

**DISEÑO DE UNA ARQUITECTURA BASADA EN AGENTES  
MÓVILES PARA COMERCIO ELECTRÓNICO**

**LUIS ANTONIO LEÓN CHACÓN**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA  
2004**

**DISEÑO DE UNA ARQUITECTURA BASADA EN AGENTES  
MÓVILES PARA COMERCIO ELECTRÓNICO**

**LUIS ANTONIO LEÓN CHACÓN**

**Trabajo de Investigación para optar por el  
Título de Magíster en Informática**

**Director**

**SERGIO F. CASTILLO C.**

**Ingeniero de Sistemas, Ph. D.**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA  
2004**

*Con todo mi amor A Dios,  
Mis padres Maria de los  
Ángeles y Luis Antonio, y a  
Mi Amor Janeth.*

## **AGRADECIMIENTOS**

El autor expresa sus agradecimientos a:

Ph. D. Sergio F. Castillo C., Director de la investigación, por apoyarme, por sus orientaciones, y por tener esa admirable paciencia.

Ing. Janeth Gissella Gómez Gualdrón, Autora del prototipo ADAM: Aplicación de los Agentes Móviles al Comercio Electrónico, por su apoyo y colaboración en todo momento.

Mi nena Jane por su amor, paciencia y ser como es. Me diste la fuerza para iniciar y lograrlo.

Mis padres: Luis Antonio, por enseñarme el valor de la dedicación al trabajo y María de los Ángeles, por apoyarme en todo instante.

Nani, Gilberto, Ledy, Sergio, Hernán, Ariel, Tomas, Carlos, Claudia, Wilson, Raúl, JuanK y Juan de Dios, mis amigos, por serlo aunque parezca que me olvido de ellos.

Los profesores que me enseñaron grandes lecciones académicas y de vida. Destacando a Hernán, Sergio y Juan de Dios.

Todas aquellas personas que de una u otra manera contribuyeron a mi formación integral y a la realización del presente proyecto.

Quienes olvido, porque siempre me pasa.

También a quienes son un contra ejemplo a seguir, por recordarme quién debo evitar ser.

Dios, por darme todo lo que necesito, alejar todo lo que no me sirve, estar siempre conmigo y no desampararme cuando estoy desfalleciendo incluso en los momentos en que mi fe se encuentra disminuida.

## TABLA DE CONTENIDO

<b>INTRODUCCIÓN.....</b>	<b>1</b>
<b>1. PLANTEAMIENTO DEL PROYECTO.....</b>	<b>5</b>
1.1. DESCRIPCIÓN DEL PROBLEMA .....	5
1.2. OBJETIVOS .....	7
1.2.1. Objetivo General .....	7
1.2.2. Objetivos Específicos.....	8
1.3. ESTRUCTURA DEL DOCUMENTO.....	8
<b>2. ESTADO DEL ARTE Y LA TÉCNICA.....</b>	<b>11</b>
2.1. AGENTES MÓVILES .....	11
2.1.1. De tipos de datos primitivos a agentes, pasando por los objetos.....	11
2.1.2. Conceptos básicos.....	13
2.1.3. Clasificación de agentes .....	15
2.1.4. Los inicios en el campo de los Agentes Móviles .....	16
2.1.5. El modelo de Agentes Móviles.....	17
2.1.6. Ventajas del Modelo de Agentes Móviles.....	18
2.1.7. Desventajas del Modelo de Agentes Móviles .....	20
2.2. PLATAFORMAS DE AGENTES MÓVILES.....	21
2.2.1. Aglets Workbench de IBM. ....	22
2.2.2. Java Agent DEvelopment Framework (JADE). ....	23
2.2.3. Voyager de ObjectSpace. ....	25
2.2.4. Odyssey de General Magic.....	26
2.3. AGENTES MÓVILES Y COMERCIO ELECTRÓNICO.....	27
2.3.1. Comercio Electrónico.....	27
2.3.2. Transacciones de comercio electrónico basadas en Agentes Móviles. ....	29
2.3.3. Sistemas basados en Agentes Móviles para Transacciones de CE. ....	35

<b>3.</b>	<b>REQUERIMIENTOS DE UNA ARQUITECTURA PARA COMERCIO ELECTRÓNICO.....</b>	<b>44</b>
3.1.	ANÁLISIS DE LOS SISTEMAS PARA COMERCIO ELECTRÓNICO ACTUALES	44
3.2.	REQUISITOS DEL SISTEMA PARA COMERCIO ELECTRÓNICO .....	45
3.2.1.	Sistema del Comprador.....	46
3.2.2.	Sistema del Vendedor.....	47
3.2.3.	Tipos de Servicios.....	47
3.3.	REQUISITOS DE LA PLATAFORMA DE AGENTES MÓVILES.....	48
3.3.1.	Seguridad básica.....	48
3.3.2.	Servicios de Agentes .....	48
3.4.	REQUISITOS DEL LENGUAJE DE PROGRAMACIÓN .....	49
3.5.	REQUISITOS DE SEGURIDAD .....	50
3.5.1.	Protección del servidor.....	50
3.5.2.	Protección de los agentes.....	50
<b>4.</b>	<b>PRESENTACIÓN DE ARAMCEL .....</b>	<b>51</b>
4.1.	ANÁLISIS Y DISEÑO GLOBAL .....	51
4.2.	DESCRIPCIÓN GENERAL DE ARAMCEL .....	53
4.3.	SISTEMA DEL COMPRADOR .....	56
4.3.1.	Funcionamiento.....	58
4.3.2.	Plataforma de Agentes.....	59
4.3.3.	Pseudocódigo de algunos Agentes.....	61
4.4.	SISTEMA DEL VENDEDOR.....	66
4.4.1.	Funcionamiento.....	67
4.4.2.	Plataforma de Agentes.....	68
4.4.3.	Pseudocódigo de algunos Agentes.....	69
4.5.	TRANSACCIONES DE COMERCIO ELECTRÓNICO EN ARAMCEL.....	74
4.5.1.	Búsqueda de información. ....	75
4.5.2.	Compra de Artículos .....	78
4.5.3.	Notificación de Ofertas .....	82
4.5.4.	Transacciones a través del Servidor de Compradores .....	85

<b>5.</b>	<b>SEGURIDAD EN ARAMCEL.....</b>	<b>87</b>
5.1.	AMENAZAS DE SEGURIDAD EN ARAMCEL.....	87
5.1.1.	Amenazas de Seguridad Inherentes al Modelo de Agentes Móviles.....	87
5.1.2.	Principales problemas del comercio electrónico.....	91
5.2.	MODELO DE SEGURIDAD DE ARAMCEL.....	93
5.2.1.	Autenticación.....	96
5.2.2.	Autorización.....	97
5.2.3.	No-Repudio.....	98
5.2.4.	Solución al problema del Agente Hostil.....	98
5.2.5.	Solución al problema del Servidor Hostil.....	99
5.2.6.	El Agente Fiscalizador.....	99
5.2.7.	Protocolo de Seguridad.....	100
<b>6.</b>	<b>VALIDACIÓN DE LA ARQUITECTURA.....</b>	<b>103</b>
6.1.	TECNOLOGÍAS UTILIZADAS.....	103
6.2.	DESCRIPCIÓN GENERAL DE ADAM.....	103
6.2.1.	Marco de trabajo de ADAM.....	105
6.2.2.	Diagrama general de clases de ADAM.....	106
6.3.	CARACTERÍSTICAS VALIDADAS POR MEDIO DE SOFTWARE.....	109
6.4.	CARACTERÍSTICAS NO VALIDADAS POR MEDIO DE SOFTWARE.....	110
6.5.	EVALUACIÓN DEL FUNCIONAMIENTO DE ADAM.....	112
6.5.1.	Pruebas de Verificación.....	112
6.5.2.	Pruebas de Validación.....	113
6.5.3.	Pruebas de Rendimiento.....	115
	<b>CONCLUSIONES.....</b>	<b>119</b>
	<b>RECOMENDACIONES PARA TRABAJOS FUTUROS.....</b>	<b>123</b>
	<b>REFERENCIAS.....</b>	<b>125</b>

## LISTA DE FIGURAS

Figura 1. El modelo Cliente/Servidor y el modelo de los agente móviles.....	16
Figura 2. Ejemplo del Modelo Cliente Servidor. ....	29
Figura 3. Ejemplo del Modelo de Agentes estacionarios.....	31
Figura 4. Ejemplo del Modelo de Agentes Móviles.....	33
Figura 5. Infraestructura y líneas de comunicación de Agora.....	35
Figura 6. Protocolo de Seguridad en MAgNET. ....	39
Figura 7. Arquitectura del Sistema E-marketplace.....	41
Figura 8. Caso de Uso Buscar Información.....	51
Figura 9. Caso de Uso Ordenar Compra / Venta. ....	52
Figura 10. Caso de Uso Notificar Compradores .....	52
Figura 11. Arquitectura ARAMCEL.....	53
Figura 12. Migración de un Agente Móvil.....	55
Figura 13. ARAMCEL - Sistema del Comprador.....	57
Figura 14. Pseudocódigo del Agente Móvil Buscador. ....	63
Figura 15. Pseudocódigo del Agente Móvil Comprador.....	65
Figura 16. ARAMCEL - Sistema del Vendedor .....	66
Figura 17. Pseudocódigo del Agente Móvil Notificador. ....	71
Figura 18. Pseudocódigo del Agente Estacionario Vendedor. ....	73
Figura 19. Pseudocódigo del Agente Estacionario Despachador. ....	74
Figura 20. Diagrama de Secuencia Búsqueda de Información (Parte 1). ....	76
Figura 21. Diagrama de Secuencia Búsqueda de Información (Parte 2). ....	77
Figura 22. Diagrama de Secuencia Búsqueda de Información (Parte 3). ....	78
Figura 23. Diagrama de Secuencia Compra de Artículos (Parte 1). ....	79
Figura 24. Diagrama de Secuencia Compra de Artículos (Parte 2). ....	81
Figura 25. Diagrama de Secuencia Compra de Artículos (Parte 3). ....	82
Figura 26. Diagrama de Secuencia Notificación de Ofertas y Novedades (Parte 1). ....	83
Figura 27. Diagrama de Secuencia Notificación de Ofertas y Novedades (Parte 2). ....	84
Figura 28. Diagrama de Secuencia Notificación de Ofertas y Novedades (Parte 3). ....	85

Figura 29. Interacción UC – AEAC en Terminales Limitadas.....	86
Figura 30. Agente Hostil. ....	88
Figura 31. Servidor Hostil. ....	90
Figura 32. Esquema de Cifrado / Descifrado. ....	95
Figura 33. La estructura de un agente en ARAMCEL.....	96
Figura 34. Protocolo de seguridad en ARAMCEL.....	101
Figura 35. Marco de Trabajo de ADAM.....	105
Figura 36. Diagrama General de Clases Sistema del Comprador. ....	106
Figura 37. Diagrama General de Clases del Sistema del Vendedor. ....	108
Figura 38. Ubicación de equipos en pruebas de rendimiento.....	116



## LISTA DE TABLAS

Tabla 1. Convenciones para Diagramas de Secuencia.....	56
Tabla 2. Descripción General de Clases – Sistema del Comprador. ....	107
Tabla 3. Descripción General de Clases – Sistema del Comprador. ....	108
Tabla 4. Características de los equipos en las pruebas de rendimiento .....	116
Tabla 5. Resultados de las pruebas de rendimiento .....	117

## RESUMEN

### TITULO:

**DISEÑO DE UNA ARQUITECTURA BASADA EN AGENTES MÓVILES PARA COMERCIO ELECTRÓNICO\*.**

### AUTOR:

LUIS ANTONIO LEÓN CHACÓN\*\*

### PALABRAS CLAVES:

Agente Móvil, Comercio Electrónico, Seguridad, Agente Hostil, Servidor Hostil, Arquitectura Software.

### DESCRIPCIÓN:

Los agentes móviles son entidades software que pueden transportar su código y datos desde un computador a otro a través de Internet; debido a esa movilidad presentan grandes ventajas al trabajar en sistemas distribuidos y colaborativos, por esas ventajas y por su capacidad de representar a un usuario en la red se estudió la posibilidad de aplicarlos en el Comercio Electrónico. En este sentido se definen las siguientes transacciones: búsqueda, compra / venta y notificación.

En la Arquitectura basada en Agentes Móviles para Comercio Electrónico "ARAMCEL" se representan los componentes y la interacción para realizar esas transacciones, donde un Usuario Comprador puede enviar un agente para que lo represente e interactúe con otros agentes en sitios remotos de la red y así obtener el artículo que busca. Para ello se especifican dos sistemas, el Sistema del Comprador y el del Vendedor, con sus respectivos componentes. Se tiene en cuenta que en las transacciones electrónicas y al usar agentes móviles, hay problemas de seguridad porque tanto los agentes móviles como los servidores con que interactúan son vulnerables a ataques y brechas de seguridad.

Se enfocan tres problemas de seguridad: Autenticación, Autorización y No Repudio; se incluye una solución al problema del agente hostil y del servidor hostil. En ARAMCEL se presenta un mecanismo de seguridad que propone una solución a esos problemas cumpliendo con la realización de las transacciones definidas. El modelo de seguridad depende de un Servidor Central, servidores confiables y la Infraestructura de llave pública (PKI). Considerando las características de ARAMCEL, es posible extenderla y enriquecerla para contrarrestar nuevos tipos de ataques que surjan en el entorno. La validación se realizó por medio del prototipo ADAM: Aplicación de los agentes móviles al comercio electrónico.

---

\* Trabajo de Investigación.

\*\* Facultad de Ingenierías Físico - Mecánicas. Maestría en Informática. Escuela de Ingeniería de Sistemas e Informática. Director, Ph.D. Sergio F. Castillo C.

## ABSTRACT

**TITLE:**

**A DESIGN OF AN ARCHITECTURE BASED ON MOBILE AGENTS FOR E-COMMERCE\* .**

**AUTHOR:**

LUIS ANTONIO LEÓN CHACÓN\*\* .

**KEYWORDS:**

Mobile agent, E-commerce, Security, Malicious Agent, Malicious Server, Software Architecture.

**DESCRIPTION:**

Mobile agents are software entities that they can transport their code program and data from a computer to another through Internet; due to that mobility they present huge advantages when they work in distributed and collaborative systems, for those advantages and for their capacity to represent an user in the net it was studied the possibility of applying them in the E-commerce. In this sense the following transactions are defined: search, buys / sale and notification.

In the Architecture based on Mobile Agents for E-commerce "ARAMCEL", the components and the interaction are represented to carry out those transactions, where a Buyer can send an agent so that it represents him and It interacts with other agents in remote places of the net and this way to obtain the article that it looks for. For that reason they are specified it two systems, Buyer's System and Supplier's System, with their respective components. In this work keeps in mind that in the electronic transactions and when mobile agents are used, there are security risks because as much the mobile agents as the servers with which they interact are vulnerable to attacks and breaches of security.

Three security problems are focused: Authentication, Authorization and No-repudiation; besides a solution to Malicious agent's problem and Malicious server is included. In ARAMCEL a security model is presented that proposes a solution to those problems fulfilling the realization of the defined transactions. The security model depends on a Central Server, reliable servers and the Infrastructure of public key (PKI). Besides considering the characteristics of ARAMCEL, it is possible to extend it and to enrich it to counteract new types of attacks that they can arise in the environment. Validation was carried out by means of the prototype ADAM: Application of the mobile agents to the e-commerce.

---

\* Research Work.

\*\* Faculty of Engineering Physical - Mechanical. Master in Computer Science. School of Systems Engineering and Computer Science. Director, PhD. Sergio F. Castillo C.

## INTRODUCCIÓN

El actual auge de las tecnologías de la información y la comunicación ha transformado la vida diaria de las personas. Gracias al Internet, por ejemplo, hay una cantidad ilimitada de información a la que los usuarios pueden acceder manualmente mediante los sitios Web disponibles o utilizando programas para descargar información de manera automática en los equipos. Internet también elimina la distancia entre la gente a través de medios como el correo electrónico, los grupos de conversación (vía texto, voz y audio visual), páginas web personales, educativas, de investigación, industriales, etc. De esta manera se provee colaboración, entretenimiento, educación, trabajo para todas las edades y profesiones; en general surge un cambio en la forma de vida, donde hay un acceso más rápido, fácil y económico a información actualizada.

Pero, Internet y sus tecnologías asociadas no sólo están cambiando la forma de vivir y pensar de los ciudadanos comunes, también las empresas están ingresando a la economía electrónica; se está revolucionando la concepción acerca de los negocios, del intercambio de información, bienes, servicios y capital. En un negocio es posible dar a los clientes un acceso controlado a los recursos del proveedor de servicios, de esta manera mediante Internet un cliente puede comunicarse con el sistema informático de una compañía sin la intervención de humanos. Pero las tecnologías informáticas no sólo han modificado las relaciones entre los clientes y las empresas, también están modificando cualquier clase de negocio o transacción comercial que involucre el intercambio de información, intercambio que ahora es más fácilmente realizado a través de Internet. Esto cubre un amplio rango de diferentes tipos de negocios, desde relaciones comerciales entre empresas, empresas con consumidores y consumidores con ellos mismos. Con lo anterior surge una forma de comercio en la cual las transacciones (como pueden ser solicitudes de información del producto, información de precios, solicitudes de descuentos) entre los diferentes actores comerciales (compradores, vendedores, intermediarios) se llevan a cabo mediante Internet y otras tecnologías de la información y la comunicación. Esta forma de comercio se denomina "Comercio Electrónico".

De esta forma, si bien las transacciones comerciales no se modifican, si se realizan en forma diferente gracias a los cambios tecnológicos. Así por ejemplo, una transacción tan simple como la petición de un precio X de un producto Y, han evolucionado desde la petición verbal de un comprador a un vendedor, a la petición telefónica, petición vía telefax, y ahora peticiones electrónicas mediante Internet y otras tecnologías. De esta forma se ha realizado un avance en la automatización de las transacciones.

Pero incluso dentro del comercio electrónico hay diferentes niveles de automatización. La consulta de un precio, por ejemplo, puede ser realizada por un comprador visitando la página web de la empresa vendedora. Este proceso puede ser mejorado en cuanto un usuario podría utilizar un programa informático que "visitara" diferentes páginas web de diversas empresas y como resultado "presentara" al usuario la página web donde encontró el mínimo precio para el producto deseado.

Sin embargo, este programa informático no podría ser totalmente determinístico, debería poder responder a situaciones ligeramente inesperadas o tomar decisiones sin tener que solicitar la intervención del usuario. Dicho programa por ejemplo, debería poder establecer que encontró un producto mejor a un precio inferior, o conocer, aunque fuera un poco, las preferencias del usuario. De esta forma, se necesita utilizar sistemas informáticos que utilicen técnicas de inteligencia artificial.

De las diferentes técnicas de inteligencia artificial que existen, los sistemas basados en agentes demostró ser la más adecuada para el problema que nos ocupa. En esencia, un agente es una entidad software (implementada mediante un lenguaje basado en objetos) que tiene la capacidad para interactuar con el medio ambiente donde se encuentra, comunicarse con otros agentes y realizar tareas de manera que alcance sus metas. Un agente puede representar a un usuario dentro de una transacción comercial, puesto que cuenta con características que lo hacen autónomo, reactivo y proactivo. Con los agentes el usuario tiene una representación de sí mismo para realizar operaciones de búsqueda de información sobre productos requeridos, e incluso dar a esos agentes la autonomía necesaria para estar en capacidad de formalizar compras utilizando su número de tarjeta de crédito.

De esta forma llegamos a realizar transacciones comerciales mediante sistemas informáticos basados en agentes. Dicho sistema puede, por ejemplo, conectarse desde la máquina de un usuario para determinar el precio X de un producto Y en sitios web de diferentes empresas, y poder responder a diferentes situaciones inesperadas gracias a las capacidades ya mencionadas de los agentes.

Sin embargo, existen dos problemas esenciales al escenario planteado. El primero es que, actualmente, la evolución de la tecnología ha cambiado la imagen tradicional de aquello que es la "máquina de un usuario". De hecho, se estima que para los próximos años la máquina más usada por los usuarios para navegar por Internet serán los considerados dispositivos móviles. Por las capacidades y por filosofía de uso, en dichos dispositivos no es posible utilizar sistemas basados en agentes tradicionales. Por otra parte, el escenario planteado no reduce uno de los problemas principales: la sobrecarga de la red de comunicaciones.

El grupo de trabajo de la presente tesis de maestría decidió entonces enfocar la investigación hacia una mejora del escenario planteado, mediante el uso de agentes móviles. Dichos agentes tienen la capacidad intrínseca de transportar su código a otro equipo a través de la red y continuar su ejecución allí; de esa manera pueden realizar operaciones directamente en la máquina donde se encuentran los recursos.

Desde el punto de vista de implementación, los agentes móviles (que serán presentados con mayor profundidad durante el texto del presente documento) pueden ser vistos como programas que viajan y se ejecutan de máquina en máquina. Aunque el principio no es novedoso, este principio ya se aplicaba en la década de los años ochenta, el uso de este principio si lo es. De tal forma que los agentes móviles permiten a los usuarios disminuir los inconvenientes de tiempo y dinero generados al tener que mantener su sesión en la red, revisando por si mismos cada paso en la interacción con los recursos que hay en las máquinas remotas.

El escenario final del trabajo aquí presentado es la realización de transacciones comerciales mediante sistemas informáticos basados en agentes móviles. En dicho sistema el usuario activa un agente (que conoce sus preferencias comerciales) y le

comunica la acción a realizar (como por ejemplo, solicitar el precio X de un producto Y a diferentes empresas). Dicho agente podría entonces comunicar estas preferencias a un agente móvil comprador que viajaría por la red de redes, visitando las máquinas de las empresas, y realizando peticiones en nombre del usuario. En cuanto termine su trabajo, regresaría a la máquina del usuario (o al sistema donde se encuentre registrado ese usuario) para informar de sus resultados. Y esto, utilizando las ventajas anteriormente mencionadas de los agentes, pero a su vez este modelo por ser una aplicación distribuida tiene que interactuar con diversos ambientes, algunos de los cuales pueden ser hostiles y por tanto conllevan riesgos de seguridad.

Con base en las situaciones planteadas, el problema de esta tesis se puede presentar de la siguiente forma: "¿Es posible realizar transacciones relacionadas con comercio electrónico mediante el uso de sistemas informáticos basados en agentes móviles, que ofrezcan las características de los agentes: representabilidad de un usuario, autonomía para tomar decisiones, y capacidad de responder a los cambios?". Así expuesto, el principal problema de esta investigación se reduce a dos subproblemas básicos: determinar cómo se puede llevar a cabo lo expuesto y establecer las condiciones para superar el principal problema de los agentes móviles: la seguridad de las transacciones.

# 1. PLANTEAMIENTO DEL PROYECTO

En este capítulo se hace la presentación del proyecto, se describe la oportunidad que el "Grupo en Ingeniería Telemática y Sistemas Inteligentes" GITSI de la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander empleó para dar origen al Diseño de una Arquitectura basada en Agentes Móviles para Comercio Electrónico (ARAMCEL). La presentación del proyecto, además de la descripción del problema, incluye los objetivos planteados y una breve descripción de la estructura de este documento.

## 1.1. DESCRIPCIÓN DEL PROBLEMA

Cada vez se puede encontrar un mayor número de personas que realizan sus compras u obtienen servicios a través de Internet. Los vendedores buscan captar nuevos clientes, contactar a los clientes antiguos y crear perfiles para ofrecer un servicio personalizado y eficiente, además los clientes demandan el acceso a bienes y servicios que se ajusten a sus necesidades reales. Esos vendedores y clientes son partes que se interrelacionan para llevar a cabo una transacción comercial; esas partes implicadas en una transacción pueden apoyarse en entidades que actúen en su nombre.

Esas entidades pueden ser personas o entidades software. Hay entidades software que tienen las funcionalidades para inspeccionar el medio ambiente donde se ejecutan, reaccionar a los eventos que se presentan, tomar decisiones y realizar acciones para alcanzar un objetivo. Una muestra de estas entidades son los agentes, éstos además de las funcionalidades anteriores se pueden comunicar con otros agentes para colaborar entre ellos y realizar una tarea encomendada.

En el caso de una transacción comercial, los agentes pueden aprovechar esas funcionalidades para representar a un usuario y realizar las labores que le han sido delegadas. Esas labores pueden incluir la selección de sitios donde buscar información sobre artículos, realizar comparación de resultados, ordenarlos y entregar un listado con los resultados más significativos. De la misma forma se le pueden asignar tareas más complejas, como es la compra de un artículo utilizando el número de tarjeta de crédito del usuario.

Esa búsqueda o compra de artículos implica una conexión permanente a Internet, esto conlleva un aumento en la congestión de la red por la cantidad de peticiones que hay que realizar para obtener un servicio que se encuentra en una máquina remota. Una solución para evitar el problema generado por la interacción de equipos en la red, es utilizar agentes con capacidad para desplazarse a un equipo remoto. Este es el caso de los agentes móviles, que poseen la funcionalidad para transportar su código a otro equipo a través de la red y continuar su ejecución allí; de esa manera pueden realizar operaciones directamente en la máquina donde se encuentra la información o los recursos que necesita.

La utilización de agentes, específicamente de los agentes móviles provee al usuario la posibilidad de disminuir las dificultades ocasionadas al tener que mantener su sesión en la red, revisando cada paso que hay que dar al usar los recursos que hay en las máquinas remotas. Entre esas dificultades está la mayor inversión de tiempo que se realiza cuando no se usan agentes, así como el costo generado por mantener equipos con conexión fija a la red.

En las transacciones realizadas en una actividad de comercio electrónico, la utilización de Agentes Móviles puede ayudar tanto a compradores como a vendedores; los agentes pueden visitar más lugares, emplear menor cantidad de tiempo y conseguir mayor información. Los Agentes Móviles pueden realizar más tareas y mostrar su eficacia, puesto que esa información la pueden entregar organizada con base en las indicaciones o preferencias dadas por el usuario.

Esa funcionalidad de los agentes móviles que les permite desplazarse a otro equipo en la red, representa una ventaja al transformar el acceso a recursos remotos en accesos locales. Este desplazamiento conlleva riesgos de seguridad implícitos en su modelo, puesto que al ser una aplicación en un sistema distribuido tiene que interactuar con diferentes ambientes y algunos de ellos pueden ser hostiles. La hostilidad puede estar dada por el mismo medio ambiente o por otros agentes, con lo cual puede ocurrir que la información en la red no permanezca privada, que un mensaje sea modificado mientras es transportado, que el autor de un documento o el remitente de un mensaje no sea quien dice ser, etc.

El trabajo de investigación que se presenta, se centra en el problema de realizar transacciones relacionadas con Comercio Electrónico mediante la aplicación del Modelo de los Agentes Móviles y cumpliendo con algunos de los aspectos más críticos en seguridad. En los siguientes capítulos se plantean los principales conceptos relacionados con el trabajo realizado.

## **1.2. OBJETIVOS**

El presente documento es el informe final de la Tesis de Maestría titulada “Diseño de una Arquitectura basada en Agentes Móviles para Comercio Electrónico”, cuyo plan de investigación fue presentado en junio de 2002 y aprobado en febrero de 2003. Los objetivos general y específicos que se plantearon son los siguientes:

### **1.2.1. Objetivo General**

El objetivo general del trabajo de investigación fue diseñar una arquitectura basada en el modelo de los Agentes Móviles que considere los principales problemas relacionados con la seguridad requerida en un contexto de Comercio Electrónico y así permita el desarrollo de aplicaciones para ese campo.

En el contexto del presente trabajo, una arquitectura software es aquella que permite:

- A unos componentes software ofrecer servicios y a otros componentes software hacer uso de esos servicios, facilitando por lo tanto el desarrollo de aplicaciones distribuidas.
- A los componentes software interactuar unos con otros sin importar los lenguajes, sistemas operativos y medios de comunicación usados para implementarlos.
- Suministrar reglas para definir nuevos componentes y específicamente como dichos componentes interactúan unos con otros [BASS03].

### **1.2.2. Objetivos Específicos.**

El objetivo general descrito genera los siguientes objetivos específicos:

- Seleccionar los criterios de diseño de la arquitectura con base en el estado de la infraestructura de llave pública (PKI) y los problemas de seguridad autenticación, autorización y no-repudio.
- Diseñar los componentes de la arquitectura, utilizando diagramas de casos y de secuencia de mensajes basados en la especificación UML (Lenguaje de Modelado Unificado).
- Realizar la validación del trabajo de investigación por medio de la implementación de un prototipo que muestre las principales características de la arquitectura. Dicha implementación se llevará a cabo por medio de proyectos de grado de estudiantes de la escuela de Ingeniería de Sistemas e Informática

### **1.3. ESTRUCTURA DEL DOCUMENTO**

A continuación se describe la organización de esta Tesis de Maestría. El documento está distribuido en nueve (9) capítulos de la siguiente manera:

El primer capítulo hace la introducción al trabajo realizado en la tesis, a través del planteamiento del proyecto; en él se realiza una breve descripción del problema y se presentan los objetivos que se plantearon en este trabajo de investigación.

El capítulo 2 introduce los conceptos generales del Modelo de Agentes Móviles, sobre el cual se explican las principales ventajas y desventajas de su utilización. Después se realiza un análisis comparativo sobre las principales plataformas de Agentes Móviles que se han desarrollado. Para finalizar el capítulo, se introduce el concepto de comercio electrónico (CE) y la forma como los Agentes Móviles pueden colaborar en la realización de transacciones de CE; además se hace el análisis de los principales sistemas para CE basados en Agentes Móviles.

El capítulo 3 incluye la descripción de los requerimientos de la Arquitectura. Para llegar a ellos se inicia con los principales puntos extraídos del análisis realizado a los sistemas basados en Agentes Móviles para comercio electrónico en el capítulo 2; con base en esa información y en el estudio del estado del arte se presentan los requerimientos desde el punto de vista de las transacciones de comercio electrónico, las necesidades de la Plataforma de Agentes Móviles que se seleccione y los requisitos de Seguridad para agentes y servidores.

Dentro del capítulo 4 se describe la arquitectura ARAMCEL propuesta en este trabajo. La primera parte ofrece la visión global del diseño y la descripción general de la arquitectura. Después se detallan los sistemas de comprador y vendedor; dentro de cada uno de estos sistemas se describen los componentes, el funcionamiento general del sistema y se incluye la explicación de los agentes estacionarios y móviles. Ese capítulo finaliza con las transacciones de comercio electrónico soportadas por ARAMCEL así como la función de cada uno de los agentes y componentes en ellas.

El capítulo 5 hace la descripción de la seguridad en ARAMCEL. Se inicia con la presentación de las amenazas de seguridad a los Agentes Móviles y los aspectos de seguridad necesarios en la realización de transacciones de comercio electrónico. Con base en estos problemas se presenta el modelo de seguridad de ARAMCEL, el cual

incluye la solución propuesta a las amenazas mencionadas con base en los aspectos de seguridad seleccionados; se finaliza con la descripción del protocolo de seguridad.

En el capítulo 6 se expone la implementación del prototipo Aplicación de los Agentes Móviles al Comercio Electrónico (ADAM), el cual fue usado para la validación de la arquitectura. La primera parte resume las tecnologías utilizadas en el desarrollo y la descripción general del prototipo; después se hace la descripción general de las principales características de ARAMCEL implementadas y no implementadas en ADAM; se finaliza con la evaluación del prototipo por medio de pruebas de validación, verificación y rendimiento.

Los últimos tres capítulos corresponden a las conclusiones del Trabajo de Investigación, las recomendaciones para trabajos futuros y las referencias bibliográficas, respectivamente.



## **2. ESTADO DEL ARTE Y LA TÉCNICA**

Este capítulo presenta el estado del arte y la técnica utilizados en el trabajo, con especial énfasis en el modelo de los agentes móviles. Se discuten los conceptos fundamentales de agentes, tipos de agentes, modelo de agentes móviles, ventajas, limitaciones, plataformas más importantes y la relación entre Transacciones de Comercio Electrónico y Agentes Móviles.

### **2.1. AGENTES MÓVILES**

Esta sección presenta el modelo de Agentes Móviles (AM), sus inicios, conceptos básicos, tipos de agentes, ventajas y limitaciones.

#### **2.1.1. De tipos de datos primitivos a agentes, pasando por los objetos.**

Uno de los principales avances de los lenguajes de programación de primera generación fue el uso de variables: dado que los métodos de programación anterior a dichos lenguajes sólo permitían almacenar información en registros específicos y predefinidos, el uso de variables nemotécnicas (que en realidad hacían referencia a direcciones de memoria) facilitó grandemente la labor de escribir los programas. El tipo de datos que dichas variables podían almacenar era predefinido y dependía del lenguaje de programación, así por ejemplo, las variables de tipo entero eran comunes a todos los lenguajes mientras que otros tipos de datos eran exclusivos de ciertos lenguajes, como el tipo de dato "real doble flotante" del lenguaje de programación Fortran IV.

Dado que un tipo de dato es una entidad software que permite almacenar las propiedades de una entidad real, con las variables y su tipo de dato asociado, un programador podía representar en un ambiente informático una simplificación funcional de las entidades de

un ambiente físico, exterior y real. Por ejemplo, una variable de tipo de dato "Alumno", contendrá la información funcional relacionada con un alumno desde la perspectiva de un sistema informático universitario: nombres, apellidos, código, fecha de nacimiento, etc. Otra información relevante del alumno en el mundo real como los nombres de sus padres, sus intereses o el nombre de su mascota, son dejados de lado en la abstracción funcional que un programador debe realizar al momento de diseñar y programar una aplicación informática.

El lenguaje Pascal introdujo un avance en los tipos de datos primitivos al permitir que el usuario definiera sus propios tipos de datos mediante una estructura que estaba conformada por tipos de datos básicos. El lenguaje de programación C fue más allá, al permitir que el usuario combinara diferentes tipos de datos originales y del usuario mediante la primitiva "*struct*". Estos avances conllevaron al importante concepto de "objeto" y a los consecuentes lenguajes de programación orientados a objetos.

Un "objeto" es una evolución de los tipos de datos, ya que permite almacenar en un ambiente informático tanto las propiedades como el comportamiento de una entidad real. La clase "Alumnos", por ejemplo, a la que pertenecen posibles objetos "alumnos", almacena la información relacionada con un alumno y además su "comportamiento", que abarca procesos como "Matricular" (proceso que incluye a un alumno en una asignatura), "CalcularPromedio" (proceso que calcula el promedio ponderado de las notas de un estudiante), etc. Los objetos son entonces entidades software más completas, puesto que contienen las propiedades y el comportamiento de entidades reales, pueden llevar a cabo procesos computacionales y tienen importantes características como la encapsulación y la herencia (que puede ser simple o múltiple).

El siguiente paso de esta lenta evolución lo constituyen los agentes. Un agente es una entidad software con las capacidades de los objetos (también contienen las propiedades y el comportamiento de entidades reales) pero, además de las características básicas de los objetos como herencia y encapsulación, poseen otras características importantes como autonomía, proactividad, sociabilidad y reactividad (las cuales son presentadas con mayor detalle en la siguiente sección).

Es necesario aclarar que el concepto de agentes se ubica en el punto de vista conceptual del diseño de un sistema informático y no desde el punto de vista de la implementación de dicho sistema: se desarrollan sistemas basados en agentes pero actualmente no hay lenguajes de programación basadas en agentes, ni bases de datos orientadas a agentes. Por facilidad de trabajo, los sistemas basados en agentes actualmente se desarrollan utilizando lenguajes de programación orientados a objetos (como es el caso en este trabajo), pero también podrían ser implementados, con un esfuerzo de programación excesivamente mayor, en lenguajes clásicos no orientados a objetos.

El concepto de agente es primordial para el diseñador de una aplicación informática, quien debe analizar el ambiente real y realizar una abstracción funcional para representarlo en un ambiente informático.

### **2.1.2. Conceptos básicos.**

El concepto de "agente" en el dominio de la informática surgió en el campo de la inteligencia artificial, a mediados de la década de los años noventa. La esencia del concepto de agente está relacionada con la propiedad que marca su nombre, la agencia. La "agencia" es la propiedad de representar a otra entidad: agente de policía, agente de ventas, agente 007, agente comprador, son entidades que respectivamente representan la autoridad civil del Estado, una empresa vendedora, su Majestad la Reina y un usuario que desea realizar unas compras. Para poder ejercer dicha representación, un agente debe poseer al menos un mínimo de autonomía, y así la "agencia" y la autonomía son las dos propiedades más importantes de un agente<sup>1</sup>.

El concepto de agente reúne aspectos de diversos campos como son la economía, la psicología, incluso la administración de empresas, y la misma inteligencia artificial. Dentro de dicho campo, un agente ha sido la evolución de conceptos como "fuente de conocimientos" (*knowledge source*), actores, demonios (*demons*), conceptos que se

---

<sup>1</sup> Para el presente trabajo se incluye la propiedad de "movilidad" como una de las características esenciales de los agentes.

remontan a la década de los años setenta. Una presentación completa de estos puntos se encuentra en [CASTILLO02].

Muchos autores han presentado diversas definiciones semi-formales del concepto de agente, cada una poniendo en relevancia la línea de trabajo del autor. Se tienen así definiciones de agente, que han sido formuladas con base en sus características, en sus requerimientos ó en la tecnología que utilizan [LANGE98]. A continuación se presentan algunas de las definiciones más apropiadas.

G.W. Lecky-Thompson [LECKY96] presenta una primera definición general: “Un agente es una pieza de software que realiza una tarea dada usando información obtenida de su entorno, que le permite actuar de forma apropiada y completar esa tarea exitosamente. El software debe ser capaz de adaptarse a sí mismo con base en los cambios que ocurren en su entorno, para que a pesar de esa variación obtenga el resultado esperado”.

J.A. King [KING95] ofrece una definición más orientada a la labor de los agentes como “representantes” de los humanos: “Un agente se considera como un sistema de computación que sustituye a una persona o proceso para realizar una actividad o cumplir con un requerimiento. La entidad sustituta ofrece capacidades de toma de decisiones similares a las descritas por las intenciones de un humano. Un agente inteligente puede operar dentro de los confines de una necesidad general o representada con precisión y entre los límites de un espacio de información dado.”

La primera definición resalta la perspectiva del software y está enfocada a la realización de las tareas; la segunda pertenece a la perspectiva del usuario final. Ambas han sido útiles para el presente trabajo, sin embargo la siguiente definición es tomada como la definición de referencia para el presente trabajo. Carlos Iglesias define un agente como una entidad software o programa informático cuyos rasgos principales son [IGLESIAS98]:

- Autonomía. Un agente es autónomo cuando opera sin la intervención directa de humanos ó de otros agentes; sus acciones y preferencias dependen de su propia experiencia y tiene cierto nivel de control sobre ellas y sobre su estado interno.

[WOOLDRIGE95]. La autonomía no es solo intuición, es el buen manejo de las percepciones, conocimientos y experiencia acumulada en la solución de problemas.

- **Sociabilidad.** Un agente debe estar en capacidad de interactuar con otros agentes, para eso utiliza un lenguaje de comunicación. Este rasgo es un aspecto importante en las comunidades o sociedades de agentes.
- **Reactividad.** Es la capacidad de adaptarse al ambiente, de percibir lo que sucede alrededor y responder a tiempo a los cambios que ocurren en él; un agente reactivo presenta comportamientos estímulo-respuesta.
- **Proactividad.** Es la capacidad de tener un comportamiento orientado por metas u objetivos; un agente proactivo toma la iniciativa (no espera a recibir una orden) para avanzar hacia el objetivo.

### **2.1.3. Clasificación de agentes**

Los agentes pueden clasificarse con base en sus características, en la tecnología que los soporta, en la función para la que fueron desarrollados, etc. En el contexto del presente trabajo se han clasificado según su movilidad. La movilidad es una propiedad ortogonal de los agentes, es decir, no todos los agentes son móviles.

- **Agentes estacionarios.** Un agente estacionario es aquel que reside en un sistema informático y permanece en él durante su ejecución. Su interacción con el mundo exterior la realiza por medio de intercambio de mensajes, de forma similar al software convencional. En esta investigación no se profundizará en este tipo de agentes.
- **Agentes Móviles.** Un agente móvil no está limitado al sistema en donde inició su ejecución; este tipo de agente se puede mover<sup>2</sup> por los nodos de una red para llevar a cabo sus tareas; cuando las finaliza regresa a su sitio de origen.

---

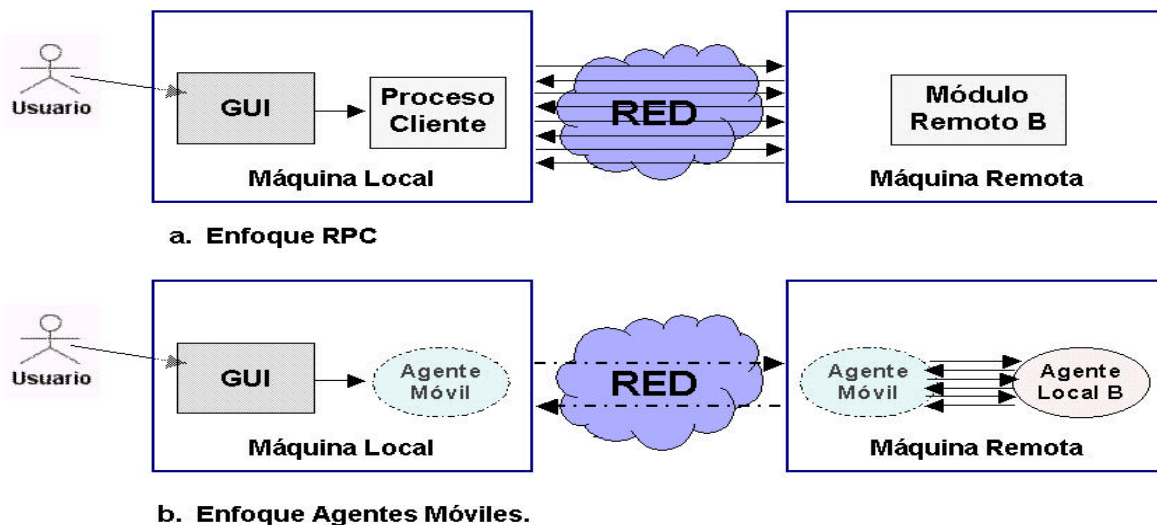
<sup>2</sup> Para llevar a cabo este desplazamiento, necesitan una plataforma de Agentes Móviles. En la sección 2.2 de este documento se presenta una descripción de las principales plataformas que se han desarrollado.

#### 2.1.4. Los inicios en el campo de los Agentes Móviles

En 1990 se funda la empresa General Magic, la cual en 1994 presenta la primera implementación de un sistema de Agentes Móviles que permite el acceso automático e interactivo a redes; en esa implementación se utiliza *Telescript* [GENERAL94]. *Telescript* es un lenguaje de programación para el desarrollo de aplicaciones distribuidas usando el paradigma de Agentes Móviles y fue diseñado para permitir el envío del código y del estado de un proceso a una máquina remota, esa era una función que ningún otro lenguaje de programación del momento proporcionaba. Ese primer sistema dio las bases y planteó los conceptos básicos del modelo de Agentes Móviles.

La Figura 1 presenta la interacción generada con el enfoque con RPC y el de Agentes Móviles. Los Agentes Móviles se representan con elipses, puesto que son más flexibles que los módulos tradicionales; el desplazamiento del agente se presenta por medio de líneas punteadas y la interacción con líneas continuas.

Figura 1. El modelo Cliente/Servidor y el modelo de los agentes móviles.



Fuente. El autor.

Otro avance en el surgimiento de los Agentes Móviles, fue dado por el reporte interno de IBM entregado en 1995 y titulado "*Mobile Agents: Are they a good idea?*" [CHESS97].

Ese reporte define los Agentes Móviles como "programas que generalmente son escritos en lenguaje *script*, los cuales pueden ser despachados desde una máquina cliente y transportados a un equipo remoto para continuar allá su ejecución"; además presenta un análisis crítico al modelo de Agentes Móviles y su aplicación específica en el campo de los dispositivos móviles, también resalta la facilidad que ofrece para la realización de búsquedas semánticas de información y las ventajas relacionadas con la interacción local en máquinas remotas.

### **2.1.5. El modelo de Agentes Móviles**

Los agentes son objetos activos que pueden correr concurrentemente y pueden tener su propio estado; cuando se comunican transfieren mensajes, esos mensajes son cualquier estructura de datos que puede ser pasada a través de un valor (parámetro). Una vez que un agente recibe un evento, ejecuta la reacción apropiada según la naturaleza del evento y de su estado actual. Un cambio en el estado del agente debido a un proceso o a una comunicación con otro agente genera una reacción. Enviar mensajes y reaccionar a los eventos es una manera para los agentes de comunicarse; además, los agentes no comparten su estado, lo que les hace entidades realmente autónomas.

Un sistema de agentes es modelado con agentes autónomos, cada uno es responsable de una conducta específica, organizado en una sociedad con reglas de conducta donde ellos tienen que comunicarse para lograr sus metas. Un solo modelo se define para todos los agentes autónomos, donde cada uno se comunica con los otros en la sociedad; pueden existir agentes básicos que sean responsables de tareas específicas. Esta descomposición del sistema en agentes con sus conductas es similar a la descomposición de tareas propuesta por [BROOKS91, según CASTILLO02], con eso se facilita la aplicación del modelo de agentes; además, la coordinación entre agentes se cumple por invocación remota de los módulos apropiados.

De tal forma el modelo de agentes se define como la colección de hechos que describen el conocimiento, racionalidad y estrategias del agente; así como los procedimientos de inferencia y heurística que permiten realizar predicciones sobre el agente que va a ser creado.

Un sistema basado en Agentes Móviles es una aplicación informática cuyo diseño e implementación se ha llevado a cabo siguiendo el modelo de los Agentes Móviles.

#### **2.1.6. Ventajas del Modelo de Agentes Móviles**

Como ya se ha mencionado un agente móvil tiene la habilidad de moverse desde un sistema a otro en la misma red. Esa capacidad le permite desplazarse a un sistema que contiene un objeto con el cual quiere interactuar y entonces toma ventaja de estar en el mismo servidor de la red que el objeto. Con base en esas características surgen al menos siete razones para utilizar el modelo de Agentes Móviles [LANGE99]:

- Los Agentes Móviles reducen la carga de la red.

Los sistemas distribuidos a menudo involucran múltiples interacciones para poder cumplir con una tarea, el resultado es gran cantidad de tráfico en la red. Con los Agentes Móviles los usuarios pueden empaquetar una conversación y despacharla a un destino, donde las interacciones se realizan localmente. No sólo pueden empaquetarse conversaciones, es posible desplazar cualquier tarea directamente a la máquina donde se encuentran los recursos, evitando tener que ocupar la red para establecer la transferencia de los datos que requieran esos procesos. La razón para el procesamiento de datos basado en Agentes Móviles es simple: mover el proceso computacional donde están los datos es mejor que mover los datos a la máquina donde está el proceso.

- Los Agentes Móviles superan los inconvenientes por la latencia de la red.

Los sistemas en tiempo real críticos, tales como sistemas de control en procesos de manufactura, necesitan responder en tiempo real a cambios en su medio ambiente. El controlar tales sistemas a través de una red involucra sustanciales retrasos. Por esta razón, en sistemas que requieran tiempo real crítico, tales latencias no son aceptables; igualmente si la red está congestionada los tiempos de respuesta se hacen más prolongados. Las entidades que por sus requerimientos no pueden perder tiempo, se benefician con el uso de agentes que realicen el trabajo y en el momento en que el usuario lo requiera entreguen los resultados obtenidos.

- Los Agentes Móviles se ejecutan de forma asincrónica<sup>3</sup> y autónoma.

En la actualidad hay trabajos que dependen de conexiones de red costosas o frágiles, por eso cuando las tareas que se llevan a cabo requieren continuamente abrir una conexión entre un dispositivo móvil y una red fija, esta situación probablemente no es factible económica ni técnicamente. Para solucionar ese problema, las tareas son empaquetadas en Agentes Móviles, los cuales se envían a la red, donde se encargan de continuar el trabajo; allí pueden actuar de manera autónoma y sin el inconveniente que se presenta con los mensajes de tipo sincrónico, donde el autor del mensaje detiene su hilo de ejecución y sólo continúa en el instante en que recibe la respuesta.

- Los Agentes Móviles encapsulan protocolos.

Cuando los datos son intercambiados en un sistema distribuido, cada servidor contiene el código que implementa los protocolos necesarios para una adecuada codificación de los datos que se envían y para interpretar apropiadamente los datos que llegan. Sin embargo, con el tiempo los protocolos son modificados para satisfacer las nuevas necesidades de eficiencia y seguridad; por esa razón, se presentan problemas debido a que los equipos en la red son heterogéneos (tienen diferencias en su Hardware y en su Software), lo cual no permite actualizar los protocolos apropiadamente. En ese sentido los agentes tienen la ventaja de poseer su propio protocolo de comunicaciones y por tanto son independientes de su entorno.

- Los Agentes Móviles se adaptan dinámicamente.

Los Agentes Móviles pueden examinar su ambiente de ejecución y reaccionar autónomamente a los cambios. El agente tiene la capacidad de detectar si el ambiente donde se encuentra cambia y de acuerdo con su grado de autonomía toma las decisiones necesarias para continuar con su trabajo de una manera eficaz. Además los Agentes Móviles tienen la capacidad única de distribuirse ellos mismos entre los servidores de la red para mantener la configuración óptima y así resolver un problema particular.

---

<sup>3</sup> Los agentes móviles son en su naturaleza asíncronos y por medio de esta característica se evita que el sistema que envía el agente se bloquee ó deba permanecer conectado a la red en espera de su respuesta.

- Son heterogéneos por naturaleza.

La computación en red es fundamentalmente heterogénea, tanto desde la perspectiva del hardware como del software. Debido a que los Agentes Móviles son por lo general independientes de la capa de transporte y de los computadores (son dependientes únicamente del medio ambiente de ejecución), proveen condiciones óptimas para la integración de sistemas sin que exista la necesidad de realizar adaptaciones.

- Los Agentes Móviles son robustos y tolerantes a fallos.

La capacidad de los Agentes Móviles para reaccionar dinámicamente a eventos y situaciones desfavorables los hace útiles para la implementación de sistemas de procesamiento distribuido fiable y tolerante a fallos. Cuando un agente va a regresar al servidor origen y encuentra que el usuario que lo creó finalizó su sesión en la red, puede almacenarse en memoria secundaria y después de cierto tiempo activarse nuevamente para regresar.

#### **2.1.7. Desventajas del Modelo de Agentes Móviles**

La seguridad ha sido tradicionalmente un inconveniente en los sistemas informáticos, pero el modelo de Agentes Móviles por sus características presenta un nuevo tipo de inconvenientes de seguridad, los cuales se convierten en su principal desventaja. Los dos problemas principales son [ZHAO00]:

- El problema del Agente Hostil.

Un agente hostil es un agente que realiza acciones perjudiciales, como el acceso no autorizado y alteración de los recursos locales (datos, llamadas al sistema), sobrecarga de los recursos del servidor o incluso daños a su integridad.

- El problema del Servidor Hostil.

Un servidor hostil es una máquina o servidor que intenta espiar ó manipular el código, los datos o el control de flujo del agente, proporciona llamadas falsas al sistema, ejecuta código del agente incorrectamente, o realiza ingeniería inversa y manipula su

código ó los secretos comerciales, ya sea mediante la clonación del agente, proporcionándole información errónea ó simplemente interceptándolo.

## 2.2. PLATAFORMAS DE AGENTES MÓVILES

Una de las causas que motivó el auge de la tecnología de Agentes Móviles fue la aparición del lenguaje JAVA [SUN04a], que dentro de sus librerías básicas incluye funcionalidades que permiten: transformar en un “archivo plano” una clase u objeto activo (operación conocida bajo el nombre de serialización), enviar ese archivo por la red (mediante RMI<sup>4</sup> – invocación de métodos remotos), recibirlo y volverlo clase u objeto activo, tal como era antes de viajar (operación conocida como des-serialización). Estas funcionalidades ofrecidas por JAVA para la programación distribuida, permitieron la creación de diversas plataformas para Agentes Móviles.

En los siguientes numerales se presentan algunas de las plataformas de Agentes Móviles que se encuentran actualmente y que son relevantes dentro del contexto del trabajo de investigación planteado. Las plataformas que se incluyen son: Aglets Workbench, JADE, Voyager y Odissey [HERRERA99, NGUYEN02].

Cada plataforma se evalúa con base en un conjunto de criterios que agrupan los principales requerimientos de una plataforma para Agentes Móviles. Los criterios mencionados son:

- Movilidad: Movilidad fuerte (migra tanto el código como el estado de ejecución) ó débil (migra sólo el código), eficiencia del método de migración, etc.
- Comunicación: Soporte para comunicación Inter – plataformas.
- Políticas de seguridad: Seguridad intra-plataforma e inter-plataforma.
- Disponibilidad y acceso.
- Documentación disponible y comunidad de usuarios.

---

<sup>4</sup> El sistema RMI (JAVA Remote Method Invocation) permite a un objeto que se está ejecutando en una Máquina Virtual JAVA (VM) llamar a métodos de otro objeto que está en otra VM, a su vez permite la comunicación remota entre programas escritos en JAVA. Para mayor información ver [SUN04b]

- Desarrollos. Aplicaciones ó proyectos que la utilizan.

### 2.2.1. Aglets Workbench de IBM.

El nombre Aglets surge de la unión de las palabras "**Agent**", que traducido del inglés es Agente y "**applets**" que significa pequeña aplicación, con esta combinación Aglets equivale a "pequeño Agente".

La herramienta de desarrollo de software Aglets (ASDK - *Aglets Software Development Kit*) es un entorno para programar Agentes Móviles en JAVA [LANGE98]. Los aglets son objetos JAVA que pueden moverse de un servidor a otro. De esa forma, un aglet que se ejecuta en una máquina puede detener su ejecución, trasladarse a una máquina remota y continuar su ejecución allí. Cuando un aglet viaja, lleva consigo tanto su código como sus datos.

Esa plataforma fue producida por IBM – Japón, y por tanto tiene el respaldo de esa compañía, aunque la empresa ha liberado el código y el proyecto se ha convertido en software de libre acceso. La última versión es la 2.0.2 [AGLETS04], la cual cuenta con una estructura clara y sencilla, de fácil acceso, buena documentación y alta aceptación por parte de los usuarios; Aglets también provee un servidor de agentes denominado Tahiti<sup>5</sup> y soporta JAVA 2.

La estructura de nombres de las clases y métodos de Aglets es bastante similar a las clases de JAVA que realizan funciones equivalentes, lo cual colabora en la programación y el aprendizaje en los usuarios que tienen experiencia en ese lenguaje, además, existe un libro que hace una introducción a la programación de agentes utilizando la API de aglets [LANGE98].

La evaluación de Aglets con base en los criterios mencionados al inicio de la sección es:

- Movilidad: Débil. Utiliza Serialización de JAVA (*ByteCode*).

---

<sup>5</sup> Tahiti es una aplicación Java que permite al usuario recibir, administrar y enviar aglets a otros computadores que también esten corriendo un Tahiti.

- Comunicación: Utiliza *Sockets*, permite el paso de Mensajes entre agentes y usa el protocolo de transferencia de agentes ATP (*Agent Transfer Protocol*), el cual soporta *HTTP tunneling*. Soporta mensajes<sup>6</sup> síncronos, asíncronos y *multicast*.
- Políticas de seguridad: La seguridad es controlada a través de Tahiti. Define tres roles (aglet, creador y propietario). Ofrece seguridad al contexto y al servidor. Los agentes son protegidos usando un objeto *proxy*. Tiene seguridad estándar a través del *JDK Keytool*.
- Disponibilidad y acceso. Su código fuente es de libre acceso y puede ser descargado del portal de Aglets [AGLETS04].
- Documentación disponible y comunidad de usuarios. Documentación variada, se dispone de un libro y es una de las plataformas de mayor aceptación en el medio de los Agentes Móviles.
- Desarrollos. Ha sido ampliamente usada en el mundo. La principal aplicación es TabiCan<sup>7</sup>, un mercado electrónico para tiquetes aéreos y paquetes de viaje en Japón, el cual ha sido diseñado para soportar miles de agentes.

### 2.2.2. Java Agent DEvelopment Framework (JADE).

JADE [TILAB04] es una plataforma de desarrollo de software orientada al desarrollo de aplicaciones de sistemas multi-agente, conforme al estándar de la Fundación para los Agentes Físicos Inteligentes (FIPA - Foundation for Intelligent Physical Agents). Su meta es simplificar el desarrollo mientras se asegura cumplir ese estándar, a través de un variado conjunto de agentes y servicios del sistema.

---

<sup>6</sup> La definición de estos tipos de mensajes es [LANGE98]:

- Síncrono: Bloquea la ejecución del agente hasta que se haya recibido una respuesta.
- Asíncrono: Permite al agente continuar su ejecución aunque no haya recibido respuesta.
- Multicast: Es un mensaje que el agente envía a todos los agentes en la plataforma.

<sup>7</sup> La mayor parte de la documentación e información sobre TabiCan se encuentra en japonés, sin embargo, una explicación de él puede ser encontrada en [LANGE98] ó en la siguiente página web: [http://domino.research.ibm.com/comm/wwwr\\_thinkresearch.nsf/pages/travel199.html](http://domino.research.ibm.com/comm/wwwr_thinkresearch.nsf/pages/travel199.html), perteneciente al equipo de investigación de IBM.

JADE puede ser distribuida sobre varias máquinas. El sistema de agentes tiene un “contenedor principal” que está compuesto por el registro RMI y por los agentes facilitador de directorio y administrador del sistema. Otros contenedores de agentes pueden ser lanzados en la misma máquina en donde se encuentra el sistema de agentes ó en máquinas remotas (un contenedor por máquina), cada uno de esos contenedores se conecta con el contenedor principal de la plataforma de Agentes, con eso se obtiene un sistema distribuido sobre varias máquinas que es una única plataforma de agentes. Los agentes pueden desplazarse entre los diferentes contenedores y por lo tanto entre máquinas remotas, pero tiene una limitación, la máquina en la que está ubicado el contenedor principal debe permanecer todo el tiempo conectada, si no es así, el sistema colapsa. La implementación de JADE es software libre, distribuida por TILab<sup>8</sup>, tiene una interfaz de usuario, buena documentación y alta aceptación.

A continuación se incluye el análisis de JADE con base en los criterios especificados:

- Movilidad: Movilidad Débil. No permite movilidad Inter-plataformas, si la plataforma se encuentra distribuida sobre diversas máquinas, es necesario que aquella en donde se encuentra el contenedor principal esté siempre conectada, de lo contrario, el sistema no funciona.
- Comunicación: Permite los mensajes Inter-plataforma y utiliza para mensajes el lenguaje de comunicación de agentes ACL (*Agent Communication Language*) y XML (*Extensible Markup Language*).
- Políticas de seguridad: Provee servicio de autenticación de conexión, validación de usuario y cifrado de mensajes RPC. El *proxy* del agente actúa como una compuerta bidireccional entre plataformas JADE y una conexión TCP/IP ordinaria.
- Disponibilidad y acceso. JADE es software libre, distribuido por TILab.
- Documentación disponible y comunidad de usuarios. Existe documentación que puede ser descargada del sitio web del proyecto y una lista de discusión en la que los usuarios pueden realizar sus preguntas. En estos momentos hay universidades y equipos de investigación en diferentes países que trabajan para incluir nuevas

---

<sup>8</sup> TILab es el Laboratorio de investigación de Telecom Italia en donde se ha desarrollado la plataforma JADE. Para mayor información consúltese [TILAB04].

características como seguridad, movilidad, etc; otros evalúan su aplicación en diferentes entornos.

### **2.2.3. Voyager de ObjectSpace.**

Voyager surgió como parte de la línea de productos de ObjectSpace para el desarrollo de sistemas distribuidos, ha sido implementada en el lenguaje de programación JAVA [GLASS99]. Voyager permite usar la sintaxis del mensaje para construir objetos remotos, envía los mensajes, y los mueve entre los programas. No es necesario ejecutar el servidor de agentes Voyager en todos los nodos de la red, debido a que los objetos virtuales pueden migrar no sólo entre agentes sino que también hacia los entornos de ejecución de JAVA de otros objetos virtuales.

Este producto incluye información sobre las características del núcleo y la arquitectura de la plataforma, incluso un Intercambiador de Solicitudes de Objetos (*Object Request Broker* – ORB) con apoyo para los objetos móviles y los agentes autónomos. En el núcleo del paquete hay servicios para la comunicación escalable de grupos, persistencia y servicios básicos de directorio.

Voyager se convirtió en una de las plataformas de Agentes Móviles más usada, debido a las ventajas que ofrece, por ejemplo, permite la creación remota de objetos y agentes. Además, el sistema tiene la capacidad de mover objetos, lo que sumado a que la documentación que se encuentra es extensa y relativamente completa, lo convierte en un sistema con grandes posibilidades.

En resumen, se puede afirmar que Voyager es una plataforma de Agentes Móviles genérica y robusta, es un sistema sólido, pero por ser una plataforma cerrada un usuario no puede utilizarla fácilmente para enriquecer su funcionalidad básica. Asimismo, las últimas versiones han dejado de ser libres y son costosas<sup>9</sup>, lo cual ha ocasionado dificultades al emplearla para fines académicos.

---

<sup>9</sup> Para mayor información, remitirse a <http://www.recursionsw.com/>, compañía que compró "Voyager" a ObjectSpace.

A continuación se incluye el resumen de la descripción de la plataforma:

- Movilidad: Migración Débil.
- Comunicación: Soporta mensajes síncronos, asíncronos y *multicast*.
- Políticas de seguridad: Divide los agentes en dos grupos: Nativos con acceso total y Extranjeros con restricciones de acceso. Es posible reemplazar el protocolo TCP utilizado para el transporte con SSL.
- Disponibilidad y acceso. Es un sistema cerrado, por lo tanto es difícil enriquecer su funcionalidad. Las últimas versiones no son de libre distribución.
- Documentación disponible y comunidad de usuarios. En sus primeras versiones fue una plataforma muy empleada. Documentación variada.

#### **2.2.4. Odyssey de General Magic.**

Como se mencionó, General Magic es una de las compañías más importantes en el dominio de los Agentes Móviles, pues además de producir *Telescript* patentó el término "Agente Móvil". Con la experiencia que la compañía obtuvo durante su trabajo con *Telescript*, creó Odyssey, un sistema de Agentes Móviles basado en JAVA [NGUYEN02]. Odyssey presenta facilidades para el diseño e implementación de programación distribuida a través de AM, puesto que los agentes que se crean en la plataforma semejan los objetos del mundo real y sus interacciones; además es posible adaptar la plataforma a las necesidades particulares de la aplicación mediante la modificación de las interfaces *AgentSystem* y *Transport*.

El paquete Odyssey es uno de los más compactos paquetes de sistemas de Agentes Móviles e incluye el código fuente de algunas de las principales clases del sistema, infortunadamente no suministra todas las funcionalidades de *Telescript*. La documentación del sistema es bastante deficiente, aunque el paquete suministra la descripción de la API, es muy superficial. Con base en los criterios mencionados puede resumirse el análisis de Odyssey así:

- Movilidad: Migración Débil.
- Comunicación: Mensajes orientados a la sesión síncronos ó asíncronos. Diversos protocolos para la transferencia de agentes: RMI, CORBA, IIOP.

- Políticas de seguridad: Controla el acceso a los recursos y realiza autenticación.
- Disponibilidad y acceso. Incluye código fuente de las clases principales.
- Documentación disponible y comunidad de usuarios. Documentación deficiente y superficial. Comunidad de usuarios escasa.

## **2.3. AGENTES MÓVILES Y COMERCIO ELECTRÓNICO**

Uno de los campos de aplicación de los Agentes Móviles es el comercio electrónico, eso es explicable pues en una transacción comercial se requiere la intervención de al menos dos partes que negocien o actúen con base en sus exigencias, necesidades, intenciones, gustos, etc. Un agente puede representar a un usuario dentro de una transacción comercial pues cuenta con características que lo hacen autónomo, reactivo y proactivo, así mismo en algunos casos para la realización de transacciones se necesita acceder en tiempo real a recursos que se hallan distribuidos en la red, de esa forma la movilidad de los agentes da una fortaleza en ese entorno.

Los siguientes numerales amplían el concepto de Comercio Electrónico, su definición y los principales tipos de transacciones existentes; después se presenta la aplicación de los Agentes Móviles al comercio electrónico y se mencionan algunos de los sistemas que existen actualmente.

### **2.3.1. Comercio Electrónico.**

Existen muchas definiciones de Comercio Electrónico, una de ellas es formulada por [PRESIDENCIA00], en ella se plantea que el Comercio Electrónico (CE)<sup>10</sup> es una forma de hacer negocios en la economía electrónica; donde existe un conjunto de aplicaciones y procesos de negocios que vinculan organizaciones, clientes, proveedores y comunidades a través de transacciones electrónicas e intercambio electrónico de información, productos y servicios mediante tecnologías como Internet, que lo hacen posible.

---

<sup>10</sup> En Inglés *Electronic Commerce, e-commerce*.

Los orígenes del comercio electrónico están marcados por la aparición de las transferencias interbancarias electrónicas de fondos (*Electronic Funds Transfer – EFT*) en la década de los años setenta y el surgimiento en los años ochenta del sistema para intercambio electrónico de datos (*Electronic Data Interchange – EDI*); EDI apoyado en el correo electrónico permite el comercio entre empresas [CLARKE98, KALAKOTA97].

Esas tecnologías colaboran en la reducción del intercambio de papeles y en el incremento de la automatización de las oficinas, con lo cual se obtiene una mejora en los procesos de negocios. La gran desventaja fue el alto costo de implantación, operación y mantenimiento que generaba para las empresas, por esa razón, la mayor parte de las pequeñas y medianas empresas (PYMES) quedaron fuera del comercio con las de mayor capital [PULIDO99].

Pero el auge de Internet en los años noventa dio una solución tecnológica de fácil acceso al problema de publicar, administrar y distribuir información y conocimiento, lo que permitió a las PYMES ingresar al intercambio electrónico y negociar con grandes empresas de diferentes países [COMISION99]. Es a partir de ese punto que el comercio electrónico empieza a utilizarse entre empresas y consumidores.

Existen varios tipos de CE, los cuales se clasifican con base en los actores que participan en las transacciones. Se han establecido tres categorías principales [BERGER00, LFCIA03]:

- **B2B (*Business to Business – Empresa a Empresa*):** En este tipo de comercio electrónico los actores de las transacciones son las empresas, las cuales utilizan tecnologías como EDI para la transferencia de datos y archivos. Este tipo de transacciones han sido las de mayor auge en los últimos treinta (30) años.
- **B2C (*Business to Consumer – Empresa a Consumidor*):** En B2C, la relación comercial se da entre empresas y consumidores finales. Aunque el movimiento de dinero que genera es inferior al del B2B, es el más conocido popularmente y por lo tanto un pilar importante en la evolución del comercio electrónico.

- **C2C (Consumer to Consumer – Consumidor a Consumidor):** En este tipo de comercio electrónico, la relación comercial se da entre particulares y es conocida popularmente como “Subastas *On-line*”.

### 2.3.2. Transacciones de comercio electrónico basadas en Agentes Móviles.

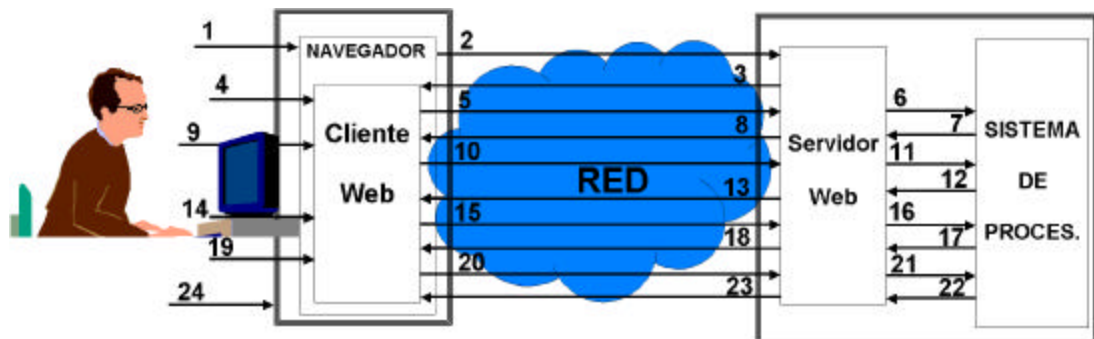
Antes de pasar a la realización de transacciones basadas en agentes, es necesario observar el proceso con base en los diferentes modelos.

#### 2.3.2.1 Modelo Cliente / Servidor.

Por ejemplo, si un usuario desea comprar un pasaje de avión, utilizando Cliente / Servidor debe conectarse a la página de la aerolínea, ingresar la fecha, número de pasajes, ciudades origen – destino y luego esperar los resultados de los posibles vuelos, cuando los obtenga, debe revisarlos y seleccionar el que más le convenga.

En este modelo el Usuario tiene que permanecer con su sesión en la red interactuando durante toda la transacción, esto implica que no puede irse sin haber finalizado el proceso. Por lo tanto, si la disponibilidad de tiempo del Usuario es insuficiente para revisar todos los resultados obtenidos, es posible que no encuentre la opción que mejor se ajuste a sus necesidades; además las interacciones en este proceso indican un elevado tráfico en la red. El proceso se presenta con mayor detalle en la Figura 2.

Figura 2. Ejemplo del Modelo Cliente Servidor.



Fuente. El autor.

1. El Usuario digita el nombre de la página de la aerolínea en el navegador.

2. A través del navegador se solicita la página de la aerolínea.
3. El Servidor Web de la aerolínea recibe la solicitud y envía los datos de la página.
4. El Usuario procede a ingresar los datos básicos de consulta, fecha de viaje, número de pasajes, clase, ciudad de origen, ciudad de destino.
5. El Cliente Web envía los datos al Servidor Web.
6. El Servidor Web entrega los datos al Sistema de Procesamiento, para que realice la consulta respectiva.
7. El Sistema de Procesamiento realiza la consulta y devuelve los resultados al Servidor Web.
8. El Servidor Web envía los resultados al Cliente Web.
9. El Usuario revisa los resultados que le presenta el Cliente Web y escoge aquellos que considera apropiados por las preferencias de horario que tiene o solicita datos de viaje con trasbordo en otra ciudad, para revisar con mayor detalle las nuevas posibilidades.
10. El Cliente Web solicita información específica o detallada.
11. El Servidor Web entrega los datos al Sistema de Procesamiento, para que realice la consulta detallada.
12. El Sistema de Procesamiento realiza la consulta y devuelve los resultados al Servidor Web.
13. El Servidor Web envía los resultados al Cliente Web.
14. En caso de estar de acuerdo con los resultados, por ajustarse a sus necesidades, el Usuario procede a aceptar y solicita iniciar el proceso de compra. Si no está de acuerdo con los resultados finaliza el proceso y vuelve a empezar con otra aerolínea.
15. El Cliente Web solicita los datos para iniciar la compra
16. El Servidor Web entrega al Sistema de Procesamiento la petición realizada por el Usuario.
17. El Sistema de Procesamiento solicita los datos de pago y entrega; Esos datos corresponden a nombre, número de identificación, número de tarjeta de crédito, dirección de correspondencia, etc.
18. El Servidor Web entrega al Cliente Web, la solicitud de datos de pago y entrega.
19. El Usuario revisa los datos que le solicitan e ingresa la información pertinente.
20. El Cliente Web envía los datos.
21. El Servidor Web entrega los datos al Sistema de Procesamiento.

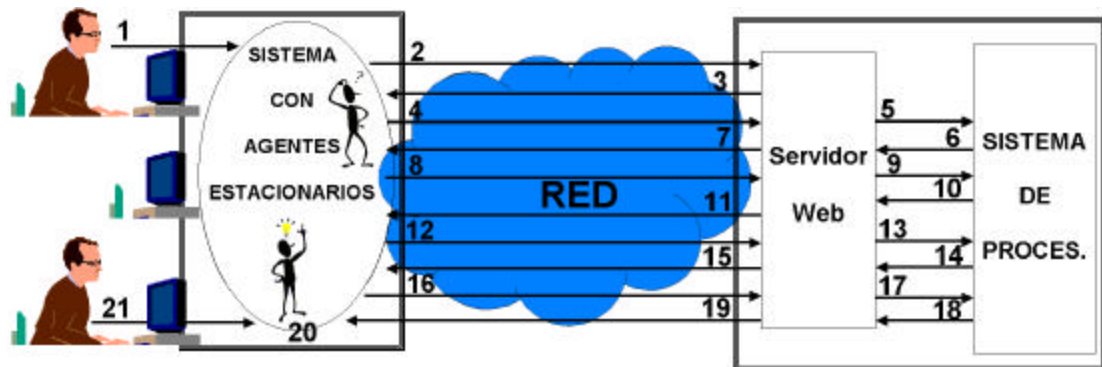
22. El Sistema de Procesamiento revisa los datos y genera una confirmación de compra, que entrega al Servidor Web.
23. El Servidor Web envía la confirmación de compra.
24. El Usuario revisa la confirmación de compra y la guarda.

### 2.3.2.2 Modelo de Agentes Estacionarios.

Con el modelo Cliente / Servidor el Usuario realiza parte del proceso con cada aerolínea hasta encontrar una con la cual efectuar el proceso completo; por esta razón surge como alternativa la utilización de Agentes Estacionarios. Con estos agentes, el Usuario ingresa la fecha, número de pasajes, ciudades origen – destino y sus preferencias en cuanto a horarios, precios, clase, etc. El agente se conecta a través de Internet con los sitios de varias aerolíneas, consulta los horarios de vuelo y precios, encuentra el mejor precio y el horario de salida más adecuado para su usuario con base en la información suministrada, realiza la reserva y dependiendo de la autonomía dada, efectúa el pago con el número de la tarjeta de crédito del Usuario.

En este modelo el Usuario mantiene su sesión en la red, pero sólo se requiere su presencia al inicio para dar la información pertinente al agente, una vez que esto ocurra puede irse sin haber finalizado el proceso. Cuando decida regresar puede revisar los resultados que obtuvo su agente; similar al modelo anterior, las interacciones en este proceso indican un elevado tráfico en la red. El proceso se presenta con mayor detalle en la Figura 3.

Figura 3. Ejemplo del Modelo de Agentes estacionarios.



Fuente. El autor.

1. El Usuario le entrega los datos al agente. Los datos corresponden a fecha de viaje, número de pasajes, clase, ciudad de origen, ciudad de destino, horarios preferibles de viaje y precio.
2. Un agente se conecta al Servidor Web de la aerolínea.
3. El Servidor web de la aerolínea recibe el enlace y presenta las opciones de consulta.
4. El agente procede a ingresar los datos básicos de consulta y los envía al Servidor Web.
5. El Servidor Web entrega los datos al Sistema de Procesamiento, para que realice la consulta respectiva.
6. El Sistema de Procesamiento realiza la consulta y devuelve los resultados al Servidor Web
7. El Servidor Web envía los resultados al agente.
8. El agente revisa los resultados que le presenta el Servidor Web y escoge aquellos que considera apropiados por las preferencias de horario que tiene o solicita nuevos datos de viaje con trasbordo en otra ciudad, para optimizar la consulta. El agente procede a realizar la nueva consulta.
9. El Servidor Web entrega los datos al Sistema de Procesamiento, para que realice la consulta detallada.
10. El Sistema de Procesamiento realiza la consulta y devuelve los resultados al Servidor Web
11. El Servidor Web envía los resultados al agente.
12. Si hay más aerolíneas se repite el proceso con cada una, para obtener el resultado que mejor se ajuste al Usuario. En caso de estar de acuerdo con los resultados, el agente procede a aceptar y solicita el inicio del proceso de compra.
13. El Servidor Web entrega al Sistema de Procesamiento la petición realizada por el agente.
14. El Sistema de Procesamiento solicita los datos de pago y entrega; Esos datos corresponden a nombre, número de identificación, número de tarjeta de crédito, dirección de correspondencia, etc.
15. El Servidor Web entrega al agente, la solicitud de datos de pago y entrega.
16. El agente revisa la solicitud que le hacen e ingresa la información concerniente.
17. El Servidor Web entrega los datos al Sistema de Procesamiento.
18. El Sistema de Procesamiento revisa los datos y genera una confirmación de compra.

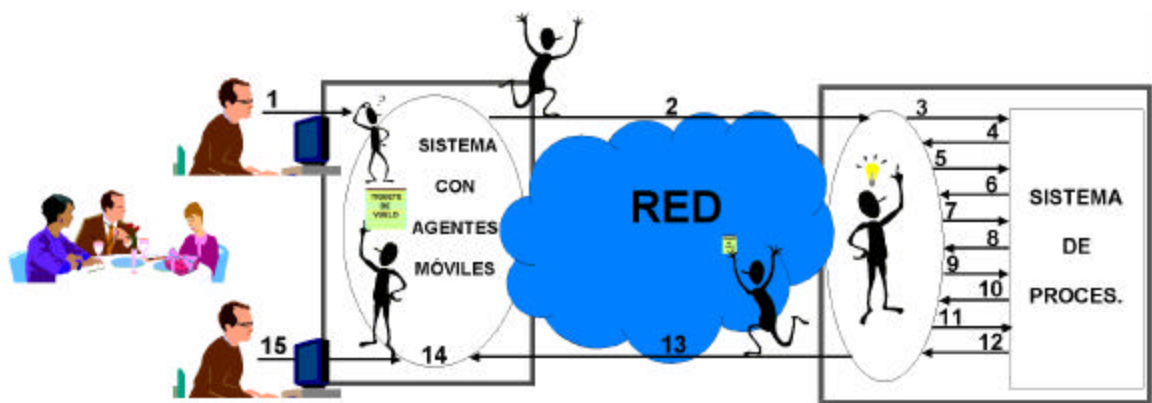
19. El Servidor Web envía la confirmación de compra
20. El agente revisa la confirmación de compra y guarda.
21. El Usuario revisa los resultados.

### 2.3.2.3 Modelo de Agentes Móviles.

Con la utilización de Agentes estacionarios se obtiene una sustancial mejora para el usuario puesto que ya no tiene que invertir tiempo revisando por si mismo cada resultado; no obstante para que los agentes estacionarios puedan realizar su trabajo, es necesario que la sesión del Usuario permanezca activa en la red.

Para disminuir los inconvenientes de los dos modelos anteriores, se ha propuesto la utilización del modelo de Agentes Móviles. Con base en este modelo el usuario entrega sus preferencias al agente, el cual visita los sitios de varias aerolíneas, consulta los horarios de vuelo y precios, encuentra el mejor precio y el horario de salida más adecuado para su usuario; el agente también puede reservar o pagar el pasaje. Con este modelo el tráfico en la red se ve disminuido, puesto que las interacciones generadas por las múltiples peticiones y respuestas relacionadas con el servidor se realizan de forma local, entre el agente y el servidor. Además presenta como ventaja para el usuario, la posibilidad de poder finalizar su sesión en la red y dedicarse a otras actividades; cuando el usuario decida regresar, activa su sesión para que el agente regrese y puede entregarle los resultados que obtuvo. El proceso se muestra con mayor detalle en la Figura 4.

Figura 4. Ejemplo del Modelo de Agentes Móviles.



Fuente. El autor.

1. El Usuario le entrega los datos al agente. Los datos corresponden a fecha de viaje, número de pasajes, clase, ciudad de origen, ciudad de destino, horarios preferibles de viaje y precio.
2. Un agente viaja a una aerolínea.
3. El agente solicita acceso a realizar consultas en la aerolínea.
4. El Sistema de Procesamiento presenta las opciones de consulta al agente.
5. El agente procede a ingresar los datos básicos de consulta al Sistema de Procesamiento de la aerolínea.
6. El Sistema de Procesamiento realiza la consulta y devuelve los resultados al agente.
7. El agente revisa los resultados que le presentan y escoge aquellos que considera apropiados por las preferencias de horario que tiene o solicita nuevos datos de viaje, para optimizar la consulta. El agente procede a realizar la nueva consulta.
8. El Sistema de Procesamiento realiza la consulta y devuelve los resultados al agente.
9. Si los resultados no satisfacen las preferencias del Usuario, el agente viaja a otra aerolínea a realizar el mismo proceso. En caso de estar de acuerdo con los resultados, el agente procede a aceptar y solicita el inicio del proceso de compra.
10. El Sistema de Procesamiento solicita los datos de pago y entrega al agente. Esos datos corresponden a nombre, número de identificación, número de tarjeta de crédito, dirección de correspondencia, etc.
11. El agente revisa la solicitud que le hacen e ingresa la información correspondiente.
12. El Sistema de Procesamiento revisa los datos y genera una confirmación de compra, que entrega al agente.
13. El agente revisa la confirmación de compra y viaja a su sitio de origen.
14. El agente en el sitio de origen guarda la confirmación en el sistema del Usuario.
15. El Usuario revisa los resultados.

Las ventajas de los agentes móviles son de gran utilidad en un entorno de comercio electrónico, por esa razón los sistemas basados en AM para CE han aumentado rápidamente y las empresas se han interesado en utilizar este tipo de aplicaciones.

### 2.3.3. Sistemas basados en Agentes Móviles para Transacciones de CE.

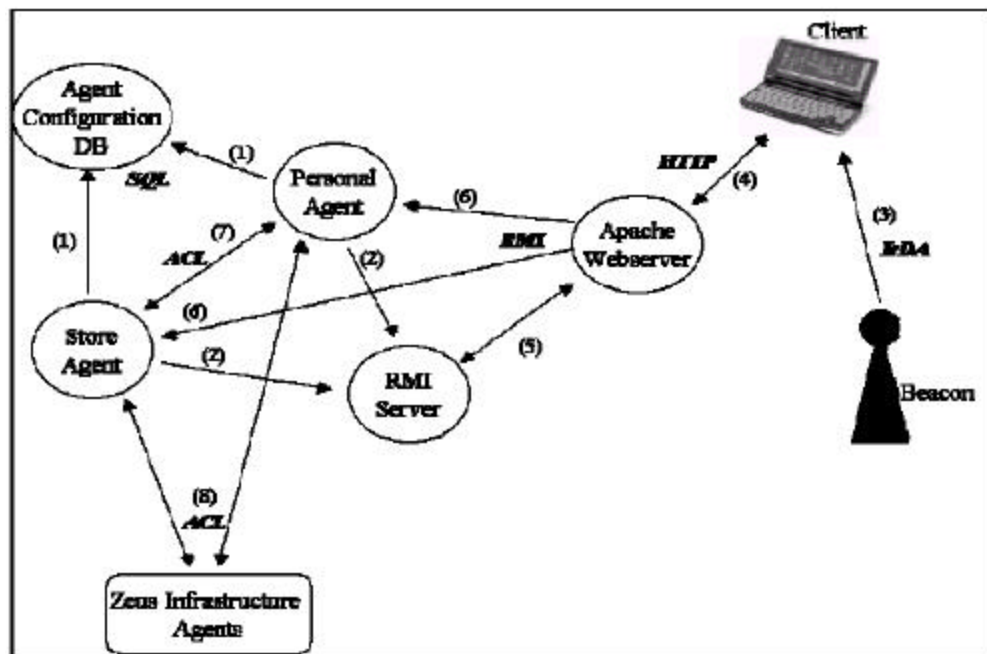
Actualmente existen sistemas basados en Agentes Móviles para el comercio electrónico, algunos de ellos son:

#### 2.3.3.1 Agora Shopping Mall.

El sistema Agora [FONSECA01] fue construido en los laboratorios de Hewlett Packard en Palo Alto, con el objetivo de estudiar los agentes, sus aplicaciones y la relación entre el comercio electrónico y los asistentes digitales personales.

Infraestructura de Agora. La infraestructura de Agora comprende cuatro subsistemas que incluyen un Servidor Web Apache, servidor RMI, la base de datos SQL (MySQL) y la plataforma de Agentes Zeus. La interacción entre esos subsistemas se presenta en la Figura 5.

Figura 5. Infraestructura y líneas de comunicación de Agora.



Fuente. [FONSECA01].

Cuando un agente es instanciado consulta en la base de datos (1) la información necesaria para realizar su trabajo; para el agente personal (PA), la base de datos contiene

la lista de productos que debe comprar y el agente vendedor (SA) busca el inventario de la tienda que representa. Además de consultar en la base de datos, los agentes deben registrarse (2) con el Servidor RMI; esto le permite al Servidor Web comunicarse (6) con los agentes ejecutando remotamente sus métodos.

Después que todos los agentes están ejecutándose, inicia el escenario de *Shopping Mall*. Los dispositivos de transmisión (3) usando comunicación a través de Infrarrojos codifican en formato XML las URLs. El Jornada ejecuta *e-squirt*, una aplicación que recibe las transmisiones del dispositivo y crea un enlace a la URL. Cuando el comprador hace clic en ese enlace, el cliente Web que se ejecuta en el PDA hace una petición HTTP (4) al servidor Web que usa su tarjeta de red PCMCIA inalámbrica. Tomcat extiende el servidor y permite mostrar el contenido de Web dinámico usando la tecnología Java Server Page (JSP).

El Servidor Web debe consultar (5) el servidor RMI para obtener las referencias a los agentes. Después de obtener estas referencias, el Servidor Web puede ejecutar (6) los métodos proporcionados por los agentes. Una vez los agentes han recibido la tarea que deben realizar, se comunican (7) usando un mensaje ACL. Los agentes usan el mismo tipo de mensajes para interactuar (8) con la plataforma de agentes Zeus.

Después de conocer la infraestructura de Agora, se incluye la descripción del funcionamiento del sistema.

Funcionamiento: Cuando el usuario se conecta a Internet, establece comunicación con su Agente Personal (PA), quien trae una lista de compras, la cual se muestra en el PDA del usuario; de la lista, el usuario selecciona el artículo que le interesa. Una vez se ha determinado el producto, el PA se comunica con un agente llamado facilitador de directorio (DF), quien le entrega la lista de las tiendas que distribuyen ese producto, este listado es mostrado nuevamente en el PDA.

Con base en esta nueva información, el usuario debe seleccionar el sitio en que desea comprar, para luego establecer contacto con el Agente Vendedor (SA). El SA le entrega

la información del producto al comprador y si está interesado establece los términos de compra. Después el PA del usuario se dirige a la subasta electrónica y se registra.

La participación del PA en la subasta, se realiza sin estrategia alguna, tan solo incrementa levemente el precio en cada oferta. Finalizada la subasta el Administrador informa quien ganó. Si el PA del usuario ganó, procede a efectuar el pago al Administrador, este último informa al SA de la venta. Con esa información se transfiere el producto al comprador y el dinero al vendedor.

Análisis Crítico: Con base en el estudio realizado de Agora, puede concluirse:

- El principal punto a favor de Agora es el soporte para dispositivos móviles.
  
- Muchas de las ventajas de los Agentes Móviles son ignoradas en el proyecto, puesto que los agentes no son totalmente autónomos y se exige la presencia del comprador en casi todo momento. Además, no se consigue la reducción esperada en la carga de la red, puesto que los listados de productos y tiendas son totalmente descargados en el PDA.
  
- Otro punto importante de señalar corresponde al itinerario, el agente no visita varias tiendas realizando búsquedas comparativas, por el contrario, sólo lleva el listado de ellas para que el usuario seleccione una.

### **2.3.3.2 MAgNET: A Mobile Agent E-commerce System**

MAgNET [DASGUPTA01] (*Mobile Agents for Networked Electronic Trading*) es un sistema basado en Agentes Móviles desarrollado en el lenguaje de programación Java y la plataforma de Agentes Móviles Aglets, propuesto como Tesis Doctoral en la Universidad de California en los Estados Unidos. MAgNET permite a los usuarios la compra comparativa de artículos ofrecidos por diferentes vendedores en la red. En MAgNET, un humano comprador crea un aglet móvil que compara cotizaciones de un artículo en los sitios de diferentes vendedores y retorna con la mejor oferta.

Funcionamiento: La operación de MAgNET puede dividirse en las siguientes fases:

1. Un comprador ingresa las preferencias en diferentes atributos incluyendo precio, tiempo de entrega y cantidad para cada ítem que desea comprar, a través de la GUI en su computador. Un agente estacionario toma la información de la GUI, actualiza la información en la base de datos del comprador y selecciona el listado de vendedores por cada producto solicitado. Para cada producto se construye un agente móvil comprador, el cual lleva la información del producto y la lista de vendedores que lo distribuyen.
2. El agente móvil comprador visita cada sitio de su itinerario<sup>11</sup>. En cada sitio interactúa con un agente estacionario que provee el servicio de búsqueda en la base de datos. Este agente entrega los datos del producto al agente móvil, quien compara esta información con las preferencias entregadas por el usuario ó con la mejor oferta recibida, si ya ha visitado otros sitios. Si la oferta actual es mejor que la anterior, la reemplaza, envía un mensaje de cancelación de la orden anterior y reserva el producto. Para mantener la reserva crea un agente en el sitio del vendedor.
3. El agente móvil comprador regresa al final de su itinerario al sitio del comprador y suministra la dirección del vendedor que entregó la mejor oferta, el humano comprador puede contactar directamente al vendedor ó cancelar la transacción.

Seguridad: MAgNET incluye un protocolo de seguridad utilizando la Infraestructura de Llave Pública (PKI – *Public Key Infrastructure*). Dicho protocolo se muestra en la Figura 6 y consta de los siguientes pasos:

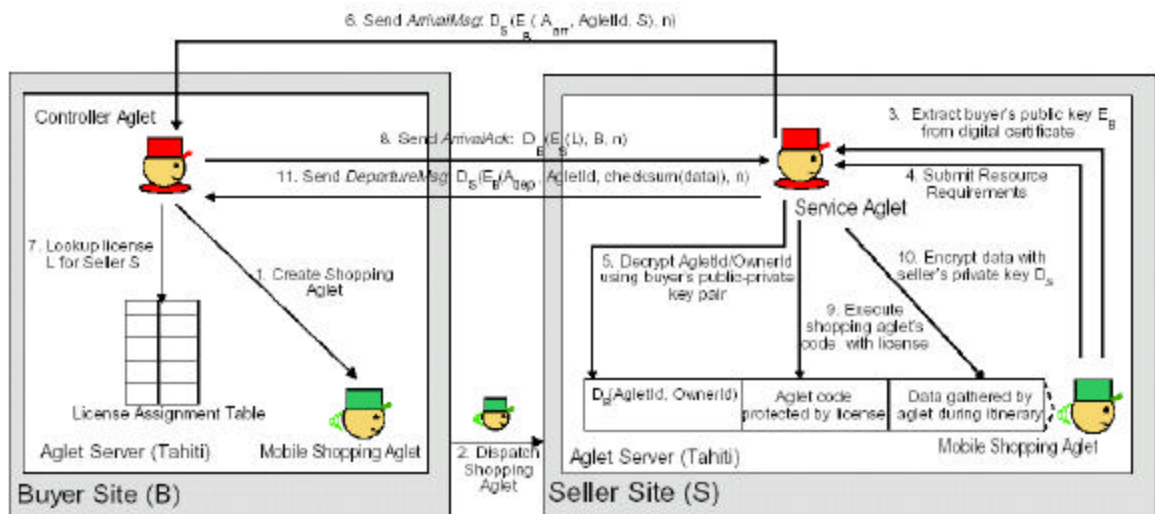
1. El Aglet Controlador (AC) crea el Aglet móvil comprador.
2. El Aglet móvil comprador se mueve al sitio del vendedor N.
3. El Agente de Servicio en el sitio del vendedor (ASV) extrae la llave pública del comprador del certificado digital que trae al Aglet móvil comprador.
4. Aglet móvil comprador solicita recursos.
5. ASV descifra el identificador de agente (*AgletID*) y el identificador de propietario (*OwnerID*), utilizando la llave pública leída en (3).

---

<sup>11</sup> El itinerario de un agente móvil contiene la dirección de los diferentes servidores que debe visitar.

6. ASV envía un mensaje a AC indicando que el agente llegó y solicitando la licencia de ejecución del agente.
7. AC recibe el mensaje y busca en el *Keystore* la licencia de ejecución para el vendedor N.
8. AC Envía la licencia a ASV.
9. ASV ejecuta el código del agente utilizando la licencia recibida.
10. ASV cifra la información que entrega al agente utilizando la llave privada del vendedor N.
11. El Aglet móvil comprador continúa el itinerario
12. ASV envía un mensaje a AC indicando que el Aglet móvil comprador salió del sitio del vendedor N.

Figura 6. Protocolo de Seguridad en MAgNET.



Fuente. [DASGUPTA01].

Análisis Crítico: Con base en estudio realizado al sistema MAgNET, se concluye que:

- MAgNET cubre gran parte de las necesidades básicas de los usuarios en un entorno de comercio electrónico, pero obliga al usuario a conocer los datos exactos del producto que desea adquirir, por lo tanto ofrece una búsqueda de información restringida.

- Otra de las limitaciones de MAgNET está relacionada con el pago, puesto que no se ofrece al comprador la opción de crear un agente que ordene el pago. La orden de compra del producto debe hacerla el usuario por sí mismo.
- Se cuenta con un protocolo de seguridad eficiente, pero que limita las ventajas de los Agentes Móviles. Puesto que dentro del protocolo se exige la conexión permanente del usuario para poder autenticar y autorizar al vendedor para ejecutar el agente móvil; la permanencia del usuario en la red también se requiere para recibir los mensajes de salida del agente de cada sitio. Con eso no se garantiza la seguridad totalmente, puesto que podría existir un ataque de hombre en medio y además obliga a que los agentes sean síncronos.
- El sistema MAgNET utiliza Aglets en su implementación y con base en las características de seguridad de esta plataforma, define gran parte de su protocolo de seguridad, por lo tanto las plataformas que no manejen un esquema similar no pueden utilizar este mismo modelo.

### **2.3.3.3 E-marketplace**

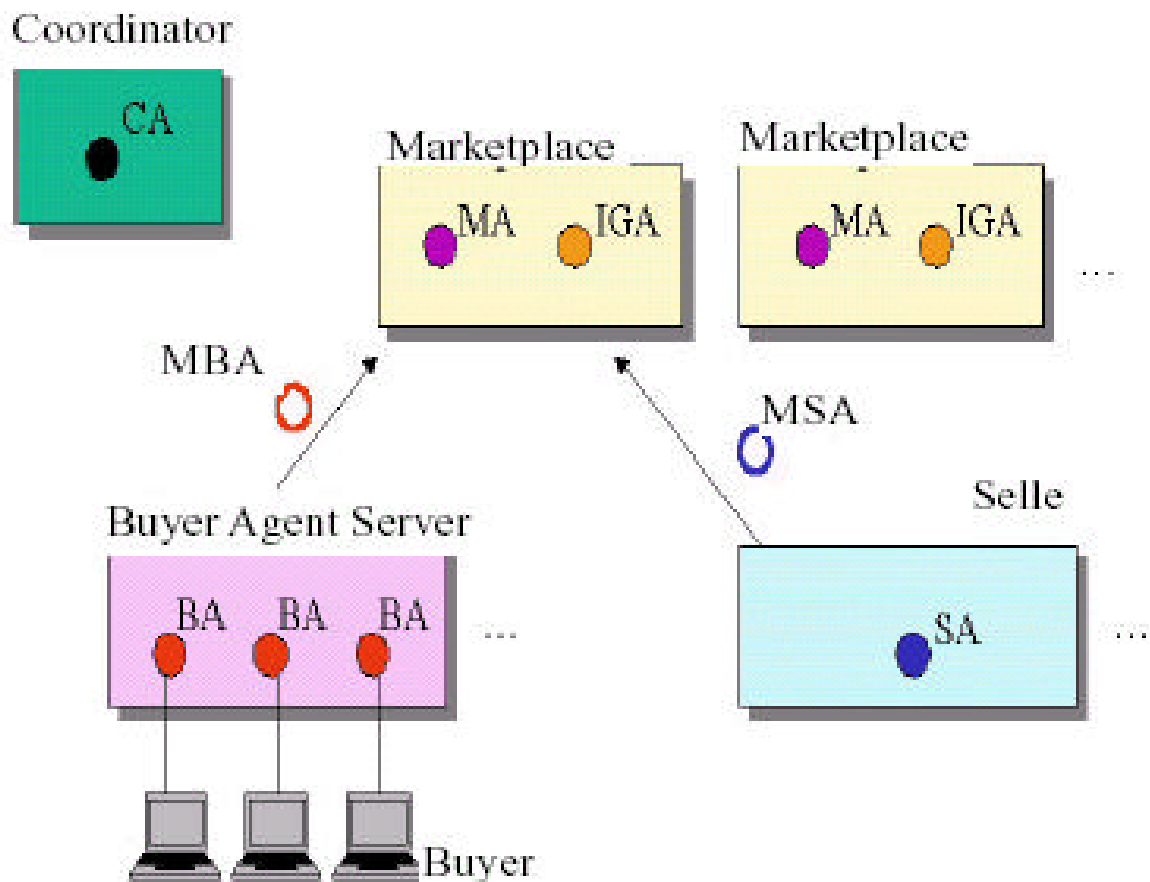
E-marketplace [WANG03] es un mercado electrónico desarrollado utilizando el lenguaje de programación Java y la plataforma de Agentes Móviles Aglets, propuesto en la Universidad Tamkang en Taiwán. Ese sistema incluye Agentes Móviles que proveen varios servicios y disponen de un mecanismo de negociación que les permite interactuar en ese mercado.

Componentes: El sistema E-marketplace (Figura 7) define cuatro tipos de servidores, cada uno de ellos contiene varios agentes estacionarios y móviles. A continuación se incluye la descripción de cada uno de ellos:

- **Servidor Coordinador:** En ese servidor se encuentra el agente coordinador (CA) que se encarga de administrar el entorno de comercio electrónico y proveer servicios de registro y autenticación.
- **Marketplace (Mercado):** Es la plataforma que provee las facilidades para que los Agentes Móviles realicen las transacciones de negocios.

- Servidor de Agentes del Comprador: Ese servidor provee una interfaz web que permite a los usuarios compradores activar y controlar sus agentes. Además se encuentra una base de datos que almacena los perfiles e información de los usuarios; los cuales se conectan a través de un agente que provee una interfaz web y se comunican con un agente estacionario al cual le expresan sus necesidades. Con esa información el agente estacionario crea el agente móvil comprador.
- Servidor del Vendedor: Cada compañía que desee participar en E-marketplace, debe tener su servidor. En ese sitio hay dos agentes, el primero es estacionario y se encarga de crear el segundo agente que viaja a los Mercados y ofrece los productos.

Figura 7. Arquitectura del Sistema E-marketplace.




Fuente. [WANG03].

Funcionamiento: Los pasos del proceso de compra en E-marketplace son los siguientes:

1. El comprador crea y envía el agente móvil comprador, especificando las tareas y la dirección del mercado que debe visitar.
2. El agente móvil migra al mercado especificado y se registra. Allí busca la información del producto que el comprador necesita y los posibles vendedores. El agente móvil envía al agente estacionario del servidor de agentes del comprador esa información y retorna al origen.
3. El comprador recibe la lista sugerida por el agente móvil y selecciona el vendedor para realizar la transacción. Con esa selección, el comprador envía el agente móvil al mercado a completar la compra.
4. Cuando el agente móvil llega al mercado contacta el agente del vendedor seleccionado y realiza la compra. Cuando la transacción ha finalizado se contacta al agente administrador y se informa del resultado, con eso se valida la operación.
5. El agente móvil regresa al origen y reporta los resultados.

Análisis Crítico: Con base en el estudio realizado a E-marketplace, se concluye que:

- E-marketplace ofrece la posibilidad de acceder al mercado a compradores que no cuentan con el sistema en sus computadores; esta opción la ofrece a través de un Servidor de Agentes.
- Una de las características principales de este sistema es la existencia de mercados (marketplace) lo cual resulta bastante útil en el momento de agrupar compradores y evitar el tránsito en la red, pero aumenta la carga de administración del sistema; además de tener que administrar el Servidor de Agentes del Comprador, debe incluirse servidores que soporten los mercados electrónicos. Con respecto al hardware, se encuentra que deben ser equipos de mayor potencia, puesto que albergarán los agentes vendedores y los agentes compradores de cada uno de los usuarios del mercado.

- Otro punto importante a tener en cuenta es la funcionalidad de los agentes vendedores en el mercado. Para responder a las necesidades de los compradores deben tener acceso a la información de los productos distribuidos por el vendedor que representan. Por el tamaño del agente es poco práctico que lleve con él la información, así que debe enviar y recibir mensajes para obtenerla ó enviar un agente al sitio de su propietario a recuperarla. Con esta interacción extra se está aumentando el tráfico en la red, en lugar de reducirlo que es el sentido del mercado.
  - Se incrementan los riesgos de seguridad, puesto que la información que lleva un agente vendedor puede ser capturada en cualquiera de sus múltiples interacciones con el servidor de su propietario.
- 

## **3. REQUERIMIENTOS DE UNA ARQUITECTURA PARA COMERCIO ELECTRÓNICO**

Este capítulo presenta el análisis de requisitos de una arquitectura basada en Agentes Móviles para realizar transacciones de comercio electrónico. Se inicia con los principales puntos que se extraen del análisis de los sistemas para comercio electrónico realizado en el capítulo 2 y con base en esos resultados y el estado del arte y la técnica se formulan los requerimientos de diseño de la arquitectura. Dentro de estos requerimientos se incluyen los requisitos que deben cumplir las principales tecnologías necesarias para la implementación de la misma y cuyos conceptos se utilizan para la formulación de la arquitectura.

### **3.1. ANÁLISIS DE LOS SISTEMAS PARA COMERCIO ELECTRÓNICO ACTUALES**

Con base en el análisis realizado a los sistemas Agora Shopping Mall, MAgNET y E-marketplace en el Capítulo 2, se extraen los principales puntos que deben tenerse en cuenta durante la formulación de los requisitos de la Arquitectura:

1. Los Agentes Móviles que realicen los proceso de búsqueda deben estar en capacidad de realizar consultas en los sitios de varios vendedores; así mismo deben contar con la autonomía necesaria para filtrar la información encontrada y entregar sólo los resultados que le son de utilidad al usuario.
2. Es necesario permitir a los Agentes Móviles hacer uso de su autonomía para obviar el alto grado de interacción con el usuario durante su itinerario, de esta manera se evita generar congestión en la red y limitar las ventajas del modelo.

3. En la transacción de búsqueda de información, es necesario brindar al usuario la posibilidad de consultar por medio de palabras claves, puesto que en la mayor parte de los casos no se conocen los datos exactos y completos del producto ó servicio.
4. En la transacción de Compra / Venta debe ofrecerse la opción al usuario de ordenar el pago por medio de un Agente Móvil que lleve el número de su tarjeta de crédito ó el equivalente a ella en el sistema; una posibilidad es usar un código autorizado de pago que identifique a ese usuario.
5. El diseño de la arquitectura debe ser independiente de la plataforma de Agentes Móviles que se seleccione para su implementación, es decir, no debe definirse la arquitectura con base en las características específicas de una plataforma; aunque es posible establecer límites ó requerimientos básicos de una plataforma para poder ser utilizada.
6. Es necesario ofrecer a los usuarios la posibilidad de acceder al sistema, aunque no cuenten con el software completo (plataforma de agentes, base de datos, sistema de agentes, etc.).
7. Se deben ofrecer características de seguridad que permitan autenticar usuarios, servidores y agentes y que con base en esa autenticación se autorice la realización de transacciones. Las políticas de seguridad que se establezcan deben ser independientes de la permanencia del usuario conectado a la red; las transacciones deben continuar, aunque no se encuentre el usuario, de no ser así se estarían limitando las ventajas del Modelo de Agentes Móviles.

### **3.2. REQUISITOS DEL SISTEMA PARA COMERCIO ELECTRÓNICO**

Un Sistema para comercio electrónico debe aprovechar la ventaja de los Agentes Móviles para disminuir la carga de la red; los agentes deben tener la autonomía para viajar a los diferentes sitios, recibir la información de los productos y filtrarla con base en

preferencias. Además las características de seguridad deben ser independientes de la plataforma de agentes utilizada para evitar reducir el diseño de la arquitectura a una simple implementación.

En el comercio electrónico se pueden destacar dos roles: el proveedor de servicios y la entidad que los usa; por esa razón pueden plantearse dos sistemas: Sistema del Vendedor y Sistema del Comprador. A continuación se describe cada uno de ellos.

### **3.2.1. Sistema del Comprador.**

En este sistema hay que tener en cuenta al usuario y el equipo de cómputo que utiliza. Los siguientes ítems ilustran esta afirmación:

- El tiempo de permanencia de un Usuario en un sistema es variable, por tanto los agentes deben tener la autonomía necesaria para actuar sin necesidad de recibir órdenes a cada paso. Además con las características de movilidad y autonomía, los agentes tienen la capacidad de desplazarse a través de la red, viajar a los sitios de los vendedores, recibir información de productos, y filtrarla con base en las preferencias del Usuario.
- Se debe permitir el acceso al Sistema del Comprador a usuarios que no tengan instalado el software en su equipos, este puede ser el caso de compradores que accedan a un equipo que se encuentra permanentemente en la red y que es proveído directamente por el Administrador del sistema ó por un proveedor de Servicios.
- El sistema debe proveer opciones de búsqueda de información sobre los productos para evitar que el usuario tenga que conocer los datos exactos del artículo que desea adquirir; además es necesario ofrecer mayor funcionalidad en la compra.
- Dar la oportunidad al Usuario de enviar un agente que ordene el pago, con su número de tarjeta de crédito o con un código autorizado de pago que identifique a ese usuario en el Sistema.

### **3.2.2. Sistema del Vendedor.**

- En este sistema la persona sólo administra resultados, nuevos inventarios y creación o eliminación de ofertas, no se requiere su presencia para el funcionamiento pues debe tener agentes que se encarguen de interactuar con los que llegan; Debe incluir la opción de palabras claves para que se realicen búsquedas dentro del inventario de la base de datos sin restringir al agente comprador para que sepa las características exactas del producto.
- Se hace necesario establecer las políticas necesarias para que se permita a los agentes de este sistema entregar información de productos a los agentes de los compradores.
- Incluir la notificación de ofertas y novedades a los compradores con base en su perfil, el cual está dado por su historial de compra.
- Proveer la opción de realizar la transacción de ordenes de Compra / Venta al recibir un número de tarjeta de crédito o un código autorizado de pago que identifique al usuario y a su vez entregar un código que indique la aceptación de la transacción.

### **3.2.3. Tipos de Servicios.**

Con base en el análisis de los dos Sistemas anteriores, se considera que los servicios que debe proveer una aplicación para comercio electrónico son:

- Búsqueda de información con base en características o palabras clave del producto requerido. Beneficia a todos aquellos usuarios que desconocen la información exacta de un producto.
- Compra de artículos, esto incluye dar la funcionalidad al sistema para que el usuario pueda delegar en un agente la tarea de compra de productos.

- Notificación de ofertas y novedades, de esta forma se beneficia al usuario vendedor, puesto que logra que los compradores conozcan esta información, a través de un agente móvil que vaya a sus sitios y les lleve un mensaje.

A continuación se presenta una breve descripción de los requerimientos para desarrollar sistemas para comercio electrónico.

### **3.3. REQUISITOS DE LA PLATAFORMA DE AGENTES MÓVILES**

Las plataformas de Agentes Móviles actuales proveen algunas características de seguridad, así como unos servicios básicos. En este numeral se presentan las necesidades de la arquitectura en cada uno de ellos.

#### **3.3.1. Seguridad básica**

No es necesaria. La seguridad será definida de tal manera que no dependa de la plataforma de agentes utilizada. Con esto se ayuda a que la arquitectura sea independiente de la plataforma de agentes móviles que se use.

#### **3.3.2. Servicios de Agentes**

Los siguientes son los servicios básicos que se requieren de una plataforma de Agentes Móviles:

- Creación de agentes. Este servicio es inherente a la plataforma, permite la instanciación de los agentes tanto móviles como estacionarios.
- Movilidad. La plataforma utilizada en la implementación de la arquitectura debe ofrecer la posibilidad a los agentes de viajar a los diferentes sitios especificados en su itinerario.

- Desactivación y activación de agentes. Cuando los agentes móviles están realizando sus tareas pueden tener problemas para regresar a su origen, una de las razones para que esto ocurra es que el usuario haya finalizado su sesión en la red; por lo tanto la plataforma debe permitir al agente desactivarse y almacenarse en memoria secundaria para después volver a activarse e intentar nuevamente el viaje.
- Comunicación. Para la realización de sus tareas los agentes tanto móviles como estacionarios necesitan comunicarse, la principal forma de hacerlo es el paso de mensajes; estos mensajes pueden ser locales ó remotos, dependiendo del agente y sus labores; por lo tanto la plataforma debe permitir el envío de mensajes con las características mencionadas.

### **3.4. REQUISITOS DEL LENGUAJE DE PROGRAMACIÓN**

El lenguaje de programación debe proveer las siguientes características o funcionalidades:

- Portabilidad. Esta característica es necesaria para lograr una ejecución idéntica del código del agente en diferentes sistemas operativos.
- Orientación a objetos. La utilización de un lenguaje de programación orientado a objetos permite la interacción con el modelo de agentes móviles y facilita el diseño.
- Persistencia. Considerando que los agentes móviles deben suspender su ejecución en un servidor, moverse a otro y reiniciar su ejecución allí, se hace necesario que el lenguaje de programación permita conservar el estado de datos del agente móvil.
- Permitir la definición de políticas de seguridad. El lenguaje de programación además de las características anteriores, debe proveer librerías que faciliten la utilización de mecanismos de seguridad para proteger tanto los agentes como los servidores.

### **3.5. REQUISITOS DE SEGURIDAD**

Un sistema para comercio electrónico basado en Agentes Móviles requiere las siguientes características de seguridad, que se pueden descomponer en protección del servidor y de los agentes.

#### **3.5.1. Protección del servidor**

- Autenticar los agentes y sus propietarios cuando llegan al sistema.
- Otorgar privilegios de forma individual a los agentes móviles, por esta razón hay que especificar mediante asignación de permisos los recursos que van a utilizar.
- Definir un protocolo de seguridad para garantizar las transacciones.

#### **3.5.2. Protección de los agentes**

- Verificar que el código del agente no presente modificaciones en su programación.
- Verificar los datos que transporta el agente móvil, para asegurar que no han cambiado.



## 4. PRESENTACIÓN DE ARAMCEL

En las siguientes secciones se presenta ARAMCEL. Se inicia con el análisis y diseño global; posteriormente se hace una descripción de los componentes de la Arquitectura, funcionamiento, pseudocódigo de los principales agentes y se explica el proceso para realizar las transacciones de comercio electrónico definidas.

### 4.1. ANÁLISIS Y DISEÑO GLOBAL

Con base en los requisitos de una Arquitectura Basada en Agentes Móviles para comercio electrónico detallados en el capítulo 3, se identifican las transacciones básicas de ARAMCEL; con ellas se realiza un estudio para determinar los principales Casos de Uso (CU) y Actores de la Arquitectura. A continuación se presentan los resultados de este análisis, en cada una de las funcionalidades identificadas:

#### **Búsqueda de Información**

El caso de uso identificado fue “*Buscar Información*” (Figura 8), a través del cual, el Usuario Comprador (UC) ingresa los datos de los artículos que desea buscar y establece las preferencias de búsqueda; con esta información un Agente Móvil Buscador (AMB) en representación suya, interactúa con el Agente Estacionario Vendedor (AEV) que representa cada sitio. Durante esta interacción el AMB solicita la cotización de los productos que le interesan al UC.

Figura 8. Caso de Uso Buscar Información.



Fuente. El autor.

### **Compra / Venta de Productos**

El caso de uso “*Ordenar Compra / Venta*” (Figura 9), permite al UC seleccionar los artículos que desea comprar con base en los resultados obtenidos de la búsqueda de información; en representación suya, el Agente Móvil Comprador (AMC) se dirige al Sitio del Vendedor seleccionado, interactúa con el AEV y ordena la compra del producto.

Figura 9. Caso de Uso Ordenar Compra / Venta.



Fuente. El autor.

### **Notificación de ofertas y novedades**

El caso de uso “*Notificar Compradores*” (Figura 10) permite al Agente Estacionario Despachador del Vendedor (AEDV) enviar un Agente Móvil Notificador (AMN) a los sitios de los compradores a entregar un mensaje al Agente Estacionario Recepcionista del Comprador (AERC) de cada uno de ellos. La información que lleva el AMN corresponde a las ofertas ó novedades de artículos en la tienda, las cuales ha sido previamente filtradas por el AEDV con base en el perfil de cada comprador.

Figura 10. Caso de Uso Notificar Compradores



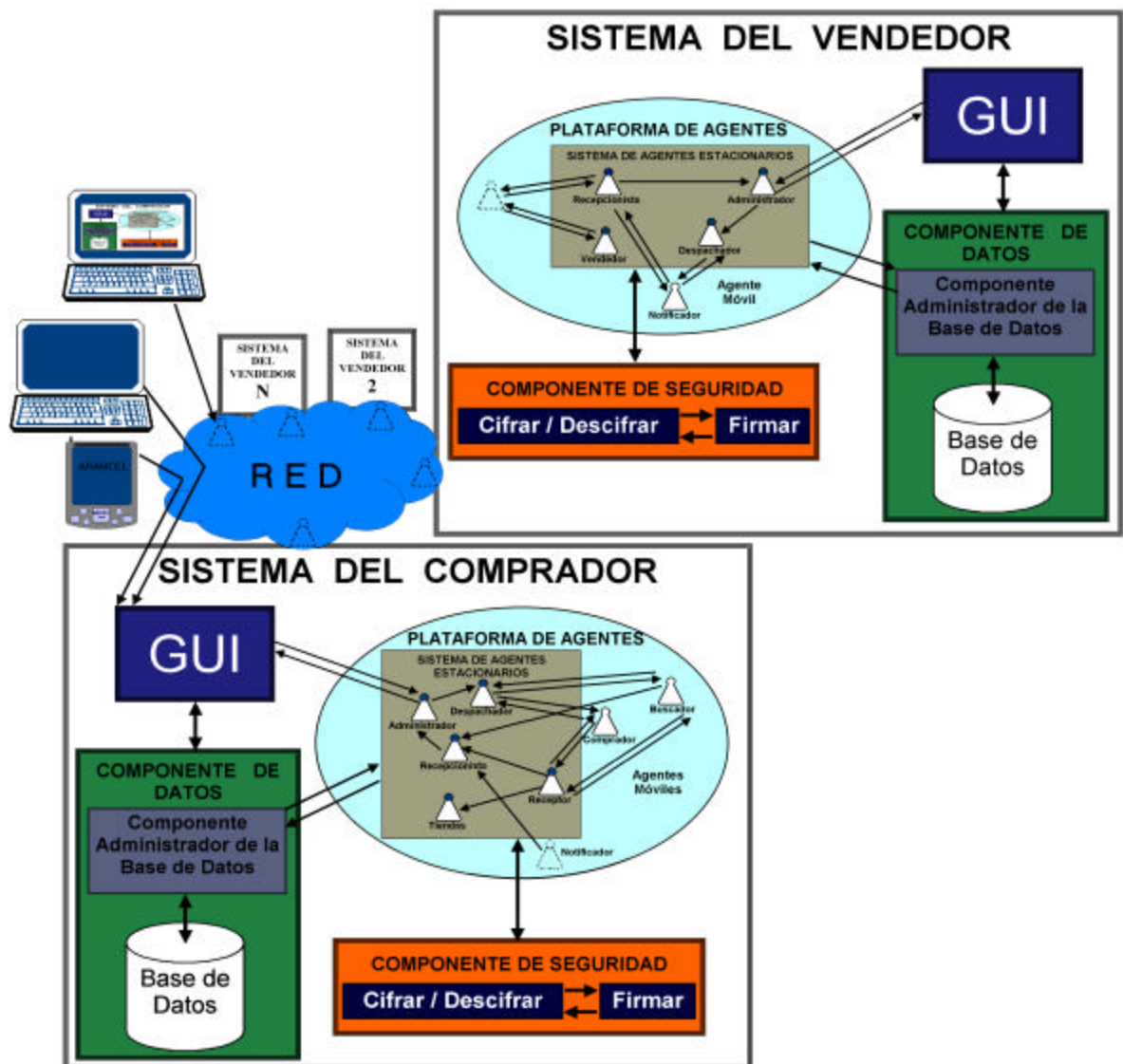
Fuente. El autor.

Hasta este punto se han identificado los principales Casos de Uso de la Arquitectura, así como los actores involucrados, con esta información se realiza la definición y diseño de los componentes de ARAMCEL.

## 4.2. DESCRIPCIÓN GENERAL DE ARAMCEL

ARAMCEL (Figura 11) es la propuesta del "Grupo en Ingeniería Telemática y Sistemas Inteligentes" GITSI de una Arquitectura basada en Agentes Móviles para realizar transacciones de comercio electrónico. La especificación de ARAMCEL está basada en el lenguaje de programación JAVA, en los conceptos del modelo de AM y en las necesidades presentes en un entorno de comercio electrónico.

Figura 11. Arquitectura ARAMCEL.



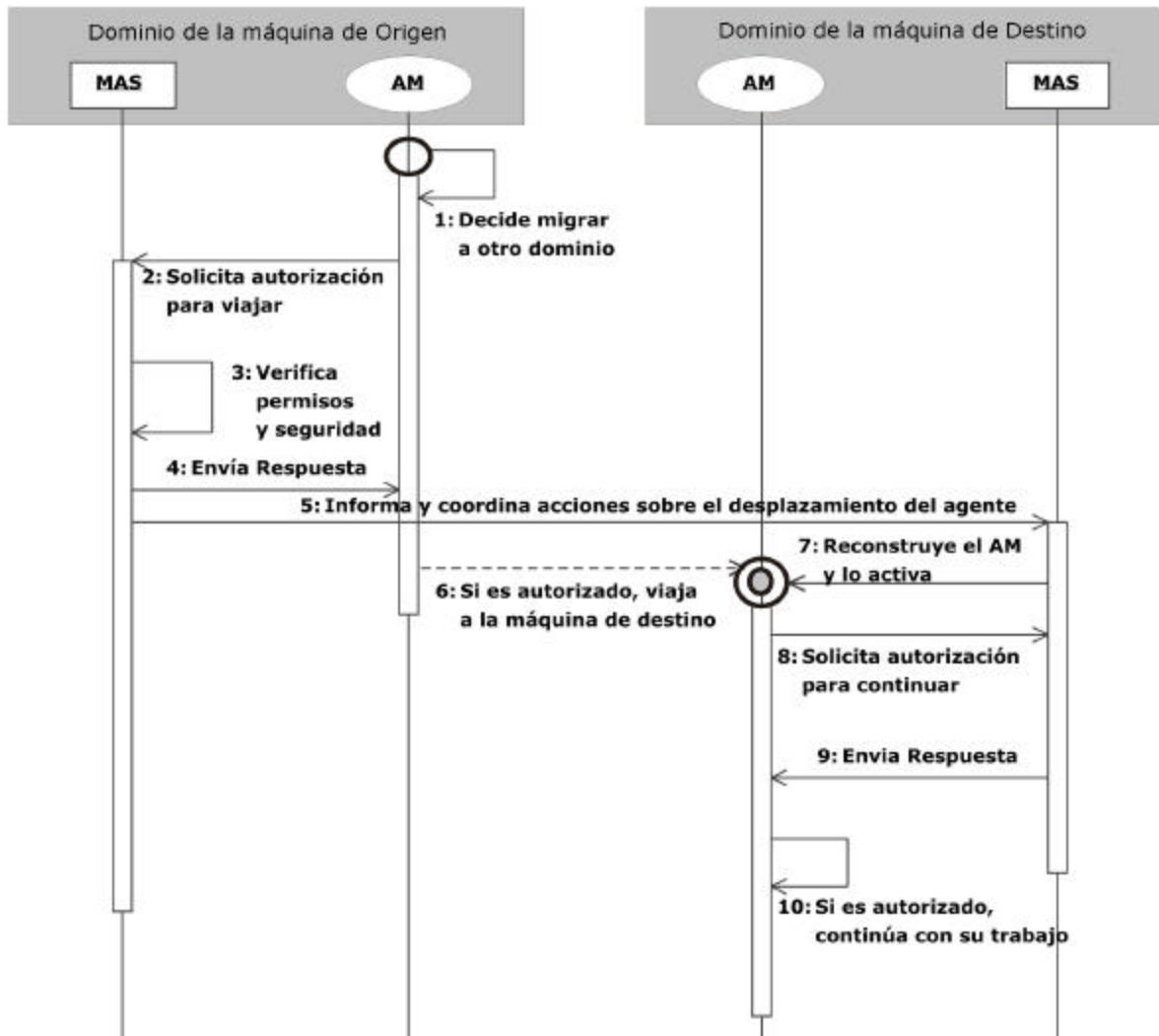
Fuente. El autor.

En la Arquitectura ARAMCEL se utiliza el modelo de negocios en el cual el Comprador, con base en sus requerimientos, envía Agentes Móviles que acceden a los productos ofrecidos por los Vendedores. Para poder enviar y recibir Agentes Móviles, es necesario que el dominio que efectúa la tarea cuente con una plataforma de agentes móviles o MAS (*Mobile Agent System* – Sistema de Agentes Móviles). El proceso de movilización o migración del agente incluye los siguientes pasos [CASTILLO02]:

1. Creación del agente. El Agente Móvil toma la decisión de migrar. Antes de hacerlo adquiere la información necesaria para realizar sus tareas y seguir su itinerario.
2. Solicitud de permiso para viajar. El Agente Móvil solicita al MAS la autorización de viaje.
3. Verificación. El MAS se encarga de verificar los permisos del agente y entregarle datos relacionados con el viaje.
4. Autorización. Si no hay problemas, el MAS le otorga al agente la autorización para viajar.
5. Comunicación entre el MAS origen y el MAS destino. Cuando se autoriza al agente, el MAS de la máquina origen contacta al MAS destino para llevar a cabo el desplazamiento del agente.
6. Transporte. El MAS transporta al agente móvil hasta la máquina destino. Para realizarlo, se serializa el código y el estado del agente, posteriormente se contacta al MAS de la máquina destino y se envía el agente serializado.
7. Reconstrucción. El MAS de la máquina de destino reconstruye y activa el agente móvil.
8. Solicitud de permisos locales. Una vez ha sido activado, el agente móvil se comunica con el MAS y solicita: autorización para utilizar los recursos locales, verificación de las condiciones locales de seguridad, inscripción en la lista de agentes disponibles, etc.
9. Autorización. Si no hay inconvenientes el MAS destino autoriza al agente móvil para continuar sus labores.
10. Continuación de labores: Cuando recibe la autorización, el agente continúa su ejecución.

El proceso descrito anteriormente puede verse en el Diagrama de Secuencia presentado en la Figura 12.





Figura 12. Migración de un Agente Móvil



Fuente. El autor.

En la Figura 12 se presentan cuatro convenciones importantes a tener en cuenta dentro de los diagramas de secuencia de la arquitectura; la Tabla 1 resume las convenciones mencionadas:

Tabla 1. Convenciones para Diagramas de Secuencia

	Representa el Agente Móvil antes de viajar
	Representa el Agente Móvil, cuando ha sido reconstruido por el MAS en el sitio de destino.
	La línea punteada representa el desplazamiento del Agente Móvil
	La elipse representa los Agentes Móviles, entidades más flexibles que los módulos tradicionales.

Fuente. El autor.

Como se presentó en la Figura 11, ARAMCEL está distribuida en dos sistemas definidos, el Sistema del Comprador y el del Vendedor, a continuación se hace la descripción de cada uno de ellos.

#### 4.3. SISTEMA DEL COMPRADOR

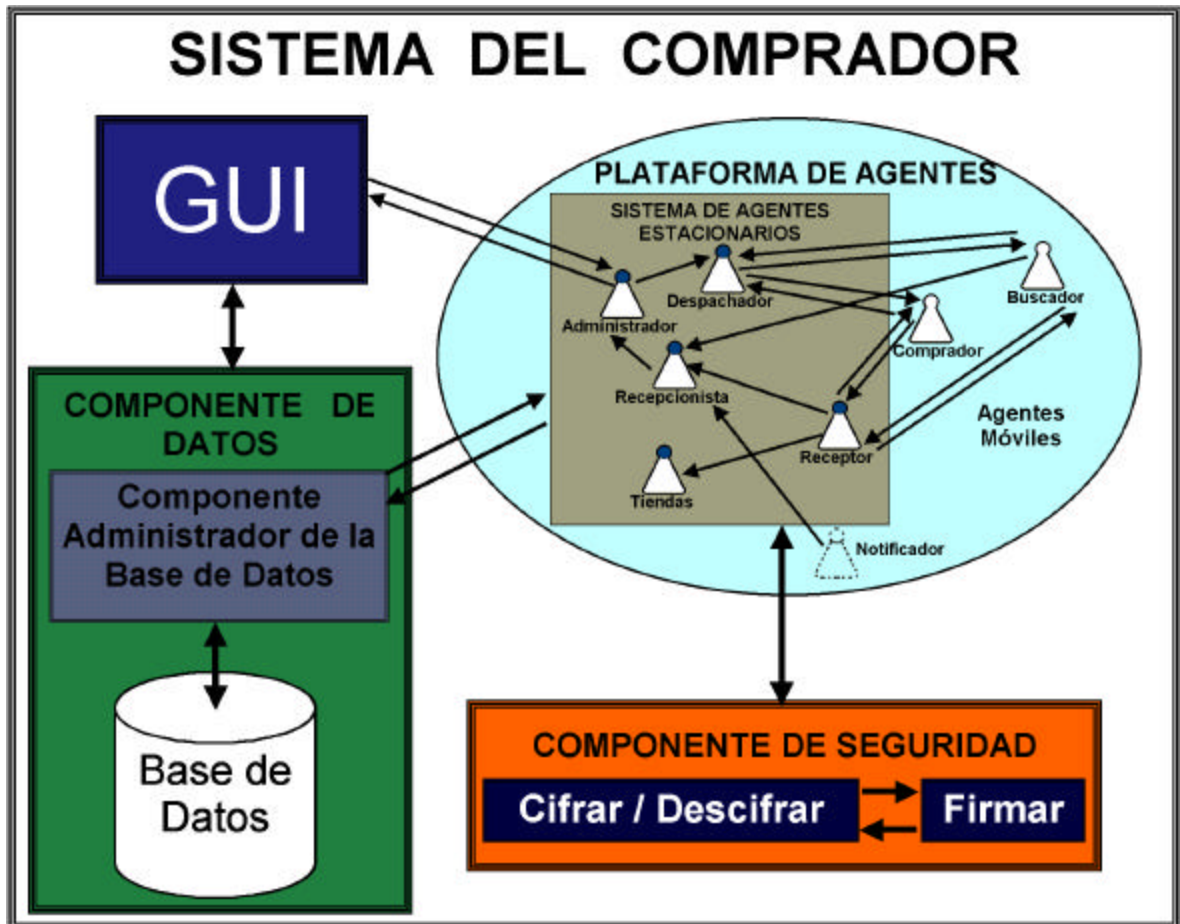
El sistema del Comprador está conformado por la interfaz gráfica de usuario, la plataforma de agentes, el componente de seguridad y el componente de datos (Figura 13).

A continuación se hace una descripción de cada uno de ellos.

- *Interfaz Gráfica de Usuario (GUI)*: Permite que el usuario ingrese información al sistema; esta información corresponde a requerimientos, preferencias sobre artículos y demás datos que sean necesarios. Además se utiliza como medio para mostrar al usuario la información que los agentes han encontrado.
- *Plataforma de Agentes*: En ella se encuentra el Sistema de Agentes Estacionarios y los Agentes Móviles. Los agentes estacionarios son aquellos que permanecen en la máquina del Comprador y cumplen funciones como el despacho de agentes a los sitios de los Vendedores y la recepción de los Agentes Móviles cuando retornan con la información que han recolectado. Los Agentes Móviles son agentes creados para la

búsqueda y compra de artículos, llevan la información de las preferencias del usuario sobre los artículos que desea y el itinerario con los sitios a visitar.

Figura 13. ARAMCEL - Sistema del Comprador.



Fuente. El autor.

- *Componente de Seguridad:* Este componente está subdividido en dos módulos, el primero se encarga del cifrado y descifrado de los datos y el segundo de los procesos de firmado y verificación de firmas. Este componente es utilizado por el Agente Despachador para solicitar el cifrado y firmado de la información que va a transportar el agente móvil. También es usado por el Agente Estacionario Recepcionista del Comprador para verificar la firma del Vendedor y descifrar la información que se recibe del agente móvil comprador.

- *Componente de Datos:* Este componente está conformado por la Base de Datos y el Componente Administrador de la Base de Datos. La primera contiene la información relacionada con las solicitudes del Comprador y el resultado de los procesos realizados por el Agente Móvil; también se almacena una lista con las direcciones de los sitios de los Vendedores y el tipo de artículos que ofrecen, ese listado es usado para la generación del itinerario del agente móvil. El Componente Administrador de la Base de Datos es el encargado de realizar las consultas y modificaciones a la información almacenada en ella.

Al Sistema del Comprador puede accederse de dos formas principalmente, la primera de ellas es a través de un equipo denominado Servidor de Compradores (SC), al cual acceden los usuarios Compradores por medio de la interfaz de usuario. Un Comprador puede acceder al sistema desde cualquier dispositivo que tenga conexión a la red (computador, asistente digital personal, celular, computador portátil, etc.). Los usuarios previamente registrados, sólo necesitan un navegador para Internet, mediante el cual se conectan al SC y pueden usar el sistema. La segunda forma corresponde a aquellos usuarios que tienen el sistema completo instalado en sus propios computadores, los cuales ejecutan la aplicación localmente, pero necesitan encontrarse conectados a Internet en el momento de enviar los agentes del sistema.

#### **4.3.1. Funcionamiento.**

Para ingresar al Sistema del Comprador, el UC se conecta a la GUI y solicita el inicio de sesión; para eso debe ingresar su nombre de usuario "*login*" y la clave de acceso "*password*". Si lo hace a través del SC, la conexión Cliente / Servidor se hace utilizando un protocolo de comunicación segura.

Una vez que sea válida la entrada, el Usuario puede hacer búsquedas en la red de los artículos que desea. El sistema mediante sus Agentes Móviles toma la información de los artículos y las preferencias del Usuario, como puede ser el tiempo de entrega del artículo, costo, cantidad de unidades, etc.; cuando se tienen esos datos, los agentes se desplazan a través de la red a cumplir con la búsqueda de información y cuando la recolectan, regresan a su sitio de origen a reportar los resultados obtenidos.

El UC puede finalizar su sesión en el sistema y después reconectarse, para revisar el listado que contiene los resultados que trajeron los agentes; con base en esa información, puede dar la orden de crear un Agente Móvil Comprador (AMC) que viaje al sitio elegido y procese la orden de compra.

#### **4.3.2. Plataforma de Agentes.**

Como ya se mencionó, la plataforma de Agentes, alberga los Agentes Estacionarios y los Agentes Móviles, los siguientes numerales presentan la descripción de cada uno de los agentes del Sistema del Comprador.

##### **4.3.2.1 Agentes Estacionarios**

Los Agentes Estacionarios en el sistema del Comprador son Despachador, Receptor, Recepcionista, Administrador de Tiendas y Administrador de la Interfaz de Usuario.

##### **Agente Despachador**

El Agente Estacionario Despachador del Comprador (AEDC) es el agente autorizado en la plataforma para crear Agentes Móviles y transferirle a cada uno la información sobre el artículo, además les incluye las preferencias del usuario. Interactúa con el Componente de Seguridad (CSC) para el cifrado y firmado de la información que va a llevar el agente móvil. Por medio del Componente Administrador de la Base de Datos del Comprador (CABDC) recibe los sitios de los vendedores que ofrecen ese tipo de artículo y se los envía al agente móvil.

##### **Agente Receptor**

El Agente Estacionario Receptor (AERP) es el que interactúa con los agentes móviles cuando regresan a la plataforma; utiliza el CSC para verificar la integridad del agente; toma los resultados de los agentes y con el CSC descifra la información y verifica la validez de la firma digital del código de aceptación y de las cotizaciones que traen. Además toma el reporte de servicio entregado por los Agentes Móviles y lo envía al Agente Administrador de Tiendas.

### **Agente Recepcionista**

El Agente Estacionario Recepcionista del Comprador (AERC) es el que tiene la función de recibir a los agentes móviles cuando regresan a la plataforma, les asigna un AERP para que interactúe con ellos; con base en su autonomía si considera que el AERP está ocupado para poder atender las peticiones de un Agente Móvil, procede a crear otro AERP y en caso de no necesitarlo lo elimina. Cuando llega un agente Notificador se encarga de recibir las ofertas que trae y se las envía al agente Administrador de la Interfaz de Usuario.

### **Agente Administrador de Tiendas**

El Agente Estacionario Administrador de Tiendas (AEAT) con base en la información recibida del AERP con respecto al servicio recibido por los Agentes Móviles en los sitios de los vendedores; procede a asignarle a cada tienda un valor que incrementa ó decrementa el puntaje de ella; de esa manera se busca darle mayor prioridad a los sitios con mejor servicio, esto se tiene en cuenta al generar el listado de sitios a visitar.

### **Agente Administrador de la Interfaz de Usuario**

El Agente Estacionario Administrador de la Interfaz de Usuario del Comprador (AEAC) es el encargado de entregar las opciones del sistema al UC, mediante él se presentan los resultados obtenidos por los agentes en sus viajes y se reciben los datos de los artículos que desea buscar el UC. Recibe la información de ofertas y si no está el usuario, las almacena mediante el CABDC para mostrarlas en otro momento.

#### **4.3.2.2 Agentes Móviles**

Los Agentes Móviles del Comprador son el Agente Móvil Buscador y el Agente Móvil Comprador.

### **Agente Móvil Buscador**

El Agente Móvil Buscador (AMB) es un agente que viaja a los Sitios de los Vendedores a buscar información sobre un determinado producto; filtra la información recibida en cada sitio con base en las preferencias del UC. El itinerario que recorre puede reordenarlo en caso que un Sistema del Vendedor no esté respondiendo, además tiene la autonomía para reintentar el viaje a cada sitio y en caso de no poder regresar al Sistema de origen,

se almacena en la memoria secundaria de un vendedor para después reactivarse e intentar regresar a la máquina de su usuario.

### **Agente Móvil Comprador**

El Agente Móvil Comprador (AMC) es el agente que viaja al Sitio del Vendedor, en donde se encuentra el artículo elegido por el UC; allí realiza con el AEV los pasos necesarios para ordenar la Compra / Venta del producto; es el encargado de entregar el código de pago al AEV y de regresar con el código de aceptación, además de realizar todos los pasos necesarios para realizar esa tarea.

La participación de cada uno de estos agentes en la transacciones de comercio electrónico definidas en ARAMCEL se mostrará en la sección 4.5, de este documento.

### **4.3.3. Pseudocódigo de algunos Agentes.**

Una vez se ha hecho la descripción general de los Agentes del Sistema del Comprador, se presenta la explicación del funcionamiento de los Agentes Móviles del Sistema junto con su respectivo pseudocódigo.

#### **4.3.3.1 Agente Móvil Buscador**

El AMB es creado por el AEDC y solicita la información correspondiente al producto que debe buscar y los requerimientos de precio máximo a pagar, tiempo de entrega aceptable y cantidad de artículos. Con esos datos el AMB solicita al AEDC acceso a la lista de potenciales vendedores para establecer su itinerario. El AMB crea un objeto en donde almacena los resultados que coinciden con el producto solicitado y que a su vez cumplan ó mejoren los requerimientos del Comprador.

El AMB visita cada sitio de los vendedores en su itinerario, si alguno de ellos no se encuentra disponible, reordena el itinerario, para intentarlo después. Cuando el AMB llega a cada uno de los sitios, establece comunicación con el Agente Estacionario Recepcionista del Vendedor (AERV), como resultado de esa interacción le asignan un Agente Estacionario Vendedor (AEV). El AMB se comunica con el AEV para obtener los datos de los productos que coinciden con el solicitado, esos datos forman un objeto

llamado Cotización. Además de esos datos básicos, la cotización incluye el tiempo de vigencia de la oferta, para indicar el periodo de tiempo durante el cual el sitio mantiene las condiciones de venta y reserva del producto. Con la información recibida el AMB clasifica los resultados, si coinciden con lo solicitado ó cumplen con las condiciones requeridas procede a almacenarlos; en cualquier caso almacena la información de la tienda y del servicio recibido, esto es para establecer el puntaje de tiendas.

Cuando el AMB finaliza el itinerario y regresa a su sitio de origen, entrega al AERP los resultados de su búsqueda y el reporte del servicio. La Figura 14 presenta el pseudocódigo del Agente Móvil Buscador.

Figura 14. Pseudocódigo del Agente Móvil Buscador.

```
// Pseudocódigo de la clase para el Agente Movil Buscador del Comprador

public class AgenteMovilBuscador extends AgenteMovil {

    private void creacion() {
        Solicitar artículo, preferencias y sitios
    }

    private void viajar() {
        Revisar el siguiente sitio en el itinerario
        Si (sitio != NULL) {
            Moverse al siguiente sitio
            Si (fallas al viajar) {
                reintentar Moverse al siguiente sitio n veces
                enviar sitio al final de itinerario y si falla n veces No reintentar y borrar sitio
            }
        }
        sino {
            Moverse al origen
            Si (fallas al viajar) {
                desactivarse(Tiempo x)
                viajar()
            }
        }
    }

    private void llegar() {
        Registrarse y revisar sitio actual
        // Cuando se registra obtiene los identificadores del sitio y del Agente Recepcionista
        Si (sitioActual == vendedor) {
            Crear mensaje de solicitud de recursos y Vendedor
            enviarMensaje(mensaje, id_Agente)
        }
        Si (sitioActual == comprador) {
            Crear mensaje de inicio de reporte de resultados
            enviarMensaje(mensaje, id_Agente)
        }
    }

    private void recibirMensaje(Mensaje mens) {
        obtener del mensaje los datos del agente
        Si (agenteEstacionarioDespachadorComprador) {
            Revisar el Mensaje recibido y extraer datos de sitios, artículos y preferencias
            Crear itinerario
            viajar()
        }
        Si (agenteEstacionarioRecepcionistaVendedor) {
            Revisar el Mensaje recibido y extraer identificador del Vendedor
            Crear mensaje para iniciar transacción
            enviarMensaje(mensaje, id_Agente)
        }
        Si (agenteEstacionarioVendedor){
            Revisar el Mensaje recibido y extraer argumentos
            Si (cotizacion) {
                Analizar cotización y guardar resultados
            }
            Analizar el servicio y guardar resultado
            viajar()
        }
        Si (agenteEstacionarioRecepcionistaComprador) {
            Revisar el Mensaje recibido y extraer identificador del Receptor
            Crear mensaje para inicio de entrega de resultados
            enviarMensaje(mensaje, id_Agente)
        }
    }

    private void enviarMensaje(Mensaje mens, Objeto id_Agente) {
        Envía el mensaje al agente
    }
}

```

Fuente. El autor.

#### **4.3.3.2 Agente Móvil Comprador**

Cuando el AEDC es informado de una solicitud de compra por parte del UC, crea un AMC. El AMC solicita la información correspondiente al producto que debe comprar, esto incluye el precio, tiempo de entrega y cantidad que aparece en la cotización entregada por el AMB<sup>12</sup>. Con esta información el AMC obtiene a través del AEDC la información sobre el Vendedor. El AMC también solicita el código del pago para el Vendedor.

Cuando el AMC decide iniciar su viaje, se desplaza al Sitio del Vendedor, al llegar, establece comunicación con el AERV y solicita recursos de ejecución y la asignación de un AEV. El AMC establece comunicación con el AEV e inicia el proceso de Compra/Venta, para esto envía la información del producto que desea comprar y la cotización que le fue entregada. El AEV le devuelve la información del producto con la que se llevará cabo el proceso, si coincide ó mejora la oferta anterior, envía el objeto que contiene el código de pago cifrado y firmado, y recibe del AEV un objeto que contiene el código de aceptación también cifrado y firmado. Una vez terminado el proceso decide volver al sitio de origen.

Cuando el AMC regresa al origen, entrega al Agente Receptor los resultados del proceso de Compra, incluyendo el código de aceptación y el reporte de servicio. La Figura 15 muestra el pseudocódigo del Agente Móvil Comprador

---

<sup>12</sup> Para mayor información sobre este proceso puede revisarse el numeral anterior "Agente Móvil Buscador".

Figura 15. Pseudocódigo del Agente Móvil Comprador.

```

//Pseudocódigo de la clase para el Agente Movil Comprador
public class AgenteMovilComprador extends AgenteMovil {

    private void creacion() {
        Solicitar artículo, cotización, sitio, datos de entrega y código de pago
    }

    private void viajar() {
        Si (sitio != NULL) {
            Moverse al sitio
            Si (fallas al viajar) {
                reintentar Moverse al sitio n veces
                Cuando sea n veces No reintentar
            }
        }
        sino {
            Regresar al origen
            Si (fallas al viajar) {
                desactivarse(Tiempo x)
                viajar()
            }
        }
    }

    private void llegar() {
        Registrarse y revisar sitio actual
        //Cuando se registra obtiene los identificadores del sitio y del Agente Recepcionista
        Si (sitioActual == vendedor) {
            Crear mensaje de solicitud de recursos y Vendedor
            enviarMensaje(mensaje, id_Agente)
        }
        Si (sitioActual == comprador) {
            Crear mensaje de inicio de reporte de resultados
            enviarMensaje(mensaje, id_Agente)
        }
    }

    private void recibirMensaje(Mensaje mens) {
        obtener del mensaje los datos del agente
        Si (agenteEstacionarioDespachadorComprador) {
            Revisar el Mensaje recibido y extraer datos solicitados
            viajar()
        }
        Si (agenteEstacionarioRecepcionistaVendedor) {
            Revisar el Mensaje recibido y extraer identificador del Vendedor
            Crear mensaje para iniciar transacción
            enviarMensaje(mensaje, id_Agente)
        }
        Si (agenteEstacionarioVendedor){
            Revisar el Mensaje recibido y extraer argumentos
            Si (búsqueda) {
                Si (nuevaCotizacion mejor o igual cotizacion) {
                    Crear mensaje con código de pago
                    enviarMensaje(mensaje, id_Agente)
                }
                sino {
                    abortar la transacción y guardar resultados
                    viajar()
                }
            }
            Si (compra) {
                Guardar código de aceptación y resultados de transacción
                viajar()
            }
        }
        Si (agenteEstacionarioRecepcionistaComprador) {
            Revisar el Mensaje recibido y extraer identificador del Receptor
            Crear mensaje para inicio de entrega de resultados
            enviarMensaje(mensaje, id_Agente)
        }
    }

    private void enviarMensaje(Mensaje mens, Objeto id_Agente) {
        Envía el mensaje al agente
    }
}

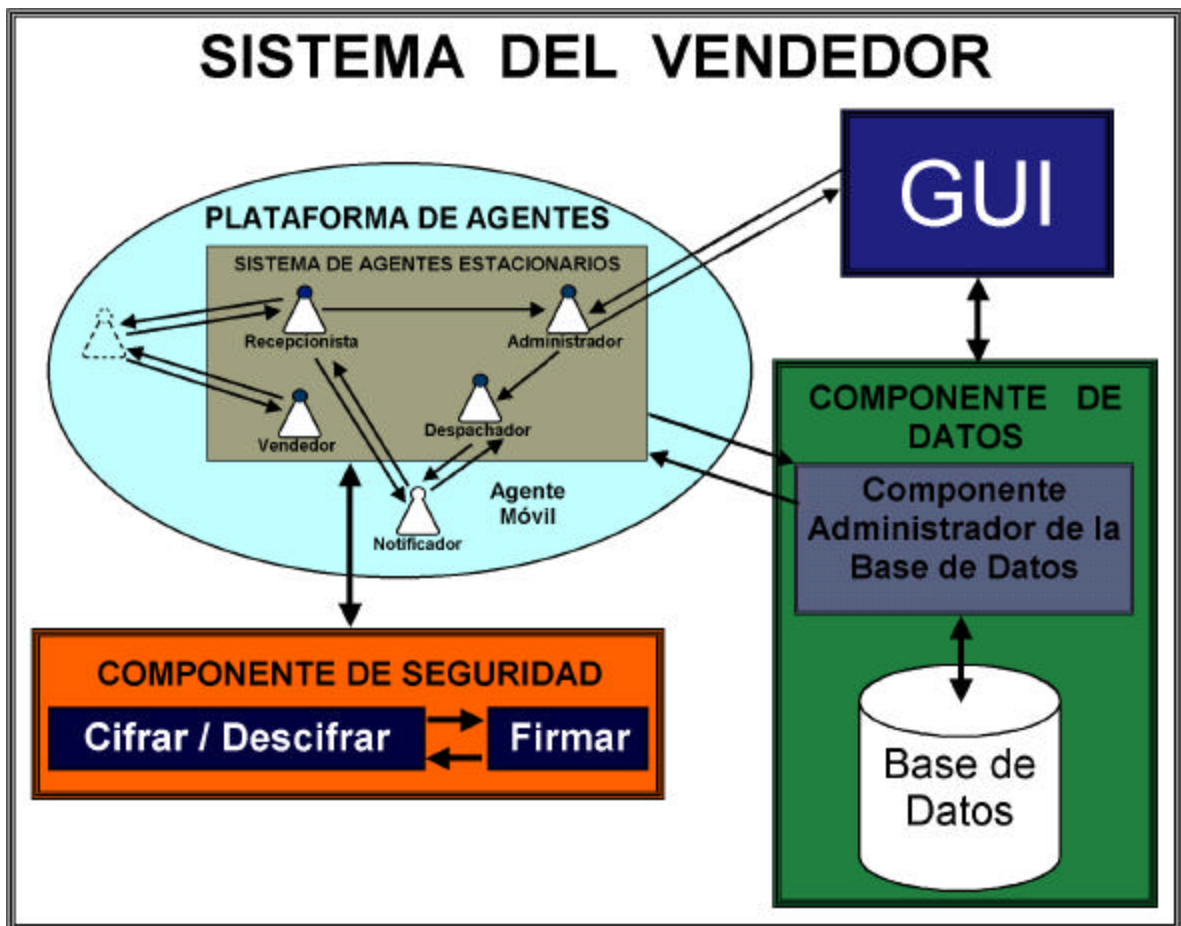
```

Fuente. El autor.

#### 4.4. SISTEMA DEL VENDEDOR

El sistema del Vendedor contiene la interfaz gráfica de usuario, la plataforma de agentes, el componente de seguridad y el componente de datos (Figura 16). El nombre de los componentes es el mismo que en el sistema del Comprador, pero cambia la funcionalidad en algunos de ellos como se describe a continuación.

Figura 16. ARAMCEL - Sistema del Vendedor



Fuente. El autor.

- *Interfaz Gráfica de Usuario (GUI)*: Permite al usuario (dueño o administrador del sitio del Vendedor) realizar modificaciones en los datos existente en ese sitio y obtener la información de las transacciones comerciales realizadas.

- *Plataforma de Agentes:* Entre los agentes estacionarios de esta plataforma, se encuentra el agente Recepcionista, encargado de asignar un agente Vendedor que interactúe con el agente móvil que hace una consulta o solicita una compra. El agente Despachador es el que crea y envía al Agente Notificador, el cual es un agente móvil del Vendedor que se encarga de informar a los Compradores sobre las ofertas vigentes en el sitio, nuevas adquisiciones y demás mensajes que desee enviar el Administrador de la tienda.
- *Componente de Seguridad:* Al igual que en el sistema del Comprador, está subdividido en dos módulos, el primero se encarga del cifrado y descifrado de los datos y el segundo de los procesos de firmado y verificación de firmas. Este componente es utilizado por el Agente Vendedor para el procesamiento de la información recibida del Agente Móvil del Comprador y la que a su vez va a ser entregada dentro del proceso de venta; además tiene la funcionalidad para revisar la integridad de los agentes.
- *Componente de Datos:* Está conformado por la base de datos que contiene la información referente a los artículos que ofrecen en ese sitio, ventas realizadas y ofertas; y por el Administrador de la base de datos que permite realizar consultas y modificaciones a la información almacenada en ella.

#### **4.4.1. Funcionamiento.**

En el caso de los sitios correspondientes a cada Vendedor, los agentes estacionarios son los encargados de interactuar con los Agentes Móviles que buscan información y con aquellos que llegan a realizar procesos de compra. De esta manera, los únicos agentes con acceso al Componente de Datos, GUI y Componente de Seguridad, son los Agentes Estacionarios del Sistema.

La interacción con el usuario dueño o administrador del sitio se hace a través de la GUI. La GUI de este sitio permite al propietario o administrador verificar los artículos existentes,

obtener información de ventas realizadas, crear ofertas, actualizar precios, modificar la cantidad de artículos cuando llega nuevo inventario y registrar promociones.

#### **4.4.2. Plataforma de Agentes.**

En el Sistema del Vendedor se encuentran presentes tanto Agentes Móviles como Estacionarios, en los siguientes numerales se incluye la descripción general de cada uno de ellos.

##### **4.4.2.1 Agentes Estacionarios**

Los agentes estacionarios presentes en la plataforma son: Agente Vendedor, Agente Recepcionista, Agente Despachador y Agente Administrador de la Interfaz de Usuario.

##### **Agente Vendedor**

El Agente Estacionario Vendedor (AEV) es el encargado de recibir del AMB los datos del artículo. El AEV con esos datos busca por medio del Componente Administrador de la Base de Datos del Vendedor (CABDV) la información relevante del artículo y la entrega al AMB mediante una cotización que va firmada por medio del Componente de Seguridad (CSV). En caso de recibir un AMC se encarga de realizar todos los pasos necesarios para procesar la transacción, esto incluye la recepción del código de pago y la entrega del código de aceptación; este último va cifrado y firmado por medio del CSV.

##### **Agente Recepcionista**

El Agente Estacionario Recepcionista del Vendedor (AERV) es el encargado en el Sitio del Vendedor de recibir a los agentes móviles de los compradores, interactúa con el CSV para revisar los agentes y una vez autenticados procede a asignarles un AEV. Si encuentra que los AEV están ocupados procede a crear otro AEV, cuando no se requiera ese nuevo agente procede a eliminarlo.

##### **Agente Despachador**

El Agente Estacionario Despachador del Vendedor (AEDV) recibe los avisos del Agente Administrador de la Interfaz de Usuario sobre las ofertas y novedades; cuando esto sucede utiliza una de sus funcionalidades para crear agentes móviles en el sitio del

vendedor; a este nuevo agente se le denomina Agente Móvil Notificador (AMN). El AEDV se comunica con el CABDV para obtener la información que necesita el AMN.

### **Agente Administrador de la Interfaz de Usuario**

El Agente Estacionario Administrador de la Interfaz de Usuario del Vendedor (AEAV) es el encargado de presentar al administrador ó dueño de la tienda las opciones para crear, eliminar ó modificar datos del inventario, ver el estado actual de ventas realizadas y además permitir la recepción ó modificación de ofertas.

#### **4.4.2 Agentes Móviles**

En ARAMCEL, para el Sistema del Vendedor sólo se definió un tipo de agente móvil, el Notificador.

### **Agente Móvil Notificador**

El Agente Móvil Notificador (AMN) es el agente que recibe del AEDV información de ofertas y novedades existentes y la ubicación de los sitios de los compradores, con base en esos datos genera un itinerario y procede a visitar cada uno de ellos para entregarles la notificación.

En la sección 4.5 de este documento se presenta la función de los Agentes del Sistema del Vendedor en las transacciones definidas.

#### **4.4.3. Pseudocódigo de algunos Agentes.**

Este numeral incluye la descripción detallada del funcionamiento de algunos agentes del sistema del vendedor, así como el pseudocódigo que facilita su comprensión y posterior implementación. Los agentes seleccionados fueron el Agente Móvil Notificador y los Agentes Estacionarios Despachador y Vendedor.

##### **4.4.3.1 Agente Móvil Notificador**

El AMN es creado por el AEDV con la información correspondiente a ofertas, novedades y sitios a visitar. Con la información de los sitios, el AMN genera el itinerario y viaja al primer destino en su lista. El AMN visita cada sitio de compradores en su itinerario, si

alguno de ellos no se encuentra disponible, con base en su autonomía decide cuantas veces reintentarlo; en caso de mantenerse la falla reordena el itinerario, para intentarlo como último comprador a visitar, si persiste el problema cancela el sitio y continúa su viaje.

Cuando el AMN llega a cada uno de los sitios, establece comunicación con el Agente AERC, como resultado de esa interacción le envía un mensaje con las ofertas y novedades. Cuando el AMB finaliza el itinerario y regresa a su sitio de origen, entrega los resultados del viaje al AEDV que lo creó. En la Figura 17 se presenta el pseudocódigo del Agente móvil Notificador.

Figura 17. Pseudocódigo del Agente Móvil Notificador.

//Pseudocódigo de la clase para el Agente Movil Notificador del Vendedor

```
public class AgenteMovilNotificador extends AgenteMovil {  
  
    private void creacion() {  
        Establecer ofertas, novedades y sitios  
    }  
  
    private void viajar() {  
        Revisar el siguiente sitio en el itinerario  
        Si (sitio != NULL) {  
            Moverse al siguiente sitio  
            Si (fallas al viajar) {  
                reintentar Moverse al siguiente sitio n veces  
                enviar sitio al final de itinerario y si falla n veces No reintentar y borrar sitio  
            }  
        }  
        sino {  
            Moverse al origen  
            Si (fallas al viajar) {  
                desactivarse(Tiempo x)  
                viajar()  
            }  
        }  
    }  
  
    private void llegar() {  
        Registrarse y revisar sitio actual  
        // Cuando se registra obtiene los identificadores del sitio y del Agente Recepcionista  
        Si (sitioActual == vendedor) {  
            Crear mensaje de entrega de resultados del viaje  
            enviarMensaje(mensaje, id_Agente)  
        }  
        Si (sitioActual == comprador) {  
            Crear mensaje de solicitud de recursos  
            enviarMensaje(mensaje, id_Agente)  
        }  
    }  
  
    private void recibirMensaje(Mensaje mens) {  
        obtener del mensaje los datos del agente  
        Si (agenteEstacionarioRecepcionistaComprador) {  
            Revisar el Mensaje recibido y extraer identificador del Recepcionista  
            Crear mensaje con ofertas y novedades  
            enviarMensaje(mensaje, id_Agente)  
            viajar()  
        }  
    }  
  
    private void enviarMensaje(Mensaje mens, Objeto id_Agente) {  
        Envía el mensaje al agente  
    }  
}
```

Fuente. El autor.

#### **4.4.3.2 Agente Vendedor**

El AEV tiene dos funciones principales en la plataforma responder a las solicitudes de búsqueda de información y realizar procesos de Compra/Venta, en ambos casos el servicio es solicitado por el agente móvil de un comprador.

Cuando llega al Sistema del Vendedor un AMB y se le asigna un AEV, el AMB se comunica con él para obtener los datos de los productos que coincidan con el solicitado. El AEV atiende la solicitud, para esto revisa el mensaje recibido y extrae los datos que ahí se encuentran, con esos datos el AEV le envía un mensaje al CABDV pidiéndole información sobre un artículo. El AEV toma los resultados que le envía el CABDV, los clasifica y forma un objeto llamado Cotización; en la cotización se incluye el precio, tiempo de entrega y la vigencia de la oferta, para indicar el periodo de tiempo durante el cual el sitio mantiene las condiciones de reserva y venta del producto. El AEV le envía la cotización al AMB en un mensaje y guarda los resultados de la transacción enviándole un mensaje al CABDV.

Si llega un AMC y tiene asignado un AEV, el AMC se comunica con ese AEV para obtener datos del artículo. El AEV atiende la solicitud, para esto revisa el mensaje recibido y extrae los datos que ahí se encuentran, con esos datos el AEV le envía un mensaje al CABDV pidiéndole información sobre el artículo. El AEV toma la información que le envía el CABDV y crea un mensaje con los resultados. El AEV le envía el mensaje al AMC, si la transacción continúa el AEV recibe un mensaje con el código de pago. El AEV procede a comunicarse con el CABDV para enviarle el código de pago y solicitarle el código de aceptación. El AEV crea un mensaje con el código de aceptación y se lo envía al AMC, además le solicita al CABDV que almacene la transacción. La Figura 18 muestra el pseudocódigo del Agente Estacionario Vendedor.

Figura 18. Pseudocódigo del Agente Estacionario Vendedor.

```
//Pseudocódigo de la clase para el Agente Estacionario Vendedor
public class AgenteEstacionarioVendedor extends AgenteEstacionario {

    private void recibirMensaje(Mensaje mens) {
        Obtener del mensaje los datos del agente
        Si (agenteMovilBuscador) {
            Revisar el Mensaje recibido y extraer argumentos
            Búsqueda de información sobre el artículo y generar cotización
            Crear mensaje con los datos obtenidos y con la cotización
            enviarMensaje(mensaje, id_Agente)
            Guardar Resultados
        }
        Si (agenteMovilComprador) {
            Revisar el Mensaje recibido y extraer argumentos
            Si (búsqueda) {
                Buscar información sobre el artículo
                Crear mensaje con los datos obtenidos
                enviarMensaje(mensaje, id_Agente)
            }
            Si (compra) {
                Guardar código de Pago
                Buscar información sobre el artículo y generar código de Aceptacion
                Crear mensaje con el código de Aceptacion
                enviarMensaje(mensaje, id_Agente)
                Guardar resultados
            }
        }
    }

    private void enviarMensaje(Mensaje mens, Objeto id_Agente) {
        Envía el mensaje al agente
    }
}
```

Fuente. El autor.

#### 4.4.3.3 Agente Despachador

Para realizar el proceso de notificación de ofertas el AEDV recibe un mensaje AEAV. El AEDV solicita al CABDV la información de ofertas, novedades y compradores registrados. Cuando el CABDV le envía los datos solicitados, el AEDV procede a analizar la cantidad de ofertas, novedades, el tamaño de cada una, así como los sitios de compradores a visitar. Con base en el análisis el AEDV define el número necesario de Agentes Móviles Notificadores.

El AEDV crea cada AMN y le define las ofertas, novedades y sitios a visitar. Aunque existe un AERV, el AEDV es el encargado de procesar los resultados que entrega el AMN cuando llega de finalizar su viaje. Con esa información que recibe, el AEDV decide las acciones necesarias para solucionar las fallas o inconvenientes que se presentaron en el viaje; una de ellas es una nueva definición del agrupamiento de las ofertas, novedades y compradores donde hubo problemas, así como la creación de los agentes necesarios. En la Figura 19 se presenta el pseudocódigo del Agente Estacionario Despachador del Vendedor.

Figura 19. Pseudocódigo del Agente Estacionario Despachador.

```
//Pseudocódigo de la clase para el Agente Estacionario Despachador del Vendedor
public class AgenteEstacionarioDespachador extends AgenteEstacionario {

    private void recibirMensaje(Mensaje mens) {
        Obtener del mensaje los datos del agente
        Si (agenteEstacionarioAdministrador) {
            Revisar el Mensaje recibido y extraer argumentos
            Si (notificacion) {
                Buscar información de ofertas, novedades y Compradores. Realizar análisis
                Con base en el análisis, definir la cantidad de agentes a crear
                Para (cantidad) {
                    crearAgente()
                }
            }
        }
    }
    Si (agenteMovilNotificador) {
        Revisar el Mensaje recibido y extraer datos del viaje
        Si (fallasViaje) {
            tomar las acciones para solucionar las fallas
            // una acción puede ser crear otro agente
        }
    }
}

private void crearAgente() {
    Crear Agente Móvil Notificador
    Inicializar el agente
}

private void enviarMensaje(Mensaje mens, Objeto id_Agente) {
    Envía el mensaje al agente
}
}
```

Fuente. El autor.

## **4.5. TRANSACCIONES DE COMERCIO ELECTRÓNICO EN ARAMCEL**

En ARAMCEL, tanto Compradores como Vendedores tienen acceso a la realización de transacciones de comercio electrónico utilizando los Agentes como sus representantes dentro del sistema. En las anteriores secciones de este capítulo, se presentaron los Agentes y Componentes incluidos tanto en el Sistema del Comprador, como el Sistema del Vendedor. Esta sección incluye la descripción detallada del proceso de cada una de las transacciones definidas en ARAMCEL, así como la interacción de esos Componentes y Agentes descritos. Para mostrar este comportamiento se incluye el diagrama de secuencia<sup>13</sup> de cada proceso.

### **4.5.1. Búsqueda de información.**

La Búsqueda de Información es una funcionalidad definida por ARAMCEL para el UC, el proceso de esta transacción se muestra en el diagrama de secuencia que se presenta en las Figuras 20, 21 y 22; la descripción de cada uno de los pasos se incluye a continuación:

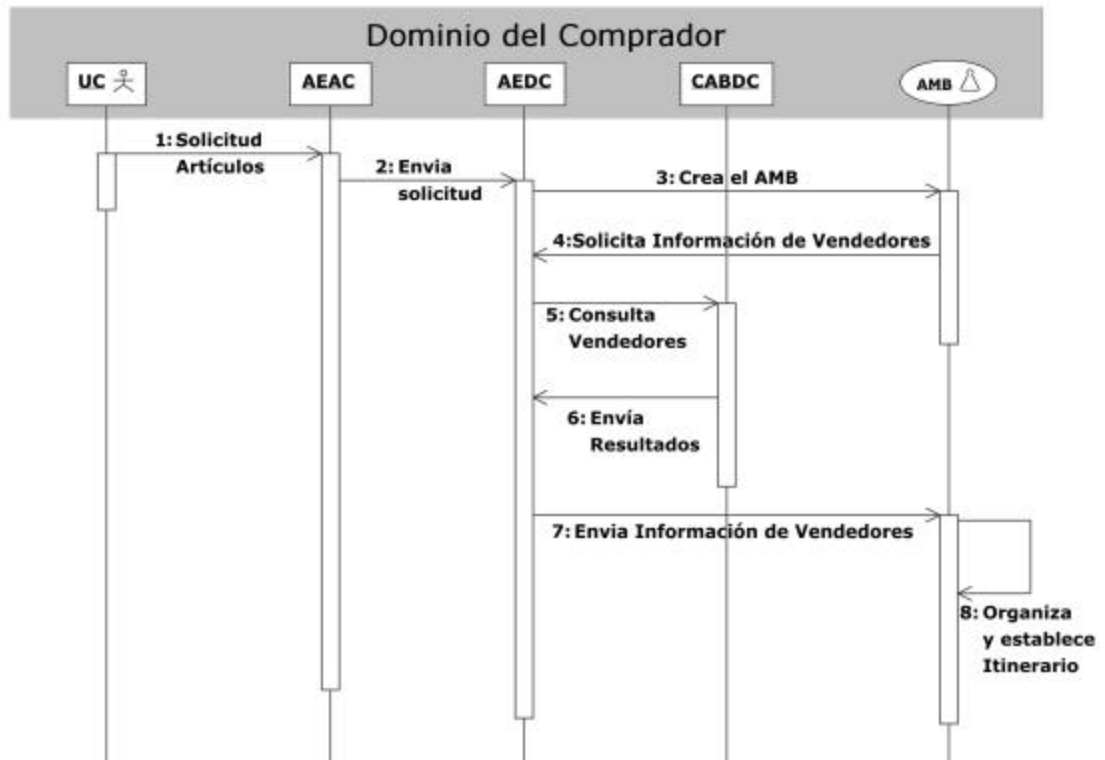
1. El UC hace una solicitud de artículos a través de la Interfaz Gráfica de Usuario (GUI).
2. El AEAC procesa la petición y le entrega al AEDC los datos de los artículos y las preferencias suministradas por el usuario.
3. El AEDC con esos datos crea un AMB por cada artículo de la lista, con las respectivas preferencias.
4. El AMB solicita al AEDC información de sitios de Vendedores que ofrezcan el tipo de artículo que necesita.
5. El AEDC consulta al CABDC la información de los potenciales vendedores y el puntaje asignado a cada uno de ellos.
6. El CABDC envía la información al AEDC, quien la procesa y la empaqueta en un mensaje.
7. El AEDC envía los datos solicitados al AMB.

---

<sup>13</sup> El diagrama de secuencia de cada transacción se ha dividido en tres imágenes para facilitar la comprensión, tomando en cuenta su tamaño.

8. El AMB recibe la información y establece su itinerario; esto se realiza evaluando los sitios por medio del puntaje que los clasifica con base en la calidad del servicio prestado.

Figura 20. Diagrama de Secuencia Búsqueda de Información (Parte 1).

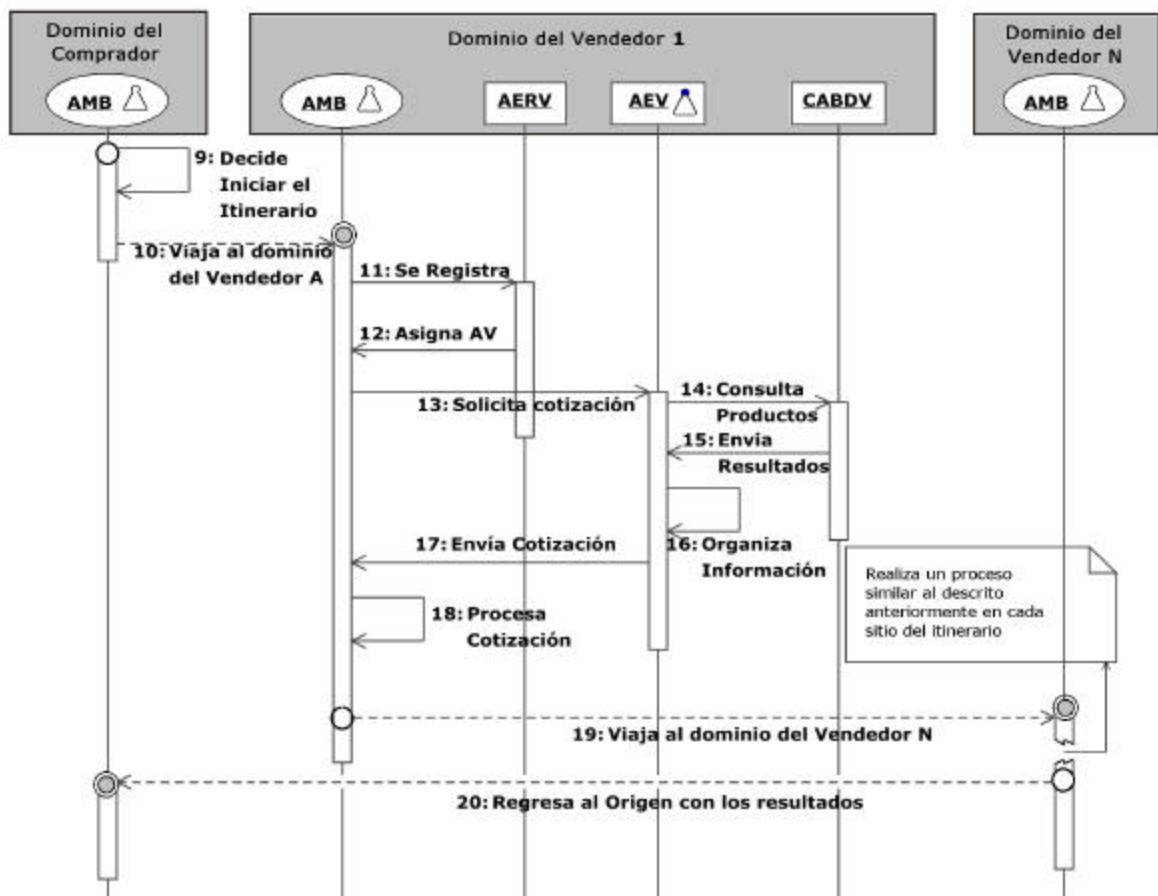


Fuente. El autor.

9. Cuando el AMB ha completado la información necesaria, toma la decisión de iniciar su itinerario.
10. El AMB se desplaza al dominio del primer vendedor (Vendedor 1).
11. Al llegar al sitio del Vendedor el AMB establece comunicación con el AERV y se registra.
12. El AERV le asigna un AEV para que interactúe con el AMB.
13. El AMB solicita el artículo al AEV y espera la cotización que incluye precio, cantidad y tiempo de entrega del artículo.
14. El AEV consulta al CABDV la información sobre los productos ofrecidos que coinciden con las características del solicitado por el AMB.

15. El CABDV envía los resultados de la consulta en la Base de Datos (BD).
16. El AEV organiza la información enviada y establece qué artículos coinciden con las palabras clave enviadas por el AMB. Los datos resultantes son organizados dentro de un mensaje denominado cotización que incluye, además de los datos de los productos, el precio, tiempo de entrega, cantidad disponible y tiempo de vigencia.
17. El AEV envía el mensaje con la cotización al AMB.
18. El AMB procesa la oferta recibida; si cumple con las preferencias del UC, la almacena. Con base en el servicio y la calidad de los resultados ofrecidos, el AMB almacena la información del servicio en la tienda. El AMB se prepara para continuar su recorrido.
19. El AMB viaja al dominio del siguiente vendedor y realiza un proceso similar al descrito en los pasos 11 al 18. Cuando finaliza el itinerario se prepara para regresar al origen.
20. El AMB regresa al origen con los resultados de la búsqueda de información.

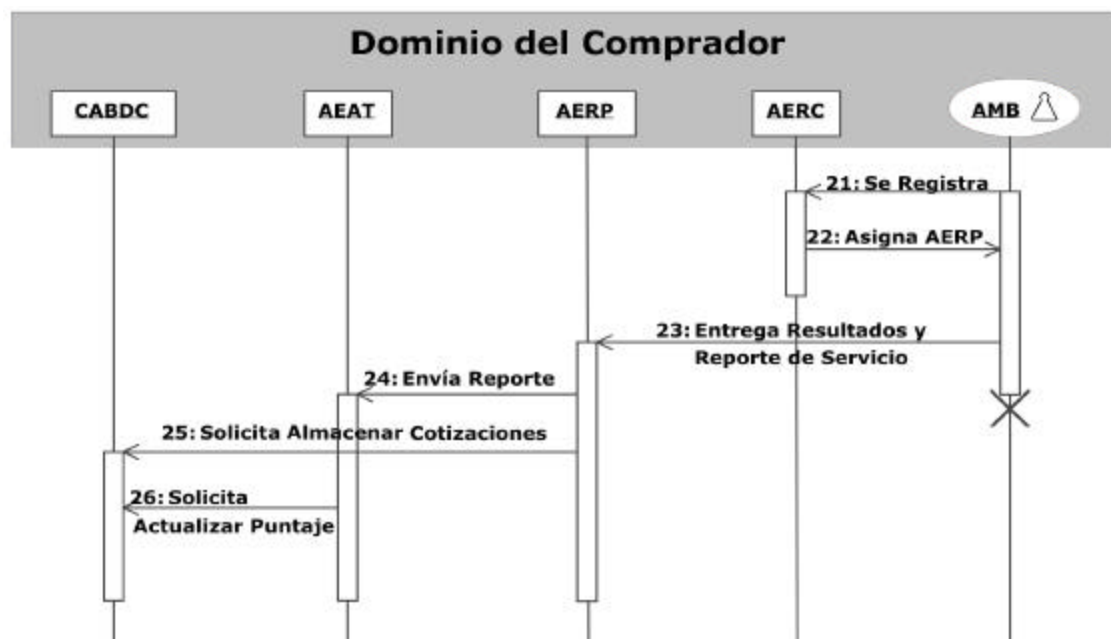
Figura 21. Diagrama de Secuencia Búsqueda de Información (Parte 2).



Fuente. El autor.

21. Cuando el AMB llega al sitio del comprador, se registra con el AERC.
22. Una vez que el AERC lo ha identificado como un agente de la plataforma, lo comunica con el AERP.
23. El AMB reporta al AERP los resultados obtenidos en su viaje; estos incluyen los datos de la búsqueda de información y el reporte de servicio.
24. El AERP envía el reporte de servicio al AEAT.
25. El AERP procesa la información recibida sobre la búsqueda de información y solicita al CABDC que la almacene. Esta información será entregada al UC en el momento que la solicite.
26. El AEAT establece con base en su autonomía el incremento ó decremento en los puntajes de cada una de las tiendas y solicita el almacenamiento de esta información al CABDC.

Figura 22. Diagrama de Secuencia Búsqueda de Información (Parte 3).



Fuente. El autor.

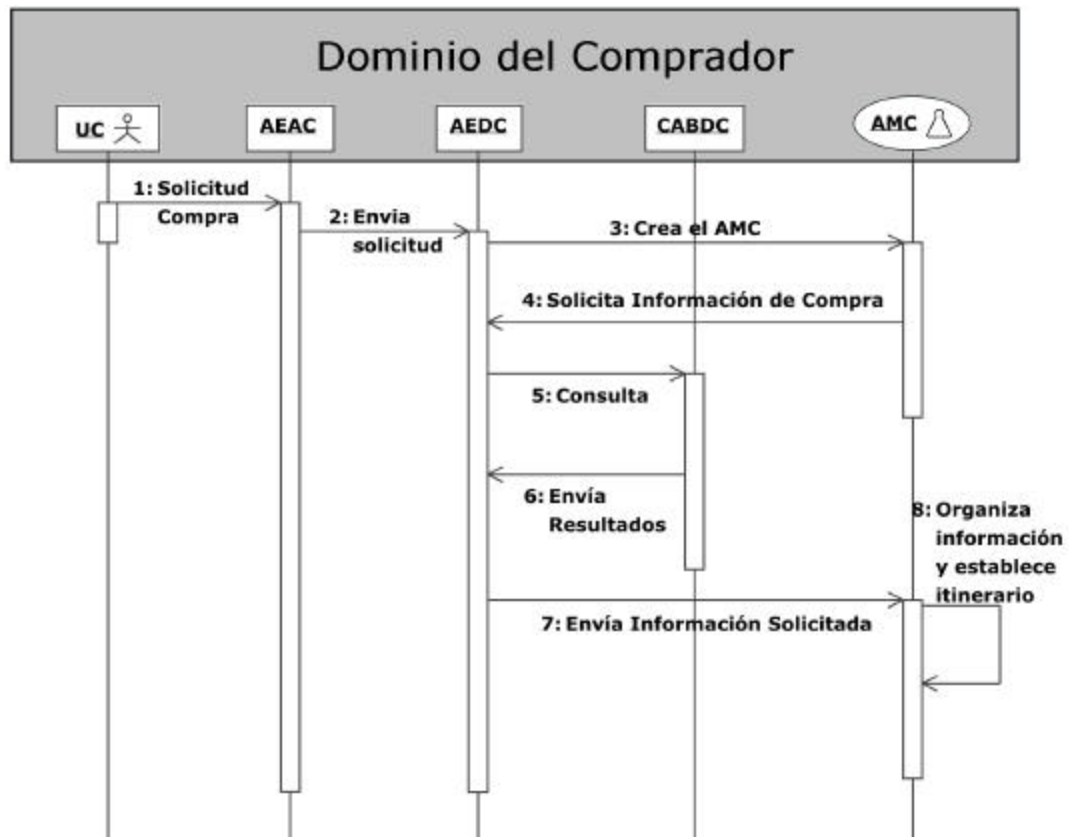
#### 4.5.2. Compra de Artículos

La Compra de Artículos es una funcionalidad definida por ARAMCEL para el UC, en la cual tiene la posibilidad de enviar a un agente para que lo represente en un proceso de

compra. El proceso de esta transacción se muestra en el diagrama de secuencia que se presenta en las Figuras 23, 24 y 25; la descripción de cada uno de los pasos se incluye a continuación:

1. El UC a través de la GUI revisa los resultados de la búsqueda de artículos y decide cuales comprar.
2. El AEAC toma la información de los artículos que desea comprar el UC y la envía al AEDC.
3. El AEDC crea un AMC por cada artículo.
4. El AMC solicita al AEDC información del sitio del Vendedor que ofrece el artículo que necesita, la cotización, los datos de entrega y el código de pago.
5. El AEDC consulta al CABDC la información que solicitó el AMC.
6. El CABDC busca en la BD y envía los resultados de la consulta.

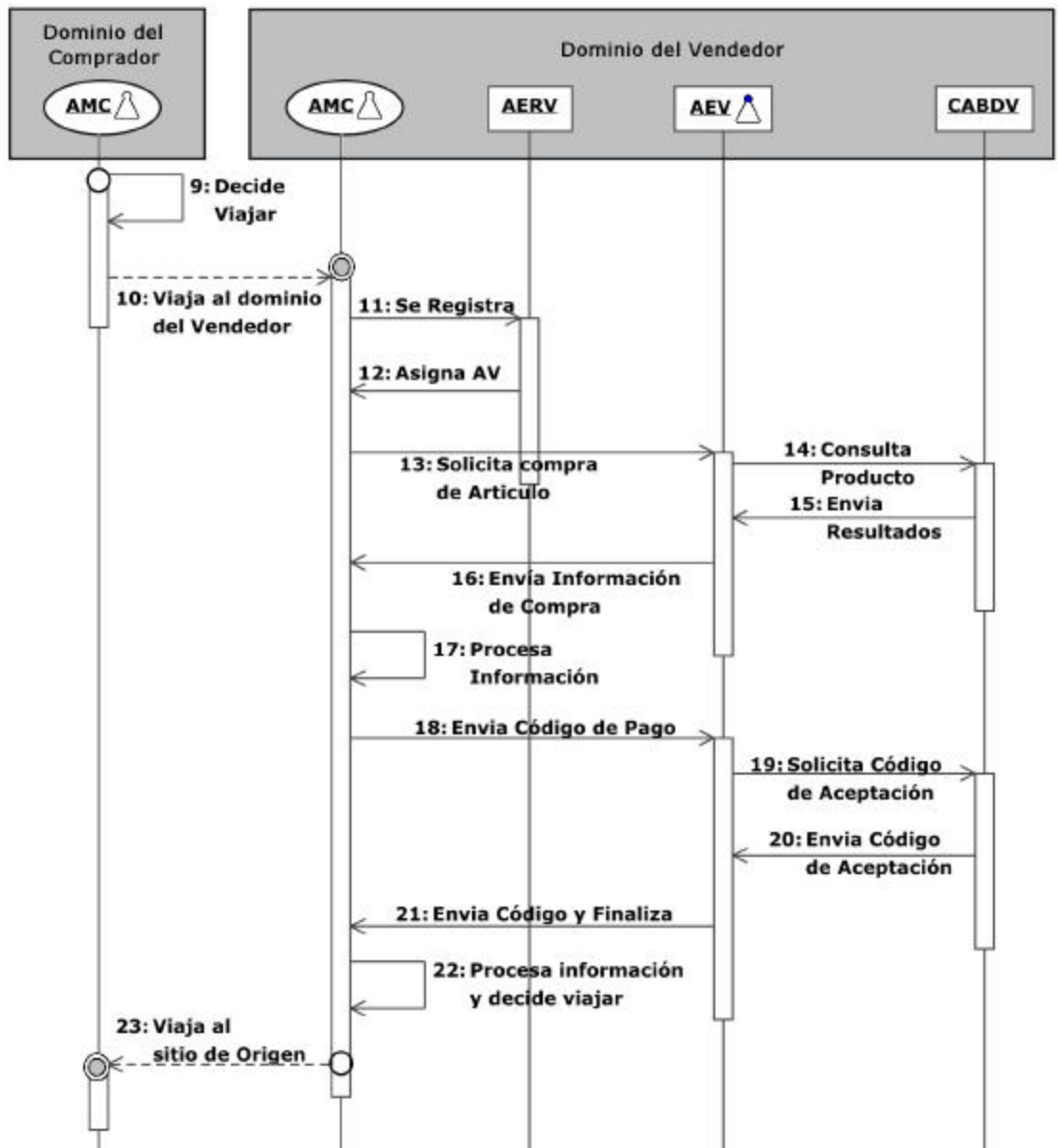
Figura 23. Diagrama de Secuencia Compra de Artículos (Parte 1).



Fuente. El autor.

7. El AEDC procesa la información recibida y la empaqueta en un mensaje; el mensaje es enviado al AMC.
8. El AMC organiza la información que recibe del AEDC, establece su itinerario y se prepara para viajar al sitio del vendedor.
9. El AMC con base en su autonomía, toma la decisión de iniciar su itinerario.
10. El AMC viaja al sitio donde se encuentra el artículo seleccionado por el UC.
11. El AMC llega y establece comunicación con el AERV; se registra.
12. El AERV procesa el registro y le asigna un AEV.
13. El AMC envía la solicitud de compra al AEV.
14. El AEV procesa la solicitud hecha por el AMC, y solicita la información necesaria al CABDV para ordenar la venta.
15. El CABDV consulta en la BD los datos solicitados y los envía al AEV.
16. El AEV organiza los datos recibidos y los envía dentro de un mensaje al AMC.
17. El AMC organiza y procesa la información. Si coincide ó mejora la oferta entregada en la cotización continúa la transacción, de lo contrario la aborta.
18. Si continúa la transacción, el AMC envía el código de pago y los datos de entrega.
19. El AEV recibe el código de pago y solicita al CABDV la generación del Código de Aceptación para la orden de venta realizada.
20. El CABDV almacena la transacción, genera el código de aceptación y lo envía al AEV.
21. El AEV entrega el código de aceptación al AMC y finaliza la transacción.
22. El AMC almacena el código de aceptación. Si la transacción fue abortada, los pasos 18 al 21 no se realizan, pero en ambos casos el AMC almacena la información del servicio recibido. El AMC se prepara para viajar y decide regresar al sitio de origen.
23. El AMC se desplaza al sitio del UC.
24. Cuando el AMC llega a su sitio de origen se registra con el AERC.
25. El AERC lo identifica como un agente de la plataforma y lo comunica con el AERP.
26. El AMC entrega los resultados del proceso de orden de compra y el reporte de servicio.
27. El AERP envía el reporte de servicio al AEAT.
28. La información de la transacción es procesada y enviada para su correspondiente almacenamiento por medio del CABDC.

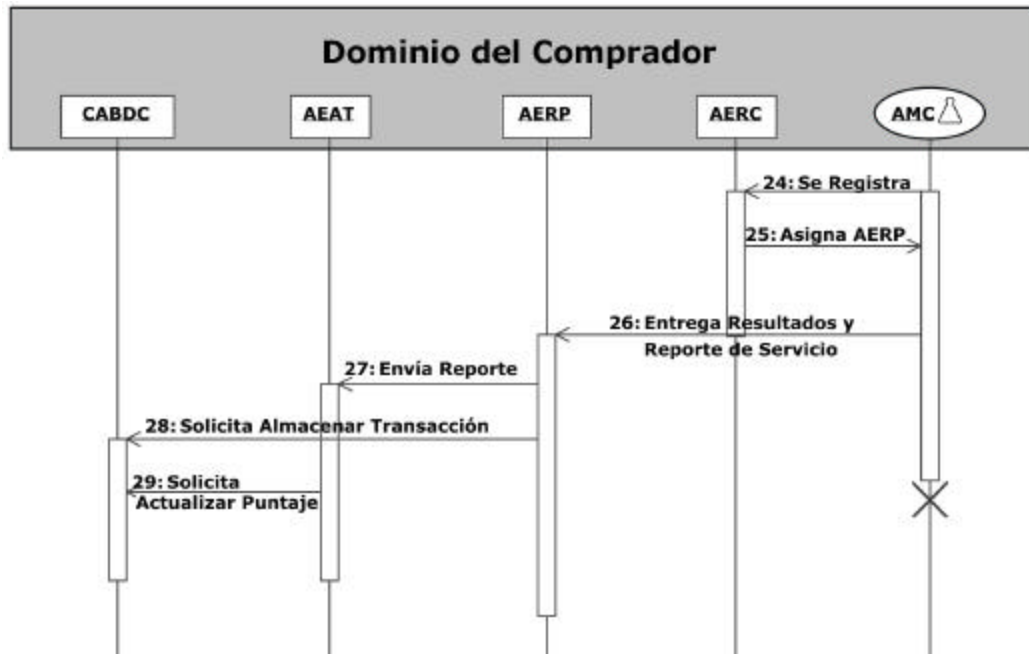
Figura 24. Diagrama de Secuencia Compra de Artículos (Parte 2).



Fuente. El autor.

29. El AEAT asigna el incremento ó decremento del puntaje al vendedor con base en el reporte de servicio entregado por el AMC. Posteriormente solicita al CABDC la actualización del puntaje en la BD.

Figura 25. Diagrama de Secuencia Compra de Artículos (Parte 3).



Fuente. El autor.

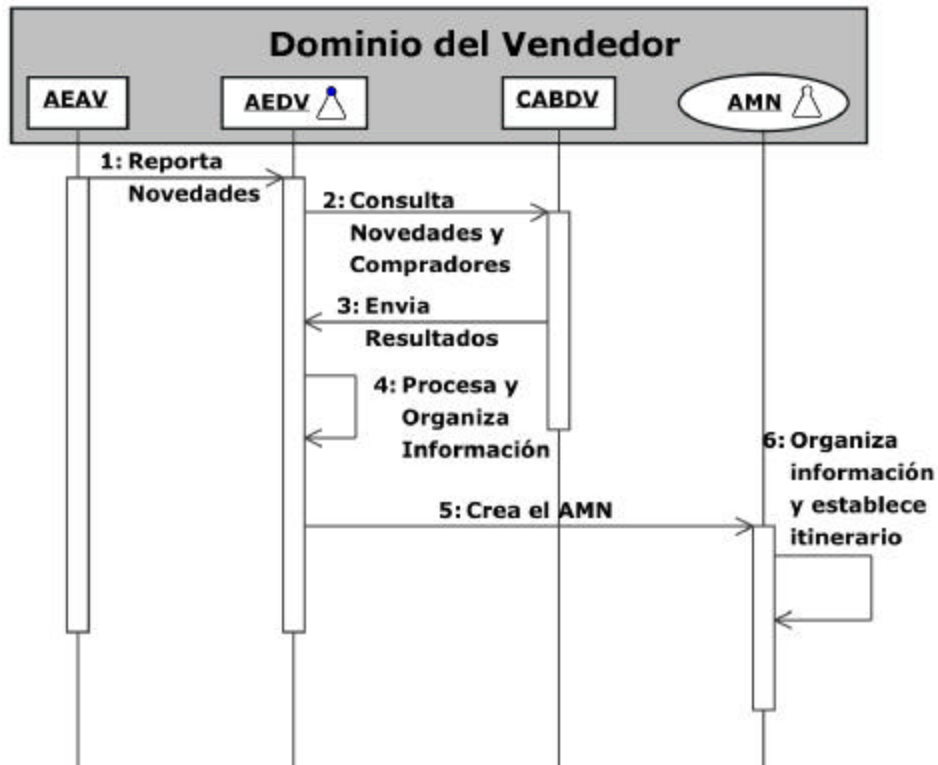
#### 4.5.3. Notificación de Ofertas

La Notificación de Ofertas es una funcionalidad definida en ARAMCEL para el Usuario Vendedor (UV); mediante la notificación se envían mensajes a los Compradores, en los cuales les avisa de la presentación de ofertas, promociones ó novedades en la tienda. Esta transacción busca incrementar las posibilidades de acceso a clientes; su proceso se muestra en el diagrama de secuencia que se presenta en las Figuras 26, 27 y 28; la descripción de cada uno de los pasos se incluye a continuación:

1. Cuando el Usuario Vendedor (UV) registra en el sistema nuevos artículos ó ingresa información de ofertas, el AEAUV le informa sobre este evento al AEDV.
2. El AEDV consulta el CABDV las novedades u ofertas registradas, así como la información de los compradores con sus respectivos perfiles.
3. El CABDV envía los resultados de la consulta al AEDV.
4. Los datos recibidos, son clasificados por el AEDV, quien con base en su autonomía, determina los compradores y mensajes que debe llevar cada AMN, así como la cantidad de agentes a crear.

5. El AEDV crea el AMN, asignándole la información de los compradores y el mensaje ó mensajes que debe entregar.
6. El AMN organiza la información recibida y establece su itinerario. Se prepara para viajar.

Figura 26. Diagrama de Secuencia Notificación de Ofertas y Novedades (Parte 1).

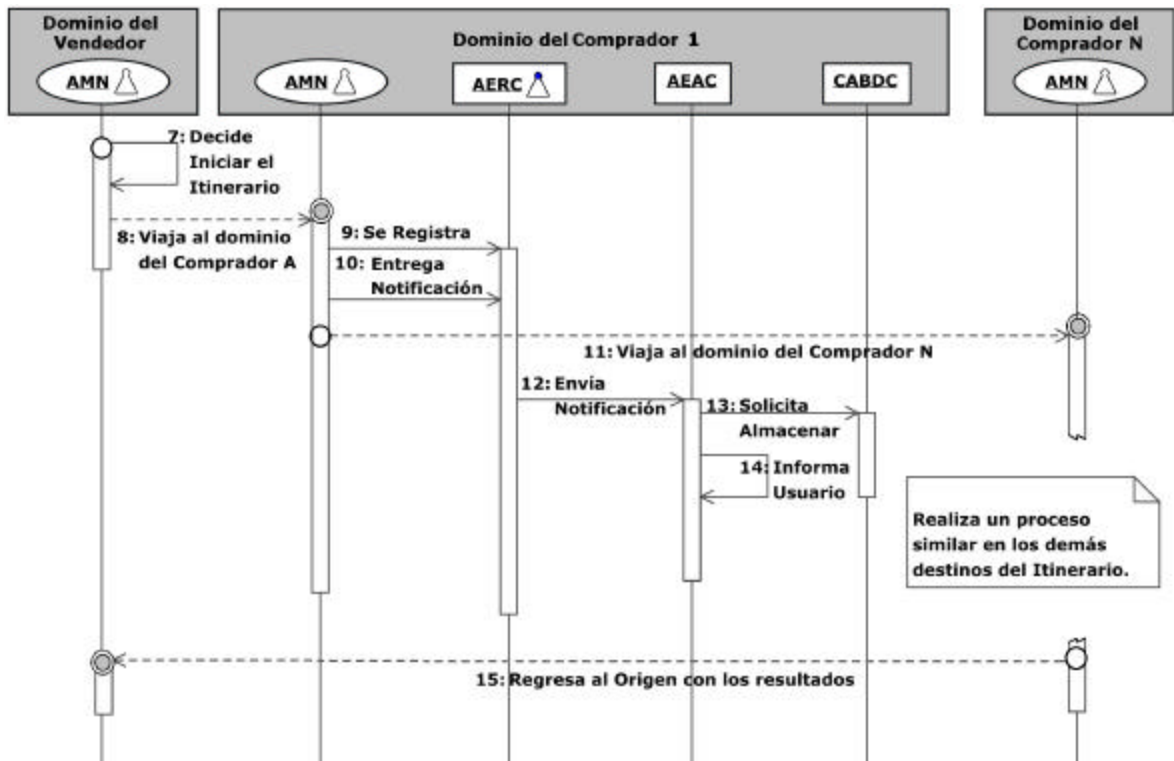


Fuente. El autor.

7. El AMN decide iniciar su itinerario.
8. El AMN viaja al dominio del primer comprador (Comprador 1).
9. Cuando el AMN llega al sitio del comprador, se registra con el AERC.
10. Después de procesar el registro, el AMN le entrega la notificación al AERC.
11. Cuando el AMN recibe el mensaje de confirmación, decide continuar su itinerario y viaja al dominio del siguiente comprador. El proceso realizado en los pasos 9 al 11 se repite en cada sitio del itinerario.
12. Una vez el AERC ha confirmado la recepción del mensaje, lo envía al AEAC.

13. El AEAC solicita el almacenamiento del mensaje para posteriormente entregarlo al UC<sup>14</sup>.
14. El AEAC le entrega la información al UC a través de la GUI.
15. Cuando el AMN ha terminado su itinerario, toma la decisión de regresar al Sitio del Vendedor y viaja.

Figura 27. Diagrama de Secuencia Notificación de Ofertas y Novedades (Parte 2).



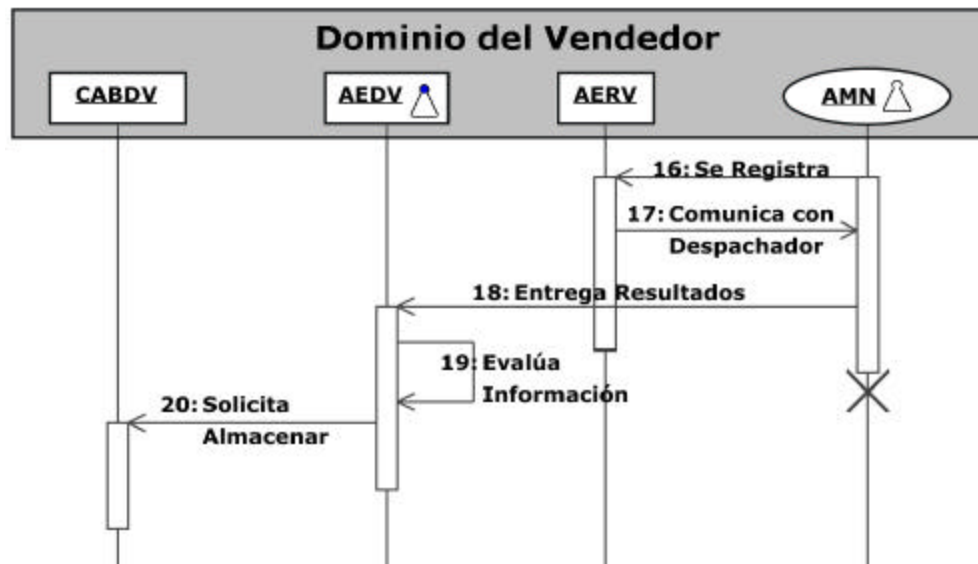
Fuente. El autor.

16. Cuando el AMN regresa al Sitio del Vendedor, se registra con el AERV.
17. Cuando el AERV lo ha identificado como un agente de la plataforma, lo comunica con el AEDV.
18. El AMN envía un mensaje al AEDV con los resultados de su trabajo de notificación.

<sup>14</sup> Este paso se realiza para usuarios que no tienen instalado el Sistema del Comprador en sus equipos y que acceden a él a través del Servidor de Compradores, puesto que sólo si han iniciado la sesión en el sistema se realizaría directamente el paso 14.

19. El AEDV evalúa la información entregada por el AMN y toma decisiones con base en la información entregada; algunas de esas decisiones son sitios que deben volverse a visitar y notificaciones pendientes.
20. El AEDV solicita al CABDV el almacenamiento de la información reportada por el AMN y la generada con base en ella.

Figura 28. Diagrama de Secuencia Notificación de Ofertas y Novedades (Parte 3).



Fuente. El autor.

#### 4.5.4. Transacciones a través del Servidor de Compradores

En ARAMCEL se ha definido un sitio denominado Servidor de Compradores, en él, se encuentra instalado el Sistema del Comprador. Para este efecto, el UC ejecuta la Interfaz Web<sup>15</sup> que administra el AEAC ubicado en el Servidor de Compradores y a través de ella accede a las diferentes opciones que ofrece ARAMCEL.

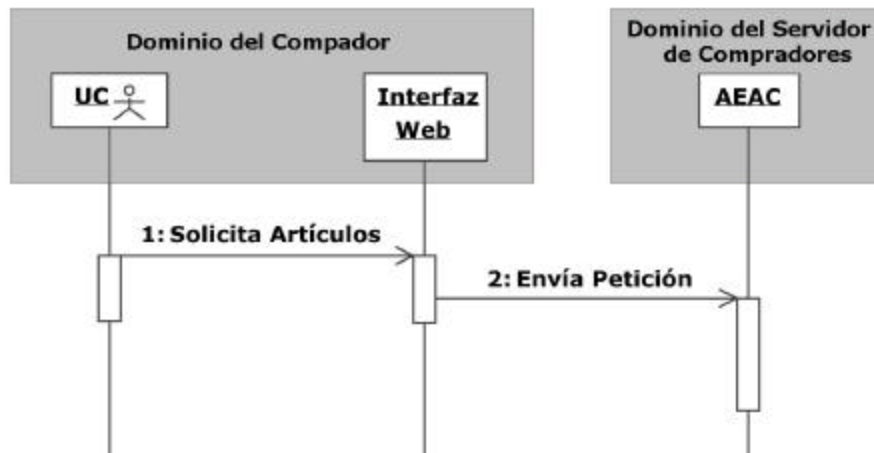
Con base en esta información, las transacciones descritas en los numerales anteriores se efectúan de la misma forma, pero cambia la distribución de dominios. La Figura 29 presenta un diagrama que muestra esta división, en él se ha incluido la clase denominada

<sup>15</sup> Para la implementación de esta interfaz puede utilizarse tecnologías como los Applets ó los Servlets definidos por el Lenguaje de Programación JAVA [SUN04a].

“Interfaz Web”, para la interacción del Usuario con el sistema. Los pasos son los siguientes:

1. El UC a través de la Interfaz Web (IW) ingresa los datos de los artículos que desea buscar ó selecciona aquellos que va a comprar.
2. La IW envía la información al AEAC.

Figura 29. Interacción UC – AEAC en Terminales Limitadas.



Fuente. El autor.

Una vez se ha realizado este proceso, las transacciones de búsqueda de información ó compra de artículos continúan a partir del paso 2 como está descrito en los numerales 4.5.1 y 4.5.2 de este documento.

Estas transacciones corresponden a un escenario sin inconvenientes de seguridad. En el siguiente capítulo se presentan las amenazas de seguridad que se pueden manifestar al realizar las transacciones de comercio electrónico definidas y se propone el Modelo de Seguridad de ARAMCEL.

## **5. SEGURIDAD EN ARAMCEL**

En este capítulo se presenta la seguridad en ARAMCEL, inicia con las amenazas inherentes al Modelo de Agentes Móviles y se describen los principales problemas del comercio electrónico. Una vez realizada esta descripción se presenta el Modelo de Seguridad con la respectiva solución a los problemas planteados como críticos en esta investigación.

### **5.1. AMENAZAS DE SEGURIDAD EN ARAMCEL**

La movilidad de agentes y la interacción con otras entidades en la red, hacen surgir unos requerimientos de seguridad en ARAMCEL que pueden ser clasificados en dos categorías. La primera de ellas se refiere a los problemas de seguridad que se presentan en el Modelo de Agentes Móviles y la segunda los aspectos que se requieren para garantizar la seguridad en un entorno de comercio electrónico. Los siguientes numerales detallan cada una de las categorías mencionadas.

#### **5.1.1. Amenazas de Seguridad Inherentes al Modelo de Agentes Móviles.**

La seguridad ha sido tradicionalmente un inconveniente en los sistemas informáticos, pero el modelo de Agentes Móviles por sus características presenta un nuevo tipo de inconvenientes de seguridad. Los dos problemas principales son [MAN01, VIGNA98, ZHAO00]:

##### **5.1.1.1 El Problema del Agente hostil.**

Cuando los Agentes Móviles de ARAMCEL llegan a cada sitio de su itinerario, necesitan algunos permisos de usuario para poder ejecutar su código. Tradicionalmente el control de acceso a los equipos de una red se hace por medio de contraseñas, en donde cada

usuario se autentica al inicio con su *login* y su *password*; pero este esquema funciona para sistemas estáticos y no para Agentes Móviles, puesto que el agente debe llevar una contraseña para cada sitio de su itinerario, lo que aumenta su tamaño y así mismo incrementa los requerimientos de seguridad para poder proteger esas contraseñas.

Entre los agentes puede existir agentes hostiles que realizan acciones perjudiciales, como el acceso no autorizado, alteración de los recursos locales (datos y llamadas al sistema), sobrecarga de esos recursos o incluso daños a la integridad del servidor. En la Figura 30 se presenta un agente hostil que llega a un sitio y ha sido diseñado para realizar labores que atentan contra la integridad del servidor, como la utilización no autorizada de los recursos para obtener información privada.

Figura 30. Agente Hostil.



Fuente. El autor.

El problema del agente hostil ha generado tres tipos de amenazas a los servidores, las cuales se presentan a continuación:

- *El acceso no autorizado a los recursos.* Cuando un agente móvil llega a cada sitio de su itinerario, ejecuta su código usando los recursos del computador del Vendedor. Por esta razón, los recursos de la máquina receptora están expuestos al ataque de un agente hostil, puesto que si no existen los mecanismos de control apropiados, ese

agente podría acceder a información privada del sistema. En el caso del comercio electrónico, la información que almacenan los Vendedores está relacionada con sus datos comerciales, de esa forma, si un agente accede a ella puede alterarla y tomar ventaja de su contenido para fines ilegales.

- *Sobrecarga de los recursos del servidor.* Es necesario proveer medidas de seguridad adecuadas para prevenir el consumo excesivo de recursos por parte de agentes hostiles, puesto que al sobrecargar el servidor hace que el tiempo de respuesta se incremente o que no haya respuesta a los agentes de los demás Compradores que buscan acceder a la información. En el caso particular de ARAMCEL los sitios de los vendedores son clasificados con base en su servicio mediante un puntaje, por lo tanto el problema mencionado ocasionaría la reducción de esa valoración.
- *Creación de agentes residentes en el sitio del Vendedor.* Otro de los ataques de los cuales puede ser víctima un servidor por parte de un agente hostil, es la creación de otros agentes que residan en el servidor. Con esos nuevos agentes puede suceder cualquiera de los dos ataques anteriormente mencionados, ó que se creen tantos agentes que también sobrecarguen los recursos. En la mayor parte de los sistemas de comercio electrónico actuales se utiliza un agente "esclavo", creado por el agente móvil, para que resida en el sitio del Vendedor y asegure la reserva de un producto; en ARAMCEL este proceso es diferente y como se mencionó en el capítulo anterior se utiliza un objeto llamado Cotización.

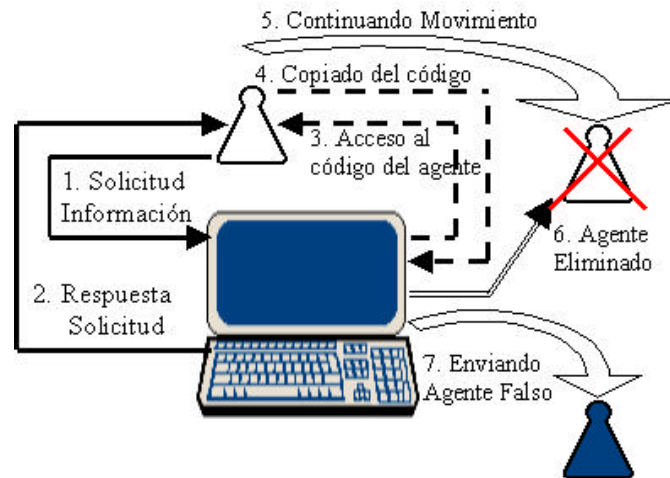
La solución tradicional al problema del agente hostil se realiza por medio de restricción de privilegios a los agentes, pero eso puede ocasionar que los agentes sean incapaces de lograr sus metas, por lo tanto es necesario buscar mecanismos alternos que provean seguridad y permitan la realización de las transacciones.

#### **5.1.1.2 El problema del servidor hostil**

Un servidor hostil es una máquina que intenta espiar ó manipular el código, los datos o el control de flujo del agente, proporciona llamadas falsas al sistema, ejecuta código del agente incorrectamente, realiza ingeniería inversa y manipula su código ó los secretos comerciales; puede hacerlo mediante la clonación del agente, proporcionando información

errónea ó interceptando el agente. Un ejemplo del ataque de un servidor hostil a un agente móvil puede observarse en la Figura 31.

Figura 31. Servidor Hostil.



Fuente. El autor.

Cuando un agente móvil llega a un determinado sitio, ya no depende de su máquina de origen, por el contrario, la máquina receptora debe instanciar la clase que dio origen al agente, leer el estado de datos que se encuentra serializado y ejecutarlo dentro de ella, por lo cual el código y datos del agente móvil quedan totalmente expuestos. Los ataques de Servidores contra agentes son de diversos tipos y pueden resumirse como sigue:

- *Clonación del Agente.* Uno de los ataques al agente móvil consiste en que un servidor captura y construye una réplica de él, de esa manera, se hace pasar por el agente original para los propósitos que ese servidor programa. En el caso del comercio electrónico un Vendedor puede clonar un agente para enviarlo a buscar información privada y de esa forma acceder a los secretos comerciales de sus competidores.
- *Negación de servicio.* Los ataques de negación del servicio pueden ser de dos tipos, el primero ocurre cuando el servidor impide que el agente continúe su itinerario, el segundo consiste en permitir al agente continuar, pero no al sitio que tiene determinado en el itinerario, sino a aquel que el servidor elige. Este segundo tipo de ataque es frecuente en los sistemas de comercio electrónico, puesto que evita que el

agente vaya al sitio del competidor y puede enviarlo al de un socio ó a otra sucursal del mismo Vendedor.

- *El ataque del hombre en el medio.* Un agente móvil puede ser interceptado y modificado mientras se desplaza de un servidor a otro. La forma de realizarlo es utilizando un programa denominado "monitor de comunicación" que se encarga de observar los mensajes intercambiados por los sistemas de agentes para extraer o modificar la información o el código de los agentes. Con este tipo de ataque un Vendedor hostil puede espiar o alterar los datos que los demás Vendedores han entregado al agente.
- *Modificación de los datos que lleva el agente.* Cuando un servidor ejecuta un agente móvil puede observar su contenido, por esta razón, si no se establecen los mecanismos de seguridad necesarios, un Vendedor hostil puede modificar los datos que le han sido entregados por otros Vendedores al agente y de esa forma competir deslealmente.

### **5.1.2. Principales problemas del comercio electrónico**

Con el auge de Internet y las comunicaciones aparecen avances en el desarrollo tecnológico que buscan aumentar la competitividad empresarial; en el caso del comercio electrónico, el modelo de agentes móviles ofrece grandes oportunidades para realizar transacciones. Así mismo se encuentran en el mercado sistemas basados en agentes móviles, se cuenta con estándares, plataformas y productos que facilitan su desarrollo, pero aún el comercio electrónico no ha llegado al auge esperado; una de las razones para que esto suceda son los problemas de seguridad existentes [DAVILA00]. Por eso se hace necesario buscar soluciones para minimizar el riesgo al realizar transacciones de comercio electrónico y la manera de lograrlo es implementar los siguientes aspectos de seguridad [TSUJI00]:

#### **5.1.2.1 Autenticación.**

Asegura que el escritor de un documento o el remitente de un mensaje es quien dice ser. Para lograr esta autenticación están surgiendo métodos de gran importancia que

conducen a un manejo efectivo de la seguridad y la confidencialidad. Algunos de los métodos utilizados son: Identificación Biométrica, Información Confidencial y Documentos de Identificación.

En la autenticación mediante información confidencial se utiliza una combinación de caracteres como contraseña, donde sólo el usuario y el proveedor de ella pueden conocerla, o se realiza mediante la utilización de una firma digital. Una firma digital es una identificación electrónica, creada por medio de un computador para representar a la persona o entidad, con el mismo efecto legal que tendría una firma hecha a mano.

#### **5.1.2.2 Autorización.**

Asegura que el usuario que va acceder a la información posee los permisos de seguridad necesarios, por tanto la información en la red permanece privada. Se puede cumplir usando métodos matemáticos complejos para realizar procesos como el cifrado y descifrado.

#### **5.1.2.3 No-repudio.**

Asegura que ninguno de los implicados en la transacción puede negar su participación en ella, así mismo, los participantes no podrán rechazar las obligaciones contractuales adquiridas en el negocio; el creador del mensaje no puede negar que lo envió, y el receptor del mensaje no puede negar que lo recibió. Este aspecto es útil principalmente por razones comerciales y legales. Hay dos clases de No-Repudio:

- De origen: Provee al receptor del mensaje con la prueba de su origen, la cual podrá emplearse para defenderse de la negación del emisor sobre el envío del mensaje. Actualmente se está empleando la firma digital para evitar el no-repudio del emisor, la firma es anexada al mensaje enviado.
- De recepción: Provee al emisor del mensaje con la prueba de su entrega, la cual podrá ser empleada para defenderse de la negación del receptor. Para evitar el no-repudio del receptor actualmente se está optando por incluir otra entidad, es decir, una tercera parte confiable, que en la mayoría de los casos es una autoridad de

certificación, que actúa como "notario" dentro de la transacción y verifica la recepción del mensaje.

#### **5.1.2.4 Confidencialidad.**

Asegura que ninguna entidad ajena a la transacción pueda acceder a los datos que utilizan, de la misma manera los actores (personas, programas, procesos, etc.) de esa transacción tan sólo conocerán aquella información necesaria para su realización.

#### **5.1.2.5 Integridad.**

Asegura que un mensaje no ha sido modificado mientras es transportado. En comercio electrónico, el contenido de una transacción no debe ser alterado por alguna de las partes implicadas en ella, ni por una tercera parte no involucrada inicialmente; así mismo, la transacción debe poder ser perfectamente reconstruible frente a terceros en caso de disputas legales. Una forma para lograrlo es mediante el uso de firmas digitales con criptografía de llave pública, en las cuales es necesario registrar la llave pública antes de la transacción.

## **5.2. MODELO DE SEGURIDAD DE ARAMCEL**

Dada la amplitud del problema de seguridad, en ARAMCEL se seleccionaron los aspectos de seguridad autenticación, autorización y no-repudio como los más críticos para el éxito de las transacciones. Esos problemas de seguridad se pueden resolver con la ayuda de técnicas para el cifrado de información. Hay dos esquemas para realizar ese cifrado:

- *Esquema común:* En este esquema la misma llave es guardada secretamente, se usa para el cifrado y descifrado de la información, se le llama cifrado simétrico. En este caso el autor del mensaje debe utilizar una clave para cifrarlo y el destinatario debe poseer esa misma clave para poder descifrarlo.
- *Esquema de llave pública:* En este esquema se usan llaves diferentes para el cifrado y descifrado, se denomina cifrado asimétrico, una llave se guarda confidencialmente y la otra es pública. El término utilizado para referirse a la infraestructura de seguridad

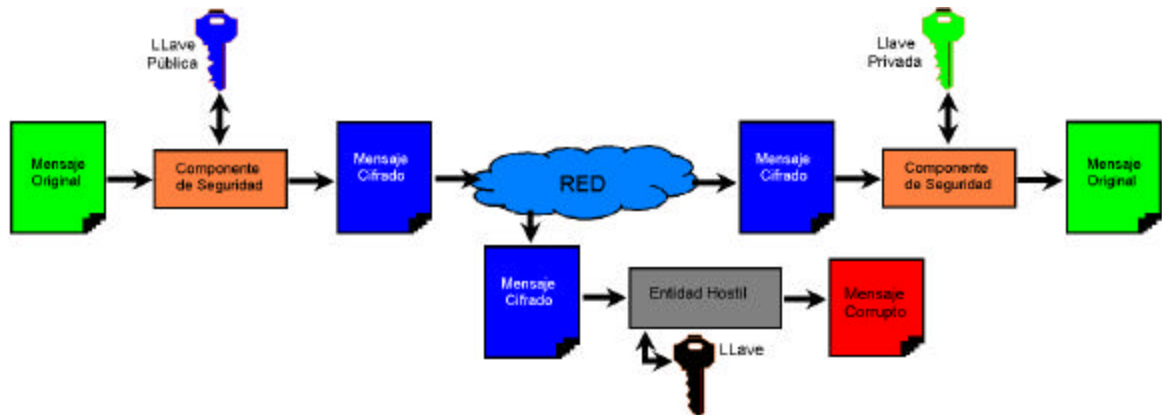
creada con base en criptografía de clave o llave pública es PKI (Public Key Infrastructure) [KABAY04, SMART04]. PKI también permite la gestión de certificados digitales, donde un certificado es un documento firmado digitalmente por una persona o entidad confiable denominada Autoridad de Certificación (CA), que enlaza cierta información perteneciente a un sujeto con su llave pública para poder asegurar que la entidad con que se está tratando es quien dice ser.

En ARAMCEL, para encontrar una solución se utiliza la infraestructura de llave pública (PKI); PKI tiene la ventaja que evita tener que compartir una misma clave entre las entidades involucradas en el intercambio de información, además permite realizar la autenticación de mensajes puesto que si una llave se utiliza para cifrar un mensaje sólo se puede descifrar con la otra.

Es importante aclarar que dentro de las políticas de ARAMCEL se tiene que los Vendedores y Compradores deben estar registrados para poder acceder al sistema, de esta forma un componente Administrador dentro del Servidor Central actúa como entidad de certificación expidiendo las llaves correspondientes y entregando la información que tanto Vendedores como Compradores deben tener. Cada sitio en ARAMCEL posee una llave privada y una llave pública, así mismo tiene acceso a las llaves públicas de los demás sitios, de esa manera los datos de respaldo generados durante la transacción son cifrados y firmados por el emisor.

Por ejemplo, el Comprador A posee una llave pública CP y una llave privada CR, el sitio del Vendedor N posee una llave pública VP y una llave privada VR. De esta forma, si un agente es enviado a realizar una transacción llevará cifrada la información que debe entregar al Vendedor, con la llave pública del Vendedor, por lo tanto, únicamente la llave privada correspondiente puede ser usada para leer la información cifrada por esa llave pública, como se muestra en la Figura 32. La utilización de estas llaves ayudan a autenticar al creador de un mensaje.

Figura 32. Esquema de Cifrado / Descifrado.



Fuente. El autor.

El modelo de seguridad de ARAMCEL también se apoya en el uso de la función HASH, como se explicará en la siguiente sección. Una función HASH [MASCAGNI04, RAMIO04] es una función matemática que realiza un resumen del mensaje al que ha sido aplicada; es usada principalmente para garantizar la autenticidad del mensaje y su origen; así mismo, que el mensaje recibido no ha sido modificado. La función HASH ha surgido como alternativa para evitar el cifrado de documentos con clave pública, puesto que este proceso es demasiado lento en documentos grandes, así que se ha optado por incluir el resumen del mensaje almacenado en una cantidad fija de bits.

Para que una función HASH [RAMIO04] sea segura debe tener las siguientes características:

1. Unidireccionalidad. Debe ser imposible encontrar por medios computacionales el mensaje  $M$  a partir del resumen  $F(M)$ .
2. Compresión. La longitud de  $F(M)$  debe ser fija, independiente de la longitud de  $M$ .
3. Difusión.  $F(M)$  debe ser una función matemática compleja de los bits de  $M$ ; si se modifica por lo menos un bit de  $M$ , el hash obtenido debe ser diferente.

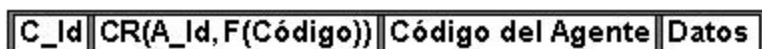
A continuación se presenta la forma como se definen en ARAMCEL los aspectos de seguridad autenticación, autorización y no-repudio; con base en ellos se soluciona el problema del agente hostil y del servidor hostil.

### 5.2.1. Autenticación

El esquema de autenticación de ARAMCEL se basa fundamentalmente en la firma digital. Cuando un Comprador envía un agente a realizar un proceso de búsqueda, este lleva consigo el identificador del usuario, cifrado con la llave privada del Comprador. Este esquema proporciona gran versatilidad y resuelve el problema de autenticación, además presenta una gran ventaja al utilizarlo con el Modelo de Agentes Móviles, ya que cuenta con la certeza que la información no ha sido alterada y se sabe con seguridad quien los envía. Para realizar la autenticación, los Agentes Móviles de ARAMCEL han sido divididos en cuatro partes como se muestra en la Figura 33, donde cada una de ellas está diseñada para apoyar el protocolo de seguridad. A continuación se presenta su descripción:

- La identidad del sitio del comprador (C\_Id) que lo creó.
- Una cabecera que contiene el identificador del agente (A\_Id) y el resultado de una función HASH aplicada al código. Esos valores están cifrados con la llave privada del Comprador, de tal manera que sólo utilizando la llave pública de ese Comprador pueden ser descifrados.
- El código del agente puede verificarse mediante esa función HASH, el resultado de ella se compara con la que lleva cifrada en la cabecera para indicar que ese agente proviene de un sitio confiable y no ha sido modificado.
- Los datos cifrados que el agente recoge de los diferentes Vendedores a lo largo de su itinerario.

Figura 33. La estructura de un agente en ARAMCEL.



Fuente. El autor.

El uso de llaves y certificados es crucial para la autenticación, por ello ARAMCEL fortalece la seguridad con base en los conceptos de la clase de Java llamada *Keystore* [OAKS01, SUN03] que representa una colección en memoria de llaves y certificados. Esa clase maneja dos tipos de entradas:

- *Llave*: este tipo de entrada del *keystore* maneja una llave confidencial de cifrado, la cual es almacenada en un formato protegido mediante una contraseña para evitar el acceso no autorizado. Generalmente, una llave almacenada en este tipo de entrada es una llave secreta o una llave privada acompañada por un certificado que corresponde a la llave pública. Las llaves privadas y los certificados son usados para que una entidad se auto autentique.
- *Certificado confiable*: este tipo de entrada contiene un certificado con una llave pública perteneciente a otra entidad. Es llamado un certificado confiable porque el propietario del *keystore* asegura que la llave pública dentro de ese certificado realmente pertenece a la entidad identificada como el propietario de la llave.

Por seguridad se tiene una contraseña para todo el contenido y además cada entrada en el *keystore* se protege por una contraseña diferente.

### **5.2.2. Autorización**

Los Agentes Móviles que llegan a cada sitio se consideran no confiables y dentro del archivo de políticas de seguridad se establece que los agentes sólo tienen acceso al envío de mensajes; de esa forma se asignan permisos que restringen a los agentes evitando acciones hostiles pero con la capacidad suficiente para interactuar con el sistema. Es necesario ser un usuario Comprador o Vendedor autorizado por el sistema con sus respectivas llaves públicas y privadas para cifrar o descifrar información; sólo quien esté autorizado, puede realizar el proceso de descifrado de la cabecera del agente y reconocer si ese agente pertenece a un usuario. Una vez el agente ha sido autenticado, se le autoriza para que use los recursos establecidos para él, con base en las políticas de seguridad del sistema.

Para el proceso de compra se requiere la previa autenticación tanto del agente Vendedor como del agente Comprador; el código de pago que lleva el agente Comprador va cifrado con la llave pública del Vendedor, para que sólo el Vendedor autorizado pueda usarlo; el

código de aceptación de la transacción, va cifrado con la llave pública del Comprador para evitar que usuarios compradores no autorizados usen ese código.

### **5.2.3. No-Repudio**

En ARAMCEL, se utilizan las firmas digitales para evitar el repudio de origen y recepción. Cuando el agente móvil entrega el identificador de su usuario y el código de pago cifrado y firmado, está entregando la prueba que acredita que el Comprador envió el agente y ordenó la compra, de esa forma se evita el repudio de origen. Por su parte, cuando el Agente Vendedor, entrega el código de aceptación de la transacción, cifrado y firmado, se puede verificar que el Vendedor aceptó la transacción y por tanto se evita el repudio de recepción.

### **5.2.4. Solución al problema del Agente Hostil**

El lenguaje de programación Java [OAKS01] provee funcionalidades que permiten implementar políticas de seguridad basadas en un mecanismo de autenticación y autorización; en ARAMCEL, los permisos de archivos restringen al agente el uso de recursos en el computador del Vendedor, al igual que las operaciones que esos agentes pueden realizar. Además en ARAMCEL el sistema del Comprador que crea el agente, le añade una cabecera que contiene el identificador del propietario y cifrados con la llave privada del Comprador, el identificador del agente y el resultado de una función HASH aplicada al código. Cuando un agente llega al sitio del Vendedor, la identidad del Comprador se verifica al aplicarle la llave pública del Comprador. El descifrado autentica al Comprador y revela la identidad del agente; en caso de no realizarse el descifrado de la cabecera del agente Comprador o si el agente intenta exceder su límite de permisos mientras se ejecuta, el Vendedor aborta la ejecución del agente y reporta al Servidor Central el ataque.

El Servidor usa las características de seguridad para definir los privilegios apropiados y restringir el acceso del agente a los recursos del sistema. Además ARAMCEL no permite la creación de agentes residentes, puesto que para reservar un artículo es suficiente con la cotización entregada y cifrada con la llave privada del Vendedor y nuevamente cifrada

con la llave pública del Comprador. Los Servidores son protegidos de los agentes hostiles, pero los agentes aún están expuestos a ataques; utilizando las opciones de seguridad, un agente solicita cierto nivel de protección, pero esa solicitud depende del servidor en el cual está ejecutándose. Por tanto, no es posible para los agentes, protegerse de servidores hostiles que ignoran los parámetros de protección pedidos por ellos; se requieren técnicas adicionales para proteger a los agentes de esos servidores hostiles.

#### **5.2.5. Solución al problema del Servidor Hostil**

Los agentes Compradores que llegan al sitio de un Vendedor ejecutan su código dentro de esa máquina, exponiéndose al problema del servidor hostil. En ARAMCEL, cada sitio de un Vendedor se encuentra en una lista de servidores confiables que están dentro de la base de datos y han sido previamente acreditados por el sistema. En caso que un servidor haya modificado los datos o el código del agente, el servidor actual puede verificarlo por medio de la información contenida en la cabecera del agente; en ese caso es tratado como hostil y eliminado por el sistema donde se encuentre. Si un agente desaparece cuando recorre el itinerario, ARAMCEL permite el envío de un agente móvil Fiscalizador que sigue el mismo recorrido y envía mensajes cuando llega y sale de cada servidor, para informar al sistema el sitio donde posiblemente ocurrió el ataque.

Al encontrar un sitio hostil inmediatamente el Servidor Central lo marcará como No Confiable y será eliminado de la lista de sitios posibles para viajar, en caso de no encontrarlo los Agentes Móviles que tengan que pasar por esos sitios serán acompañados por agentes fiscales, durante la cantidad de viajes que el Administrador del Servidor Central con base en su autonomía considere necesario.

#### **5.2.6. El Agente Fiscalizador**

Cuando un agente móvil regresa a la plataforma y se comunica con el Agente Estacionario Receptor del Comprador (AERP), éste informa al Agente Estacionario Despachador del Comprador (AEDC). El AEDC establece un plazo para que los agentes que creó regresen de su viaje; si se ha superado ese límite de tiempo y no ha recibido

notificación del AERP, el AEDC procede a crear un nuevo tipo de agente, el Agente Móvil Fiscalizador (AMF). Cuando éste es creado, solicita la información que llevaba el agente desaparecido, incluyendo el artículo a buscar y los destinos que tenía asignados; con base en esa información establece su itinerario.

El AMF visita cada sitio de los vendedores en su itinerario y al llegar a la respectiva plataforma envía un mensaje al AEDC confirmando su ingreso. Después realiza un proceso de búsqueda similar al que debía efectuar el agente desaparecido, cuando ha finalizado y va a salir del sitio, lo reporta al AEDC por medio de un mensaje.

Si el agente logra finalizar el itinerario y regresa al sitio del comprador, se comunica con el AERP para entregar los resultados de la búsqueda y con el AEDC para informar la finalización de su viaje, sin encontrar el sitio hostil. Con esa información el AEDC se prepara para enviar un AMF con cada AMB que salga de la plataforma y deba visitar alguno de los posibles sitios hostiles.

Si el agente no logra finalizar su itinerario, el AEDC revisa el registro de mensajes de entrada y salida de cada sitio; al comparar con el itinerario que llevaba el AMF determina el sitio hostil, lo marca para evitar volver a utilizarlo como vendedor. Además le informa al Servidor Central para que el Administrador del Sistema tome las medidas necesarias.

El proceso utilizado en el Sistema del Comprador, se realiza de forma análoga en el Sistema del Vendedor para determinar los sitios hostiles que puedan atacar al Agente Móvil Notificador (AMN).

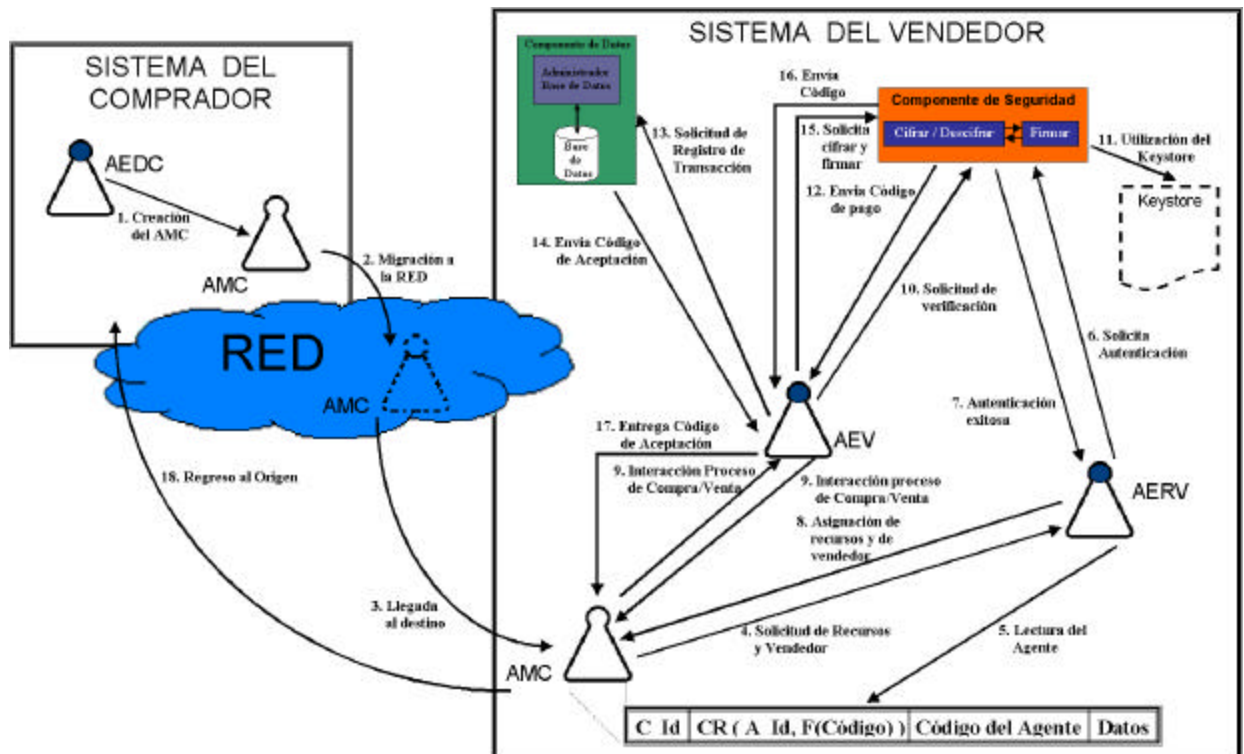
### **5.2.7. Protocolo de Seguridad**

Los aspectos más relevantes del protocolo se presentan en la Figura 34 y su proceso, en forma general, se describe a continuación.

En el Sistema del Comprador el Agente Estacionario Despachador de Comprador (AEDC) crea un Agente Móvil Comprador (AMC) (1), el AMC viaja al dominio del Vendedor (2). Cuando el Agente Móvil del Comprador (AMC) llega al sitio del Vendedor (3) solicita la

asignación de un agente vendedor y recursos en el servidor (4). Antes de cumplir con la solicitud, el Agente Estacionario Recepcionista del vendedor (AERV) lee los datos del agente (5) y se los envía al Componente de Seguridad, para que realice la autenticación del agente y del Usuario Comprador (6). El Componente de Seguridad busca el certificado digital del comprador y extrae la Clave Pública (11), confirma la existencia del usuario y revisa la integridad del código del agente; descifra el A\_Id y la F(código), aplica la función HASH al código del agente y compara la función obtenida con la descifrada. Cuando el AMC es autenticado (7), se le asignan recursos y un Agente Estacionario Vendedor (AEV) (8); con este último, el AMC realiza la interacción del proceso de Compra/Venta (9).

Figura 34. Protocolo de seguridad en ARAMCEL.



Fuente. El autor.

El AMC entrega el código de pago, cifrado y firmado al Agente Estacionario Vendedor (AEV). El AEV envía el código de pago al Componente de Seguridad y le solicita la verificación de la firma (10). Ese componente utiliza la información almacenada en el Keystore (11) para descifrar el código de pago y verificar la firma; después de eso,

devuelve el código de pago descifrado al AEV (12). Este último hace la solicitud de registro de la transacción al Componente de Datos (13), el cual genera y almacena el código de aceptación y envía al AEV (14). A su vez el AEV solicita al Componente de Seguridad el cifrado y firmado del código (15), cuando esto se ha llevado a cabo (16), el código es entregado al AMC (17), quien regresa a su origen (18).



## **6. VALIDACIÓN DE LA ARQUITECTURA**

La validación de la arquitectura se llevó a cabo por medio de la implementación del prototipo software ADAM – Aplicación de los Agentes Móviles al comercio electrónico. La implementación se realizó como un proyecto de pregrado de la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander. ADAM fue realizado con base en las especificaciones detalladas de esta tesis. Para profundizar en la implementación de ADAM consultar [GOMEZ04]. A continuación se incluye un resumen de los principales aspectos relacionados con el desarrollo de ADAM.

### **6.1. TECNOLOGÍAS UTILIZADAS.**

Las tecnologías utilizadas para el desarrollo de ADAM fueron: Arquitectura ARAMCEL, plataforma de Agentes Móviles Aglets [LANGE98], lenguaje de programación JAVA [SUN04a, CEBALLOS00], proveedor de seguridad Cryptix [CRYTIX04] y motor de base de datos relacional MySql [MYSQL04].

### **6.2. DESCRIPCIÓN GENERAL DE ADAM**

El prototipo software Aplicación de los Agentes Móviles al comercio electrónico – ADAM se restringió a la realización de dos de las transacciones de comercio electrónico propuestas en ARAMCEL. Las transacciones referidas son: la búsqueda de información con base en las preferencias del usuario y la realización de órdenes de Compra/Venta de productos. Además para realizar la implementación las transacciones de comercio electrónico se limitaron a un caso particular conformado por un sistema de tiendas que distribuían libros y discos compactos.

Los Usuarios Compradores en ADAM pueden realizar las siguientes acciones:

- Acceso al sistema y solicitud de información sobre productos que les interesen. Un usuario comprador ingresa al sistema un conjunto de palabras claves sobre los datos del producto que requiere; esos datos son: título, autor, artista ó ISBN. El usuario puede establecer preferencias como: precio máximo a pagar ó tiempo límite de espera por ese producto.

Con base en las palabras clave y las preferencias del usuario, el sistema construye los Agentes Móviles Buscadores que viajan a las diferentes tiendas registradas y buscan la información que solicita el usuario, cuando finalizan su búsqueda, regresan al sitio de origen y entregan los resultados. Si el usuario finaliza su sesión en el sistema, el Agente Móvil Buscador de ADAM espera que se reestablezca la conexión y regresa con los resultados obtenidos en el viaje.

- Selección de artículos que desea comprar e ingreso de los datos de entrega. Con esa información ADAM construye y envía un Agente Móvil Comprador (AMC) que se desplaza al sitio que distribuye el producto seleccionado por el usuario y ordena su adquisición. Este proceso se realiza utilizando un AMC por cada producto.
- Consulta de la información histórica de compras realizadas por el usuario.

ADAM también ofrece funcionalidades para los usuarios Vendedores; ellos pueden realizar las siguientes acciones:

- Registro en su propia Base de Datos de la información de los productos que cada uno ofrece. Esa información la utilizan los Agentes Vendedores de ADAM para responder a las consultas de los usuarios compradores y realizar el proceso de gestión de órdenes de Compra/Venta.
- Mantenimiento de la información en la Base de Datos. Los usuarios vendedores no intervienen en los procesos de negociación, entrega de información ni orden de Compra/Venta.

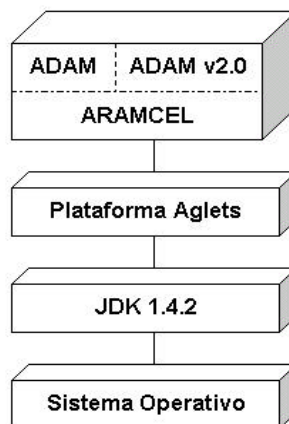
- Ingreso al sistema de la información de promociones para los usuarios compradores. Con estos datos ADAM construye y envía un Agente Móvil Notificador (AMN) a los sitios de los compradores. Ese AMN una vez finalizado el recorrido se auto elimina.
- Consultar la información histórica de las ventas realizadas.

### 6.2.1. Marco de trabajo de ADAM

El marco de trabajo de ADAM está compuesto de cuatro capas principales como se muestra en la Figura 35. Las capas inferiores corresponden al sistema Operativo encargado de controlar los recursos del sistema y al entorno de ejecución que provee el JDK1.4.2. Sobre ellos se encuentra la implementación de la plataforma de Agentes Móviles Aglets que se encarga de ejecutar los agentes y administrar su movilidad.

La última capa está constituida por el prototipo basado en Agentes Móviles para comercio electrónico ADAM, en el se encuentran los agentes estacionarios y móviles definidos para los sistemas del comprador ó del vendedor, según sea el caso. En la representación gráfica de esta capa se aprecia la relación existente entre ARAMCEL y ADAM, puesto que el prototipo y sus futuras versiones implementan las funciones definidas por la arquitectura.

Figura 35. Marco de Trabajo de ADAM



Fuente. El autor.

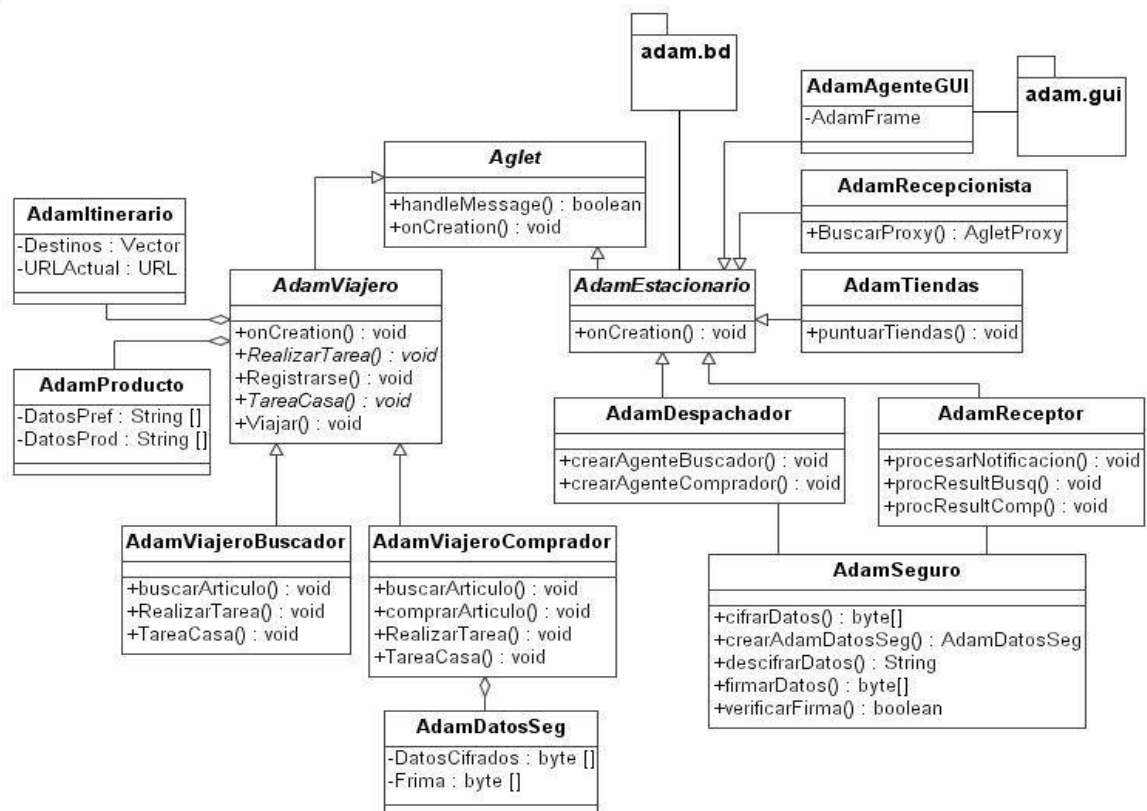
## 6.2.2. Diagrama general de clases de ADAM

Una vez descrito el funcionamiento general y el marco de trabajo, se hace un acercamiento al diagrama general de clases que muestra las principales clases del sistema. Además se incluye la descripción general de cada una de las clases. Los diagramas hacen referencia a los paquetes “adam.bd” y “adam.gui”, cuya descripción y diagrama se ha omitido por no ser relevantes dentro del desarrollo de esta investigación.

### 6.2.2.1 Sistema del Comprador.

La Figura 36 muestra el diagrama de clases del sistema del comprador y la descripción general de las clases se observa en la Tabla 2.

Figura 36. Diagrama General de Clases Sistema del Comprador.



Fuente. [GOMEZ04].

Tabla 2. Descripción General de Clases – Sistema del Comprador.

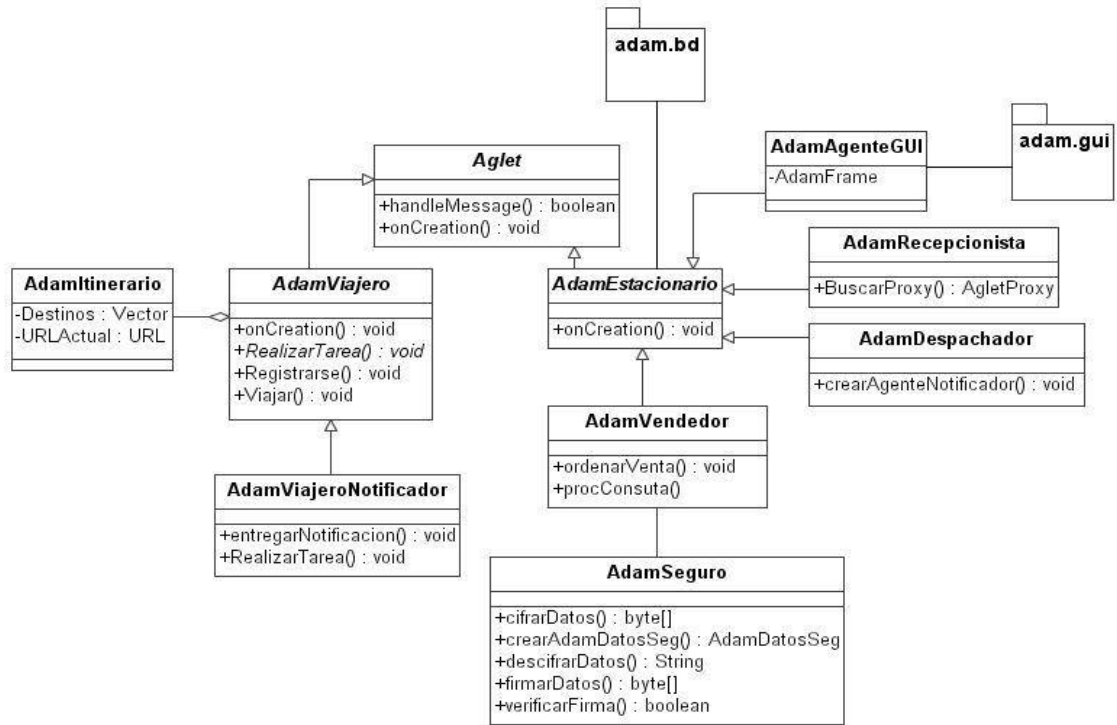
<b>Clase</b>	<b>Descripción</b>
<b>Aglet</b>	Clase Abstracta que define la funcionalidad de los Agentes Móviles de la plataforma Aglets.
<b>AdamViajero</b>	Clase Abstracta que define la funcionalidad principal de los Agentes Móviles de ADAM.
<b>AdamViajeroBuscador</b>	Agente Móvil que realiza procesos de búsqueda de información.
<b>AdamViajeroComprador</b>	Agente Móvil que realiza procesos de ordenar Compra/Venta.
<b>AdamEstacionario</b>	Clase Abstracta que define la funcionalidad principal de los Agentes Estacionarios de ADAM.
<b>AdamAgenteGUI</b>	Agente administrador de la interfaz de usuario.
<b>AdamRecepcionista</b>	Agente que recibe a los Agentes Móviles que llegan al sitio del comprador
<b>AdamTiendas</b>	Agente que procesa los reportes de servicio de las tiendas entregados por los Agentes Móviles.
<b>AdamReceptor</b>	Agente que procesa los resultados entregados por los Agentes Móviles.
<b>AdamDespachador</b>	Agente que interactúa con los Agentes Móviles antes que inicien su itinerario.
<b>AdamItinerario</b>	Almacena el itinerario del Agente Móvil.
<b>AdamProducto</b>	Almacena los datos y las preferencias del producto que el Agente Móvil va consultar u ordenar comprar.
<b>AdamDatosSeg</b>	Un objeto AdamDatosSeg almacena datos cifrados y su correspondiente firma digital.
<b>AdamSeguro</b>	Realiza los procesos de cifrado, verificación de firma y descifrado de datos.

Fuente. [GOMEZ04].

### 6.2.2.2 Sistema del Vendedor.

La Figura 37 muestra el diagrama de clases del sistema del vendedor y la descripción general de las clases se observa en la Tabla 3.

Figura 37. Diagrama General de Clases del Sistema del Vendedor.



Fuente. [GOMEZ04].

Tabla 3. Descripción General de Clases – Sistema del Comprador.

Clase	Descripción
<b>Aglet</b>	Clase Abstracta que define la funcionalidad de los Agentes Móviles de la plataforma Aglets.
<b>AdamViajero</b>	Clase Abstracta que define la funcionalidad principal de los Agentes Móviles de ADAM.
<b>AdamViajeroNotificador</b>	Agente Móvil que realiza procesos de notificación
<b>AdamEstacionario</b>	Clase Abstracta que define la funcionalidad principal de los Agentes Estacionarios de ADAM.
<b>AdamVendedor</b>	Agente que procesa las solicitudes de consulta y compra de los Agentes Móviles del comprador.
<b>AdamAgenteGUI</b>	Agente que administra la interfaz de usuario.
<b>AdamRecepcionista</b>	Agente que recibe a los Agentes Móviles del Comprador y les asigna un vendedor.
<b>AdamDespachador</b>	Agente que interactúa con los Agentes Móviles antes que inicien su itinerario.
<b>AdamItinerario</b>	Almacena el itinerario del Agente Móvil.
<b>AdamDatosSeg</b>	Un objeto AdamDatosSeg almacena datos cifrados y su correspondiente firma digital.
<b>AdamSeguro</b>	Realiza los procesos de cifrado, verificación de firma y descifrado de datos.

Fuente. [GOMEZ04].

### **6.3. CARACTERÍSTICAS VALIDADAS POR MEDIO DE SOFTWARE.**

Como ya se mencionó la validación de la arquitectura se realizó por medio de la implementación del prototipo software ADAM. A continuación se presentan las principales características de ARAMCEL implementadas en ADAM:

- La transacción búsqueda de información. Por medio de esta transacción, ADAM ofrece a un usuario comprador la posibilidad de consultar sobre libros y discos compactos en un sistema de tiendas utilizando un agente móvil buscador como su representante en el proceso.
- Almacenamiento de datos y resultados de búsqueda. En ADAM se almacenan por medio del componente de Datos, los datos ingresados por el usuario al sistema en el momento de solicitar la búsqueda de información sobre un determinado artículo; así mismo, se guardan los resultados obtenidos por los agentes durante el proceso.
- La transacción orden de compra / venta. El prototipo ADAM permite a un usuario comprador consultar los resultados de la búsqueda de información y seleccionar el libro ó disco compacto que desea adquirir, con base en esa información, se crea un agente móvil comprador que viaja a la tienda y ordena la compra del producto.
- Registro de compra / ventas realizadas. En ADAM se almacena, por medio del componente de Datos, la información referente a las compras por cada usuario, de esta forma las tiendas pueden establecer el perfil del usuario con base en su historial; de la misma manera, en el sitio del comprador se guarda la información de las compras realizadas, para permitirle al usuario comprador la supervisión de los resultados de sus transacciones.
- Clasificación de tiendas. Para realizar esta clasificación se asignó un puntaje inicial a cada tienda, que se incrementa ó decrementa con base en los resultados entregados

por los agentes al realizar los procesos de búsqueda de información ó compra de artículos.

- Proceso de notificación básico. Este proceso se implementó en ADAM restringiéndolo a la entrega de un mensaje denominado “Notificación” a los compradores; este mensaje es directamente ingresado por el usuario vendedor al sistema.
- Autenticación y Autorización del usuario durante el proceso de compra / venta. En ADAM para que el agente móvil del comprador lleve a cabo un proceso de compra / venta, se autentica primero al usuario que representa, de esta forma se autoriza su acceso a la transacción.
- No-repudio de la transacción orden de compra / venta. En ADAM se utilizó dentro del componente de seguridad, el proveedor de seguridad Cryptix, que permite el firmado y cifrado de la información relacionada con el pago y la aceptación de la transacción de compra / venta. Por medio de este proceso se cumplió con el aspecto de seguridad No-repudio.
- Control de acceso al sistema de archivos. Este control se realizó de la forma definida en ARAMCEL, por medio de las políticas de seguridad ofrecidas por la plataforma Aglets, utilizada durante el desarrollo del prototipo. No fue necesario realizar implementaciones adicionales a este respecto.

#### **6.4. CARACTERÍSTICAS NO VALIDADAS POR MEDIO DE SOFTWARE.**

El prototipo de validación ADAM no implementó la totalidad de las características de ARAMCEL. A continuación se resumen las principales funcionalidades no implementadas:

- El acceso a opciones de compra a través de un Servidor de Compradores para usuarios que no cuentan con el sistema completo instalado en sus equipos.

- La gestión de asignaciones para controlar el consumo de memoria RAM, puesto que el lenguaje de programación JAVA no ofrece un método ó interfaz que permita conocer el consumo de memoria por hilo.
- El firmado de la información que recibe el Agente Móvil Buscador en cada sitio de su itinerario.
- El componente Administrador del sistema.
- El agente fiscalizador, porque surgió posterior a la implementación del prototipo y con base en la experiencia adquirida durante el desarrollo.
- El proceso de notificación, tan sólo se incluyó como un simple envío de mensajes desde el Usuario Vendedor hacia los Usuarios Compradores.
- El almacenamiento del perfil de cada Usuario Comprador, se limitó a guardar la información de las compras realizadas, sin incluir la opción de personalización del servicio con base en esa información.
- La división de los agentes móviles con base en la estructura presentada en el capítulo 5 de este documento.
- El proceso de autenticación y autorización total del agente móvil y del usuario comprador. Estos aspectos se limitaron a lo necesario para autenticar el usuario comprador y autorizar el acceso del agente móvil a la transacción de compra / venta.
- El doble cifrado definido para el código de aceptación y el código de pago. En ADAM se utilizó un cifrado sencillo.

## **6.5. EVALUACIÓN DEL FUNCIONAMIENTO DE ADAM.**

El prototipo ADAM fue sometido a pruebas de Validación y Verificación, realizadas durante el desarrollo, la información detallada de diseño, ejecución y resultados de estas pruebas puede ser encontrado en [GOMEZ04]. Una vez finalizada y entregada la primera versión de ADAM, se realizaron pruebas de rendimiento dirigidas por el autor de esta investigación.

### **6.5.1. Pruebas de Verificación.**

Las pruebas de verificación son un conjunto de actividades que aseguran que el software implementa correctamente una función específica [PRESSMAN02]. En el caso del prototipo software ADAM, las pruebas de verificación se realizaron en dos etapas, la primera correspondió a las pruebas aplicadas por la desarrolladora del prototipo y la segunda a las efectuadas por miembros de GITSI. A continuación se resumen los principales aspectos de las pruebas de verificación.

#### **6.5.1.1 Ambiente necesario para las pruebas**

Las pruebas de verificación se realizaron en equipos de GITSI y cuyas especificaciones de hardware y software son:

##### **Hardware**

- **Comprador**

Se instaló el sistema del comprador en un equipo con las siguientes características:

- Procesador Pentium MMX de 200MHz
- Memoria RAM de 48MB
- Disco Duro de 10G
- Tarjeta de Red

- **Vendedor**

El sistema del vendedor se instaló en un equipo con las siguientes características:

- Procesador Pentium IV de 1.6GHz

- Memoria RAM de 256MB
- Disco Duro de 40G
- Tarjeta de Red

### **Software**

Los equipos utilizados contaban con las siguientes especificaciones de software:

- Sistema Operativo Windows'98
- Máquina Virtual de Java JDK1.4.2
- Administrador de Base de Datos MySql 4.0.14b

#### **6.5.1.2 Información General necesaria para las pruebas**

Para realizar la prueba se almacenó en la base de datos del sistema la información de dos (2) tiendas.

#### **6.5.1.3 Procedimiento y Resultados**

El procedimiento se fijó con base en el proceso que fuese a ser probado. Los usuarios debían utilizar las funciones ofrecidas por el sistema y generar el ambiente necesario para el surgimiento de posibles errores con base en las instrucciones que les fueron entregadas. Los resultados de esas pruebas se utilizaron para corregir las fallas del prototipo; además las sugerencias entregadas sirvieron para el mejoramiento del mismo.

#### **6.5.2. Pruebas de Validación.**

La validación se refiere a un conjunto diferente de actividades que asegura que el software construido se ajusta a los requisitos del cliente [PRESSMAN02]. Las pruebas de validación del prototipo software ADAM se realizaron por personal ajeno al desarrollo del proyecto, en este caso se selecciono un grupo de estudiantes de "Ingeniería Industrial" pertenecientes a la asignatura "Programación de Computadores", de la Universidad Industrial de Santander. A continuación se resumen los aspectos principales de las pruebas.

### **6.5.2.1 Ambiente necesario para las pruebas**

Los sistemas de comprador y vendedor se instalaron en diferentes computadores del Laboratorio de Informática Luis Eduardo Arias de la Universidad Industrial de Santander.

Las especificaciones de los 16 equipos utilizados son:

#### **Hardware**

- Procesador Pentium IV de 1.6GHz
- Memoria RAM de 128MB
- Disco Duro de 20G
- Tarjeta de Red

#### **Software**

- Sistema Operativo Windows'XP
- Máquina Virtual de Java JDK1.4.2
- Administrador de Base de Datos MySql 4.0.14b

#### **Sistema del Vendedor**

Para el desarrollo de las pruebas se instaló el software del vendedor en seis (6) equipos. Cada uno de los equipos contaba con una base de datos que contenía la información de los productos ofrecidos. La distribución de los equipos con base en el tipo de producto fue:

- Dos (2) máquinas correspondían a vendedores que distribuían libros.
- Dos (2) máquinas correspondían a vendedores que distribuían discos compactos.
- Dos (2) máquinas correspondían a vendedores que distribuían libros y discos compactos.

#### **Sistema del Comprador**

Se instaló el sistema del comprador en diez (10) equipos.

### **6.5.2.2 Procedimiento**

Los diez computadores que tenían el sistema del comprador estaban disponibles para ser usados por los estudiantes que colaboraron con la prueba.

Los estudiantes estaban encargados de utilizar el software como usuarios finales y diligenciar una encuesta sobre la utilización del mismo, rendimiento, tiempos de respuesta y seguridad básica; así mismo, se solicitó hacer una evaluación de los posibles usos de un software de este tipo, para que fuese analizado con base en su experiencia en transacciones electrónicas y en los conocimientos de Administración y Comercio adquiridos en su carrera.

### **6.5.2.3 Resultados**

Con base en el análisis y en las observaciones generales recibidas se concluyó que el sistema tuvo buena aceptación y cumple con gran parte de las expectativas y necesidades de los usuarios para la realización de transacciones de comercio electrónico; además recibió buena aceptación la utilización de los Agentes Móviles y se recibieron comentarios favorables sobre los beneficios para los usuarios de este tipo de sistemas.

### **6.5.3. Pruebas de Rendimiento**

Para evaluar los tiempos de respuesta del prototipo software ADAM se diseñó un experimento utilizando cuatro equipos y contando con la colaboración de miembros de GITSI en la ciudad de Bucaramanga y estudiantes miembros del Semillero de Investigación en Inteligencia Artificial de la Universidad del Valle - Sede Tuluá.

#### **6.5.3.1 Ambiente necesario para las pruebas**

Los cuatro equipos utilizados en el experimento se distribuyeron geográficamente como se muestra en la Figura 38; las características técnicas de cada máquina se resumen en la Tabla 4.

Figura 38. Ubicación de equipos en pruebas de rendimiento



Fuente. El autor.

Tabla 4. Características de los equipos en las pruebas de rendimiento

Característica	Computador A	Computador B	Computador C	Computador D
<b>Ubicación</b>	Universidad del Valle. Tuluá, Valle del Cauca.	Universidad del Valle. Tuluá, Valle del Cauca.	Residencia Miembro GITSI. Bucaramanga, Santander.	Residencia Miembro GITSI. Bucaramanga, Santander.
<b>Hardware</b>	Intel PentiumIV 1,6 GHz	Servidor Intel Xeon	Intel Pentium IV 1,5 GHz	Intel Pentium MMX 200 MHz
<b>Software</b>	Windows XP JDK 1.4.2 ASDK 2.0.2 MySql 4.0.14b ADAM-Comprador	Linux Mandrake 10 JDK 1.4.2 ASDK 2.0.2 MySql 4.0.14b ADAM-Vendedor	Windows 98 JDK 1.4.2 ASDK 2.0.2 MySql 4.0.14b ADAM-Vendedor	Windows 98 JDK 1.4.2 ASDK 2.0.2 MySql 4.0.14b ADAM-Vendedor

Fuente. El autor.

Cada uno de los equipos en los que se instaló el Sistema del Vendedor contaba con una base de datos que contenía la información de los productos ofrecidos. Las tres vendedores distribuían libros y discos compactos. En cada servidor Tahití se incluyó un

Agente Estacionario<sup>16</sup> encargado de monitorear las horas de ingreso y salida del agente en la plataforma; este agente al finalizar la prueba entregó un reporte con base en el *AgletID* de cada agente.

### 6.5.3.2 Procedimiento

La hora del sistema de los equipos utilizados en la prueba fue sincronizada, para evitar las diferencias que pudiesen afectar los resultados. Una vez se configuraron los equipos, se solicitó a los estudiantes del Semillero que hicieran búsquedas de información utilizando el software. Se limitó el experimento a la búsqueda de información, puesto que es esta transacción la que requiere de un itinerario y por lo tanto facilita la comparación de los tiempos con base en las características y ubicación de los equipos utilizados. Se realizaron 20 procesos de búsqueda de información.

### 6.5.3.3 Resultados

Al finalizar el experimento se obtuvieron los reportes generados por los Agentes Estacionarios ubicados en cada plataforma. En la Tabla 5 se presenta el tiempo para el viaje entre servidores y el tiempo invertido por los agentes móviles al llegar a cada servidor. Los tiempos incluidos en la Tabla corresponden al promedio de los tiempos generados en cada caso.

Tabla 5. Resultados de las pruebas de rendimiento

Tarea	Tiempo (Promedio)	Ubicación Geográfica
Viaje Comp. A hacia Comp. B	1,3 Segundos	Tuluá – Tuluá
Proceso en Computador B	1,1 Segundo	Tuluá
Viaje Comp. B hacia Comp. C	12 Segundos	Tuluá – Bucaramanga
Proceso en Computador C	1,2 Segundos	Bucaramanga
Viaje Comp. C hacia Comp. D	4,6 Segundos	Bucaramanga – Bucaramanga
Proceso en Computador D	5 Segundos	Bucaramanga
Viaje Comp. D hacia Comp. A	16 Segundos	Bucaramanga - Tuluá
<b>TOTAL ITINERARIO</b>	<b>41,2 Segundos</b>	

Fuente. El autor.

<sup>16</sup> Este agente se implementó específicamente para el experimento y no está incluido dentro de los sistemas comprador y vendedor.

#### **6.5.3.4 Análisis de Resultados**

El análisis de los resultados obtenidos se hizo con base en los tiempos utilizados en los desplazamientos y en el tiempo de procesamiento utilizado en cada uno de los equipos.

##### **Tiempo de Procesamiento**

Con base en los resultados obtenidos puede observarse que el tiempo de procesamiento en cada equipo está directamente relacionado con las capacidades del mismo, puesto que los tiempos en los computador B y C son similares, pero en el D, se observa que el tiempo es casi cinco (5) veces mayor que en las demás máquinas. Esto ocurre, porque se trata de un computador con un procesador de menor capacidad y memoria RAM limitada, lo cual a su vez retarda las búsquedas y procesos hechos en la plataforma.

##### **Tiempo de Viaje**

Al revisar los tiempos de viaje se observa la amplia diferencia entre los obtenidos en interacciones en la misma ciudad, con aquello que incluyen viajes Bucaramanga – Tuluá ó viceversa. Al igual que en el análisis del tiempo de procesamiento, en este caso se obtienen tiempos mayores en aquellas interacciones realizadas con el computador C.

Al sumar los tiempos de viaje y procesamiento se obtuvo el tiempo promedio total del itinerario, el cual fue *41,2 Segundos*. Este proceso se realizó en un entorno “casi ideal”, puesto que sólo había un sistema del comprador; es importante señalar que los tiempos pueden aumentar con base en la cantidad de compradores accediendo al sistema.

Al analizar las diferencias de tiempo obtenidas con base en la capacidad de hardware de los equipos podría recomendarse a los vendedores que utilicen el sistema, la adquisición de máquinas con altas capacidades de procesamiento para que no conviertan sus equipos en cuellos de botella.

## CONCLUSIONES

El principal logro de este trabajo de investigación ha sido el diseño de la arquitectura ARAMCEL, la cual ofrece soporte para tres tipos de transacciones básicas de comercio electrónico mediante la utilización del modelo de Agentes Móviles; las transacciones en ARAMCEL son búsqueda de información, realización del proceso de Compra / Venta y notificación de información. ARAMCEL ofrece además un modelo de seguridad que soporta autenticación, autorización y no-repudio, con los cuales se proporciona protección a agentes y servidores.

El diseño de ARAMCEL constituye un avance en la aplicación del Modelo de los Agentes Móviles al Comercio Electrónico y por lo tanto facilita la implementación de aplicaciones en dicho campo. Asimismo colabora en el desarrollo de la línea de investigación en Agentes Móviles del Grupo de Investigación en Ingeniería Telemática y Sistemas Inteligentes – GITSI, adscrito a la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander. ARAMCEL también se constituye en un aporte a las necesidades de un mundo en proceso de globalización y provee un entorno seguro de ejecución para ese tipo de sistemas.

Con el diseño de ARAMCEL además de cumplirse los objetivos planteados al inicio de la investigación se obtuvieron conclusiones sobre los Agentes Móviles, las transacciones de comercio electrónico, los aspectos de seguridad informática y la arquitectura misma, que se presentan en los siguientes párrafos:

- La utilización del Modelo de Agentes Móviles colabora en la reducción del tráfico en la red, al permitir transformar la mayoría de accesos remotos en accesos locales. La autonomía de los agentes, los capacita para tomar decisiones que combinadas con la subdivisión del trabajo, permite tener agentes especializados en cada tarea colaborando entre ellos para alcanzar sus metas.

- Los Agentes Móviles debido a las ventajas que poseen presentan fortalezas para trabajar de manera eficiente en sistemas distribuidos, pero también tienen inconvenientes de seguridad; los cuales son una oportunidad al generar un campo de acción en investigación para generar soluciones.
- En este documento, se han discutido las amenazas de seguridad existentes en un sistema para comercio electrónico basado en Agentes Móviles, los problemas del agente hostil y servidor hostil. Con base en esas amenazas, en los aspectos de seguridad autenticación, autorización y no-repudio y en PKI se presentó el modelo de seguridad de ARAMCEL que ofrece una solución a esos problemas.
- Aunque se ha presentado una solución a algunos de los problemas de seguridad, existen otros que son objeto de investigación en el mundo, además de aquellos que se presenten a medida que los sistemas basados en Agentes Móviles para comercio electrónico se hagan más populares. Sin embargo, considerando las características del modelo de seguridad presentado es posible extenderlo y enriquecerlo para contrarrestar nuevos tipos de ataques que surjan en el entorno.
- La transacción de búsqueda de información diseñada en ARAMCEL, permite al usuario consultar información sin conocer los datos exactos de los productos, además los resultados que le son entregados al usuario han sido previamente filtrados por el agente móvil buscador, por tal razón sólo se obtienen aquellos considerados como relevantes; esta característica de búsqueda no sólo evita que el comprador deba perder tiempo en la red buscando por sí mismo, sino que disminuye el tiempo revisando ofertas, lo cual sumado a la posibilidad de obtener resultados de varios vendedores le facilita la comparación de las opciones de compra. Otro de los valores agregados es el puntaje de tiendas, puesto que de esta forma se establece una clasificación de vendedores que facilita esa búsqueda, evitando invertir más tiempo del necesario y por lo tanto los usuarios pueden obtener resultados de mejor calidad y en menor tiempo.
- ARAMCEL también permite ordenar la Compra / Venta de artículos; por medio de ella el usuario puede seleccionar el producto a comprar y enviar un agente a que realice la

transacción. Las condiciones para la realización de una compra implican igualar o mejorar las seleccionadas por el usuario, en caso contrario el agente aborta la transacción. Por la razón anterior y debido al concepto de puntaje con base en el servicio, la tienda debe garantizar las condiciones de compra; esto sumado al aspecto de seguridad no-repudio hace que el proceso de Compra / Venta tenga grandes ventajas y ofrezca garantía en la transacción para el usuario.

- La tercera transacción definida en ARAMCEL es la notificación de ofertas y novedades, por medio de ella los vendedores pueden hacer conocer a sus compradores descuentos en artículos, todo tipo de promociones y nuevas adquisiciones. La gran ventaja para el vendedor es que se realiza sin su intervención directa, puesto que en la plataforma existe un agente encargado de revisar la información de la base de datos y con base en los productos, ofertas y perfiles de los usuarios, les envía Agentes Móviles Notificadores que entreguen la información. Al realizar la clasificación con base en perfiles, se obtiene que los compradores reciben sólo información que les interesa y los vendedores logran además de informar, llegar con mayor posibilidad de aceptación a los potenciales compradores.
- La implementación del prototipo software ADAM siguió detalladamente las indicaciones de diseño de ARAMCEL y desarrolló dos de las transacciones de comercio electrónico definidas por la arquitectura, la búsqueda de información y la orden de compra / venta de artículos. Estas transacciones fueron adaptadas al caso particular de un sistema de tiendas que distribuían libros y discos compactos; a su vez ADAM implementó de forma superficial algunas otras características presentadas en ARAMCEL, con el objeto de probar su funcionalidad, así como establecer alcances y limitaciones de ellas.

Las pruebas realizadas a ADAM permitieron validar experimentalmente algunas de las características de ARAMCEL, establecer su utilidad dentro del desarrollo de aplicaciones en el área y comprobar las ventajas de utilizar el Modelo de Agentes Móviles en este tipo de transacciones.

- La flexibilidad del diseño de ARAMCEL, le permite ser utilizada en el desarrollo de aplicaciones independientes de la plataforma de agentes móviles que soporte el proceso; aunque se definieron una serie de servicios exigibles para la plataforma, son funciones básicas ofrecidas por la mayor parte de ellas. La validación de esta característica se llevó a cabo dentro del desarrollo del prototipo ADAM, puesto que la selección de la plataforma se realizó con base en métricas establecidas con base en el estado del arte de las plataformas [GOMEZ04], independientemente de las características de ARAMCEL.

El resultado principal de la investigación es una arquitectura que obtiene beneficios de las ventajas del modelo de Agentes Móviles para aplicarlo en la realización de tres tipos de transacciones básicas del comercio electrónico, así mismo dispone de un modelo de seguridad versátil y eficiente que ofrece autenticación y autorización de vendedores, compradores y agentes; y que a su vez evita que los actores de las transacciones nieguen su participación en ellas. De igual forma, las especificaciones de ARAMCEL y los resultados de este Trabajo de Investigación pueden extenderse a otros sistemas para comercio electrónico basados en el mismo modelo que se encuentren actualmente en el mercado, de tal manera que al combinarlos con las características propias del sistema permita mejorar el servicio a los usuarios y proporcionar mayor seguridad en las transacciones.

## RECOMENDACIONES PARA TRABAJOS FUTUROS

Con base en la formalización de ARAMCEL, el estudio realizado sobre el estado del arte y la técnica y los resultados obtenidos en el desarrollo de esta Investigación, se presentan las siguientes recomendaciones para trabajos futuros:

- Se hace necesario ampliar el modelo de seguridad de ARAMCEL para proporcionar una solución a los aspectos de seguridad Integridad y Confidencialidad, y así ofrecer seguridad integral a las transacciones definidas; el uso de PKI dentro de la definición de la arquitectura facilita la extensión del modelo a estos aspectos.
- El modelo de seguridad de la arquitectura puede ampliarse y hacerse más robusto, realizando estudios para proveer técnicas específicas y especializadas para cada uno de los aspectos de seguridad: autenticación, autorización, no-repudio, integridad y confidencialidad.
- ARAMCEL provee la clasificación de tiendas para mejorar el servicio ofrecido a los usuarios compradores, de la misma forma puede ofrecerse una asignación de puntaje a los compradores con base en el volumen de sus compras; con esta utilidad los vendedores pueden profundizar en la realización de ofertas personalizadas u ofrecer alguna clase de servicio especial<sup>17</sup> a los clientes de alta prioridad.
- Definir los agentes necesarios en la plataforma del Sistema del Vendedor, así como los mecanismos de negociación apropiados para permitir la búsqueda de productos a vendedores, que respondan a los requerimientos de compradores, en casos de inventario agotado ó para mantener un servicio eficiente al realizar transacciones con grandes compradores.

---

<sup>17</sup> Dentro de los negocios ya se maneja un concepto de servicio especial, algunas empresas lo denominan V.I.P, Platinum, Oro, etc.

- Apoyándose en técnicas de Inteligencia Artificial puede diseñarse un algoritmo de negociación que le permita a los agentes contar con mayor inteligencia al realizar las transacciones y por lo tanto aprovechar de forma más eficiente su autonomía y ofrecer mejores resultados a los usuarios.
- El diseño de la arquitectura puede extenderse para soportar la transacción del tipo subasta electrónica, en la cual si un producto presenta alta demanda y poca oferta, varios compradores puedan intentar acceder a él. El establecimiento de esa clase de competencia, favorece a los vendedores y permite a los compradores que no tienen la opción por compra directa, intentar adquirir el producto. Para esto es necesario, además de enriquecer el diseño de componentes, incrementar la inteligencia de los agentes participantes en la subasta.
- De la misma forma que la arquitectura contempla un Servidor de Compradores, puede realizarse un estudio de requerimientos para ampliar el diseño y proveer un Servidor de Vendedores agrupados en un Mercado Electrónico; de esta manera se da a través de un Proveedor de Servicios acceso a vendedores que no manejan gran cantidad de productos y así realizar transacciones sin necesidad de establecer un sitio en Internet para cada uno de ellos.

## REFERENCIAS

- [AGLETS04] THE AGLETS PORTAL. [En línea]. US: 2004. <http://sourceforge.net/projects/aglets/>. (Consulta: mayo de 2004).
- [BASS03] BASS, Len; CLEMENTS, Paul and KAZMAN, Rick. Software Architecture in practice, Second Edition. Addison-Wesley. 2003.
- [BERGER00] BERGER, Sandy. B2B, B2C, C2B. [En línea]. US: 2000. <http://www.compukiss.com/ck/columns/article.cfm?id=48&year=2000> (Consulta: agosto de 2001).
- [BROOKS91] BROOKS, Rodney. Intelligence without representation. Artificial Intelligence, p.139-59, 1991.
- [CASTILLO02] CASTILLO, Sergio. Composición de servicios mediante el Modelo de Agentes Móviles. Tesis Doctoral (Ph.D. en Telecomunicaciones). Departamento de Ingeniería de Sistemas Telemáticos. Universidad Politécnica de Madrid, 2002.
- [CEBALLOS00] CEBALLOS, Francisco Javier. Java 2, Curso de Programación. Editorial Ra-Ma. Madrid, España. 2000.
- [CHESS97] CHESS, David; HARRISON, Colin and KERSHENBAUM, Aaron. Mobile Agents: Are they a good idea?. Mobile object systems: Toward the programmable Internet, Springer-Verlag, Lecture Notes in Computer Science No. 1222: 25-45. 1998

- [CLARKE98] CLARKE, Roger. Electronic Data Interchange (EDI): An Introduction. Xmax Consultancy Pty Ltda, Australian National University, 1998.
- [COMISION99] COMISION EUROPEA. E-Commerce – An Introduction. [En línea]. 1999. <http://www.ispo.cec.be/ecommerce/answers/introduction.html> (Consulta: octubre de 2003)
- [CRYPTIX04] THE CRYPTIX FOUNDATION LIMITED. Cryptix. [En Línea]. US:2004. <http://www.cryptix.org/> (Consulta: enero de 2004).
- [DASGUPTA01] DASGUPTA, P; MOSER, L.E. and MELLIAR-SMITH, P.M. MAgNET: A Mobile Agent Ecommerce System. Department of Electrical and Computer Engineering University of California, Santa Barbara, CA 93106. 2001.
- [DAVILA00] DAVILA, Jorge. El Comercio Electrónico todavía está por llegar. Asociación Española de Criptología y Seguridad de la Información. España. 2000.
- [FONSECA01] FONSECA, Steven; GRISS, Martin and LETSINGER, Reed. An Agent-Mediated E-Commerce Environment for the Mobile Shopper. Software Technology Laboratory. HP Laboratories Palo Alto. 2001
- [GENERAL94] GENERAL MAGIC, Inc. Telescript technology: The foundation for the electronic marketplace, 1994. White paper.
- [GLASS99] GLASS, Graham. ObjectSpace Voyager Core Package Technical Overview. Mobility: process, computers and agents, Addison-Wesley, 1999.

- [GOMEZ04] GOMEZ, Janeth; CASTILLO, Sergio y LEON, Luis. ADAM: Aplicación de los Agentes Móviles al Comercio Electrónico. Tesis de Pregrado (Ingeniería de Sistemas). Universidad Industrial de Santander. Colombia. 2004.
- [HERRERA99] HERRERA, Marisol; VEGAS, Andrés y VILLENA, Julio. Agentes Móviles y Recuperación de Información. Universidad Politécnica de Madrid, Departamento de Sistemas Telemáticos. 1999.
- [IGLESIAS98] IGLESIAS, Carlos. Definición de una metodología para el desarrollo de sistemas multiagente. Tesis Doctoral (Ph.D. en Telecomunicaciones). Dpto. de Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid, 1998.
- [KABAY04] KABAY, Michel. A Primer on Public Key Infrastructures. Program Master of Science in Information Assurance. Division of Business & Management. Norwich University. US. 2004
- [KALAKOTA97] KALAKOTA, Ravi and WHINSTON, Andrew. Electronic Commerce, a Manager's Guide. Addison-Wesley. 1997.
- [KING95] KING, James. Intelligent Agents: Bringing Good Things to Life. AI Expert, February 1995.
- [LANGE98] LANGE, Danny and OSHIMA, Mitsuru. Programming and Deploying Java Mobile Agents with Aglets. Addison Wesley. US. 1998.
- [LANGE99] LANGE, Danny and OSHIMA, Mitsuru. Seven Good Reasons for Mobile Agents. Communications of the ACM, vol.42, No.3, pp.88-89, Mar.1999.

- [LECKY96] LECKY-THOMPSON, Guy. University of Maryland Baltimore County UMBC. AgentWeb. 1996.
- [LFCIA03] LABORATORY OF FOUNDATIONS IN COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE – LFCIA. Conceptos básicos de Internet y Comercio Electrónico. University of Corunna. 2003.
- [MAN01] MAN, Mo Chun and WEI, Victor. Man M. C. , Wei V. K. A Taxonomy for Attacks on Mobile Agent. IEEE EUROCON'2001 Trends in Communications, International Conference. 2001.
- [MASCAGNI04] MASCAGNI, Michael. Complexity and Analysis of Data Structures and Algorithms. Department of Computer Science. Florida State University. US. 2004.
- [MYSQL04] MYSQL AB. The MySQL Database. [En Línea]. US:2004. <http://www.mysql.com/> (Consulta: febrero de 2004).
- [NGUYEN02] NGUYEN, Giang and DANG T. Tung Agent Platform Evaluation and Comparison. Institute of Informatics. Slovak Academy of Sciences. . 2002.
- [OAKS01] OAKS, Scott. Java Security, Second Edition. O'reilly & Associates, Inc. US. 2001.
- [PRESIDENCIA00] PRESIDENCIA DE LA REPUBLICA, Ministerio de Comercio Exterior. Política Nacional para la Productividad y Competitividad. Colombia. 2000.
- [PRESSMAN02] PRESSMAN, Roger. Ingeniería del Software, Un enfoque práctico. McGraw Hill, 5ª Edición, Madrid, España. 2002

- [PULIDO99] PULIDO, Karla. EDI y XML en el Comercio Electrónico entre empresas. Tesis (Magíster en Ciencias de la Computación). ITESM - CCM, 1999.
- RAMIO04 RAMIÓ, Jorge. Seguridad Informática y Criptografía. Versión 3.2. [Libro Electrónico]. Universidad Politécnica de Madrid. 2004. Disponible en: <http://www.criptored.upm.es/> (Consulta: mayo de 2004)
- [SMART04] SMART, Nigel. Public Key Infrastructure. Department of Computer Science. University of Bristol, UK. 2004.
- [SUN03] Sun Microsystems, Inc. Java 2 SDK, Standard Edition Documentation. Version 1.4.2. US:2003. <http://java.sun.com/>. (Consulta: agosto de 2003).
- [SUN04a] Sun Microsystems, Inc. Java Technology. [En Línea]. US:2004. <http://java.sun.com/> (Consulta: abril de 2004).
- [SUN04b] Sun Microsystems, Inc. Java Remote Method Invocation (Java RMI). [En Línea]. US:2004. <http://java.sun.com/products/jdk/rmi/> (Consulta: abril de 2004).
- [TILAB04] TILAB. JADE - Java Agent DEvelopment Framework. [En línea] Italia, 2003. <http://jade.tilab.com/> (Consulta: abril de 2004).
- [TSUJI00] TSUJI, Hidekazu. Security Technology for Electronic Commerce. Electronic Commerce Promotion Council. Japan. 2000.
- [VIGNA98] VIGNA, Giovanni. Mobile Agents and Security, Lecture notes in Computer Science. Springer-Verlag, 1998.

- [WOOLDRIGE95] WOOLDRIDGE, Michael and JENNINGS, Nick. Intelligent Agents: Theory and Practice. [En línea] 1995. <http://www.doc.mmu.ac.uk/STAFF/mike/ker95/ker95-htm>. (Consulta: mayo de 2002).
- [WANG03] WANG, Ying-Hong; WANG, Ching-Ling; HUANG, Hui-Hua and LO, Wen-Hung. Applying Mobile Agents to E-business. Department of Computer Science & Information Engineering. Tamkang University. Taiwan. Tamkang Journal of Science and Engineering. 2003.
- [ZHAO00] ZHAO, Jian and BLUM, Thomas. Next-Generation E-Commerce: XML+Mobile Agent+Trust. Fraunhofer CRCG, Providence, Rhode Island, USA. 2000.