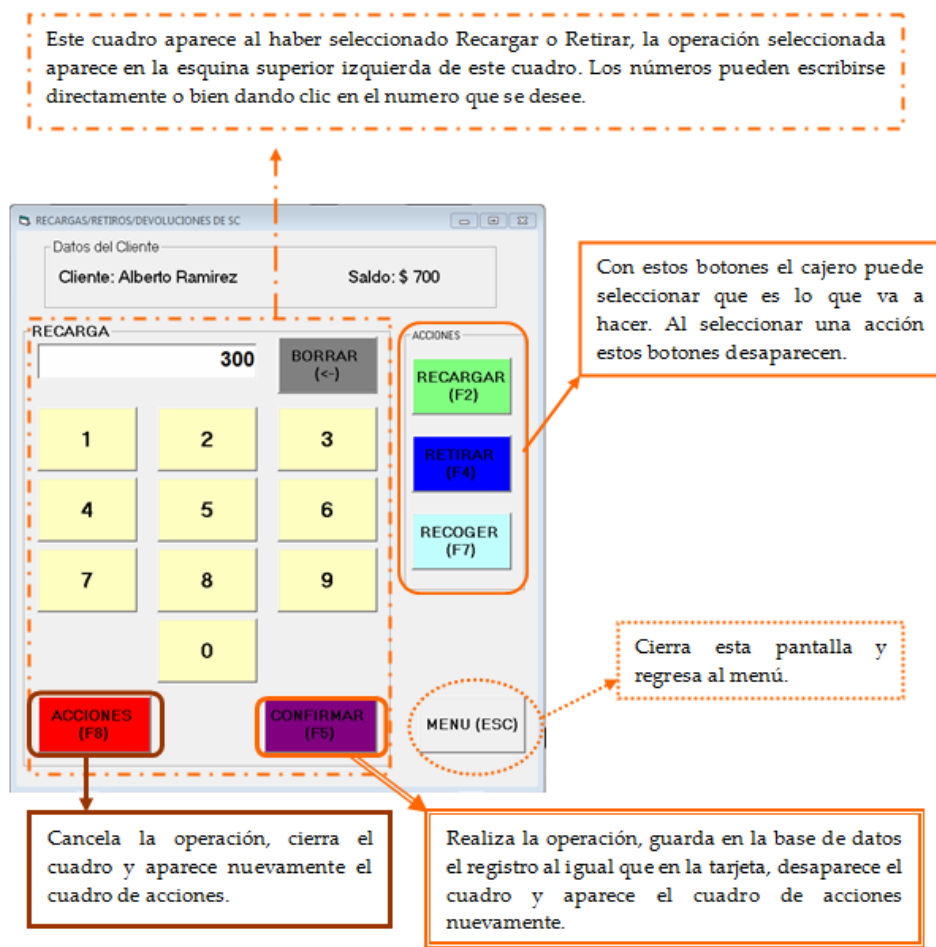


Fig. 13 Forma de Recargas/Retiros/Devoluciones

CONFIRMAR

Calcula cifra de auditoría

Graba operación en la TI y afecta el saldo (resta o suma el saldo)

Graba operación en la base de datos (tabla OPERACIONES_DE_CAJERO)

ACCIONES

En caso de que el cajero se hubiera equivocado o bien el usuario se arrepintiera de realizar la operación, al presionar este botón se cierra el cuadro con los botones numéricos y ofrece nuevamente el menú de acciones inicial.

El resto de las funciones de este modulo no se documentó en la tesis.

7.3.5. MÓDULO ADMINISTRADORES

En este módulo no se describe en la tesis puesto que es similar al de CAJEROS, aunque naturalmente usa otras tablas, etc.

En este modulo son especialmente importantes las funciones

- **Recibe-fondos-de-cajero**
- **Pago_a_puesteros**

7.3.6. MÓDULO CONTAGIOS

CONTAGIO_TI_A_BASE_DE_DATOS

Le graba a la tabla contagio de ventas el lote “más viejo”, son 5 operaciones de venta. Le graba a la tabla contagio de operaciones-de-caja el lote “más viejo”, son 5 operaciones de caja.

CONTAGIO_DE_BASE_DE_DATOS_A_TI

Le graba a la TI las últimas 5 ventas de la tabla VENTAS, pone el lote en el recientemente desocupado, cambia el indicador del “más viejo”, le graba a la TI las últimas 5 operaciones de la tabla operaciones-de-cajero, pone el lote en el recientemente desocupado, cambia el indicador del “mas viejo”

7.3.7. MÓDULO TODO-LECTORAS-Y-TARJETAS

Este modulo contiene todas las funciones referentes a la interacción del paquete BAZAR con la lectora y las tarjetas, las cuales son:

- Conexión de la lectora: busca si hay una lectora conectada a la PC.
 - a. De ser así, procede la comunicación con la lectora y si todos los protocolos están correctos, se realiza la conexión con la lectora.
 - b. Si no invoca la función manda_error
- Conexión con la tarjeta: verifica si hay una tarjeta, de ser así, revisa los protocolos de comunicación y verifica si la tarjeta corresponde con el modelo que se indico, si lo es, la conexión procede.
- Desconexión de tarjeta: indica a la lectora que deje de enviar energía a la tarjeta y la desconecta.
- Desconexión de la lectora

El código de las funciones aquí descritas se muestra en la Fase IV.

7.3.8. MÓDULO CONCENTRACIÓN DE DATOS

Observación fundamental (aplica a todo el módulo); nunca se procesa un registro sin validarlo antes.

7.3.8.1. RECOLECCIÓN DE DATOS

- Pide el dispositivo en el que se grabara
- Trae un archivo con los datos “anteriores” un archivo de transporte (no documentado) con los datos “anteriores”
- Se “empalman” los pares de tablas de ventas y de operaciones de caja
- La lista resultante de registros se empalman con los datos anteriores que traía el archivo del dispositivo
- Se actualiza la cifra de control del archivo

7.3.8.2. CONCENTRACION DE DATOS

OPCION se pueden usar dos archivos, uno para ventas y otro para operaciones de cajero.

Nunca se procesa un registro sin antes validarlo (audit).

Se indica el dispositivo y el nombre del archivo (que será en general siempre el mismo).

Se abre el archivo y se valida la cifra de control (se cuentan los registros que contiene el archivo). También se valida el audit number de la cifra de control.

Si hubiera alguna violación, se invoca la alarma correspondiente.

Se procesan los registros del archivo y se incluyen en el archivo o los archivos correspondientes (VENTAS u OPERACIONES-DE_CAJERO)

Naturalmente, no se incluyen duplicados.

Lógica sugerida al programador: Se construye una lista de todas las ventas registradas en esa computadora (uniendo – disyunción – las de la tabla de venta y la de contagios de ventas). A continuación se unen las que ya estaban en el archivo del transporte, y la lista resultante se graba en el archivo. Lo mismo se hace con operaciones de cajero.

7.3.9. MÓDULO CONSULTAS-DIVERSAS

No se documentaron en la tesis, debido al incremento de tamaño de la misma que resultaría de la inclusión de todos los programas.

7.4. LA BASE DE DATOS DE LA PRIMERA VERSIÓN DEL PAQUETE

La versión 1.0 de la base del BAZAR estaba dividida en 2 bases (la de puestero y la de cajero) y un catalogo que es el de productos. Sin embargo para la versión 2 se decidió que para efectos de hacer mas eficiente el contagio se unificarían las bases y el catalogo a una sola base de datos, dando como resultado BazarBase (descrita anteriormente).

En lo que a las tablas se refiere, en realidad no sufrieron grandes cambios, salvo la tabla de cajeros, que anteriormente estaba dividida en 4 tablas. Por tal motivo no se incluyó la descripción detallada de la versión 1.0.

CAPÍTULO 8: FASE IV

Implementación (programación, instalaciones y pruebas)

Los relatos y detalles de esta fase no se incluyeron en la tesis, no sólo por la extensión de los mismos, sino que se siguieron métodos clásicos para todas las componentes de los materiales a producir. Solo se incluyeron los programas o rutinas que pudieran servir a un investigador en un sentido concreto, ya sea para entender los algoritmos o para usarlos en alguna aplicación.

8.1. LOS PROGRAMAS

Se elaboran, prueban y depuran los programas definidos en la fase anterior. Se muestran las rutinas necesarias para el manejo de lectoras de tarjetas y la grabación y lectura de las tarjetas mismas.

8.1.1. RUTINA DE CONEXIÓN CON LA LECTORA

```
Public Function initialize_to_card_reader() As Boolean
```

```
deshabilitada = False
```

```
If deshabilitada Then
```

```
    manda_error
```

```
    Exit Function
```

```
End If
```

```
initialize_to_card_reader = False
```

```
If Not (get_context) Then
```

```
    MsgBox "NO SE ENCUENTRA LA LECTORA DE TARJETAS O NO SE  
PUDO CONECTAR, INTENTE DE NUEVO, SI NO FUNCIONA  
PROBABLEMENTE LA LECTORA ESTA DAÑADA", , "BAZAR"
```

```
    Exit Function
```

```
End If
```

```
retCode = SCardConnect(hContext, "ACS ACR38U 0", SCARD_SHARE_DIRECT,  
0, hcard, Protocol)
```

```
If retCode <> SCARD_S_SUCCESS Then Exit Function
```

```

SendLen = 4
SendBuff(0) = &H8
RecvLen = 262
retCode = SCardControl(hcard, ioctl_smartcard_set_card_type, SendBuff(0),
SendLen, RecvBuff(0), RecvLen, nbytesret)
If retCode <> SCARD_S_SUCCESS Then Exit Function
retCode = SCardDisconnect(hcard, SCARD_UNPOWER_CARD)
If retCode <> SCARD_S_SUCCESS Then Exit Function
initialize_to_card_reader = ConnActive
End Function

```

8.1.2. RUTINA DE CONEXIÓN CON LA TARJETA

```

Public Function conect_to_card() As Boolean
If deshabilitada Then
    manda_error
    Exit Function
End If
If ConnActive Then
    conect_to_card = True
    Exit Function
End If
conect_to_card = False
ConnActive = False
retCode = SCardConnect(hContext, "ACS ACR38U 0",
SCARD_SHARE_SHARED, SCARD_PROTOCOL_T0 Or
SCARD_PROTOCOL_T1, hcard, Protocol)
If retCode <> SCARD_S_SUCCESS Then Exit Function
ConnActive = True
conect_to_card = True
End Function

```

8.1.3. RUTINA DE DESCONEXIÓN DE LA TARJETA

```
Public Function disconnect_card()  
If ConnActive Then  
    retCode = SCardDisconnect(hcard, SCARD_UNPOWER_CARD)  
    ConnActive = False  
End If  
End Function
```

8.1.4. RUTINA DE DESCONEXIÓN DE LA LECTORA

```
Sub disconnect_card_reader()  
If ConnActive Then  
    retCode = SCardDisconnect(hcard, SCARD_UNPOWER_CARD)  
    ConnActive = False  
End If  
retCode = SCardReleaseContext(hContext)  
End Sub
```

8.2. DOCUMENTACIÓN

Se documentan los programas y procesos. En general se complementa la documentación recibida como parte de la definición de los programas. Especialmente se incluye una lista de módulos (de clase o .bas) de formas.

Se entrega una descripción de la lógica, incluyendo

- La navegación del programa
- Los archivos y tablas de bases de datos que utiliza.

Los detalles de algoritmos deben estar presentes en la definición recibida de la fase III.

8.2.1. LA LISTA DE FORMAS DEL PRYBAZAR

En esta sección se enlistan todas las formas que utilizó el paquete BAZAR versión 1.0, debido a que la versión 2 no está del todo programada.

- fadministracion
- falta_cliente
- fcaja
- fcajero
- fconsulta_movimientos_adm
- fconsulta_ventas
- fcorte_de_caja
- fidentificate
- flista_cajeros
- flista_cortes
- flista_productos
- flista_puesteros
- fmenu
- fproductos
- fpuestos
- frecargas_retiros
- freportes
- fventas

8.3. INSTRUCTIVOS DE OPERACIÓN

Se elaboran instructivos de uso de los programas. Esencialmente serán explicaciones de las formas y cómo se las usa, pero también se indicarán claramente las situaciones anormales que se pudieran presentar y cuáles son las acciones que debe tomar el usuario en cada una de ellas.

No se muestran instructivos en esta tesis.

8.4. INSTALACIÓN PARA PROBAR EL SISTEMA

Se instalan y prueban las componentes físicas de todo tipo. Consiste en probar físicamente todo lo necesario: lectoras, tarjetas, computadores, etc.

8.5. DETALLES DE LOS PROCEDIMIENTOS DE EMERGENCIA Y EXCEPCIÓN

Se elaboran los procedimientos de emergencia, tratando de cubrir cualquier excepción que se pudiera presentar y mostrando al usuario en donde se encuentra el error.

8.6. DATOS PARA LA DEMOSTRACIÓN DEL PAQUETE

Se preparan datos para la demostración del sistema. En el caso del BAZAR, se preparó un evento simulado, con algunos datos de cada tipo. Esto permitirá a los encargados de la fase siguiente probar el sistema en vivo.

CAPÍTULO 9. FASE V

Instalación del sistema

Como se trata de un paquete, y no un sistema a medida, las actividades difieren en ciertos aspectos.

- **CAPACITACIÓN:** no se capacita a usuarios, sino se prepara el material para que los futuros usuarios puedan usar el paquete.
- **MATERIALES PARA CAPACITACIÓN:** se hace de acuerdo a la naturaleza del paquete. Como el BAZAR tiene pocos elementos que tienen que ser explicados, puesto que las formas que presenta se explican solas, salvo algunos detalles, en lugar de material para capacitación se dota a los programas de “ayudas” e instructivos de operación.
- **MATERIALES PARA INSTANCIACIÓN:** este es el aspecto más importante de esta fase. Deben estar perfectamente detalladas las actividades, y en especial, las circunstancias especiales que pueden hacer que una instancia presente diferencias comparada con otras.

Los pasos para una instanciación (preparar el sistema que usará un bazar, usando el paquete BAZAR)

1. Se determina dónde estarán las computadoras, y se consiguen en número suficiente. Aunque es conveniente que haya una computadora de refacción, no es imprescindible.
2. Se instalan (físicamente) las computadoras. Muchas veces esta actividad la hará personal del organismo que ofrece el bazar.
3. Se instalan las lectoras (una en cada una de las computadoras).
 - a. Se instala el “driver” de la lectora.
 - b. Se conecta la lectora (en general serán tipo USB).
4. Se carga el programa ejecutable BAZAR.exe. Este paso incluye otros archivos, pero son para bitácoras y otros propósitos.
5. Se prueba la lectora (con un programa que lleva el instalador). Si no funciona, se determina el motivo y se corrige.

6. Se prepara una base de datos para el evento. En general se hará en la computadora del proveedor de servicio de sistemas para el evento.
 - a. Se genera el parámetro único de la instancia.
 - b. Se proporcionan ciertos datos (el nombre del evento entre ellos)
 - c. Se especifican las opciones para esa instancia
 - d. Se crea un usuario inicial, y se le asigna una palabra clave, misma que se le proporciona al “primer administrador”, que será el encargado de agregar otros usuarios al sistema.
 - e. Se entrega la base de datos a los administradores del sistema.
7. Los administradores introducen los datos básicos de su evento
 - a. Los puestos
 - b. Los productos que se ofrecerán, cada uno con un precio
 - c. Las cajas
 - d. Se Introducen los usuarios
 - i. cajeros
 - ii. puesteros
 - iii. otros administradores.
 - e. Se asignan roles a los usuarios.
 - f. En general proporcionarán a los usuarios una palabra clave (se sugiere que sea el “número de usuario” que les tocó) y se invita a que ellos mismos la modifiquen antes de que se les entregue la TI de usuario.
 - g. Se emiten las TI de usuario y se entregan a las personas.
 - h. Se actualizan los catálogos cajero-caja y puestero-puesto.
8. Se instala (copia) la base de datos en todas las computadoras. En cada una, se pone el Número inicial de operaciones de venta o de caja (puede ser una serie de 10,000 o 100.000 números reservados para esa computadora).
9. Se prueba el sistema; por ejemplo, se invoca una VENTA y se cancela antes de grabarla.

10. Se hace una demostración de los procesos a los actores involucrados (las personas que tendrán ciertos roles en el evento).
11. Se les aclara que no podrán iniciar una sesión sin introducir su tarjeta de usuario. Si el evento lo exige (es una opción de instanciación) tendrán que proporcionar su palabra clave. Si no tienen su tarjeta, deben acudir al administrador para que determine el motivo (la perdieron, etc.) y solicitar una nueva (esto le asignará un nuevo número de usuario). Si fue pérdida o hay sospecha de hurto, el administrador enviará a alguien a que elimine el usuario anterior en *todas las computadoras*. Si la tarjeta se rompió o ya no puede ser leída, la entrega al administrador y éste le proporciona otra.
12. Se explica a los puesteros que pueden usar el sistema con o sin mouse, y con o sin el teclado, y se les muestra ambos modos y cómo se cambia de un modo a otro (aunque no hace falta indicarlo, puesto que siempre están planeados ambos dispositivos).
13. Se hace una demostración al personal que recolecta los datos de las diversas computadoras, y se informa de las excepciones que pudieran surgir. Entre otras, un puestero o cajero “no los deja” trabajar. Le informarán que su actividad es prioritaria, y que de todos modos no tardarán mucho: se inserta una memoria USB, se ejecuta un programa, y se van (el programa es casi instantáneo).
14. Se les informa claramente qué deben hacer cuando:
 - a. se detecte una violación al sistema
 - b. se descomponga su lectora
 - c. no pueda leer una tarjeta de Visitante
 - d. el sistema no puede leer su tarjeta de usuario
 - e. no funciona la computadora
 - f. no hay energía eléctrica
 - g. un visitante protesta por algo relacionado con el sistema
 - h. un visitante pide “fiado”: véndeme aunque no tenga saldo, luego cargo mi tarjeta y te pago

- i. un visitante desea pagar en efectivo por algo (está prohibido o no, es decisión del organizador del bazar)

Y naturalmente, las instrucciones que les quieran dar los organizadores.

Comentarios sobre la “capacitación” de los usuarios

Con cierta frecuencia, que tiende a disminuir, habrá actores del sistema que nunca han usado una computadora, o en particular, un sistema de información (sólo usan el correo electrónico). Se debe tener preparada una presentación corta y puntual para estos casos, en la cual se explica el BAZAR pero para personas que no conocen la terminología ni los conceptos de computación. No les digan “botón” sin especificar que es como un timbre... Y expliquen lo que es un “menú”, lo que significa “confirmar” o “cancelar”, etc. Naturalmente las personas que tengan que actualizar los catálogos no tendrán esas características: si no hay nadie que lo pueda hacer, lo hará el proveedor del servicio de sistemas.

CONCLUSIONES

Se elaboró un paquete denominado BAZAR con el objetivo principal de ilustrar el uso de tarjetas inteligentes como medio para transportar (copiar) datos entre computadoras. El sistema demostró que el efecto contagio puede disminuir los temores de los actores del sistema de que se produzcan circunstancias que los perjudican. Se explicó el modo en el que se seleccionaron los tipos de tarjetas y lectoras apropiadas para el uso del paquete, tomando en cuenta las especificaciones y comparándolas con las características que aprovecharía el paquete, pero incluyendo consideraciones de costo de los equipos. En especial se mostraron los aspectos en los que el contagio contribuye a la reducción de tales dudas. Se explicaron los dispositivos de seguridad que se agregaron al paquete para prevenir o detectar violaciones a los controles incorporados al sistema para tal efecto. Se usó la descripción del desarrollo del paquete, especialmente el diseño del mismo, como ejemplo de la aplicación del método de las 5 fases del desarrollo de un sistema, y del artificio utilizado para obtener un diseño adecuado de la base de datos del mismo.

Se analizaron situaciones de otro tipo, no sólo los bazares, en los cuales podría ser aplicado el paquete, o alguna variación menor de sus funciones. La investigación futura incluirá la elaboración completa de la versión 2 del paquete, así como su generalización para otras aplicaciones. Inicialmente, se considera que el paquete podría funcionar en un mercado donde los clientes pagarían con una tarjeta inteligente, pero se encontraron circunstancias ajenas al mundo de la informática que pudieran constituir argumentos contra su uso en algunos mercados, especialmente relacionados con temas de control de alguna autoridad sobre los puestos. También se estudiaron otras aplicaciones del efecto contagio proporcionado por el uso de tarjetas inteligentes en otras situaciones, por ejemplo para obviar ciertos dispositivos para respaldo de la información. En general, este efecto puede permitir lo que se ha denominado una red sin red por los miembros del equipo de investigación que participaron en el proyecto que se describió en este trabajo.

REFERENCIAS

ACS Advanced Card Systems Ltd (2010). ACR38 Smart Card Reader [Online]. Available ACS web site:

http://www.acs.com.hk/index.php?pid=product&prod_sections=0&id=ACR38

ACS Advanced Card Systems Ltd (2010). AET63 BioTRUSTKey [Online]. Available ACS web site:

http://www.acs.com.hk/index.php?pid=product&prod_sections=0&id=AET63

ACS Advanced Card Systems Ltd (2010). PC-Linked Smart Card Readers [Online]. Available ACS web site:

<http://www.acs.com.hk/index.php?pid=products&id=2>

ACS Advanced Card Systems Ltd (2010). Smart Card and Smart Cards Operating Systems [Online]. Available ACS web site:

<http://www.acs.com.hk/index.php?pid=products&id=1>

ATMEL (2010). Serial EEPROM AT24C64 [Online]. Available ATMEL web site:

http://www.atmel.com/dyn/products/product_card.asp?part_id=2482

Bauer Mengelberg, J.R., "Teaching System Access Control", *Issues in Informing Science and Information Technology*, vol. 2, pp 139-158, 2005.

Dictionary.com (2010). Bazar [Online]. Available Dictionary.com web site:

<http://dictionary.reference.com/browse/bazar>

Eslami, Y., Sheikholeslami, A., Gulak, P.G., Masui, S. and Mukaida, K., "An Area-Efficient Universal Cryptography Processor for Smart Cards", *IEEE Transactions On Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 1, January 2006.

Fletcher, P. "Europe holds a winning hand with smart cards", *Electronic Design*, vol. 47, no. 1, pp 106-107, January 1999.

Hansmann Uwe, S. Nicklous Martin, Schack Thomas, Schneider Achim and Seliger Frank, *Smart Card Application Development Using Java*, 2th ed, Boblingen, Germany: Springer, 2002.

Neumann, G. and Strembeck, M., "An approach to engineer and enforce context constraints in an RBCA", *Proceedings of the 8th ACM Symposium on Access Control Models and Technologies (SACMAT)*, Como, Italy, 2003.

Nevado, M.V., *Introducción a las bases de datos relacionales*, Madrid: Vision Libros, 2010.

Nitsche, U., Holbein, R., Morger, O. and Teufel, S., "Realization of a context-dependent access control mechanism on a commercial platform", *Proceedings of the 14th International Information Security Conference (IFIP/Sec'98, part of the 15th IFIP World Computer Congress)*(Vienna/Budapest, Austria/Hungary, 1998), Austrian Computer Society, Vienna, pp. 160-170.

Shelfer, K.M. and Procaccino, J.D., "Smart Card Evolution", *Communications Of The ACM*, vol. 45, no. 7, pp. 83-88, July 2002.

Wikipedia (2010) Tarjeta Inteligente [Online]. Available: Wikipedia Web site:

http://es.wikipedia.org/wiki/Tarjeta_inteligente

Wikipedia (2010). Java Card [Online]. Available: Wikipedia Web site:

http://es.wikipedia.org/wiki/Java_Card

Wolfgang Rank, *Smart Cards Applications: Design Models for using and programming smart cards*, Germany: Wiley, 2007.

Zoreda, J.L. and Otón, J.M., *Smart Cards*, n.p.: Artech House, 1994.

ANEXOS (CD)

En el disco compacto que se entrega junto con la tesis (o en el mismo disco en caso de ser la versión digital de la tesis) se encuentran dos módulos en extensión .bas programados en VB6:

- módulo de lectora: éste contiene todas las funciones referentes a la lectora (conexión, desconexión, manejo de errores, etc.)
- módulo de tarjeta: éste contiene las funciones de grabación de una tarjeta, lectura de la misma y las funciones de conversión de distintos tipos de datos a byte para poder enviarlos a la tarjeta así como de las funciones de conversión inversas al leer la tarjeta.