

Universidad de las Ciencias Informáticas

Facultad 3



Título: “Especificación de un Sistema Multi-Agente para la Gestión de Ventas Personalizadas.”

Trabajo de Diploma para optar por el título de
Ingeniero Informático

Autores:

Ana Margarita Gutierrez Carbonell

Silvia María Llarch Leyva

Tutor:

Ing. Rolando Pérez Pinto

Ciudad Habana, Mayo 2008

*“Todo aquello que puede concebir la mente de un hombre, se puede
lograr”*

W. Clement Stone

DECLARACIÓN DE AUTORÍA

Declaramos que somos las únicas autoras de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmamos la presente a los ____ días del mes de _____ del año _____.

Ana Margarita Gutierrez Carbonell

Silvia María Llarch Leyva

Firma del Autor

Firma del Autor

Ing. Rolando Pérez Pinto

Firma del Tutor

AGRADECIMIENTOS

De Ana:

“A mi familia: mis abuelos que en todo momentos se que han estado muy cerca, cuidando y guiando mis pasos, a mi mamá y mi papá que siempre me han brindado gran apoyo, y han sabido guiarme por el camino correcto, y a mis cinco hermanos a quienes agradezco toda su preocupación, cariño y ternura.

A mi novio Eric, que significa mucho para mí y siempre ha estado a mi lado, apoyándome en los momentos difíciles y compartiendo mis alegrías.

A Silvia que ha sido una buena compañera de tesis y ha compartido los malos ratos y las preocupaciones durante este tiempo.

A todas aquellas personas que de una forma u otra han aportado un granito de arena a mi formación personal y al desarrollo de este trabajo.”

De Ana y Silvia:

“Agradecemos a nuestro tutor, el Ing. Rolando Pérez Pinto que con gran empeño ha sabido transmitir sus experiencias y conocimientos. Gracias a su dedicación nos ha sido posible realizar este trabajo.”

De Silvia:

“Mis agradecimientos estarán ordenados cronológicamente incluyendo a todas aquellas personas que en algún momento de mi vida influyeron aportando un granito de arena para que yo pudiera llegar hasta este preciado momento que es el de graduarme.

A mi mamá Silvia Leyva Leyva y a mi papá Arturo Llarch Rodríguez pues ustedes son los que siempre han estado a mi lado y me han apoyado en todo momento, me han aconsejado y me han guiado siempre por el buen camino, ustedes son los mejores padres del mundo. . .

A Sarita por ser mi hermanita, la única que tengo.

A mi abuela Nenita por ser la que más me extraña cuando no estoy en la casa, y por haberme dormido tantos años en sus piernas.

A Francis Pérez Perdomo mi querido profesor guía de primer año quien siempre me ayudó en ese difícil año.

A mi amiga Yissel y a sus padres María y Mario por todo el apoyo que siempre me han brindado.

A mi novio Alejandro por ayudarme siempre a estudiar y por darme la sorpresa mas linda del mundo.

A Mercedes por estar siempre en contacto conmigo (aunque sea vía telefónica) y por estar siempre pendiente de todo lo que me pasa y ayudándome en todo momento.

A Fabriciano por todo el tiempo que empleó en revisarme la Tesis y a Tania porque gracias a ella termine el trabajo en tiempo.

A AM por tener que soportarme tanto tiempo sentada a su lado mientras hacíamos la tesis.

A Barbara por ser mi compañera en los interminables viajes a Guantánamo en tren y en guagua.

A Roig, e Isledis, Daniel y Mairelis, Adonis y Daymi.

A todos mis compañeros, los que empezamos desde primer año(Linnet, Marbelis, Dagnalia, Yusel, Jessie, Henry, Denny, Yunieski, Yurita, Xiosmel, Leo, Lisandra, Yoanki, Carlos M, Fidel, Yunier, Rete, Lisbany, Sandy, El fula, Ana Leiris) y en especial a Maribel por brindarme siempre su hombro para llorar en los peores momentos, a Rolando y Ariel que siempre he podido contar con ellos para lo que necesite y porque conocimos los teatros juntos, a Baby por siempre acompañarme a la Habana, a Yitselis por siempre buscarme agua en el comedor ,a Yene por el café que le quité mientras trabajaba en la tesis, para la próxima te ganarás el nombre completo.”

DEDICATORIA

De Silvia:

Esta tesis esta dedicada a la persona más especial de mi vida, mi abuelo Vicente Llarch Aviñó, que aunque en estos momentos no esté presente físicamente, si lo está espiritualmente conmigo en todo momento y es por eso abuelito y por todas las cosas lindas que siempre me haz enseñado que quiero que sepas donde quiera que estés, que tu fuiste, eres y serás siempre mi ídolo, la persona que más yo he admirado desde que naci.

Te quiero mucho....

De Ana:

A mis seres más queridos quienes representan lo más especial en mi vida; ellos son: mi familia (mis abuelos, mi mamá, mi papá y mis hermanos) y mi novio.

Son mi gran tesoro...

RESUMEN

La Inteligencia Artificial ha logrado numerosos avances en los últimos diez años. Así el Paradigma Orientado a Agente, como parte de esta ciencia de la computación, ha hecho numerosos aportes en el desarrollo de complejos sistemas. Los Agentes Inteligentes, y específicamente su integración con los Sistemas Multi-Agentes, han sido un elemento esencial en el desarrollo de aplicaciones destinadas a la robótica y al sector de los servicios dado su nivel de autonomía y su factibilidad en complejos procesos, lo que favorece la creación de estos sistemas.

El presente trabajo propone un acercamiento a la tecnología de agentes. Se trata del desarrollo de una arquitectura de un Sistema Multi-Agente para el Comercio Electrónico, específicamente para la empresa Cubalse.

Se propone el diseño de un sistema que garantiza el trato personalizado a los clientes, mediante sugerencias de productos, recolector de información especializada y el aprovechamiento de las fluctuaciones de precios. Para la obtención de la arquitectura planteada se desarrollan todas las fases del ciclo de vida que plantea la metodología PASSI, la cual fue seleccionada luego de un estudio de las metodologías existentes para el diseño de Sistemas Multi-Agentes.

Los resultados de la investigación quedan ilustrados en la obtención de una guía que abarca las fases de Análisis, Diseño e Implementación del sistema y que ha sido validada por personalidades dedicadas al estudio de la Inteligencia Artificial especialmente de Sistemas Multi-Agentes.

PALABRAS CLAVE

Sistemas Multi-Agentes, Comercio Electrónico, Ingeniería de Software Orientada a Agentes

Tabla de Contenidos

AGRADECIMIENTOS	I
DEDICATORIA	III
RESUMEN	IV
INTRODUCCIÓN	1
CAPÍTULO 1: INGENIERÍA DE SOFTWARE ORIENTADA A AGENTES APLICADA AL SISTEMA DE COMERCIO ELECTRÓNICO PARA LA EMPRESA CUBALSE	6
1.1. Introducción.....	6
1.2. Los Agentes Inteligentes.....	6
1.3. Sistemas Multi-Agentes.....	9
1.3.1. Interacción en Multi-Agentes.....	10
1.3.2. Cooperación en Multi-Agentes.....	11
1.3.3. Colaboración y Coordinación entre agentes.....	12
1.3.4. Comunicación entre agentes.....	12
1.4. Sistemas Multi-Agentes en aplicaciones de Comercio Electrónico	13
1.5. Características de la empresa Cubalse.....	17
1.6. La Ingeniería de Software Orientada a Agentes	18
1.6.1. El Paradigma Orientado a Agentes	18
1.6.2. Análisis de las principales metodologías de desarrollo de Sistemas Multi-Agentes.....	20
1.6.2.1. Metodología Prometheus.....	21
1.6.2.2. Metodología PASSI.....	23
1.6.2.3. Metodología Ingenias.....	25
1.7. Selección de la Metodología de Desarrollo	28
1.8. Conclusiones.....	30
CAPÍTULO 2: LA METODOLOGÍA PASSI APLICADA AL SISTEMA DE COMERCIO ELECTRÓNICO PARA LA EMPRESA CUBALSE	31
2.1. Introducción.....	31
2.2. Descripción del proyecto Plataforma de Comercio Electrónico B2B.....	31
2.3. Modelo de Requerimiento del Sistema.....	32
2.3.1. Descripción de Entorno.....	33
2.3.2. Descripción de Dominio.....	34

2.3.3.	Identificación de agentes	39
2.3.4.	Identificación de Roles	47
2.3.5.	Especificación de Tareas	59
2.4.	Modelo de Sociedades de Agentes	63
2.4.1.	Descripción de Ontología	63
2.4.2.	Descripción de Roles	68
2.4.3.	Descripción de Protocolos	70
2.5.	Modelo de Implementación de Agente	71
2.5.1.	Definición de Estructura del Sistema Multi_Agente	71
2.5.2.	Definición de Estructura del Agente	74
2.5.3.	Descripción de Conducta del Sistema Multi_Agente	76
2.6.	Modelo de Código	79
2.6.1.	Biblioteca de Código Reutilizable	79
2.6.2.	Perfeccionamiento del Código Básico	82
2.7.	Modelo de Despliegue	82
2.7.1.	Configuración de Despliegue	82
2.8.	Conclusiones	85
CAPÍTULO 3: ANÁLISIS DE LOS RESULTADOS OBTENIDOS		86
3.1.	Introducción	86
3.2.	Análisis de los resultados obtenidos	86
3.3.	Método de validación de expertos	89
3.4.	Validación de expertos mediante el método Delphi aplicado a la modelación del sistema de Comercio Electrónico para la empresa Cubalse90	
3.4.1.	Planificación del criterio de expertos	91
3.4.2.	Elaboración y aplicación de las encuestas	93
3.4.3.	Procesamiento y análisis de la información.	94
3.5.	Conclusiones	96
CONCLUSIONES		97
RECOMENDACIONES		98
BIBLIOGRAFÍA		99
ANEXOS		¡ERROR! MARCADOR NO DEFINIDO.
Anexo 1a). Fragmento de código XML		¡Error! Marcador no definido.

Anexo 1b). Fragmento de código RDF..... ¡Error! Marcador no definido.

Anexo 1c). Código JAVA generado a partir del Diagrama Descripción de Ontología de Dominio. ¡Error! Marcador no definido.

Anexo 2: Protocolo FIPA Request (Demanda) ¡Error! Marcador no definido.

Anexo 3: Biblioteca de Código Reutilizable..... ¡Error! Marcador no definido.

Anexo 4a). Aval del Experto..... ¡Error! Marcador no definido.

Anexo 4b). Análisis para la selección de expertos. ¡Error! Marcador no definido.

Anexo 5: Framework de evaluación de metodologías orientadas a agentes¡Error! Marcador no definido.

Anexo 6a). Encuesta aplicada a los expertos con un valor cuantitativo según posibles respuestas. ¡Error! Marcador no definido.

Anexo 6b). Encuesta realizada a uno de los expertos..... ¡Error! Marcador no definido.

Anexo 7: Obtención del Coeficiente de Concordancia..... ¡Error! Marcador no definido.

INTRODUCCIÓN

El Comercio Electrónico se ha convertido en los últimos años en una de las mayores expresiones del comercio mundial tanto de Empresa-Empresa o en inglés Business to Business (B2B) como de Empresa-Usuario o en inglés Business to Consumer (B2C). Desde sus inicios se ha manifestado la necesidad de mejoras y el desarrollo de las tecnologías ha sido a lo largo de los años un factor esencial en la búsqueda de soluciones. En este sentido se han logrado numerosos avances, pues se lucha por la satisfacción plena del cliente.

Son varios los factores que pueden incidir en el éxito de aplicaciones destinadas al Comercio Electrónico, así lo reflejan varios sitios dedicados a esta rama comercial, entre ellos: Ciber-ventas(1), Alta Voz(2), Smart Sales(3), Mercado Libre(4), etc. A continuación se mencionan algunos de estos factores:

- Proporcionar ventajas al cliente: se refiere a ofrecer una línea de productos de atracción para los clientes a un precio competitivo, como ocurre en el entorno no electrónico.
- Proporcionar servicio y ejecución: ofrecer una compra amigable interactuando con el cliente, semejante a una compra cara a cara.
- Proporcionar una página Web atractiva: el uso de colores, gráficos, fotografías y distintas tipografías puede aumentar el éxito en este sentido.
- Incentivar a los clientes: se trata de lograr atraer al cliente y convertirlo en un seguidor del sitio. Las promociones de ventas pueden incluir ofertas especiales y descuentos. Las Webs unidas por links y los programas de publicidad pueden ayudar en este aspecto.
- Proporcionar atención personal: la Web personalizada, debe incluir sugerencias de compra y ofertas especiales personalizadas, logrando así un sitio más interactivo con el cliente y más cercano a sus gustos.
- Optimizar los procesos del negocio: el uso de las tecnologías debe proporcionar rapidez y optimización en los procesos de compra.

- Sistemas de ayuda: darle oportunidad al cliente de aclarar sus dudas a través de un sistema de ayuda en línea.
- Construir una organización con suficiente agilidad y sistemas de alerta para responder rápidamente a los cambios en el entorno económico, social y físico de una empresa.
- Proporcionar confianza y seguridad: lograr que el cliente sienta seguridad en la red, para ello se debe hacer uso de servidores paralelos, redundancia de hardware, tecnología de seguridad ante fallas, encriptación de la información y cortafuegos.

Muchas son las aplicaciones de software que se han desarrollado con el fin de proporcionar un mejor servicio. Se cuenta con sistemas automatizados capaces de realizar transacciones online, recomendaciones a clientes, seleccionar el mejor producto, entre otras; aún así resta mucho por hacer, fundamentalmente en el factor personalización. Aunque actualmente existen sistemas que brindan la posibilidad de modificar la interfaz del sitio o de elegir el mejor proveedor se hace necesario lograr mejor interacción con los clientes de una forma autónoma. Se trata de sugerencias de productos acorde con las acciones de los clientes, las características de sus compras, sus preferencias y las variaciones de precios, garantizando el propio aprendizaje del sistema en función del usuario.

En Cuba el Comercio Electrónico ha manifestado un lento crecimiento desde sus inicios en 1996(5). Existen algunas empresas que promueven su desarrollo, entre ellas la Empresa de Tecnologías de la Información y Servicios Telemáticos Avanzados(CITMATEL), que brinda a las empresas nacionales la posibilidad de comercializar sus productos y servicios dentro del país y hacia el exterior a través de la creación de tiendas virtuales específicas, con las cuales se realiza una labor de posicionamiento, además de garantizar todos los mecanismos adecuados para asegurar de manera eficiente el pago electrónico(6); pero la necesidad de avanzar en esta esfera es evidente . Se requieren sistemas que se centren en el comercio cubano y las características de su mercado, debido a que: circulan dos monedas, existen entidades priorizadas por políticas estatales, lo que afecta los compromisos con diversos clientes. Esta situación conduce a la búsqueda de soluciones

encaminadas a resolver el siguiente Problema Científico: “¿Cómo guiar el desarrollo de un Sistema Multi-Agente mediante técnicas de Ingeniería de Software Orientada a Agentes que garantice un trato personalizado a los clientes y el aprovechamiento de las fluctuaciones de precios? ”

Objeto de estudio:

La Ingeniería de Software Orientada a Agentes.

Campo de acción:

Modelación de un Sistema Multi-Agente de Comercio Electrónico para la empresa Cubalse.

Objetivo General:

Modelar un Sistema Multi-Agente para la Gestión de Ventas Personalizadas en el proyecto Plataforma de Comercio Electrónico Empresa–Empresa (B2B) para la empresa Cubalse.

Tareas a desarrollar:

Con el fin de dar cumplimiento a los objetivos trazados se proponen las siguientes tareas a desarrollar:

- Realizar un levantamiento del estado del arte de la aplicación de Sistemas Multi-Agentes en el desarrollo de sistemas para el Comercio Electrónico.
- Realizar un estudio de las metodologías existentes para el modelado de los Sistemas Multi-Agentes y seleccionar una de ellas.
- Desarrollar los pasos necesarios según la metodología seleccionada y obtener los artefactos que esta propone.

- Realizar un análisis de los resultados obtenidos y el aporte a la solución del problema planteado.

Estrategias y Métodos de investigación a utilizar:

Estrategia de Investigación:

En el desarrollo de la investigación se utiliza la *Estrategia Descriptiva* para describir el fenómeno relacionado con la personalización de aplicaciones para el Comercio Electrónico y el empleo de Sistemas Multi-Agentes.

Métodos de Investigación:

Para llevar a cabo esta investigación se abordan los principios relacionados con los métodos teóricos específicamente los que a continuación se describen:

Analítico Sintético: mediante el cual se realiza una investigación previa sobre la situación del Comercio Electrónico actualmente, específicamente en el caso de Cuba, lo que permite analizar y estudiar documentos importantes y extraer algunos elementos relevantes relacionados con el objeto de estudio.

Método de la Modelación: el cual permite la creación de modelos para la descripción del sistema, y con ello encaminar la solución del problema, así como contribuir al logro de los resultados esperados.

Método Sistémico: permite el estudio de los diferentes componentes de los sistemas de Comercio Electrónico y sus características, específicamente en Cuba, lo que determina la estructura y dinámica del sistema.

Método Particular de Encuesta: se utiliza para la recuperación de información relacionada con la evaluación del trabajo realizado, en especial para la validación mediante la opinión de un grupo de expertos en el tema a desarrollar.

Resultados Esperados:

Al concluir este trabajo se debe disponer de una guía que abarque todos los artefactos generados durante las etapas de análisis, diseño e implementación según

lo plantee la metodología de desarrollo de software utilizada, de manera que conduzca a la implementación del sistema propuesto.

El presente trabajo ha sido organizado de la siguiente manera:

Capítulo 1: En este capítulo se hace un estudio general de aspectos necesarios para la comprensión teórica analizando desde el Paradigma Orientado a Agentes y su influencia en el Comercio Electrónico, hasta el análisis de las metodologías que propone la Ingeniería de Software Orientada a Agentes haciendo una valoración crítica de las más importantes y la selección de una de ellas.

Capítulo 2: Comprende una serie de artefactos como resultado de aplicar la metodología seleccionada a un sistema para Comercio Electrónico en Cuba, específicamente para la empresa Cubalse. Estos artefactos incluyen además la documentación requerida que permite a los programadores un mejor desempeño y la mejor comprensión por parte de los clientes.

Capítulo 3: En este capítulo se realiza un análisis detallado del cumplimiento de los objetivos propuestos, los resultados obtenidos como parte de la investigación realizada y el aporte del presente trabajo a la solución del problema planteado; así como la validación de los modelos propuestos mediante un criterio de validación de experto.

CAPÍTULO 1: INGENIERÍA DE SOFTWARE ORIENTADA A AGENTES APLICADA AL SISTEMA DE COMERCIO ELECTRÓNICO PARA LA EMPRESA CUBALSE

1.1. Introducción

El paradigma de agentes y Sistemas Multi-Agentes(SMA) constituye actualmente un área de creciente interés dentro de la Inteligencia Artificial , entre otras razones, por ser aplicable a la resolución de problemas complejos no resueltos de manera satisfactoria mediante técnicas clásicas(7). Para una mejor comprensión de esta tecnología, se abordan conceptos y aspectos relevantes del paradigma y su aplicación al Comercio Electrónico, llegando finalmente a la elección de una metodología que permita modelar un sistema para la realización de ventas personalizadas.

1.2. Los Agentes Inteligentes

En la Inteligencia Artificial ha surgido un paradigma conocido como «paradigma de agentes», el cual ha tomado un gran auge entre los investigadores. Existe una idea generalizada de que se trata de la revolución de una gran tecnología, la de Agentes Inteligentes que permite abordar de una manera más apropiada la creación de sistemas inteligentes(8).

Los Agentes Inteligentes constituyen un avance significativo en el mundo de la informática, los cuales están enfocados a la realización de complejas tareas que requieren de cierto grado de inteligencia, además pueden ser considerados sistemas autónomos que se centran en su propio aprendizaje y resultan de gran interés por contribuir a la resolución de problemas complejos de difícil solución mediante técnicas clásicas.

Se pueden encontrar propuestas en la literatura un gran número de definiciones del concepto de agente, sin que ninguna de ellas haya sido plenamente aceptada por la comunidad científica, siendo quizás la más simple la de Russell, que considera un agente como una entidad que percibe y actúa sobre un entorno(8). Además, se mencionan importantes definiciones como: “un proceso computacional capaz de

realizar tareas de forma autónoma y que se comunica con otros agentes para resolver problemas mediante cooperación, coordinación y negociación. Habita en un entorno complejo y dinámico con el cual interacciona en tiempo real para alcanzar un conjunto de objetivos”(9), otra definición y una de las más populares es la planteada por Wooldridge y Jennings desde sus inicios. “Un Agente Inteligente es una entidad software que, basándose en su propio conocimiento, realiza un conjunto de operaciones para satisfacer las necesidades de un usuario o de otro programa, bien por iniciativa propia o porque alguno de estos se lo requiere”(10) y finalmente se puede tomar como “una entidad de software capaz de actuar en un ambiente (comunicándose directamente con otros agentes mediante un Lenguaje de Comunicación de Agentes), que se rige por un conjunto de tendencias (bien en forma de objetivos individuales o de una función de satisfacción/supervivencia), que posee recursos propios y que es capaz de percibir su entorno en un contexto limitado. Incorpora una representación parcial de este ambiente, posee habilidades, y puede ofrecer servicios. Puede reproducirse, y su comportamiento tiende a satisfacer sus propios objetivos, tomando en cuenta los recursos y habilidades disponibles a él en dependencia de su percepción, representación y las comunicaciones que recibe”(11), esta última definición planteada por la Fundación para Agentes Físicos Inteligentes FIPA (del inglés FIPA, Foundation for Intelligent Physical Agents) es una de las más completas.

A pesar de las divergencias entre los distintos conceptos, muchos son los autores, (8), (9),(10),(11),(12) que coinciden en que los Agentes Inteligentes poseen las siguientes características:

- Autonomía: los agentes pueden actuar sin la intervención de un ser humano o de otros agentes, y tienen algún tipo de control sobre sus acciones y su estado interno, tienen la capacidad de operar independientemente de la existencia de otros agentes, de sobrevivir en ambientes dinámicos sin la supervisión de un control externo, y de tomar decisiones sobre su comportamiento considerando sólo las percepciones, conocimientos y representaciones de que dispone.

- Sociabilidad o Cooperación: los agentes tienen capacidades para interactuar con otros agentes humanos o no, a través de algún tipo de lenguaje de comunicación.
- Reactividad: los agentes tienen capacidades para percibir estímulos de su entorno, (que puede ser el mundo físico, un usuario a través de una interfaz gráfica, una colección de otros agentes, Internet, o todos combinados), y para responder de manera oportuna a los cambios que suceden en él.
- Pro-actividad o Iniciativa: los agentes no responden únicamente a su entorno, sino que son capaces de dirigir su comportamiento, de forma emprendedora, respecto a sus propias metas u objetivos; es decir, son capaces de “tomar la iniciativa”.

Si bien es importante conocer estas características, también se debe señalar que, aunque no son de tal relevancia, se tratan otros aspectos que responden al comportamiento de los Agentes Inteligentes, pues estos son algo más específico.

- Continuidad Temporal: se considera un agente un proceso sin fin, ejecutándose continuamente y desarrollando su función.
- Racionalidad: el agente siempre realiza «lo correcto» a partir de los datos que percibe del entorno.
- Adaptabilidad: está relacionado con el aprendizaje que un agente es capaz de realizar y si puede cambiar su comportamiento basándose en ese aprendizaje.
- Movilidad: capacidad de un agente de trasladarse a través de una red telemática.
- Veracidad: seguridad de que un agente no comunica información falsa a propósito.
- Benevolencia: certeza de que un agente está dispuesto a ayudar a otros agentes si esto no entra en conflicto con sus propios objetivos.

Los agentes proporcionan una serie de ventajas respecto de otras técnicas informáticas distribuidas:

- Los usuarios que utilizan sistemas informáticos con baja capacidad de proceso, los que tienen un bajo ancho de banda en la red o los que disponen de poco espacio de almacenamiento pueden utilizar agentes para solucionar sus necesidades, desconectando su sistema informático de la red o pidiendo que otro sistema realice los cálculos y recogiendo posteriormente los resultados.
- Aplicaciones como el Comercio Electrónico, mejoran sus prestaciones, ya que el usuario puede dar instrucciones a un agente para que realice la compra cuando se cumplan las condiciones óptimas. El proceso es más rápido y económico porque evita tener que hacer comunicaciones periódicas para comprobar el estado del producto(13).

Se debe mencionar que en la mayoría de las ocasiones los Agentes Inteligentes no se desarrollan de forma independiente, sino que actúan como un sistema. Es en ese momento que manifiestan en su totalidad características antes expresadas como por ejemplo la sociabilidad/cooperación y la benevolencia.

1.3. Sistemas Multi-Agentes

Cuando un agente convive con otros agentes en un ambiente compartido, el concepto de acción adquiere un nuevo significado. El agente ya no puede ser considerado una entidad aislada y única dentro de un ambiente que responde exclusivamente a sus acciones. Ahora el ambiente incluye explícitamente a otros agentes, que pueden compartir o no sus intereses. Estos agentes también actúan y las influencias recíprocas que tienen sus acciones (interacciones), necesitan ser consideradas explícitamente. Se podría afirmar por lo tanto que así como el concepto de acción constituye la base de la metáfora de agentes, la idea de interacción es la esencia fundamental de los SMA(14).

Un SMA puede ser definido como una red débilmente acoplada de agentes que trabajan conjuntamente para resolver problemas que superan las capacidades individuales o conocimientos de cada uno de ellos(15).

Los SMA realizan una serie de acciones basadas en la coordinación inteligente entre

un grupo de agentes, se trata de coordinar conocimientos, metas, propiedades y planes para tomar una decisión o resolver un problema, dado que se trata de sistemas autónomos compuestos por un conjunto de elementos:

1. Un entorno
2. Un conjunto de objetivos. Estos objetivos se encuentran integrados en el entorno, es posible en un momento dado asociar uno de estos objetivos con un lugar en el entorno. Estos objetivos son pasivos, pueden ser percibidos, creados, destruidos y modificados por agentes.
3. Un conjunto de agentes que se considera como objetos especiales que representan las entidades activas del sistema.
4. Un conjunto de relaciones que unen objetos y por lo tanto, agentes.
5. Un conjunto de operaciones que hacen posible que los agentes perciban, produzcan, consuman, transformen y manipulen objetos.
6. Operadores que representan la aplicación de operaciones sobre el mundo y la reacción de este al ser alterado. Estos operadores se pueden entender como las leyes del universo(16).

1.3.1. Interacción en Multi-Agentes

Sin lugar a dudas, un aspecto fundamental a tener en cuenta en el análisis de un SMA es el relacionado con la interacción de los agentes. Es a partir de la complejidad y flexibilidad de estas interacciones, que los SMA obtienen un valor adicionado como un todo, que no se podría lograr si los agentes trabajaran en forma aislada. A continuación se exponen tres formas de interacción que toman como eje los objetivos personales de los agentes, sus interdependencias y su relación con los objetivos de la sociedad como un todo.

Interacciones cooperativas: este tipo de interacción se caracteriza fundamentalmente por la existencia de una tarea u objetivo común compartido por todos los agentes. Las interacciones de los agentes están dirigidas directa o indirectamente por el cumplimiento de este objetivo común, por lo que el SMA puede ser considerado

como un equipo motivado exclusivamente por el interés social.

Interacciones antagónicas: en este tipo de situaciones los agentes son antagónicos, es decir, tienen objetivos opuestos. En este caso, el objetivo de un agente es derrotar al otro, o lograr que el otro agente no cumpla su objetivo. La comunicación ya no es importante, y no se asume ningún tipo de colaboración por parte del oponente.

Interacciones individualistas: los objetivos sociales no son la única componente que incluye el comportamiento de los agentes. Cada agente puede ser diseñado por personas o instituciones diferentes con sus propios objetivos privados. Los agentes se comportarán como representantes de estas instituciones e intentarán defender estos objetivos, actuando en forma egoísta, si esto le produce algún beneficio(14).

1.3.2. Cooperación en Multi-Agentes

En general no existe una visión global de la cooperación, y más bien son estudios sobre la ventaja de la cooperación, abordada desde una perspectiva referida a cómo llevar a cabo la cooperación o a cómo relacionar los agentes para que estos cooperen.

- La cooperación como una postura intencional: es una postura que adoptan intencionalmente los agentes, por lo que esta definición de cooperación no es aplicable a los agentes reactivos.
- La cooperación desde el punto de vista del observador: se considera que un conjunto de agentes cooperan cuando un observador externo puede describir como cooperativas las acciones realizadas por los agentes.
- La cooperación como el incremento en la capacidad de supervivencia: cada agente tiene una capacidad de supervivencia que puede verse incrementada al existir un conjunto de agentes que se organizan para llevar a cabo las tareas de supervivencia.
- La cooperación como una manera de mejorar las prestaciones del sistema: en este caso, un observador externo evalúa el resultado del grupo de agentes de manera que si aparece una mejora del grupo sobre la suma de individuos se dice que los agentes están cooperando.

- La cooperación como una manera de evitar los conflictos: si los agentes de una comunidad se ven inmersos en conflictos al acceder a recursos comunes y son capaces de resolverlos, se habla de una situación cooperativa.

1.3.3. Colaboración y Coordinación entre agentes

La colaboración entre los agentes es la forma en la que se distribuyen las tareas entre ellos. Esta función de distribución debe realizarse en función de las capacidades de cada agente (capacidades cognitivas, habilidades, conocimientos que posee, compromisos del agente, etc.), de la naturaleza de las tareas, de la estructura social de la organización en la que se encuentran inmersos los agentes. La distribución se realiza en función de los contratos que se acuerdan entre los agentes, y dicha distribución se mantiene porque el agente se compromete a realizar una determinada tarea y ese compromiso pasa a ser un objetivo que debe cumplir el agente.

1.3.4. Comunicación entre agentes

La comunicación entre los agentes les permite sincronizar acciones, enviar y recibir conocimiento, resolver conflictos en la resolución de una tarea, etc. (17). La comunicación es el soporte de la cooperación, coordinación y de los procesos de negociación que se dan entre los agentes. El proceso de comunicación permite que los agentes desarrollen sus acciones con una visión menos local, tras un proceso de sincronización con el resto de los agentes. Sin embargo, una excesiva comunicación puede dar lugar a agentes cuya sobrecarga de comunicación sea mayor que el trabajo efectivo realizado.

Es fundamental que exista un mecanismo adecuado de comunicación entre agentes sobre todo en comunidades Multi-Agentes. Incluso se ha llegado a sugerir que una entidad es un agente de software si y sólo si se comunica correctamente en un lenguaje de comunicación de agentes(18). Para que esta interacción e interoperación entre agentes de software sea significativa, constructiva e inteligente se requieren tres componentes fundamentales y distintos:

- Un protocolo de transporte: formado por el mecanismo de transporte usado para la comunicación (por ejemplo: TCP, SMTP, HTTP, etc.).
- Un lenguaje común: es el medio por el cual se realiza el intercambio de información. Este lenguaje indica cual es el contenido de la comunicación y si es una aseveración, una pregunta o una solicitud.
- Un protocolo de interacción: se refiere a la estrategia que sigue el agente para interactuar con otros agentes, la cual puede ir desde esquemas de negociación y protocolos basados en teoría de juegos hasta protocolos muy simples en los que cada vez que el agente no sabe algo, busca otro que lo sepa y le pregunta(19).

El diseño de SMA, generalmente, se aborda pensando en los agentes como entes con motivación. En lugar de modelar un sistema con componentes que ejecutan métodos, el desarrollador tiene que pensar en los objetivos que los componentes deben alcanzar y en las tareas necesarias para que lo consigan(20).

Los SMA proporcionan una gran ventaja: permiten la integración de agentes existentes en un gran sistema. Por tanto, la solución de un problema no requiere el diseño y desarrollo de un nuevo agente especializado, en su lugar, puede ser utilizado el conocimiento de los agentes existentes combinándolos en un SMA y permitiendo que trabajen conjuntamente para resolver el problema.

1.4. Sistemas Multi-Agentes en aplicaciones de Comercio Electrónico

A medida que la complejidad de las aplicaciones comerciales aumenta, en particular desde la aparición de Internet, existe una necesidad de incorporar abstracciones organizacionales que faciliten su diseño, desarrollo y mantenimiento. Las instituciones electrónicas están en el trasfondo de esta aproximación. Estas proporcionan un análogo computacional de las organizaciones humanas en las que agentes, tanto software (21) como humanos, que adoptan roles diversos interactúan para conseguir objetivos individuales y organizacionales. En este escenario, la tecnología de agentes es candidata a ayudar a la reducción de costes operacionales y a la aceleración del ciclo de desarrollo del producto, al permitir a las compañías representadas por agentes, un funcionamiento más ágil y coordinado (22).

Los agentes son entidades software que poseen la suficiente autonomía e inteligencia como para poder encargarse de tareas específicas con poca o ninguna supervisión humana(23). Existe la necesidad de algún grado de autonomía que permita a los componentes responder dinámicamente a circunstancias cambiantes mientras tratan de conseguir sus objetivos sin la necesidad de intervención del usuario.

Los sistemas de Comercio Electrónico implementan una serie de etapas que se corresponden con el proceso de compra tradicional: se identifican las necesidades, se elige el producto a comprar así como el vendedor a quien se le va realizar la compra. Luego se lleva a cabo un proceso de negociación, compra y envío del producto, finalizando con el uso y evaluación del mismo(12). Además contiene con una realidad preponderantemente digital, para la cual es posible “encapsular” tecnologías de Inteligencia Artificial ya disponibles y aplicarlas en problemas suficientemente acotados para hacerlas exitosas(24). Los diferentes tipos de agentes creados para el Comercio Electrónico constituyen una realidad vigente enfocada en lo anteriormente expresado. A continuación se mencionan algunas de sus funciones y un ejemplo específico donde se han puesto en práctica:

Agentes notificadoros: su misión es avisar a sus usuarios cuando encuentran un producto acorde al que dicho usuario necesita.

Jango(25): Utiliza Internet, para mediante el ingreso del nombre de un producto, buscar automáticamente en las tiendas y sitios más relevantes. Muestra al usuario un informe con todos los datos e incluso ayuda a cumplimentar el formulario de pedido, en caso de que se realice la compra.

Agentes de recomendación: Se basan en el perfil y el conocimiento del contexto de negocio, para poder recomendar a los usuarios los productos de su interés.

FireFly(26): aprende de sus opiniones e influencia y utiliza esa información para cubrir mejor sus necesidades. Usa los gustos, opiniones, preferencias e idiosincrasia para sugerirle una nueva música que pudiera gustarle. Es el más usado y exacto, llegando a ser uno de los mejores. Firefly actualmente funciona en Yahoo.

Agentes de compra comparativa: Los agentes de compra son capaces de hacer comparaciones y encontrar el mejor precio para un artículo. Los vendedores al por

menor pueden protegerse de ellos dando a sus productos nombres únicos o bloqueando el acceso a los agentes de búsqueda

Bargain Finder(27): Agente que busca en varias tiendas los precios más baratos para CD's y cintas. Para realizar la búsqueda a un determinado comerciante, este debe estar suscrito y pagar por aparecer como referencia a los clientes. La búsqueda se realiza por el nombre exacto del artista o álbum, lo cual hace imprescindible que el usuario sepa exactamente lo que desea comprar.

Agentes de negociación: Se encargan, mediante la existencia de agentes compradores y vendedores de llegar a acuerdos para la compra de un producto, como los procesos de negociación que existen para realizar una transacción comercial.

Kasbah(28): es un sistema basado en la Web, a través del cual los usuarios pueden crear agentes autónomos para comprar y vender productos. El cual ha sido diseñado para ayudar a los usuarios en la búsqueda y elección del producto que mejor satisfaga sus necesidades. Así como la posterior negociación entre comprador y vendedor con el fin de obtener el mejor precio posible. Actualmente negocia con 9 clases de productos, tales como CD's y cámaras, y tiene la ventaja de no requerir que los usuarios permanezcan todo el tiempo a la espera de la negociación.

Agentes de subasta: Permite a los compradores competir por un producto en la red.

Michigan AuctionBot: es un servidor experimental de subastas, cuyo principal propósito es permitir que los agentes de los usuarios compitan en la red para conseguir productos en las mejores condiciones posibles. AuctionBot implementa diferentes tipos de subastas y las gestiona paralelamente(29).

Otros agentes

MAGMA(30) (Minnesota Agent Marketplace Architecture) es una arquitectura extensible para un mercado virtual basado en agentes, que incluye todos los elementos requeridos para simular un mercado real. Incluye la infraestructura de comunicaciones, mecanismos para el almacenamiento y la transferencia de los bienes, transacciones bancarias y monetarias, y los mecanismos económicos para las transacciones entre productor y consumidor.

Infilebot Project: recibe una petición del usuario y es capaz de moverse por Internet buscando información relacionada con un tema determinado. También filtra la información obtenida, de acuerdo con las necesidades del usuario y obtiene documentos relevantes para el usuario. Aprende de sus propias experiencias pudiendo entonces descubrir el modo más adecuado de alcanzar objetivos y entiende las peticiones del usuario en lenguaje de alto nivel(31).

La gran cantidad de sitios de Comercio Electrónico que ofertan millones de productos para su venta, se convierte en un caos de información sin solución cuando se necesita realizar una compra eligiendo entre todas las opciones. Los sistemas de recomendación surgen como solución a este problema. Un sistema de recomendación recibe información del usuario acerca de productos en los que se encuentra interesado y le recomienda aquellos cercanos a sus necesidades. Dotar a los sistemas de recomendación de técnicas que personalicen las recomendaciones más allá de la simple petición de productos por parte del usuario, constituye actualmente un importante campo de investigación en el área del Comercio Electrónico, aunando a profesionales de áreas tan diversas como el marketing, la interacción persona-ordenador, la minería de datos y la Inteligencia Artificial (32). Todo lo que se haga a favor de ello será vital para el éxito de aplicaciones de Comercio Electrónico.

La atención personalizada a los clientes, puede obtenerse de varias formas, desde las aplicaciones basadas en navegador y marketing directo, hasta la asistencia al cliente interactiva y en tiempo real. Puede abarcar los siguientes aspectos:

- *Personalización del sitio Web*: mostrar el contenido deseado en una gran variedad de formatos. Por ejemplo, que el cliente pueda seleccionar el contenido en un idioma concreto o según determinados temas.
- *Recomendaciones sobre los productos*: venta asesorada por expertos en productos, en editores o en el comportamiento de otras personas con perfiles similares.
- *Servicio personal*: hacer posible que los clientes interactúen con una persona en tiempo real.
- *Soporte*: ofrecer acceso al cliente a la información sobre sus cuentas y sobre el

estado de los pedidos(33).

1.5. Características de la empresa Cubalse

Cubalse es una empresa de comercio y servicios geográficamente representada en Cuba e integrada por diversas líneas de negocios, que incluyen la comercialización de artículos para el consumo personal, para el hogar y para las empresas; además de la prestación de servicios inmobiliarios, constructivos, gastronómicos, de inversiones y de tecnologías. Recientemente creó la división de programas especiales entre los que se encuentran la Ciudad Tarará y Hostalsalud. Tiene la misión de atender de manera especializada al cuerpo de diplomáticos, residentes extranjeros, agencias de prensa, y organismos internacionales acreditados en el país, a través de las tiendas especializadas, servicios de decoración, empaque y desempaque y la empleomanía a este sector.

La empresa Cubalse incide significativamente en el desarrollo económico del país con ventas aproximadas a los 400 millones de CUC al año. Su fuerza laboral la integran más de 16000 trabajadores. Se caracteriza por la orientación hacia el cliente y el trabajo en equipo.

Brinda servicios de comercio minorista y mayorista en las más de 370 tiendas de la cadena Meridiano; actividad comercial de gastronomía y recreación mediante Continental, cafeterías Cubanitas, heladerías Bim Bom, panaderías dulcerías Doña Neli, Servicentros Oro Negro, actividad comercial con Peugeot, Vedado y Multimarcas a través de Automotriz, negocios inmobiliarios por la División Inmobiliaria, construcciones, inversiones y servicios inmobiliarios con Inverco, servicios de mantenimiento con DITA, servicios de control de plagas con Biocontrol, servicios veterinarios con Almiquí, servicios aduanales con Arandia, servicios publicitarios con Grafos, servicios de empleos con la División Empleadora, servicios de lavandería ASTER.

1.6. La Ingeniería de Software Orientada a Agentes

1.6.1. El Paradigma Orientado a Agentes

La Ingeniería de Software ha derivado un entendimiento progresivo acerca de la complejidad de las aplicaciones. La producción de software actual no se concibe sin un proceso de diseño controlable, documentado y reproducible a su vez por los diversos desarrolladores, que permita definir etapas evolutivas sobre la base de abstracciones y de herramientas(34).

Una metodología para el desarrollo de software debe imponer reglas que rijan la evolución del mismo, así como definir las abstracciones y artefactos necesarios para su modelación. Toda metodología debe, de una forma u otra, satisfacer las siguientes etapas del desarrollo: análisis, diseño, implementación y prueba.

La Programación Orientada a Objetos cuenta con varios procesos de Ingeniería de Software para la modelación de las aplicaciones sustentadas bajo este paradigma, entre los que se destacan como los más estandarizados el “Proceso Unificado de Desarrollo”, (del inglés RUP: Rational Unified Process), y en la rama de Metodologías de Desarrollo Ágil “Programación Extrema”, XP (del inglés XP: eXtreme Programming).

Dado que el Paradigma Orientado a Agentes adopta novedosas concepciones y formas de concebir el comportamiento, los procesos estandarizados en la Ingeniería de Software Orientada a Objetos (ISOO) no son aplicables a la modelación de aplicaciones basadas en SMA. Uno de los mayores retos para el desarrollo de esta tecnología es la necesidad de emplear metodologías para desarrollar sistemas que enfrentan problemas que no se presentan en la implementación de los sistemas orientados a objetos, como es el hecho de que al ser los agentes autónomos no tienen que necesariamente responder a las solicitudes (como ocurre con las llamadas a objetos), y la presencia de procesos de conversación, cooperación y negociación(9).

Si bien el Paradigma Orientado a Agentes cuenta con el potencial suficiente para convertirse en la corriente principal de los paradigmas modernos de Ingeniería de Software (propiciado por la creciente complejidad de los sistemas actuales que en sentido general requieren operar en complejos entornos distribuidos) (35) y el

creciente interés en la implementación de agentes y SMA ha propiciado el surgimiento de nuevas metodologías de desarrollo basadas en dichos conceptos, no es menos cierto que la Ingeniería de Software Orientada a Agentes (ISOA) no cuenta con un proceso afianzado que rijan el desarrollo de las aplicaciones. Aún así, se han propuesto y desarrollado un amplio número de metodologías y procesos de modelado. Entre las más referenciadas MAS CommonKADS (36), metodología desarrollada en 1996 que extiende los modelos definidos en CommonKADS (metodología de Ingeniería del Conocimiento que puede ser considerada como un estándar europeo) a el modelado de SMA, adicionando técnicas de metodologías de ISOO (OOSE y OMT) y de Ingeniería de Protocolos para la descripción de los Protocolos de Comunicación de Agente (SDL y MSC96); MaSE (37), desarrollada en 1998, que parte de un conjunto inicial de requerimientos y análisis, incorporando además la etapa de diseño, extendiendo los conceptos del Paradigma Orientado a Objetos al Paradigma Orientado a Agentes; GAIA (9), desarrollada en el 2000 para el análisis y diseño de SMA, que concibe este tipo de sistemas como una organización computacional constituida por diversos roles en interacción; Tropos (38), que deviene de la Ingeniería de Requerimientos, y potencia la captura de requerimientos mediante la adopción del framework de modelado; Prometheus (39), desarrollada en el 2002, sustenta el diseño, la documentación y la implementación de SMA; PASSI (40), que en español se corresponde con Proceso para la Especificación e Implementación de Sociedades de Agentes (41), desarrollada también en el 2002, que propone un proceso paso a paso para el diseño e implementación de SMA integrando conceptos del Paradigma Orientado a Objetos y de Inteligencia Artificial; e Ingenias (42), que propone un proceso de Ingeniería de Software convencional, incorporando conceptos del Paradigma Orientado a Agentes. Además se pueden mencionar otras que aunque no tan referenciadas, también integran la gran lista de metodologías para el desarrollo de SMA entre ellas: MESSAGE, ZEUS, Vowel Engineering, RoMAS, RAP.

1.6.2. Análisis de las principales metodologías de desarrollo de Sistemas Multi-Agentes

Entre la gran variedad de metodologías antes mencionadas se han tomado algunas para realizar un mejor análisis que permita llegar a conclusiones y a la especificación de una en particular, para ser aplicada a un sistema real.

La Tabla 1 expone una comparación entre estas metodologías atendiendo fundamentalmente al proceso de aplicación y al modelado de las características del paradigma.

Metodología	Ciclo de Vida	Fases	Atributos que Modela
MAS CommonKADS	Secuencial	Análisis, Diseño	Inteligencia, Interacción
MaSE	Secuencial	Análisis, Diseño	Interacción
GAIA	Secuencial	Análisis, Diseño	Interacción, Organización
Tropos	Secuencial	Requisitos, Análisis, Diseño	Inteligencia, Interacción
Prometheus	Iterativo	Análisis, Diseño, Implementación	Interacción, Inteligencia, Entorno
PASSI	Iterativo- Incremental	Análisis, Diseño, Implementación	Autonomía, Sociabilidad, Cooperación
Ingenias	Iterativo	Análisis, Diseño, Implementación	Entorno, Interacción, Autonomía

Tabla 1: Comparación entre metodologías de ISOA.

Teniendo en cuenta que Prometheus, PASSI, e Ingenias abarcan un conjunto mayor de característica que son necesarias en el desarrollo de software, entre ellas un ciclo de vida más avanzado y fases tan importantes como análisis, diseño e implementación, es que se propone un mejor estudio de dichas metodologías.

1.6.2.1. Metodología Prometheus

Esta metodología fue concebida como un proceso detallado para la especificación, diseño e implementación de SMA. La modelación de cada agente tiene lugar en términos de metas, creencias, planificaciones y eventos. Provee soporte para el tránsito de especificaciones a diseño e implementación mediante artefactos de modelado y procesos que rigen la trazabilidad entre estos.

Prometheus presenta mecanismos de estructuración jerárquicos que permiten que el diseño sea concebido en múltiples niveles de abstracción. Estos mecanismos juegan un papel crucial en el diseño de aplicaciones de gran tamaño.

El ciclo de vida que presenta es iterativo. Mientras la primera iteración se corresponde en su totalidad a actividades asociadas a la fase de especificación del sistema, las iteraciones subsecuentes involucran una mezcla de actividades de las restantes etapas, lo cual se potencia a medida que el proceso gana en madurez.

La metodología consta de tres fases representadas en la Figura 1. La primera es la Especificación del Sistema, que se centra en la identificación de las funcionalidades básicas del SMA, de las entradas (percepciones), de las salidas (acciones) y de cualquier otra fuente de datos.

Esta fase involucra las etapas Identificación de metas y submetas, Desarrollo de escenarios de Casos de Uso (CU), Identificación de la interfaz del sistema hacia el entorno en términos de acciones, percepciones y datos externos, Identificación de funcionalidades, Identificación de lectura/escritura de datos por funcionalidades, y Elaboración de los esquemas de funcionalidades (nombre, acciones, percepciones, generación o utilización de datos, interacciones y metas).

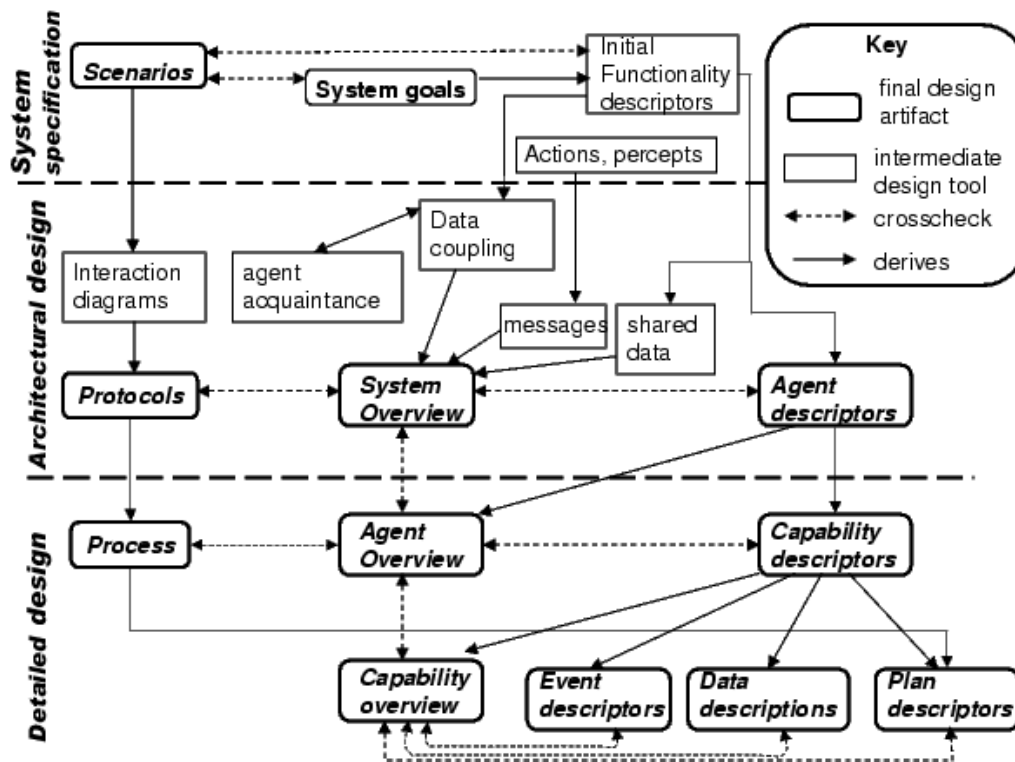


Figura 1: Fases del ciclo de vida de la metodología Prometheus (39).

La segunda fase es el Diseño Arquitectónico, fase que sobre la base de las salidas de la etapa anterior determina los agentes que compondrán el sistema y el modo en que tendrá lugar la interacción entre estos.

Intervienen en su desarrollo las etapas: Agrupación de funcionalidades para determinar tipos de agentes, Definición de tipos de agentes y desarrollo de descriptores, Descripción de la estructura global del sistema, y Desarrollo de los protocolos de interacción.

La fase de Diseño Detallado se enfoca en la estructura interna de los agentes, y el modo en que estos darán cumplimiento a sus tareas dentro del sistema global.

Esta última fase incluye las etapas: Desarrollo de diagramas de proceso, Desarrollo de diagramas de apreciación de agentes en términos de capacidades, eventos, datos y planificaciones, Refinamiento de la capacidad interna del SMA, Planificación del manejo de eventos, Definición detallada de eventos, Definición detallada de planificación, y Definición detallada de creencias.

Prometheus sustenta su desarrollo sobre las herramientas JACK Development Environment (JDE) y Prometheus Design Tool (PDT). Estas no soportan la fase de

especificación de sistema actualmente.

1.6.2.2. Metodología PASSI

Según investigadores de la Universidad de Palermo (Massimo Cossentino, Antonio Chela, Piermarco Burrafato, entre otros), el proceso de diseño de SMA puede concebirse de manera análoga a otros procesos de diseño de software sobre la base de la identificación de las entradas de dicho proceso y las salidas que el mismo genera. Con el objetivo de profundizar en la funcionalidad descriptiva del proceso, se propone un análisis de las actividades a realizar a la par del análisis de las entradas del proceso; así como la automatización de tantas etapas como sea posible(43).

El resultado de este análisis es precisamente PASSI; un proceso que ha evolucionado de un largo período de construcción teórica y experimentos en el desarrollo de aplicaciones de la Robótica para convertirse en un acercamiento más comprensivo a dicha disciplina que abarca una arquitectura de visión flexible y un modelado extenso del dominio y de la ontología¹.

La metodología constituye un modo razonado de desarrollo que permite el tránsito de requerimientos a código para diseñar e implementar SMA. La misma opera sobre la base de integración de Modelos de Diseño y conceptos de ISOA lo cual unido a las tendencias actuales de la Inteligencia Artificial en apoyo con el nivel de especificación que brinda el lenguaje de modelación UML, le atribuye un nivel de descripción acertado del objeto de aplicación (41).

PASSI incorpora cinco modelos que responden a diferentes niveles de diseño, los que a su vez se subdividen en quince fases (ver Figura 2) que resultan en el desarrollo del ciclo de vida de un SMA. Los mecanismos que implementa permiten la personalización de la representación del diseño orientado a agentes, evitando con ello la adopción de un lenguaje de modelación totalmente nuevo.

¹ Aplicación al SMA: representa un vocabulario común para que los agentes compartan la información mediante los mensajes que se intercambian.

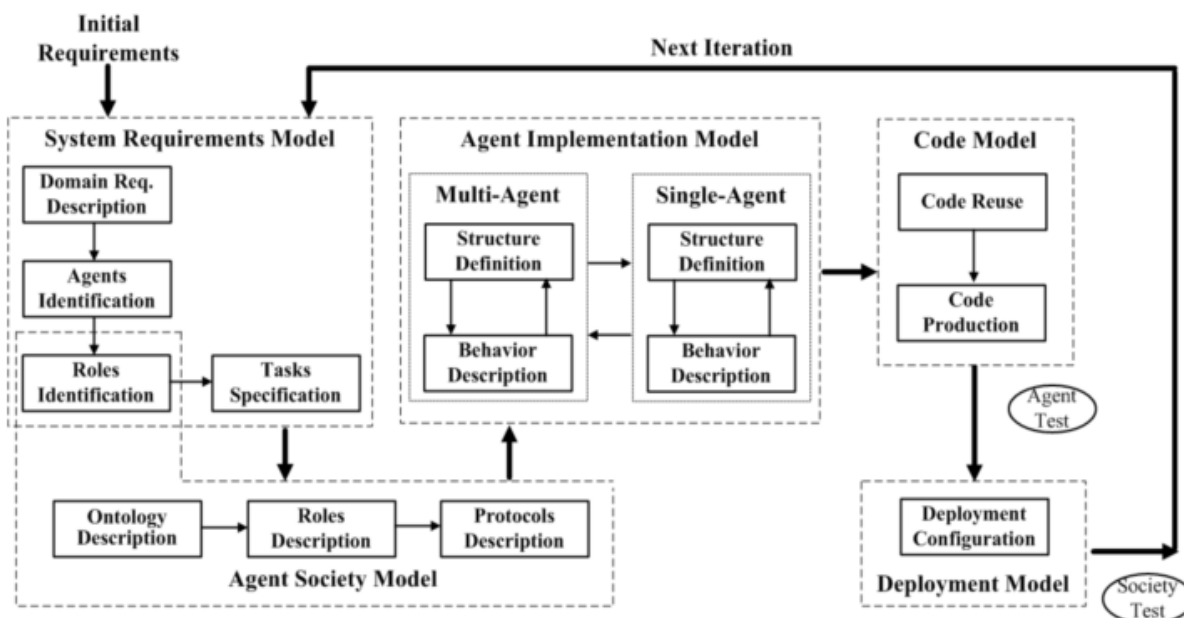


Figura 2: Fases del ciclo de vida de la metodología PASSI(41).

Los niveles y fases de PASSI abarcan una representación total de los requerimientos del sistema (entiéndase por esto punto de vista social, arquitectura de la solución, generación de código y reusabilidad del mismo; así como soporte para la configuración del despliegue de agentes).

Los desarrolladores de la metodología le atribuyen gran importancia a la reutilización de patrones de agentes existentes; de ahí que haya sido concebida para ser soportada por una herramienta CASE, el PASSI Tool-Kit (PTK), capaz de compilar modelos correspondientes a diversos niveles de aplicación del proceso; y una herramienta capaz de brindar la reusabilidad de patrones de agentes de diversas aplicaciones, el Agent-Factory.

En PASSI, un agente es una unidad significativa de software con niveles concretos y abstractos de diseño. Acorde a esta visión, un agente es una instancia de una clase agente. Así, este es la implementación de una entidad autónoma capaz de cumplir un objetivo a través de sus propias decisiones, acciones y relaciones sociales. Un agente puede emprender varios roles funcionales durante interacciones con otros agentes para lograr sus metas. Un rol es una colección de tareas realizadas por el agente siguiendo una submeta. Una tarea, a su vez, es definida como una unidad determinada de conducta individual o interactiva (41).

El proceso de aplicación de la metodología conlleva a la interpretación de un conjunto de roles por parte de los desarrolladores. Estos roles se encuentran en correspondencia funcional con los resultados de las diversas etapas de la aplicación del proceso. La Tabla 2 muestra la asignación de los diversos roles a los niveles correspondientes.

Rol del Desarrollador	Nivel Correspondiente
Experto en Dominio	Modelo de Requisitos del Sistema.
Analista de Sistema	
Experto en Ontologías	Modelo de Sociedades de Agentes.
Diseñador de Agentes	
Analista de Sistema	
Experto en Ontologías	Modelo de Implementación
Diseñador de Agentes	
Analista de Sistema	
Programador	Modelo de Código
Analista de Sistema	
Diseñador de Agentes	Modelo de Despliegue

Tabla 2: Asignación de roles por niveles (44).

1.6.2.3. Metodología Ingenias

Ingenias es una evolución de la metodología MESSAGE (del inglés MESSAGE: Methodology for Engineering Systems of Software Agents), la cual trata de integrar los resultados de las metodologías de ISOA anteriores. MESSAGE propone el

análisis y diseño de SMA desde cinco puntos de vista para capturar los diferentes aspectos del mismo: el de Organización; el de Tareas/Objetivos; el de Agente; el de Dominio; y el de Interacción (42).

Ingenias adopta la definición de agente de Newell; definición influenciada por la Inteligencia Artificial donde se asume que existen diferentes niveles de abstracción en el diseño de sistemas, entre ellos el nivel de símbolos y el nivel de conocimiento. Del mismo modo que un programa convencional maneja símbolos en el nivel de símbolos, un agente, que equivale a un programa en el nivel del conocimiento, maneja únicamente conocimiento y se comporta de acuerdo con el principio de racionalidad. Este principio estipula que un agente emprende acciones porque está buscando satisfacer un objetivo.

La metodología mejora MESSAGE en tres aspectos fundamentales:

- Integración de las vistas de diseño del sistema.
- Integración de resultados de investigación.
- Integración con el ciclo de vida de desarrollo de software.

Esta metodología propone un lenguaje de especificación de SMA así como su integración en el ciclo de vida. Siguiendo el ejemplo de MESSAGE, el lenguaje se especifica con meta-modelos y lenguaje natural.

El método de desarrollo de SMA propuesto en Ingenias concibe el SMA como la representación computacional de un conjunto de modelos. Cada uno de estos modelos muestra una visión parcial del SMA: los agentes que lo componen, las interacciones que existen entre ellos, cómo se organizan para proporcionar la funcionalidad del sistema, qué información es relevante en el dominio y cómo es el entorno en el que se ubica el sistema a desarrollar(42).

En la construcción de estos meta-modelos se integran resultados de investigación en forma de entidades y relaciones entre entidades. La instanciación de los meta-modelos produce diagramas, los modelos, similares a los que se usa en UML, con la diferencia de que estos diagramas se han creado exclusivamente para definir el SMA.

La Figura 3 representa los cinco meta-modelos que adopta la metodología Ingenias,

los cuales giran alrededor de dos entidades: la organización y el agente.

El Meta-Modelo de Agente describe agentes particulares y los estados mentales en que se encontrarán a lo largo de su vida. Los agentes realizan tareas o persiguen objetivos, responsabilidades y control.

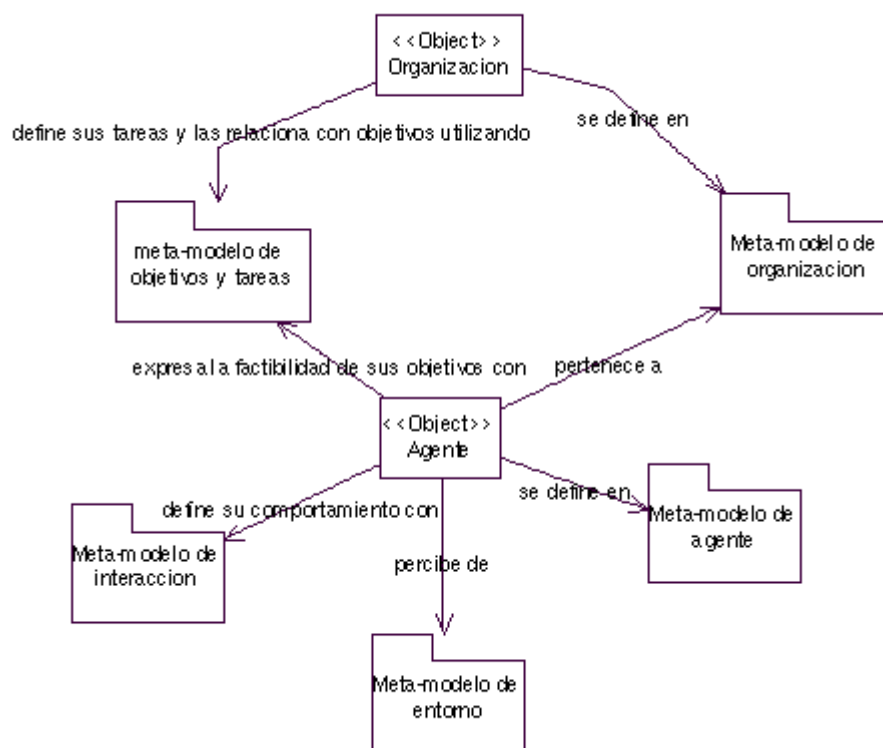


Figura 3: Relaciones entre los diferentes meta-modelos y las dos entidades principales, la organización y el agente (45).

El Meta-Modelo de Tareas y Objetivos permite asociar el estado mental del agente con las tareas que ejecuta, además de la identificación de objetivos generales y la descomposición de estos en objetivos más concretos que se pueden asignar a cada agente.

El Meta-Modelo de Organización define cómo se agrupan los agentes, la funcionalidad del sistema y qué restricciones hay que imponer sobre el comportamiento. Este modelo puede considerarse como el equivalente a la arquitectura de un sistema convencional. Permite definir a alto nivel la organización de los elementos del sistema para hacer posible los objetivos comunes a los agentes que participan en la organización.

El Meta-Modelo de Interacción detalla cómo se coordinan y comunican los agentes, así como las interacciones existentes entre estos y sus roles.

Por último el Meta-Modelo de Entorno, que define los elementos existentes alrededor del nuevo sistema y cómo los perciben cada agente. Modela las entidades y relaciones del SMA con el entorno(42).

El uso de meta-modelos como elemento de construcción proporciona la independencia entre las herramientas de soporte y la metodología, lo cual facilita portar la metodología a diferentes herramientas, ya que cualquiera que soporte meta-modelado podría servir como herramienta de soporte de desarrollo.

El proceso de instanciación de los meta-modelos no es trivial. Existen muchas entidades y relaciones a identificar, además de dependencias entre los distintos modelos.

Es por ello que en Ingenias se persigue un proceso de desarrollo similar al adoptado por otras herramientas, como Rational Rose, TogetherJ o Paradigma. En estas herramientas, el usuario tiene libertad para elaborar diagramas incompletos o generar sus propias vistas del sistema, tomando elementos de diferentes modelos e incorporándolos a un nuevo diagrama.

1.7. Selección de la Metodología de Desarrollo

Una vez analizadas estas tres metodologías se ha llegado a la conclusión que la más completa de todas y por tanto la que se utilizará en la modelación del sistema de Comercio Electrónico para la empresa Cubalse es PASSI, ya que presenta de forma general aspectos positivos que la ubican en uno de los lugares privilegiados de las metodologías existentes.

PASSI es una metodología de especificación e implementación de SMA que presenta un ciclo de vida iterativo e incremental y que es independiente del dominio de aplicación, por lo que se considera diseñada para la modelación de cualquier SMA. Las iteraciones hacen referencia a pasos en el flujo de trabajo, y los incrementos, al crecimiento del producto (46). Esta visión constituye una sólida representación del proceso de construcción de un sistema de software, pues permite

la definición de secciones a implementar acopladas funcionalmente, y con ello la disminución de la complejidad del trabajo.

Esta metodología se encuentra subdividida en niveles y fases, que a su vez poseen diferentes estados de completitud. Con relación a la Vista de Modelos presenta gran claridad y facilidad de uso de los estereotipos, además de un lenguaje de modelado fácil de usar y con grandes prestaciones, el UML.

En PASSI se atribuye gran importancia al proceso de reutilización de patrones de agentes existentes en un depósito. Para ello cuenta con una herramienta: el Agent Factory, que permite el acceso a los modelos almacenados y al código correspondiente. La metodología soporta la reutilización en un gran número de las fases de aplicación. Esta concepción permite disminuir considerablemente el tiempo de desarrollo.

PASSI es soportada por una herramienta CASE, el PASSI Tool-Kit, que constituye un sólido sostén de la metodología. Esta versión se encuentra gratis en Internet, y puede ser accedida a través de la página principal de la metodología (41). Esta herramienta divide los artefactos en los paquetes: Vista de Casos de Uso y Vista Lógica. En estos paquetes se encuentran los modelos obtenidos en cada una de las fases de la metodología.

El hecho de ser orientada a una herramienta de desarrollo atribuye a la metodología la capacidad de automatizar procesos repetitivos, mantener la información estructurada de forma coherente, gestionar grandes volúmenes de información; y por ende aumentar la productividad de los desarrolladores y la calidad del producto.

PASSI incorpora además los principales conceptos del paradigma de agentes, como son: *Recurso, Usuario, Rol, Escenario, Tarea e Interacción*.

El aspecto más cómodo de PASSI está dado por su similitud con RUP, pues este está estandarizado entre los desarrolladores de ISOO; y en la adopción de UML como lenguaje de modelado, pues es bastante difundido. Existe documentación en inglés precisa que posibilita su aplicación. Además, PASSI define con claridad roles para el equipo de desarrollo, y cubre la gran mayoría de las etapas de desarrollo de un SMA.

1.8. Conclusiones

A partir de las diferentes definiciones de Agentes Inteligentes existentes, se ha realizado un breve recorrido por temas relacionados con este paradigma, transitando por su integración en los SMA y la capacidad de funcionamiento en varias ramas especialmente el Comercio Electrónico, al que se prestó mayor atención. El diseño de todo sistema requiere de la utilización de una metodología de desarrollo basada en determinados conceptos y reglas que la rigen. En especial el Paradigma Orientado a Agentes ha desarrollado gran número de estas, lo que ha conllevado a un estudio más detallado de las mismas y a partir de ahí seleccionar una que permita desarrollar la aplicación deseada: un sistema para el Comercio Electrónico con una atención personalizada al cliente.

CAPÍTULO 2: LA METODOLOGÍA PASSI APLICADA AL SISTEMA DE COMERCIO ELECTRÓNICO PARA LA EMPRESA CUBALSE

2.1. Introducción

Atendiendo a la necesidad de la creación de un sistema de Comercio Electrónico para la empresa Cubalse y dada las potencialidades que brinda la tecnología de agentes, se ha decidido realizar la documentación de dicho sistema mediante la metodología PASSI. Es precisamente en este capítulo donde se realiza una descripción detallada del sistema y se le aplica la metodología PASSI que permite obtener una guía precisa que facilita el tránsito desde las fases iniciales como la de requerimiento hasta la conformación y despliegue de un sistema debidamente probado.

2.2. Descripción del proyecto Plataforma de Comercio Electrónico B2B

El proyecto Plataforma de Comercio Electrónico B2B se está desarrollando en la Universidad de Ciencias Informáticas (UCI), específicamente en la Facultad 3. Este proyecto está diseñado para la realización de un sistema de Comercio Electrónico para la venta mayorista, partiendo de la política de la dirección del país de promover el Comercio Electrónico en Cuba. Para poner en práctica este sistema se escoge la empresa Cubalse dentro de la cual será utilizado inicialmente en La Copa (tienda mayorista de Ciudad de la Habana), y en el futuro se extenderá al resto de las tiendas mayoristas de la empresa Cubalse.

Se pretende diseñar un sistema que reciba productos de otro sistema de inventario, muestre a los clientes una interfaz Web donde puedan explorar los productos disponibles, adicionarlos a un carro de compras y luego procesar los pedidos haciendo una transacción con el banco. El sistema debe exportar las facturas a otro sistema de facturación.

La compra-venta mayorista de productos puede tener lugar esencialmente de dos formas fundamentales: vía Web, y vía Mostrador.

Cliente Mostrador: Cuando un cliente pretende efectuar una compra vía Mostrador, se presenta en la unidad de venta. Esta unidad brinda servicios exclusivamente a personas jurídicas registradas como clientes de la empresa Cubalse. En el área de venta el cliente realiza el pedido de los productos, dándosele la posibilidad de solicitar una prefactura. En caso que el cliente lo desee, puede solicitar se le reserven los productos seleccionados, lo cual se realiza durante un tiempo variable definido por la administración. En este período el cliente puede realizar acciones sobre la base de la mercancía reservada sin riesgo alguno de que la misma sufra alteraciones. Además, este intervalo permite hacer las gestiones pertinentes sobre el modo de pago. En esta modalidad de compra el pago puede realizarse mediante tres vías diferentes: por cheque, por letra de cambio, carta de crédito interna o por transferencia bancaria. Una vez efectuado el pago, se procede al despacho de la mercancía, culminando de esta forma el proceso. Cabe señalar que en caso de que el cliente no proceda a la compra de la mercancía una vez concluido el tiempo destinado a la reserva, el mismo pierde todo el derecho sobre la mercancía.

Cliente Web: Siempre que el cliente proceda a hacer la compra vía Web, el mismo una vez hecha la solicitud tendrá la posibilidad de solicitar una prefactura, donde se reflejarán las mercancías, precios y cantidades seleccionadas. Luego de consultada la prefactura emitida, el cliente puede solicitar se le reserven las mercancías contenidas en esta. Las mercancías se reservarán al igual que en el caso anterior por un tiempo definido por la administración, durante el cual se mantiene la integridad de la solicitud realizada. En este período en caso de que el cliente decida comprar la mercancía, tiene que efectuar el pago de la misma online mediante una cuenta bancaria. Una vez efectuada la transacción el cliente selecciona el método de entrega de la mercancía según se especificó en el contrato.

2.3. Modelo de Requerimiento del Sistema

Este modelo se confecciona partiendo de la empresa, sus necesidades y propósitos. El desarrollo del mismo involucra las siguientes fases: Descripción de Entorno, Descripción de Dominio, Identificación de Agentes, Identificación de Roles y Especificación de Tareas.

2.3.1. Descripción de Entorno

Este modelo proporciona una visión del sistema al máximo nivel de abstracción mediante un Diagrama de Casos de Uso UML, donde se representan los actores internos y externos, así como sus relaciones con el sistema, considerado este como un todo único (como se muestra la Figura 4). El mismo será la traza de inicio del Diagrama Descripción de Dominio, a la vez que constituye el punto de partida de la descripción funcional del sistema (47).

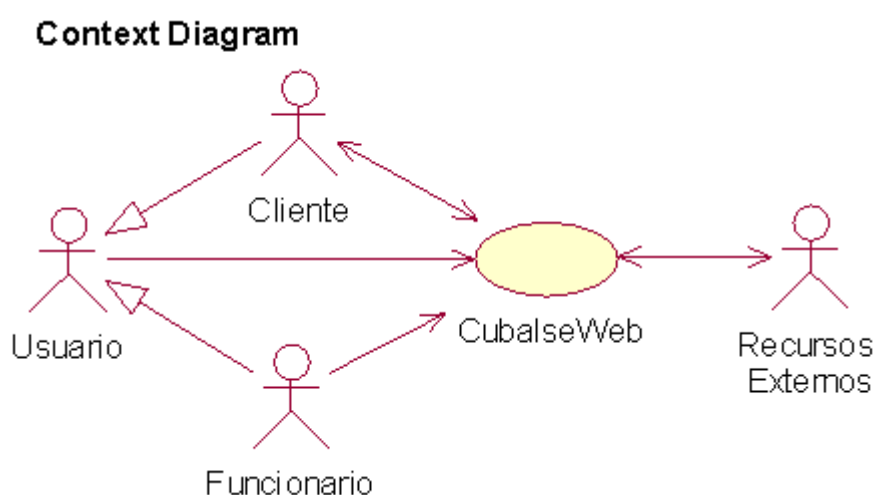


Figura 4: Diagrama de Entorno.

El Usuario que representa un actor interno se relaciona con el sistema al autenticarse, de él heredan el Cliente y el Funcionario. El Cliente interactúa con el sistema usando el comercio vía Web, mientras que el Funcionario se refiere a un trabajador de la empresa Cubalse que trabaja con los pedidos vía Mostrador, además de realizar otras funciones según el Rol (Admin, Admin_Usuarios, Admin_Comercial, Admin_Juridico, Comercial_Vendedor) que desempeña en la empresa, ambos son usuarios registrados.

Recursos Externos es un actor genérico que representa las diferentes fuentes de información necesarias para realizar las funcionalidades de la aplicación, se hace referencia a la base de datos y sistemas externos como: un sistema bancario y un sistema de facturación.

El CU denominado CubalseWeb muestra el sistema a realizar, representado como un todo único.

2.3.2. Descripción de Dominio

Esta fase ofrece una descripción funcional del sistema usando diagramas de CU convencionales mediante estereotipos importados del UML estándar. La misma describe los requerimientos del sistema en una serie jerárquica de diagramas de CU. El diagrama resultante (Figura 5) constituye el punto de partida de desarrollo del sistema, pues en el mismo tiene lugar una primera aproximación a la identificación de los agentes involucrados en la solución.

En correspondencia a las funcionalidades que se espera que cumpla el sistema se han identificado una serie de CU que se muestran a continuación.

1. CU Mostrar_Informacion
2. CU Administrar_Informacion
3. CU Mostrar_Opciones
4. CU Gestionar_Usuario
5. CU Autenticar_Usuario
6. CU Gestionar_Productos
7. CU Explorar_Productos
8. CU Solicitar_Compras
9. CU Solicitar_Prefactura
10. CU Agregar_Carro_Compras
11. CU Mostrar_Prefactura
12. CU Reservar_Productos
13. CU Consultar_Mercancia
14. CU Realizar_Transaccion
15. CU Exportar_Facturas

16. CU Sugerir_Productos

17. CU Recuperar_Prod_Reservados

18. CU Buscar_Inf_Especializada

19. CU Recopilar_Opinion

20. CU Procesar_Opinion

Debido a la importancia que tienen dentro de las funcionalidades del sistema y su influencia en la personalización del mismo, a continuación se describen los siguientes CU: Sugerir_Productos, Recuperar_Prod_Reservados, Buscar_Inf_Especializada y Procesar_Opinion.

CU: Sugerir_Productos		
Descripción:	Este CU es el responsable de crear una lista de productos acordes con las características del cliente registrado y sus preferencias, así como sugerir productos en rebaja o con un precio más factible. Es el CU que brinda mayor personalización a los clientes.	
Comunicaciones:	Iniciador	Participante
	1. Explorar_Productos	1. Sugerir_Productos

Tabla 3: Especificación del CU Sugerir_Productos.

CU: Recuperar_Prod_Reservados	
Descripción:	En caso de que algún producto reservado por un cliente, sea comprado por una entidad priorizada y el mismo se haya agotado, el sistema realizará una búsqueda en otros sistemas de Comercio Electrónico para reponer dicho producto, en caso de que no sea encontrado, se sugieren ofertas similares

	con el fin de satisfacer la necesidad del cliente.	
Comunicaciones:	Iniciador	Participante
	1. Consultar_Mercanci a	1. Recuperar_Prod_Reserv ados

Tabla 4: Especificación del CU Recuperar_Prod_Reservados.

CU: Buscar_Inf_Especializada		
Descripción:	En caso de que el cliente requiera de una descripción más detallada de un producto determinado, el sistema realizará una búsqueda interactuando con SMA recuperadores de información de Internet, para mostrar vínculos a sitios Web que brinden datos específicos.	
Comunicaciones:	Iniciador	Participante
	1. Explorar_Productos	1. Buscar_Inf_Especializad a

Tabla 5: Especificación del CU Buscar_Inf_Especializada.

CU: Procesar _Opinion		
Descripción:	Una vez recuperada la opinión de los clientes (mediante encuestas o mensajes de correo electrónico) con respecto a los servicios brindados, el sistema la procesa con el fin de obtener coincidencias, descubrir posibles dificultades y contribuir a la mejora continua del sistema. Este análisis se realiza mediante SMA orientados a la minería de opiniones, y a la minería Web.	
Comunicaciones:	Iniciador	Participante
	1. Recopilar_Opinion	1. Procesar_Opinion

Tabla 6: Especificación del CU Procesar_Opinion.

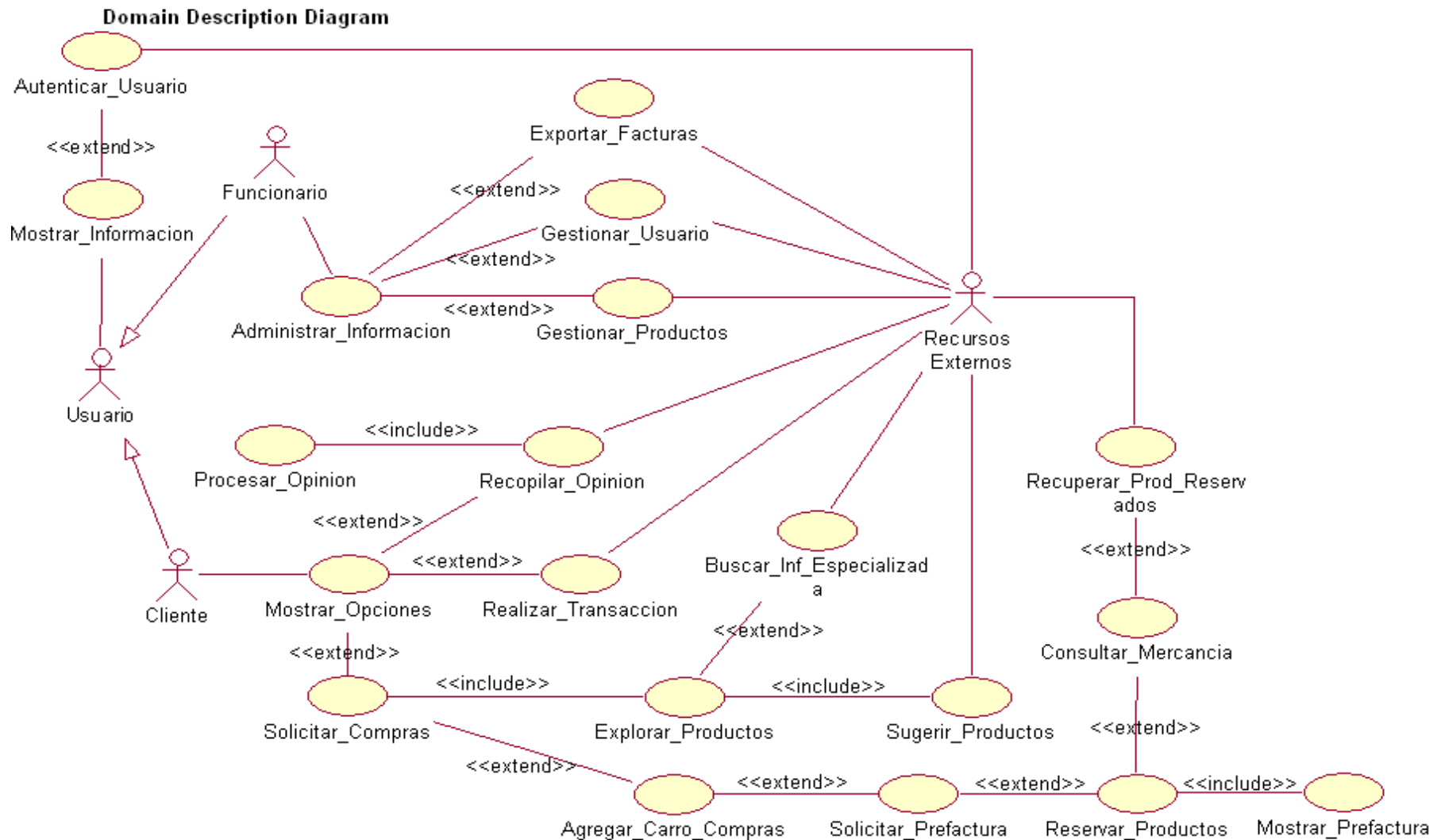


Figura 5: Diagrama Descripción de Dominio.

2.3.3. Identificación de agentes

Esta etapa permite la separación de responsabilidades en agentes, representados como paquetes de UML estereotipados. Acorde a la definición de agente adoptada, es posible ver a un agente como un CU o un paquete de CU provenientes de la descomposición funcional de la fase anterior. A partir de un diagrama suficientemente detallado de las funcionalidades del sistema, se agrupan uno o más CU en paquetes estereotipados para formar un nuevo diagrama. De esta forma, cada paquete define las funcionalidades de un agente específico. Las relaciones entre los CU del mismo agente siguen la sintaxis usual y estereotipos de UML; aunque las relaciones entre CU de diferentes agentes son estereotipadas como comunicación.

A continuación se describen una serie de agentes que serán los responsables de garantizar la funcionalidad del sistema.

Agente_Interfaz

1. CU Mostrar_Informacion
2. CU Administrar_Informacion
3. CU Mostrar_Opciones
4. CU Consultar_Mercancia
5. CU Mostrar_Prefactura

Para agrupar estos CU como un Agente_Interfaz (como se muestra en la Figura 6) se tuvo en cuenta que los mismos se referían a la comunicación con el usuario desde que entra al sitio o una vez autenticado donde al actuar como cliente o funcionario estos pueden realizar alguna acción sobre el sistema. También se encarga de mostrar una prefactura emitida o el conjunto de mercancías reservadas por el cliente luego de haber seleccionado dichas opciones. Moderando de esta forma la interacción del usuario con el sistema a nivel de interfaz.



Figura 6: Agente_Interfaz (fragmento del diagrama Identificación de Agentes).

Agente_Compra

Este agente incorpora los siguientes CU:

1. CU Solicitar_Compras
2. CU Solicitar_Prefectura
3. CU Agregar_Carro_Compras
4. CU Reservar_Productos

Estos CU fueron agrupados atendiendo a la acción de realizar una compra (como se muestra en la Figura 7), el agente engloba una serie de funcionalidades, una vez que el cliente desea realizar una compra debe solicitarla, agregar al carro de compras los productos seleccionados, luego solicitar la prefactura y en caso de que desee efectuar la compra reserva los productos expuestos en la prefactura.



Figura 7: Agente_Compra (fragmento del diagrama Identificación de Agentes).

Agente_Persona

Este agente incorpora los siguientes CU:

1. CU Gestionar_Usuario
2. CU Autenticar_Usuario

Estos CU fueron agrupados (como se muestra en la Figura 8) atendiendo a que los mismos interactúan directamente con la información referente a los usuarios del sistema. Por ello se propone este agente como moderador de la comunicación entre el sistema y los recursos externos en que se encuentra almacenada dicha información.



Figura 8: Agente_Persona (fragmento del diagrama Identificación de Agentes).

Agente_Producto

Este agente incorpora los siguientes CU:

1. CU Explorar_Productos
2. CU Gestionar_Productos
3. CU Sugerir_Productos
4. CU Recuperar_Prod_Reservados

El Agente_Producto agrupa todos los CU relacionados con las acciones que se realizan sobre los productos (ver Figura 9) como es el caso de insertarlo, modificarlo o eliminarlo, también es el agente que brinda mayor personalización al cliente creando una lista de productos acordes con sus características y preferencias, así como sugiriéndole productos en rebaja o con un precio más factible. Además en caso de que algún producto reservado por un cliente, sea comprado por una entidad priorizada y el mismo se haya agotado, realiza una búsqueda en otros sistemas de Comercio Electrónico para reponer dicho producto, en caso de que no sea encontrado, busca ofertas similares con el fin de satisfacer la necesidad del cliente.



Figura 9: Agente_Producto (fragmento del diagrama Identificación de Agentes).

Agente_ Información

Este agente incorpora los siguientes CU:

1. CU Buscar_Inf_Especializada

El Agente_Información (como se muestra en la Figura 10) en caso de que el cliente requiera de una descripción más detallada de un producto determinado, es el encargado de realizar una búsqueda interactuando con SMA recuperadores de información de Internet para mostrar vínculos a sitios Web que brinden datos específicos sobre dicho producto.



Figura 10: Agente_Información (fragmento del diagrama Identificación de Agentes).

Agente_Transacción

Este agente incorpora los siguientes CU:

1. CU Realizar_Transaccion

Este agente (ver Figura 11) se comunica con un sistema externo para realizar las transacciones con el banco de dos formas diferentes: cuando la acción se realiza vía Web el cliente paga mediante una tarjeta de crédito de forma online realizándose la transacción directamente, en caso de que sea vía Mostrador el cliente paga de diferentes formas: por cheque, por letra de cambio, carta de crédito interna o por transferencia bancaria, luego el agente debe actualizar dicho pago con el sistema bancario correspondiente.

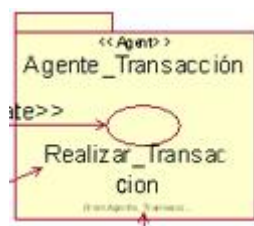


Figura 11: Agente_Transacción (fragmento del diagrama Identificación de Agentes)

Agente_Factura

Este agente incorpora los siguientes CU:

1. CU Exportar_Facturas

Este agente (ver Figura 12) se encarga de comunicarse con un sistema de facturación externo cuando algún funcionario solicita exportar una factura determinada.



Figura 12: Agente_Factura (fragmento del diagrama Identificación de Agentes).

Agente_Opinión

Este agente (ver Figura 13) incorpora los siguientes CU:

1. CU: Recopilar_Opinion
2. CU: Procesar_Opinion

Los clientes tienen la posibilidad de expresar sus opiniones acerca del sistema, sus ofertas, facilidades, o sugerencias que puedan beneficiar los servicios brindados por el mismo. A medida que se obtiene dicha información, con ayuda de SMA especializados en la minería de datos y la minería Web el agente procesa la información conformando un resumen que facilite mejorar los servicios.

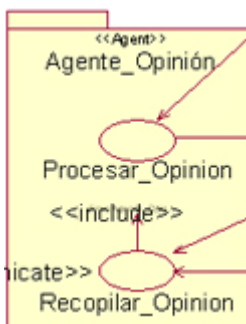


Figura 13: Agente_Opinión (fragmento del diagrama Identificación de Agentes).

A continuación se presenta el Diagrama Identificación de Agentes (Figura 14) generado automáticamente por la herramienta PASSI Tool-Kit una vez asignados los diversos CU a los agentes responsables de satisfacer los mismos. Es altamente recomendado evitar la realización de cambios en el modelo generado, proponiéndose para ello la reestructuración de la fase anterior (47).

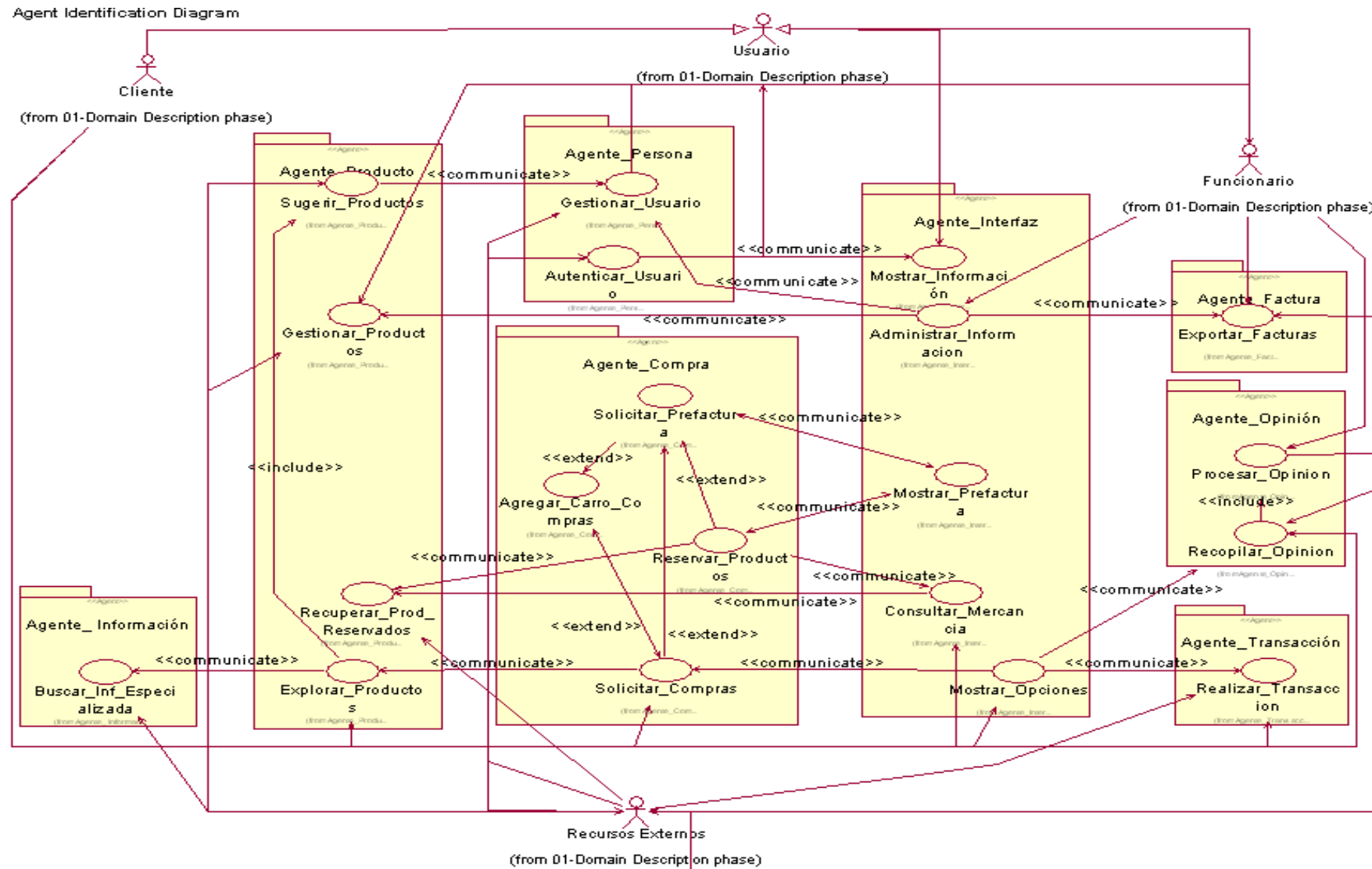


Figura 14: Diagrama Identificación de Agente.

2.3.4. Identificación de Roles

En esta etapa se exploran las responsabilidades de cada agente a través de escenarios específicos para cada rol mediante diagramas de secuencia. Cada rol es obtenido componiendo varias tareas en una conducta resultante. Esto evidencia las capacidades de un rol particular, así como su posible utilidad en la identificación de modelos reutilizables.

A cada escenario identificado corresponde un diagrama, en el que cada objeto representa un rol específico de un agente. Estos diagramas se corresponden al flujo normal de eventos del escenario mostrado; suponiendo con ello la realización satisfactoria de las tareas involucradas.

El alcance de esta etapa es modelar el ciclo de vida de cada agente, identificando los roles que él puede jugar, las colaboraciones que necesita, y las comunicaciones en que participa.

De los agentes propuestos se identifican los siguientes roles:

Agente_Interfaz

1. Rol Visualizar.

Agente_Compra

1. Rol Solicitar
2. Rol Agregar Carro Compras
3. Rol Reservar Productos

Agente_Persona

1. Rol Gestionar Usuario
2. Rol Autenticar

Agente_Producto

1. Rol Gestionar Productos
2. Rol Personalizar
3. Rol Recuperar Producto Reservados

Agente_ Información

1. Rol Buscar Información

Agente_ Transacción

1. Rol Realizar Transacción

Agente_ Factura

1. Rol Exportar Facturas

Agente_ Opinión

1. Rol Recopilar Opinión
2. Rol Procesar Opinión

Estos roles a su vez intervienen en los escenarios: Gestionar Productos, Exportar Factura, Analizar Opinión, Gestionar Usuario, y Realizar Compras; los tres últimos serán descritos a continuación debido a su complejidad y la importancia que representan para el sistema, además de la especificación con el diagrama de secuencia que supone la metodología (de la Figura 15 a la Figura 19).

Escenario: Gestionar Usuario	
Descripción:	Este escenario muestra el curso normal que sigue un usuario autenticado (cuyo privilegio corresponde a un funcionario) para realizar cualquier acción sobre un usuario del sistema, ya sea adicionar, modificar o eliminar.
CU Asociados	CU Autenticar_Usuario, CU Gestionar_Usuario, CU Mostrar_Informacion, CU Administrar_Informacion
Roles Asociados	Descripción
Rol Visualizar	Este rol es responsable de mostrar la información requerida por un usuario y una vez autenticado como funcionario, muestra las acciones que el mismo puede

	<p>realizar sobre el sistema. Las principales actividades que realiza son: mostrar información y mostrar las opciones de administrar información del sitio (gestionar usuario en este caso).</p>
<p>Rol Autenticar (Agente_Persona)</p>	<p>Este rol es encargado de autenticar los usuarios registrados en la base de datos, proporcionándoles los privilegios establecidos. Las principales actividades que realiza son, una vez introducido usuario y contraseña, verificar la existencia de los datos.</p>
<p>Rol Gestionar Usuario (Agente_Persona)</p>	<p>Este rol es responsable de la inserción de un nuevo usuario al sistema, así como modificar sus datos o eliminarlo de la base de datos, una vez verificado que quien realiza la acción sea un funcionario. Sus principales actividades son :</p> <p>Insertar Usuario: Introducir los datos de los usuarios y enviarlos a la base de datos.</p> <p>Modificar Usuario: Seleccionar un usuario de la base de datos y modificar sus campos.</p> <p>Eliminar Usuario: Seleccionar un usuario de la base de datos y eliminarlo de la misma.</p>

Tabla 7: Especificación del escenario Gestionar Usuario.

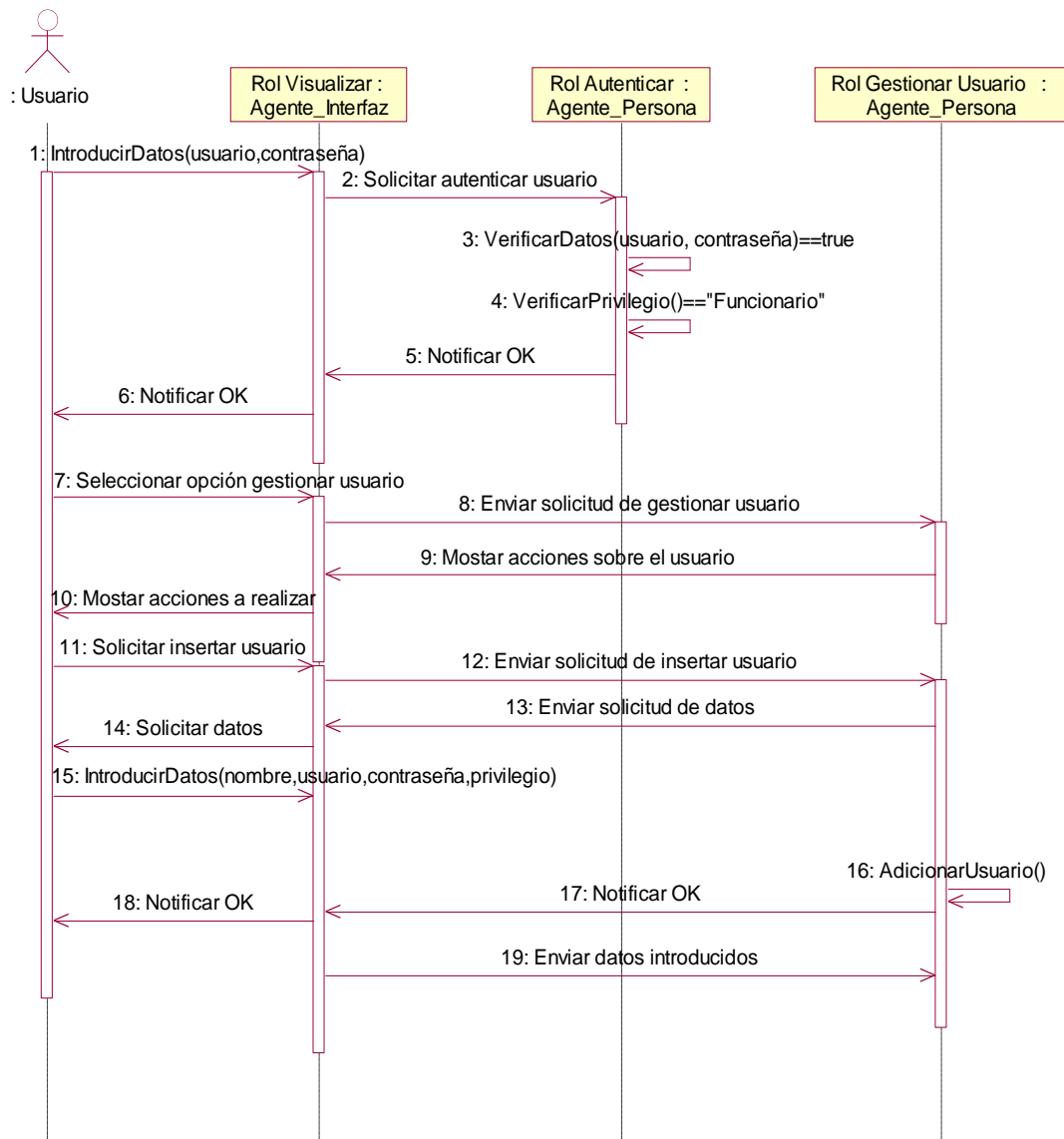


Figura 15: Diagrama Identificación de Roles del escenario Gestionar Usuario: Sección Insertar Usuario.

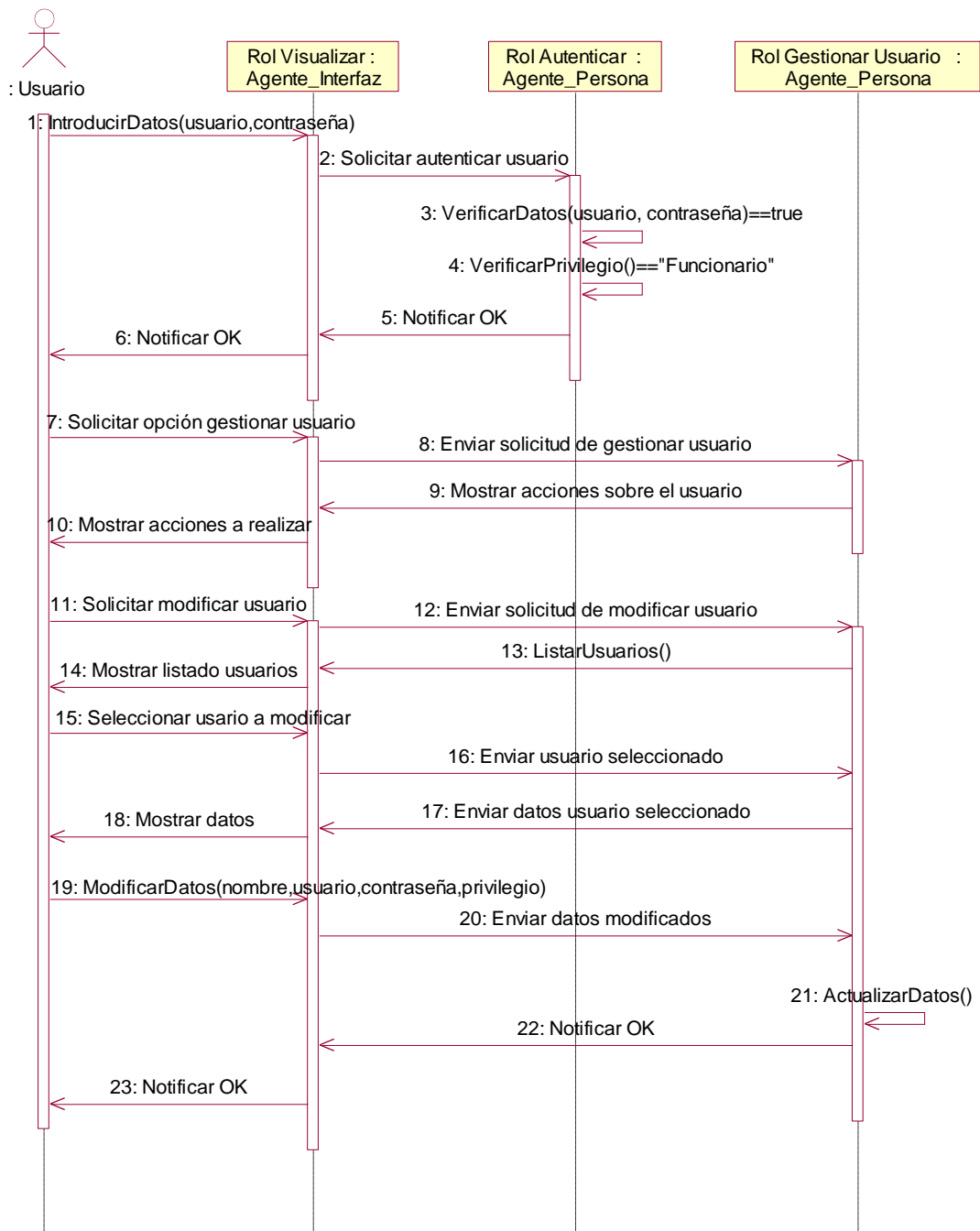


Figura 16: Diagrama Identificación de Roles del escenario Gestionar Usuario: Sección Modificar Usuario.

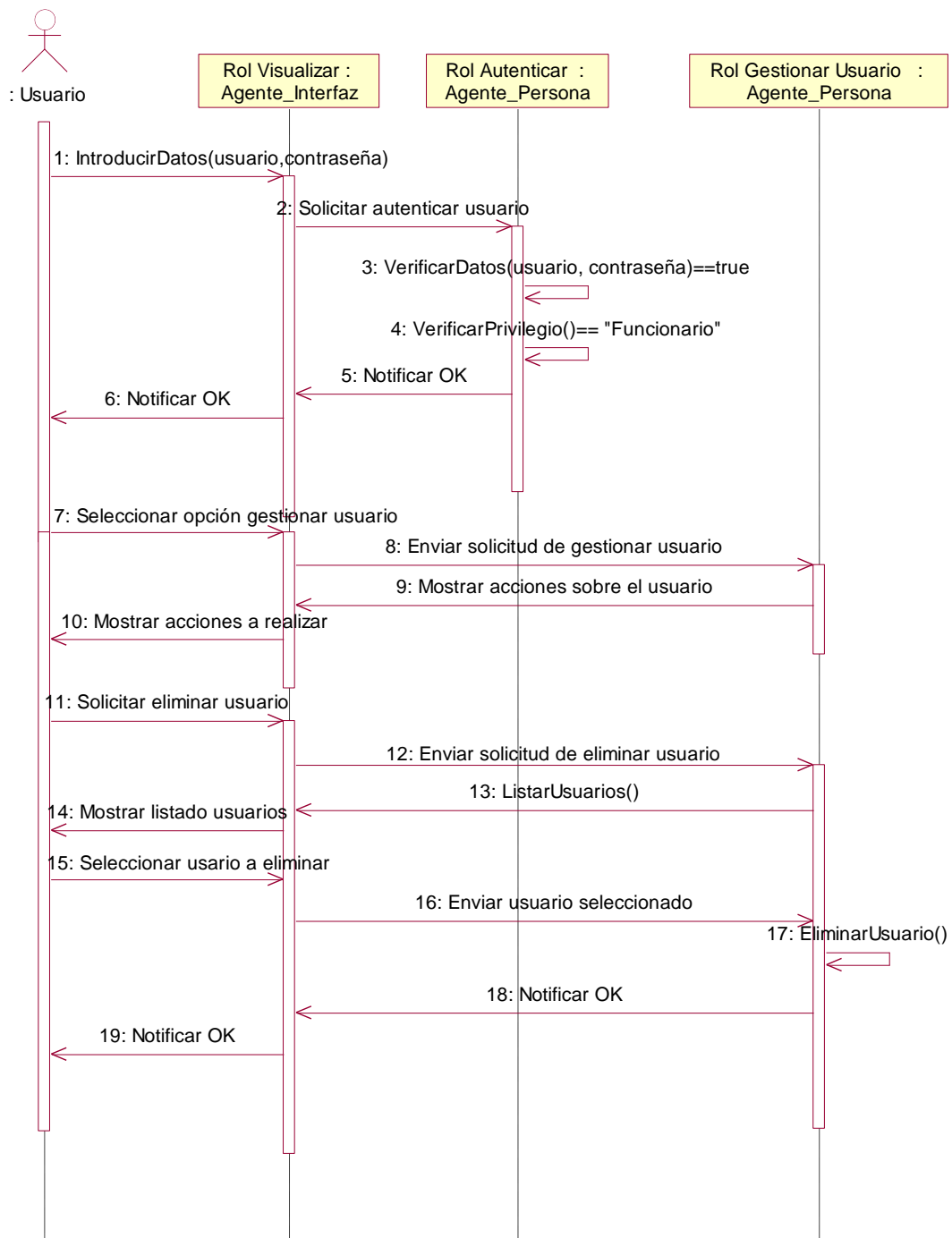


Figura 17: Diagrama Identificación de Roles del escenario Gestionar Usuario: Sección Eliminar Usuario.

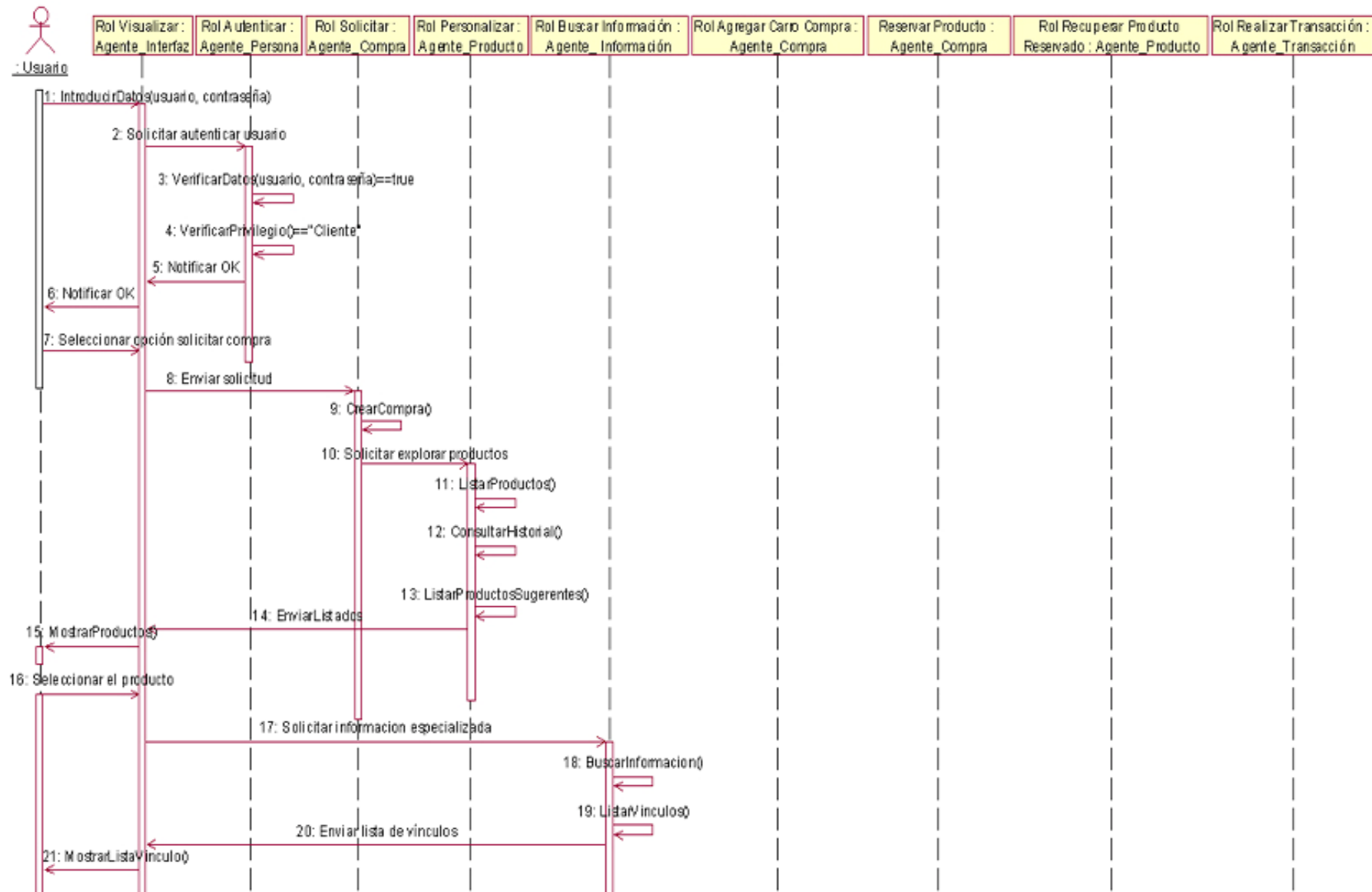
Escenario: Realizar Compras	
Descripción:	Este escenario muestra el curso normal que sigue un usuario autenticado (cuyo privilegio corresponde a un cliente) para realizar una compra, ya sea vía Mostrador o vía Web.
CU Asociados	CU Mostrar_Opciones, CU Autenticar_Usuario, CU Solicitar_Compras, CU Solicitar_Prefactura, CU Agregar_Carro_Compras, CU Reservar_Productos, CU Explorar_Productos, CU Sugerir_Productos, CU Mostrar_Prefactura, CU Recuperar_Prod_Reservados, CU Buscar_Inf_Especializada, CU Realizar_Transaccion, CU Consultar_Mercancia.
Roles Asociados	Descripción
Rol Visualizar. (Agente_Interfaz)	<p>Este rol es responsable de mostrar la información requerida por un usuario y una vez autenticado como cliente, muestra las acciones que el mismo puede realizar sobre el sistema, en este caso solicitar realizar una compra y al seleccionar las opciones:</p> <p>Consultar Mercancía: Muestra la mercancía reservada por un cliente y los detalles de la misma desde la reservación hasta la compra del producto.</p> <p>Mostrar Prefactura: Muestra la prefactura emitida.</p> <p>Las actividades que realiza son visualizar la información necesaria para que el cliente pueda seleccionar la acción a realizar, además de mostrar la mercancía y la prefactura emitida.</p>

<p>Rol Autenticar (Agente_Persona)</p>	<p>Este rol es encargado de autenticar los usuarios registrados en la base de datos, proporcionándoles los privilegios establecidos. La actividad principal que realiza es: una vez introducido usuario y contraseña, verificar la existencia de los datos.</p>
<p>Rol Solicitar (Agente_Compra)</p>	<p>Este rol es el responsable de solicitar realizar una compra y solicitar una prefactura. Las actividades que realiza son: iniciar una nueva compra y emitir una prefactura con los productos agregados al carro de compras así como el costo total de los mismos.</p>
<p>Rol Personalizar (Agente_Producto)</p>	<p>Este rol es el responsable de brindarle una mayor personalización al cliente, sugiriéndole de los productos disponibles una lista de aquellos relacionados con su perfil, aquellos que se encuentren en rebaja y los de un precio más factible. Las actividades que realiza son: consultar perfil e historial del cliente registrado, buscar coincidencias y mostrar los productos disponibles, acordes con las coincidencias encontradas, además de aquellos que hayan sufrido alguna rebaja de precio para finalmente crear una lista de los productos a sugerir.</p>
<p>Rol Buscar Información (Agente_Información)</p>	<p>Este rol es el responsable en caso de que el cliente desee ver una información más detallada de un producto en específico, de comunicarse con el SMA recopilador de información y mostrar vínculos a sitios Web con mayor información, siendo estas las actividades que realiza.</p>
<p>Rol Agregar Carro Compras</p>	<p>Este rol es el responsable de agregar productos al carro de compras según los seleccione el cliente. Las actividades que realiza son: seleccionar los productos y</p>

(Agente_Compra)	agregarlos al carro de compras.
Rol Reservar Productos (Agente_Compra)	Este rol es el responsable de que una vez agregados los productos y actualizada la prefactura, si el cliente desea reservar los productos incluidos en ella los mismos sean reservados. Esta reserva se mantendrá durante un tiempo seleccionado por el funcionario de la empresa responsable de dicha tarea. Las actividades que realiza son: cambiar el estado de los productos de disponibles a reservados.
Rol Recuperar Producto Reservados (Agente_Producto)	Este rol es el responsable de comunicarse con otros sitios de Comercio Electrónico para buscar productos que necesite recuperar para reponerlos a un cliente que los haya reservado, de no ser encontrados recomienda productos similares. Las actividades que realiza son: listar los productos a recuperar, realizar búsqueda, recuperar los productos encontrados y en caso de que todos no sean encontrados hacer sugerencias.
Rol Realizar Transacción (Agente_Transacción)	Este rol es el responsable de comunicarse con un sistema externo para realizar las transacciones con el banco. Sus principales actividades son: comunicarse con el sistema bancario, verificar los datos de la transacción y realizar la misma.

Tabla 8: Especificación del escenario Realizar Compras.

Capítulo 2



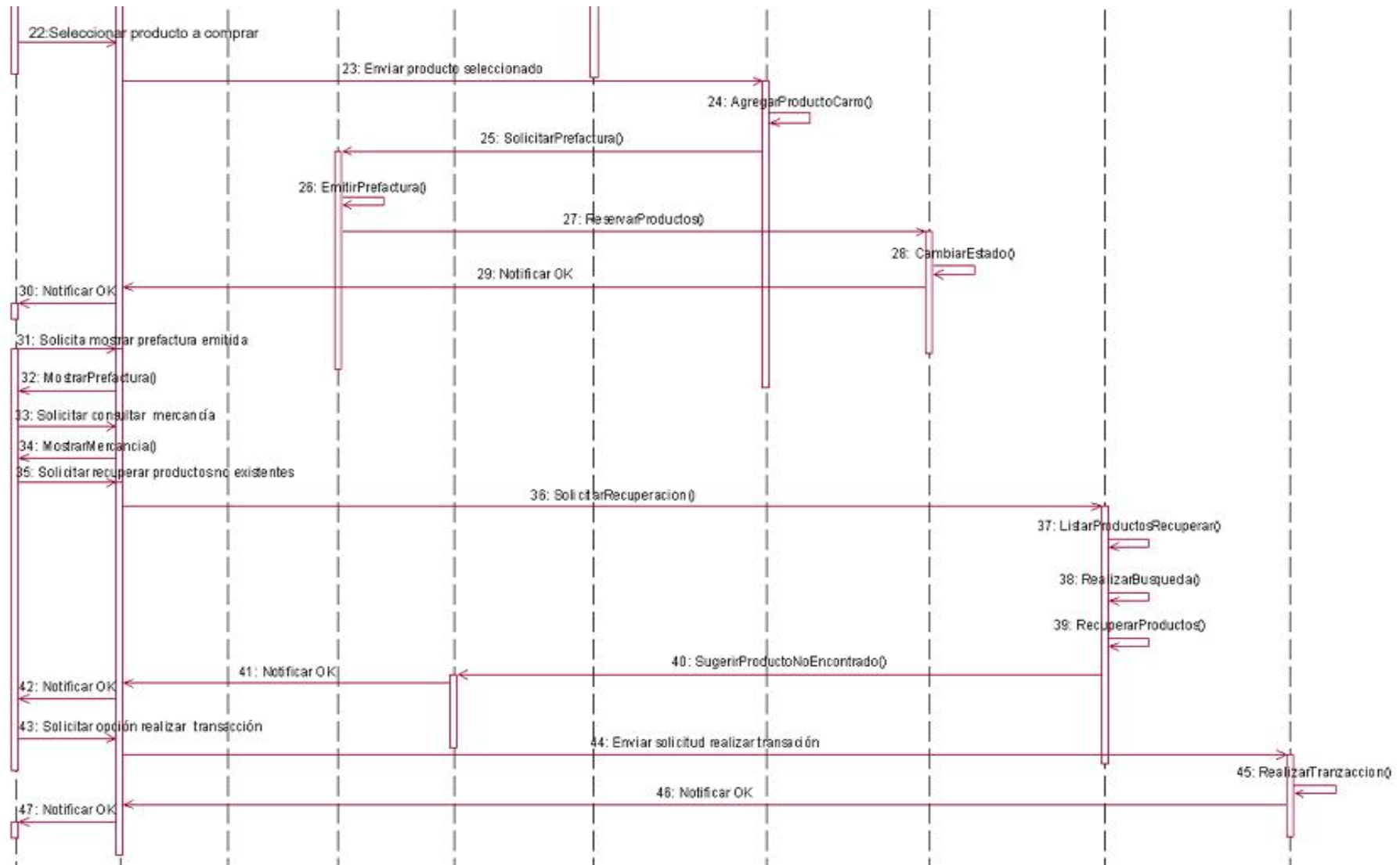


Figura 18: Diagrama Identificación de Roles del escenario Realizar Compra.

Escenario: Analizar Opinión	
Descripción:	Este escenario es el que recopila el estado de opinión de los usuarios utilizando diferentes medios como por ejemplo la encuesta, luego auxiliándose de SMA orientados a la minería de opiniones, y a la minería Web se realiza un análisis de dicha información para obtener posibles soluciones y mejoras de los servicios brindados por el sistema.
CU Asociados	CU Mostrar_Opciones, CU Recopilar_Opinion, CU Procesar_Opinion
Roles Asociados	Descripción
Rol Visualizar. (Agente_Interfaz)	Este rol es responsable de mostrar la información requerida por un usuario y una vez autenticado como cliente muestra las acciones que el mismo puede realizar sobre el sistema, en este caso emitir una opinión.
Rol Recopilar Opinión (Agente_Opinión)	Este rol es el encargado de recopilar las opiniones de los clientes mediante encuestas o mensajes de correo electrónico. La principal actividad que realiza es: una vez introducido el estado de opinión por parte de un cliente actualizar la base de datos de las opiniones.
Rol Procesar Opinión (Agente_Opinión)	Este rol es responsable de que una vez actualizada la base de datos de las opiniones los agentes de minería de datos procesen la información para buscar coincidencias y ofrecer un resumen con información de interés para los funcionarios.

Tabla 9: Especificación del escenario Analizar Opinión.

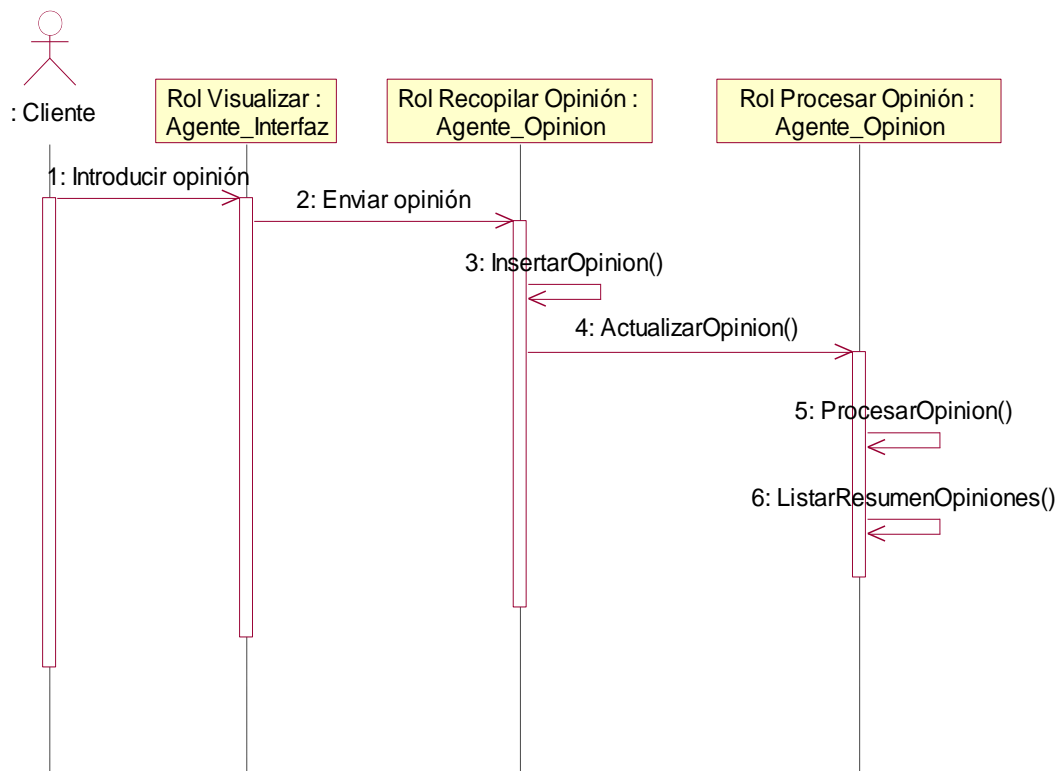


Figura 19: Diagrama Identificación de Roles del escenario Analizar Opinión.

2.3.5. Especificación de Tareas

Esta etapa especifica a través de diagramas de actividades las capacidades de cada agente. El alcance de esta fase es modelar el ciclo de vida de cada agente, reflejando las colaboraciones que necesita, y las comunicaciones en que participa; de ahí que estos diagramas se obtengan a partir del Diagrama Identificación de Roles explorando todos los escenarios en que participa el agente. Se analizan todas las interacciones y acciones internas que el agente realiza para lograr el propósito del escenario en cuestión. De cada Diagrama Identificación de Roles se obtiene una colección de tareas relacionadas, de cuya apropiada agrupación resulta el Diagrama Especificación de Tareas.

Se enfoca además en la conducta de cada agente descomponiéndola en tareas. Las tareas generalmente encapsulan alguna funcionalidad que forma una unidad lógica de trabajo.

Para cada agente en el modelo, se confecciona un diagrama de actividad que se compone de dos calles: la primera contiene una colección de actividades que simbolizan las tareas del agente, mientras que la segunda contiene actividades que representan a los otros agentes entrelazados (47).

A continuación se muestran los Diagramas de Actividades correspondientes a los agentes: Agente_Compra, Agente_Persona, Agente_Opinión y Agente_Información como se muestra de la Figura 20 a la Figura 23.

Agente_Compra

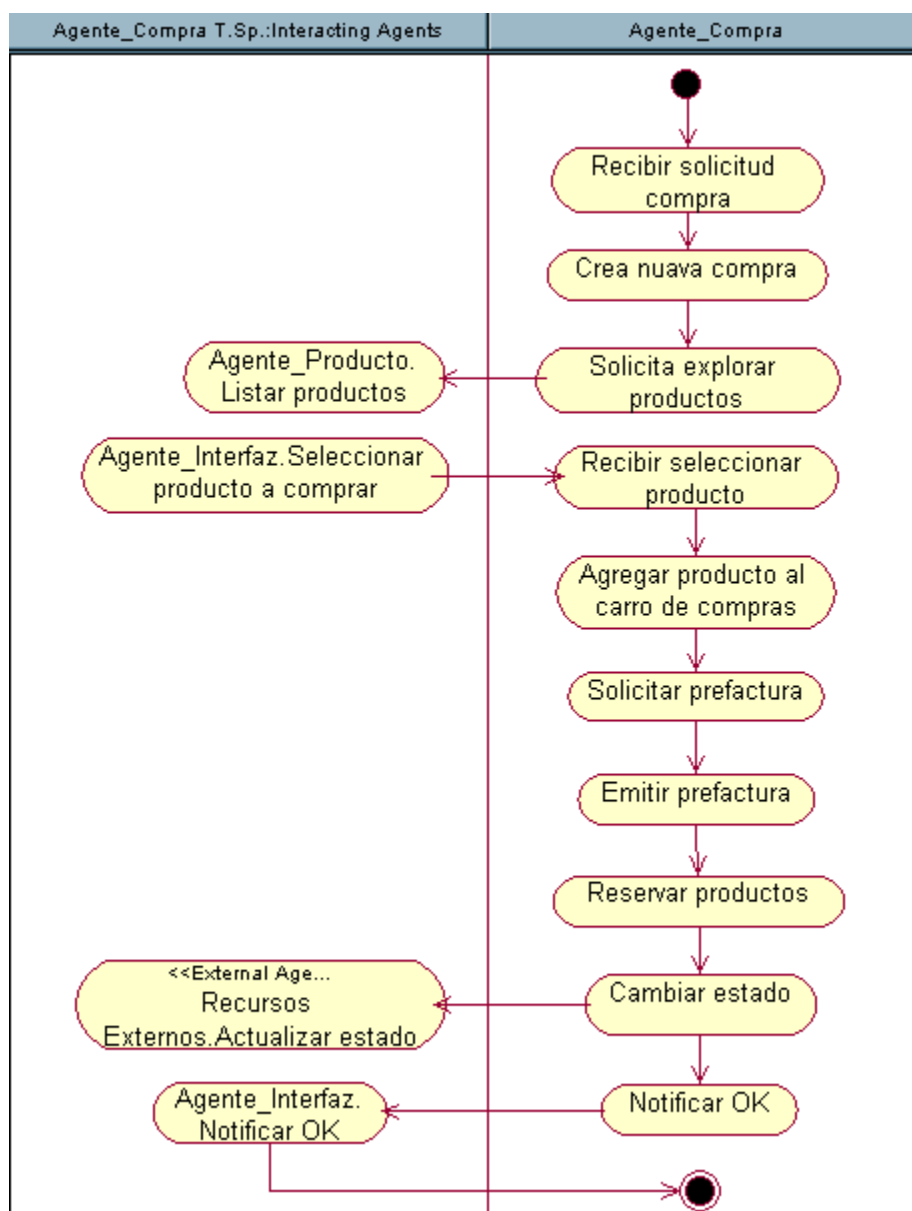


Figura 20: Diagrama Especificación de Tareas: Agente_Compra.

Agente_Persona

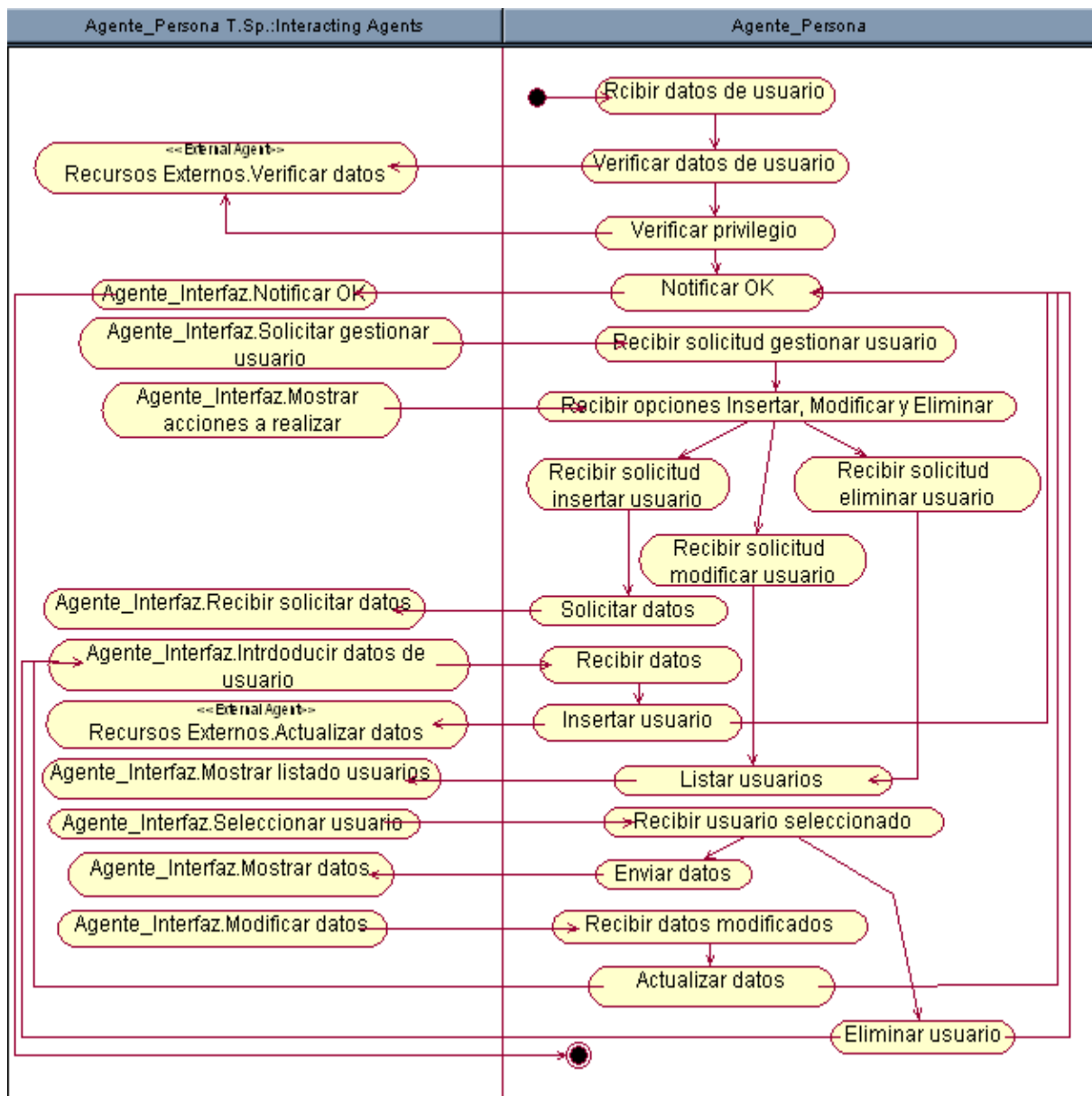


Figura 21: Diagrama Especificación de Tareas: Agente_Persona.

Agente_Información

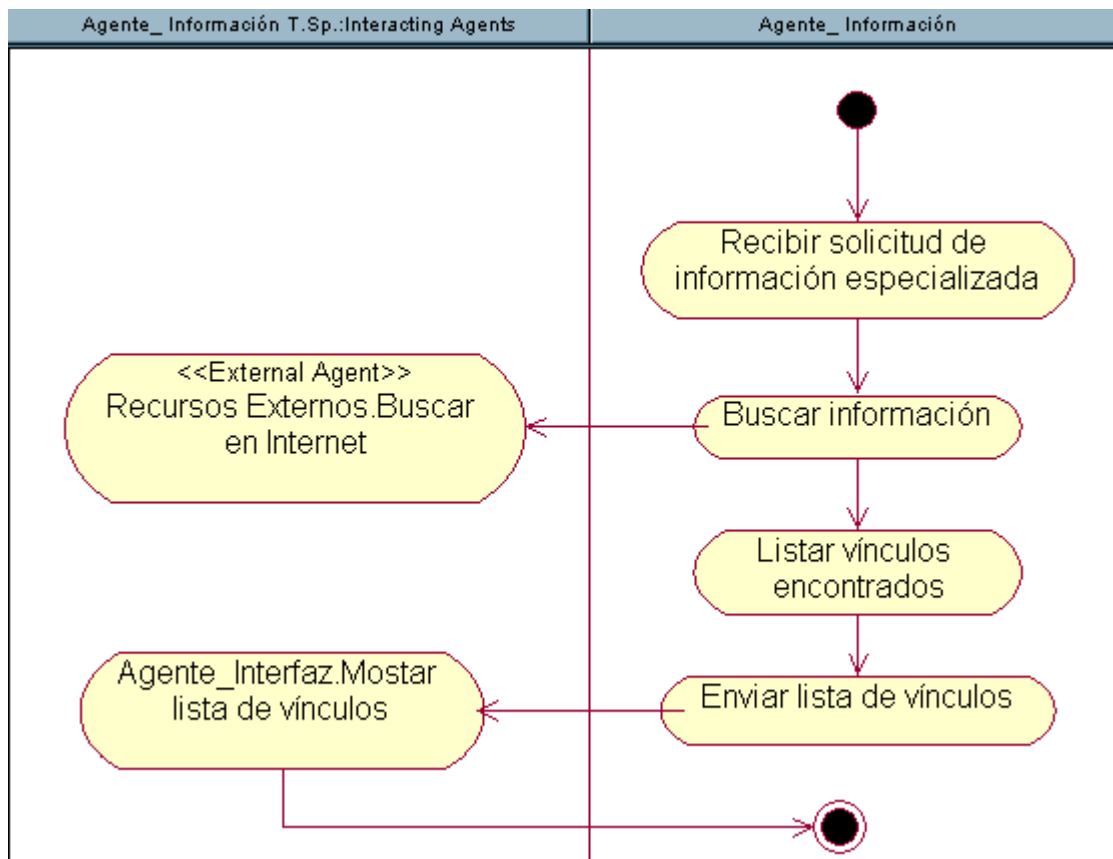


Figura 22: Diagrama Especificación de Tareas: Agente_Información.

Agente_Opinión

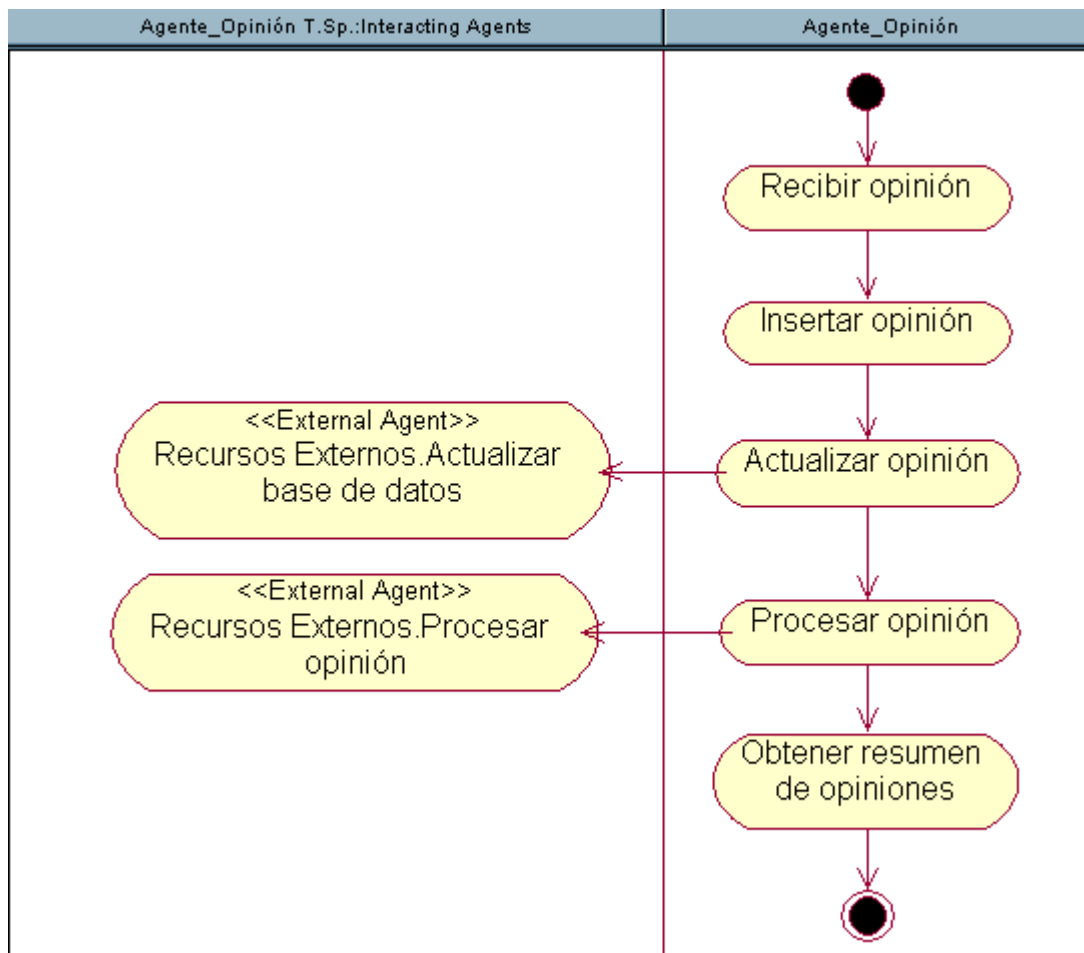


Figura 23: Diagrama Especificación de Tareas: Agente_Opinión.

2.4. Modelo de Sociedades de Agentes

Este nivel modela las interacciones sociales y dependencias entre los agentes involucrados en la solución del sistema. Las fases de este nivel son las siguientes: Descripción de Ontología, Descripción de Roles y Descripción de Protocolos.

2.4.1. Descripción de Ontología

Para detallar la ontología resultante de la solución del sistema se introduce esta etapa, en la que se confeccionan dos diagramas de clases:

- Diagrama Descripción de Ontología de Dominio: Este diagrama describe la ontología del dominio que representa a las entidades a través de las clases (Figura 24).
- Diagrama Descripción de Ontología de Comunicación: Este enfoca el conocimiento de los agentes y las relaciones comunicativas entre los mismos (Figura 25).

En el diseño de estos diagramas se parte de los resultados de la fase Identificación de Agentes. Para cada agente identificado se introduce una clase, así como una asociación para cada comunicación entre dos agentes (ignorando las distinciones sobre los roles de agentes).

Del nivel de descripción obtenido en el Modelo de Requerimientos del Sistema, se obtiene una serie de clases clasificadas en términos de conceptos, acciones y predicados, que representan todos los diversos conceptos con que interactúan los agentes identificados (47).

El alcance de esta etapa es establecer relaciones directas entre estos conceptos, para con ello garantizar un estándar en la comunicación de los agentes, ya sea interna en el sistema o con recursos ajenos a este.

La Descripción de Ontología de Dominio detalla las clases antes mencionadas; agrupando en el término conceptos todo el conocimiento que operan los agentes. Según lo anterior un concepto es una representación esquemática en forma del estereotipo clase de UML de las diversas entidades físicas y lógicas identificadas en el sistema. Las acciones representan, a grandes rasgos, las principales funcionalidades de los conceptos; mientras que los predicados hacen referencia a expresiones alternativas, que definen posibles cursos y que tienen un papel relevante para la arquitectura identificada.

Se propone la diferenciación de estas clases en particular mediante el uso del color amarillo que asigna por defecto la herramienta Rational Rose en el caso de los conceptos, el azul claro para los predicados, y el blanco para las acciones (47).

Los elementos que constituyen la ontología del dominio serán descritos como un esquema XML para garantizar la generación automática de código, a la mayor escala posible, obteniendo automáticamente una descripción XML del volumen de los mensajes expresado en uno de los posibles idiomas de comunicación de agentes

(48). Finalmente, la ontología descrita será exportada a código RDF y a código JAVA mediante la herramienta PASSI Tool-Kit.

Para cada comunicación es imprescindible especificar tres elementos: ontología, lenguaje y protocolo de interacción. Los lenguajes y protocolos de interacción son (en mayoría) estandarizados por FIPA. La ontología, al relacionarse estrictamente al problema, será definida como consecuencia de la propia aplicación.

La línea de asociación que representa cada comunicación entre agentes va dirigida del iniciador al participante, como puede deducirse de la descripción de su interacción realizada en la fase Identificación de Roles. Esta descripción permite la definición simultánea de dos elementos relacionados: el conocimiento del agente y su especificación de comunicaciones.

Communication Ontology Description Diagram

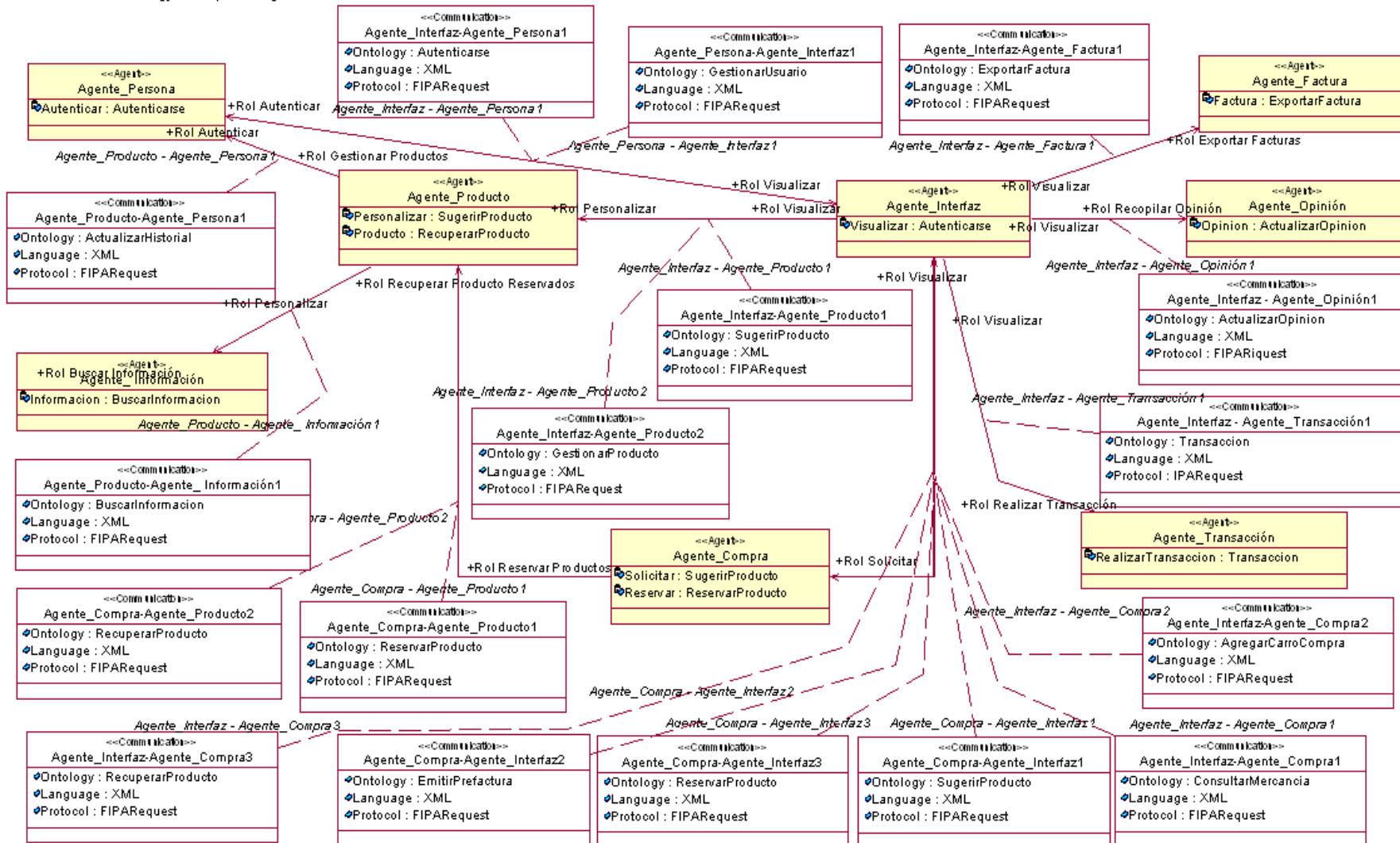


Figura 25: Diagrama Descripción de Ontología de Comunicación.

En el Anexo 1a) puede observarse un fragmento del fichero XML correspondiente a las entidades identificadas en esta fase. De igual forma el Anexo 1b) presenta un segmento de la representación de la ontología identificada en formato RDF y una sección del código JAVA generado que se expone en el 1c).

2.4.2. Descripción de Roles

El resultado de esta etapa es un diagrama compuesto por paquetes que corresponden a cada agente identificado en la fase Identificación de Agentes, y que incluyen un artefacto (clase de UML) para cada rol identificado. En estas clases correspondientes a cada rol se introducen las tareas que el rol en sí debe efectuar. De igual forma se especifican las dependencias entre los diversos roles y se establecen las comunicaciones entre roles de diferentes agentes como se especifica en el Diagrama Descripción de Ontología de Comunicación.

Parte de este diagrama es generado automáticamente por el PASSI Tool-Kit, o sea, los paquetes que representan a los agentes y los diversos roles que se identificaron en la fase Identificación de Roles.

El convenio adoptado para representar las interacciones entre roles es indicar las mismas mediante asociaciones.

Los agentes son autónomos, así que ellos pudieran negarse a proporcionar un servicio o un recurso. Por esta razón, el diseño necesita un esquema que exprese las condiciones que permitan explorar maneras alternativas de lograr las metas. Para comprender semejante esquema, se introducen en el Diagrama de Descripción de Roles (Figura 26) algunas relaciones adicionales que expresen los siguientes tipos de dependencia:

- *Dependencia de servicio.* Un rol depende de otro para provocar una meta.
- *Dependencia de recurso.* Un rol depende de otro para la disponibilidad de una entidad.
- *Dependencia Suave de Servicio/Recurso.* El servicio/recurso pedido es útil o deseable, pero no esencial para lograr la meta de un rol determinado.

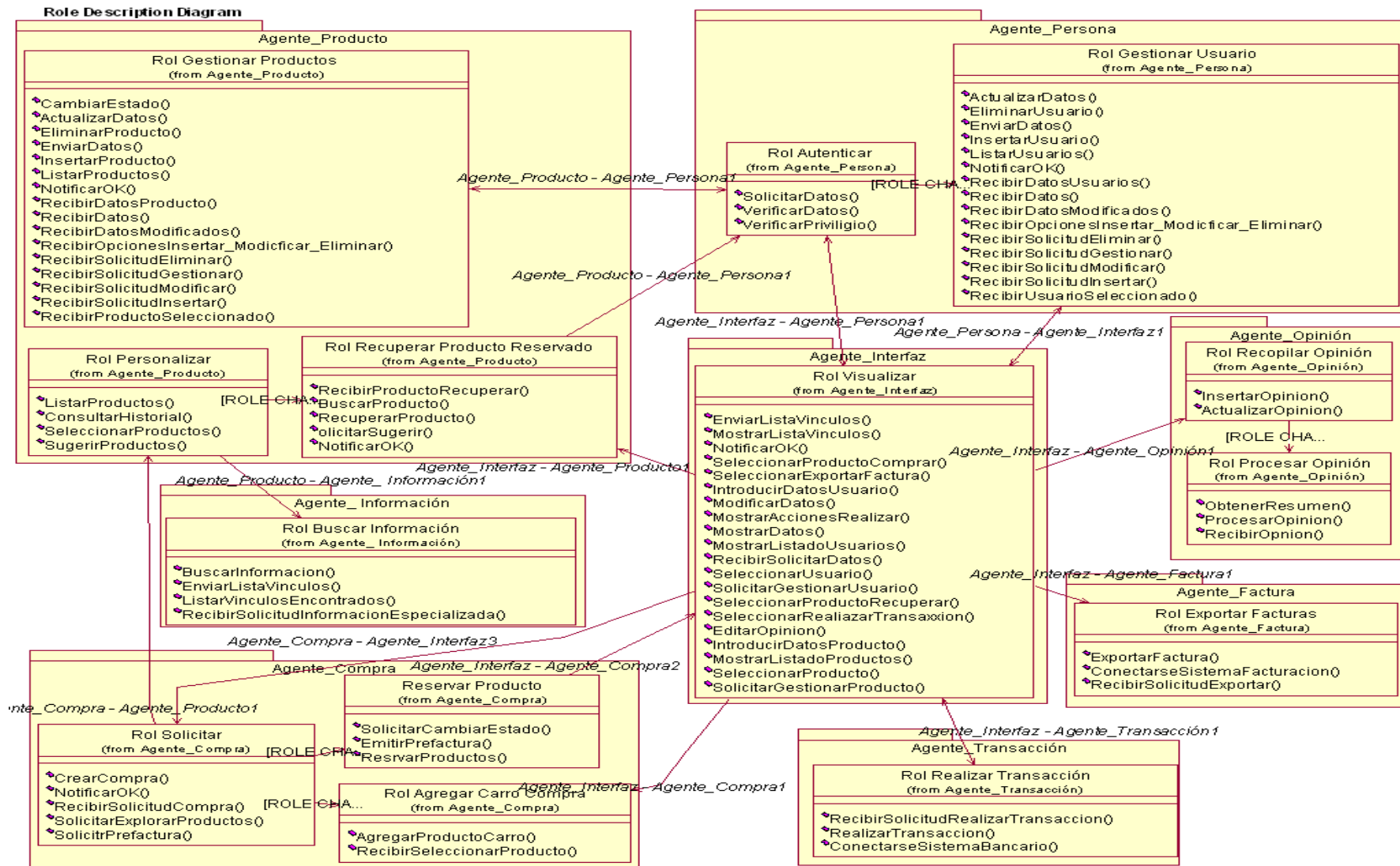


Figura 26: Diagrama de Descripción de Roles.

2.4.3. Descripción de Protocolos

Esta fase especifica la gramática de cada protocolo de comunicación en términos de “actos del habla” mediante el uso de diagramas de secuencia.

Como se ha visto en la fase Descripción de Ontología y especificado por la arquitectura de FIPA, se ha utilizado un protocolo para cada comunicación. Normalmente la documentación relacionada a dichos protocolos se expresa en forma de diagramas de secuencia AUML (extensión de UML para agentes) (41), de ahí que el diseñador no necesite especificar los mismos, incorporando en su diseño los estándares propuestos por FIPA. No obstante, en algunos casos, dichos estándares existentes no son adecuados y por consiguiente algunos protocolos especializados necesiten ser diseñados propiamente. Para ello es posible auxiliarse en la propia documentación de FIPA (49). En este caso se utiliza el protocolo FIPA Request (49) para moderar la comunicación entre los agentes. Este protocolo permite a un agente solicitar un servicio y/o recurso a otro para satisfacer una necesidad determinada. El participante procesa la demanda, y determina si está en condiciones de satisfacerla o no (ver Anexo 2). A continuación se presenta el diagrama de secuencia correspondiente a este protocolo (Figura 27).

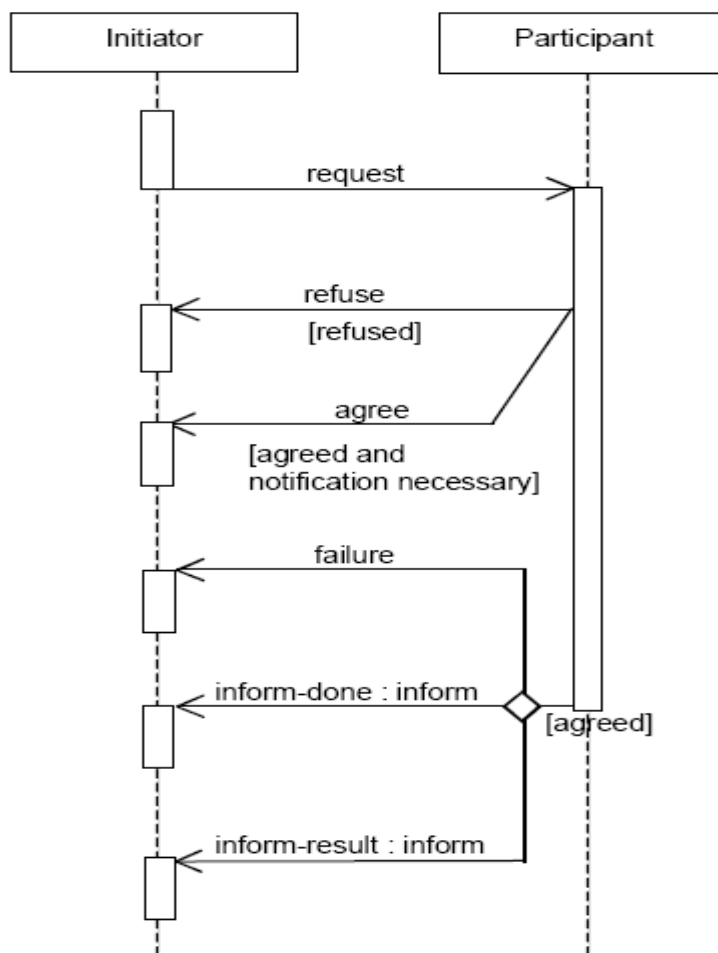


Figura 27: Diagrama Descripción de Protocolos.

2.5. Modelo de Implementación de Agente

Este modelo proporciona una modelación de la arquitectura del sistema en términos de clases y métodos. Su desarrollo conlleva las siguientes fases:

2.5.1. Definición de Estructura del Sistema Multi_Agente

En esta fase se describe la estructura de las clases de agente mediante el uso de diagramas de clases convencionales. La definición de estructura del agente produce diagramas de clases subdivididos lógicamente en SMA y Sistema de Agente. En el primero se atribuye gran importancia a la arquitectura general del sistema, identificando agentes y sus tareas en él. El Sistema de Agente se enfoca en la

estructura interior de cada agente y se revelan todos los atributos y métodos de la clase agente con sus clases internas (es decir, sus tareas).

- *Definición de Estructura del SMA:* Esta sub-etapa representa la estructura del SMA en conjunto mediante un diagrama de clases, donde cada clase simboliza uno de los agentes identificados en la fase Identificación de Agentes. El conocimiento de cada agente puede representarse mediante compartimientos de atributos (como se vio en el Diagrama de Ontología), considerando el uso de compartimientos de operaciones para la identificación de las tareas del agente. Este diagrama se muestra en la Figura 28.

- *Definición de Estructura del Agente:* En esta se ilustra la estructura interior del agente a través de las clases que lo constituyen, o sea, la clase principal del agente y las clases internas que identifican sus tareas. Se utiliza un diagrama de clases para cada agente, declarando los métodos y atributos del mismo, y las clases responsables de las tareas. Los diagramas correspondientes a los agentes: Agente_Compra, Agente_Persona y Agente_Opinión se muestran desde la Figura 29 hasta la Figura 31.

El resultado de esta etapa es obtener una estructura detallada del software, listo para ser implementado casi automáticamente.

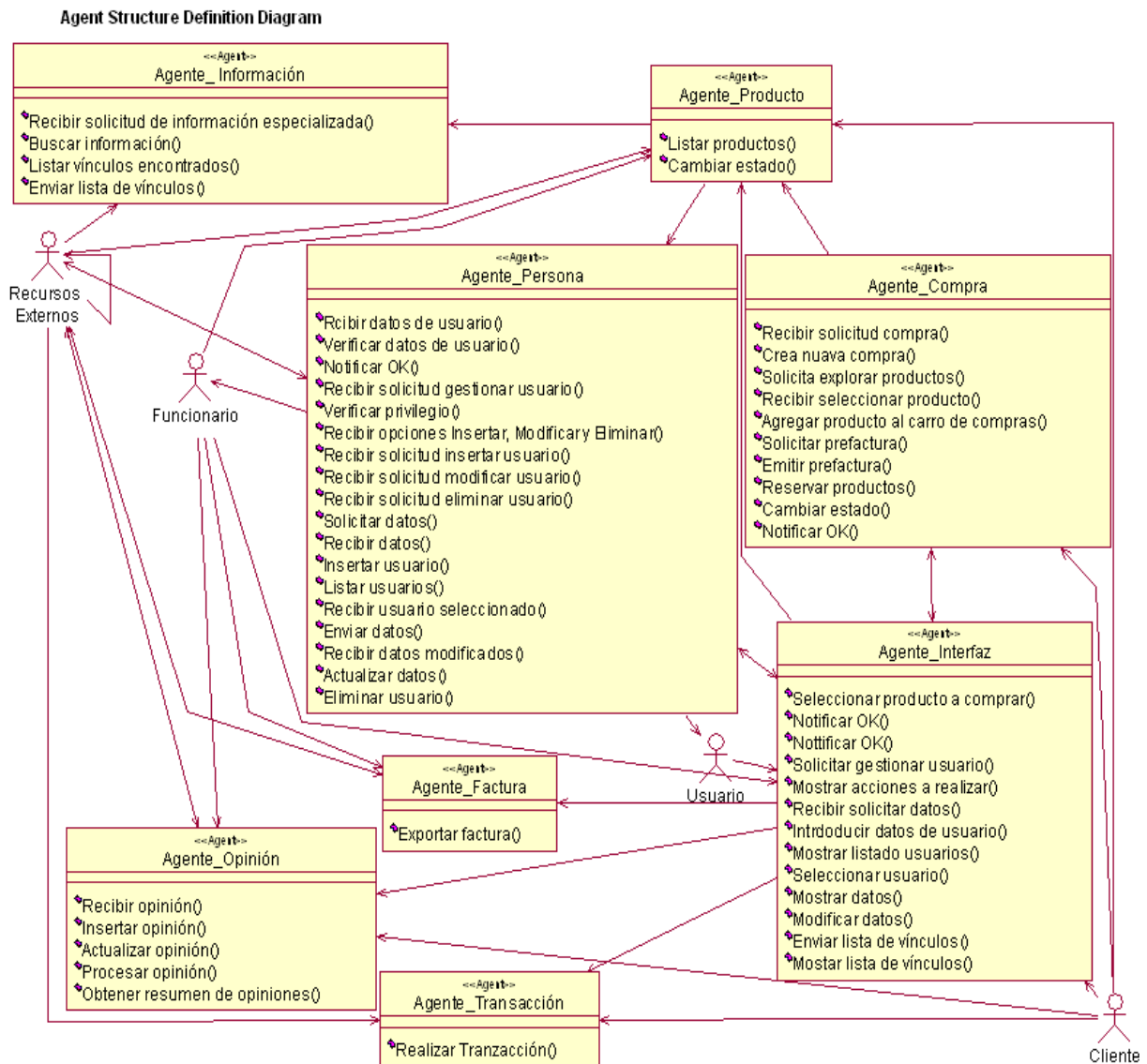


Figura 28: Diagrama Definición de Estructura del SMA.

2.5.2. Definición de Estructura del Agente

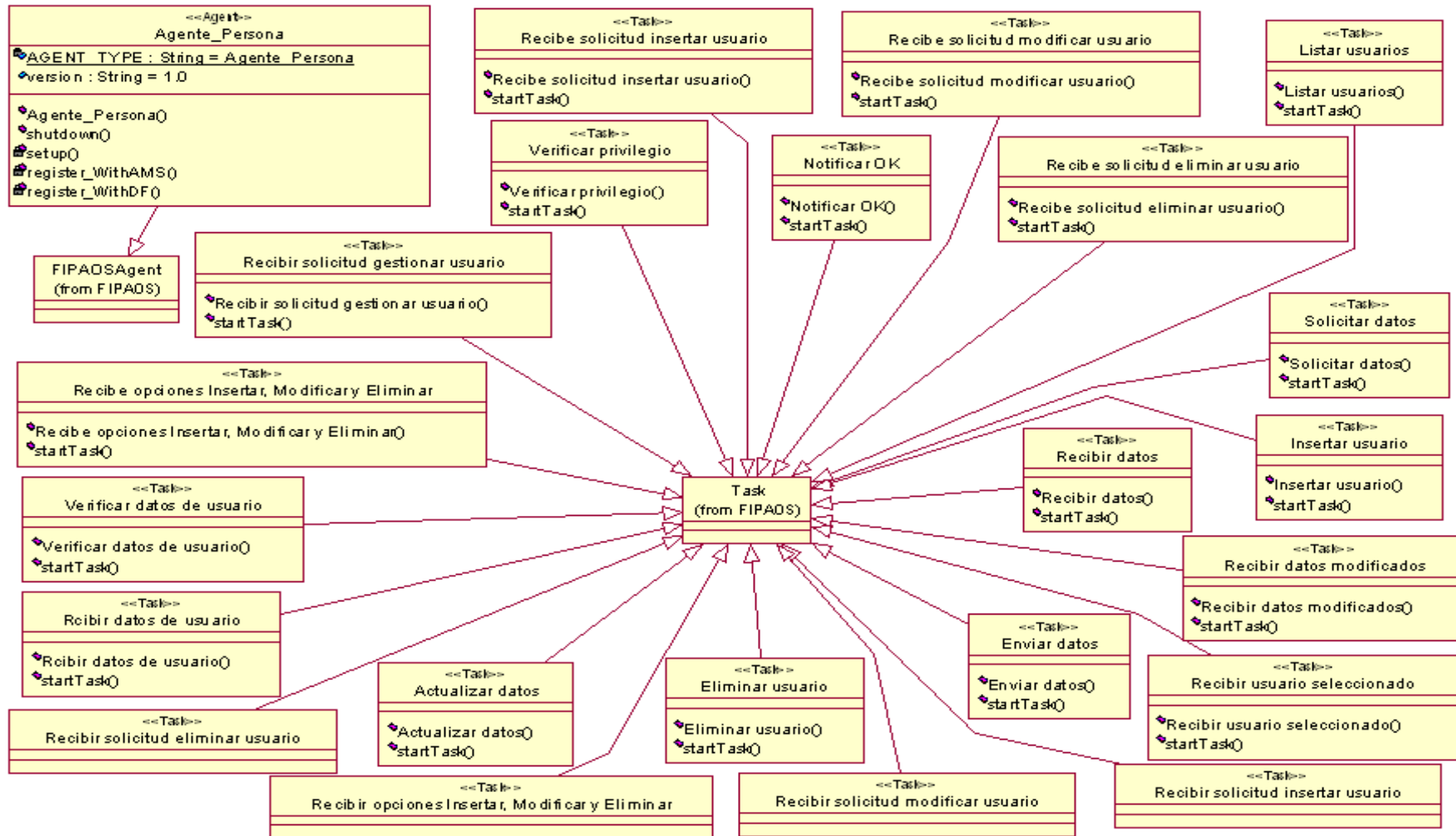


Figura 29: Diagrama Definición de Estructura del Agente_Persona.

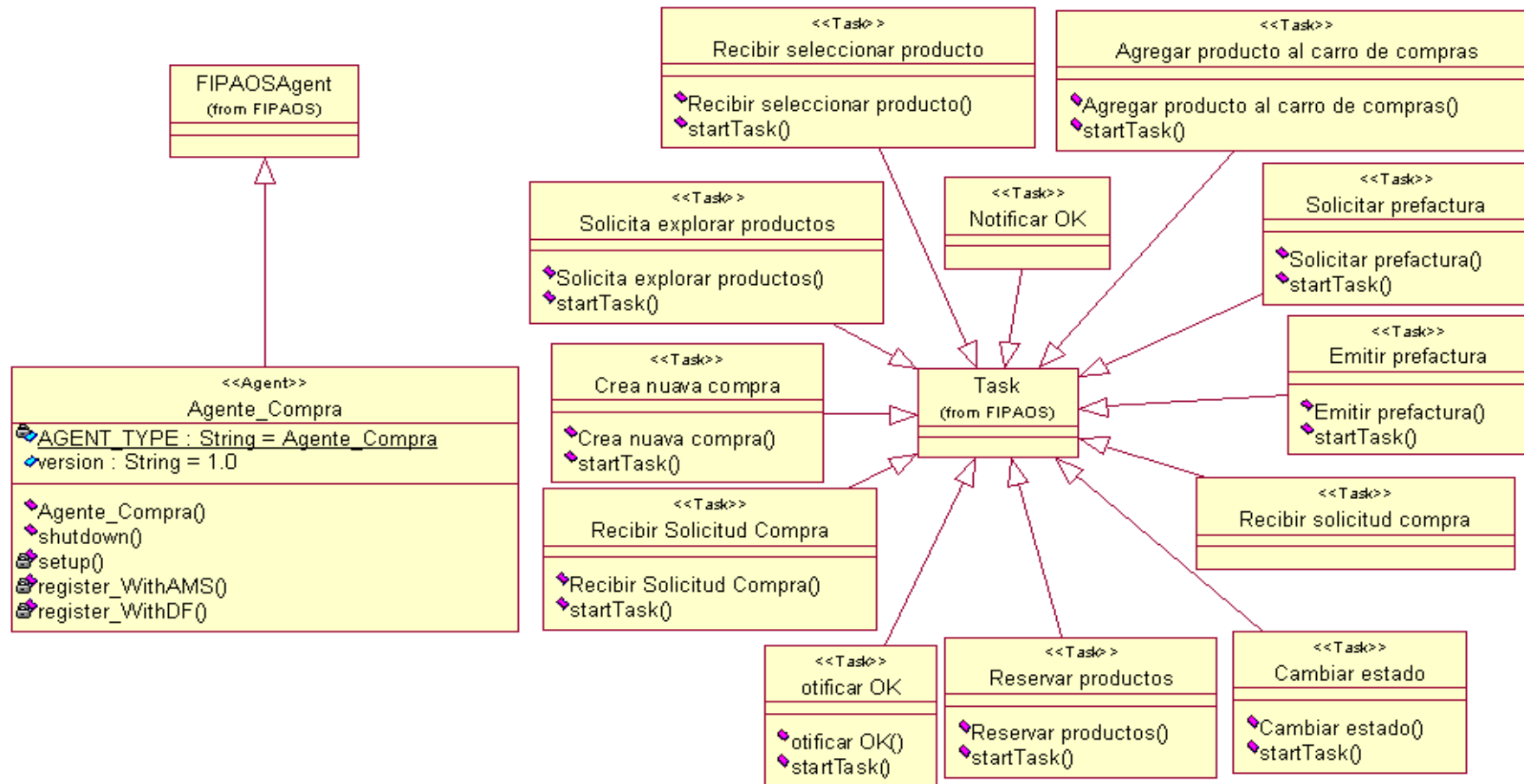


Figura 30: Diagrama Definición de Estructura del Agente_ Compra.

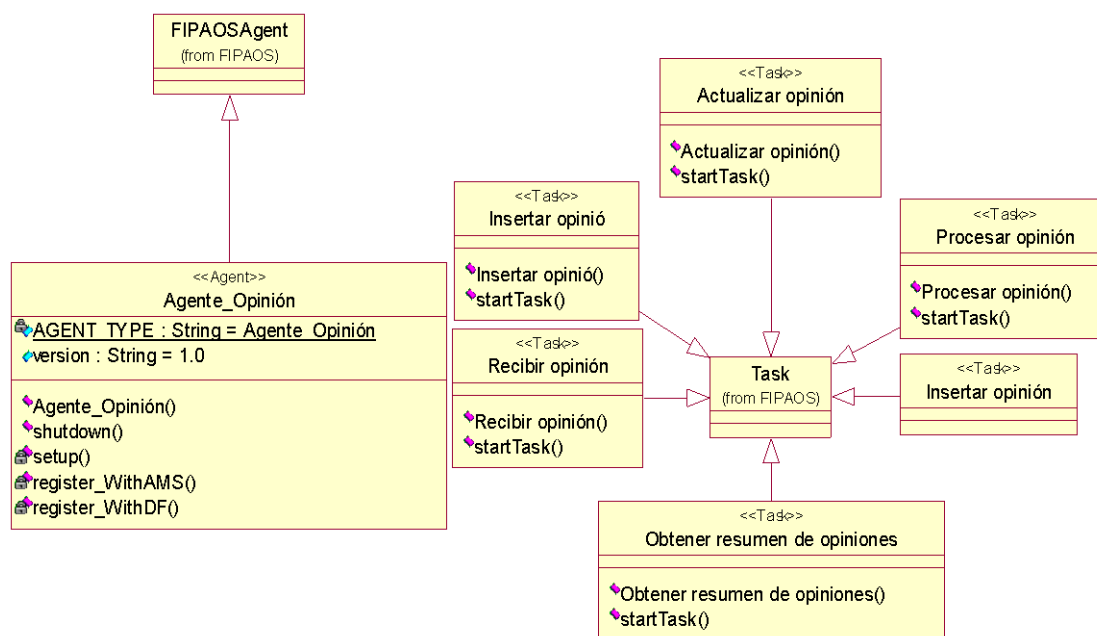


Figura 31: Diagrama Definición de Estructura del Agente_Opinión.

2.5.3. Descripción de Conducta del Sistema Multi_Agente

Al igual que la fase anterior, esta etapa produce dos diagramas de clases subdivididos lógicamente, donde el correspondiente al SMA diseña el flujo de eventos por invocación de los métodos e intercambio de mensajes, mientras que el referente al agente en particular, especifica los métodos anteriores.

- *Descripción de Conducta del SMA:* En esta sub-etapa se muestra el flujo de eventos entre y dentro de las clases de los agentes principales y sus clases internas (representando sus tareas) mediante uno o más diagramas de actividades. La convención adoptada para la realización de estos diagramas es diseñar una calle para cada agente y para cada tarea. Las actividades dentro de la calle indican los métodos de la clase relacionada. Se establecen transiciones usuales de la norma de UML para significar cualquier evento (por ejemplo, un mensaje entrante o una conclusión de la tarea) o invocación de métodos.
- *Descripción de Conducta del Agente:* En esta se incorpora la implementación de los métodos introducidos en los diagramas Definición de Estructura del Agente. La manera de describirlo es libre; o sea, la más apropiada a consideración del

diseñador (por ejemplo, trazas de flujo, diagramas de estado o descripciones de texto).

Por lo extenso del diagrama de actividades correspondiente a la Descripción de Conducta del SMA sólo se muestra el flujo correspondiente a las acciones de un cliente y este ha sido subdividido lógicamente en las acciones Compra, Opinión y Transacción, expuesto en la Figura 32 a),b),c) y d) para el caso de la Compra, mientras Opinión y Transacción se muestran en la Figura 33 y en la Figura 34 respectivamente.

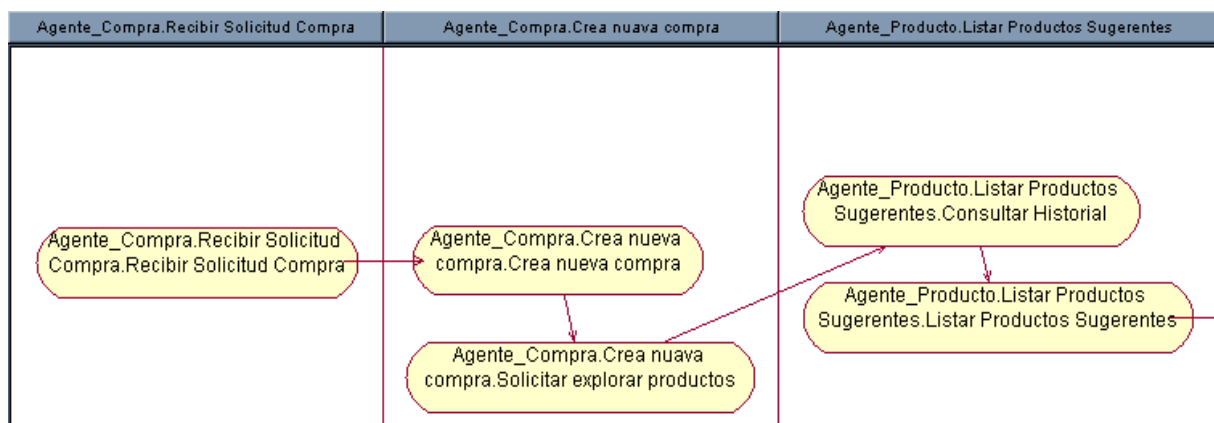


Figura 32. a)

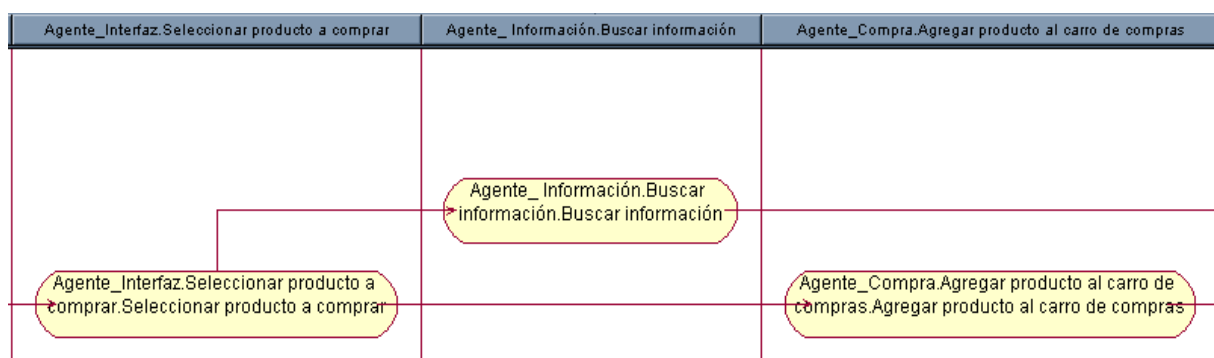


Figura 32. b)

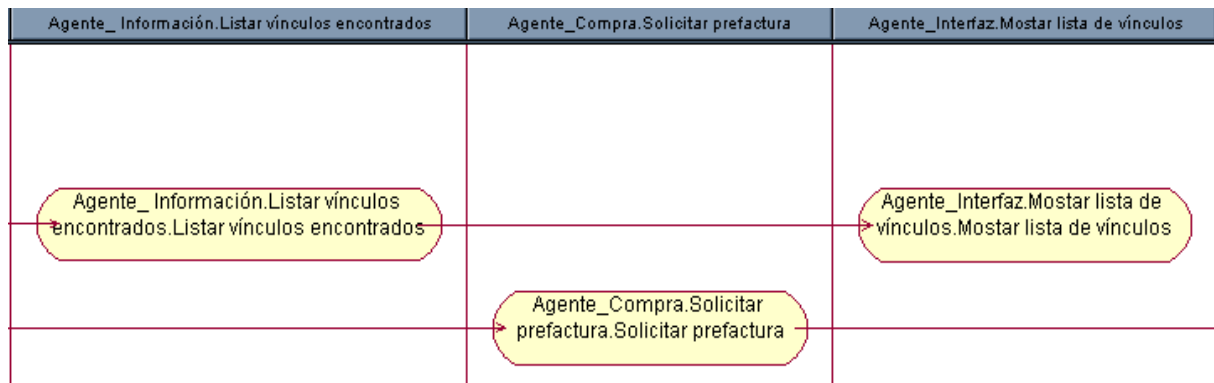


Figura 32. c)

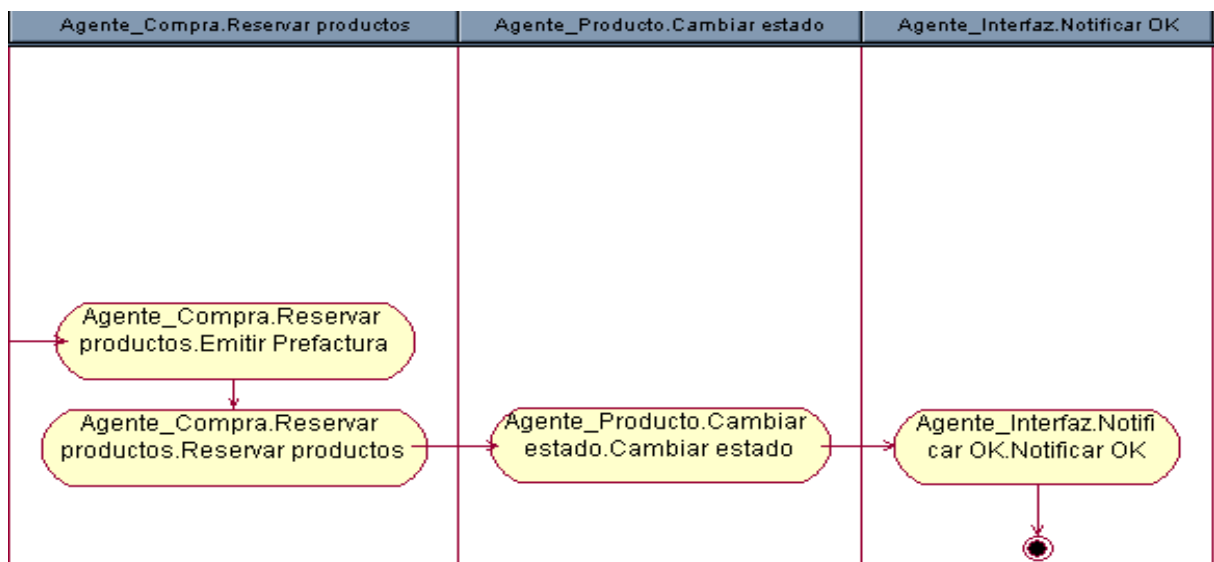


Figura 32.d)

Figura 32: Diagrama de Descripción de Conducta del SMA: Compra.

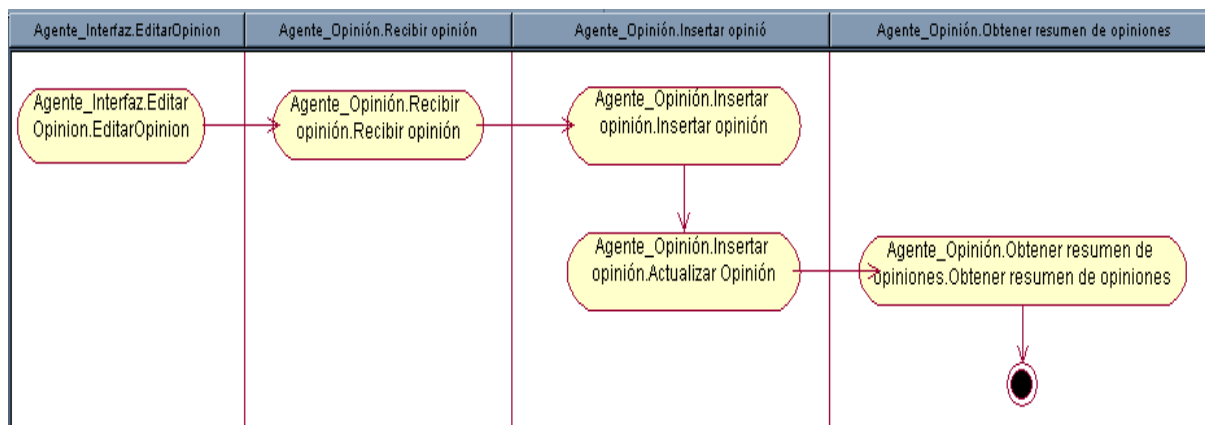


Figura 33: Diagrama de Descripción de Conducta del SMA: Opinión.

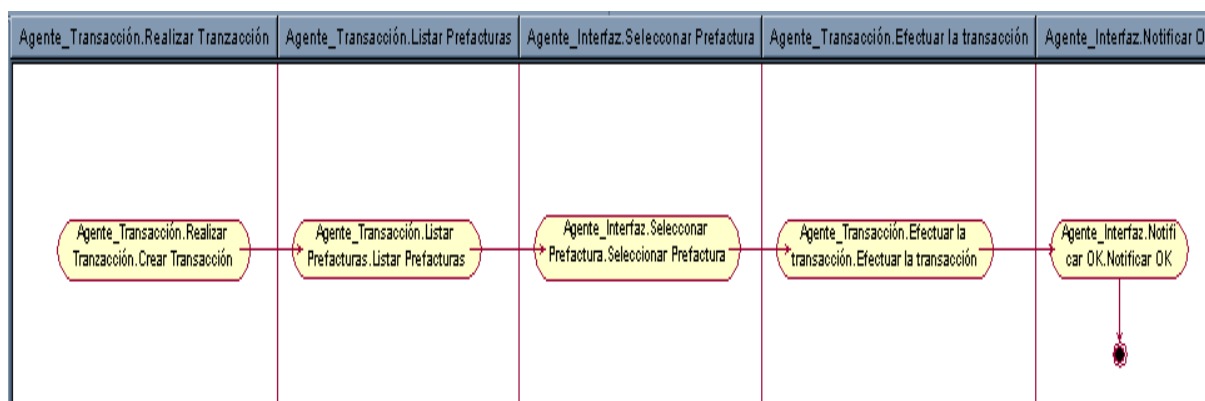


Figura 34: Diagrama de Descripción de Conducta del SMA: Transacción.

2.6. Modelo de Código

Este nivel constituye un modelo de la solución a nivel del código. Su elaboración exige las fases siguientes: Biblioteca de Código Reutilizable y Perfeccionamiento del Código Básico

2.6.1. Biblioteca de Código Reutilizable

En esta fase se trata de reutilizar patrones existentes de agentes y tareas. No es correcto hablar solamente de modelos de código, porque el diseñador utiliza diagramas que detallan la biblioteca de modelos y no a su código directamente; funcionalidad brindada mediante la herramienta PASSI Tool-Kit.

Los modelos por consiguiente no son sólo líneas de código, sino que también son partes de diseño (de agentes y tareas), lo cual puede ser reutilizado para la implementación y diseño de nuevos sistemas.

El mejor ambiente para intentar reutilizar modelos es el ambiente de diseño. La herramienta PASSI Tool-Kit ha demostrado ser muy eficiente en la conversión de elementos del diseño (clases, métodos, atributos...) en código compilable.

Los modelos más útiles son aquellos que podrían ser clasificados como modelos de la interacción, debido a la estructura de la plataforma FIPA compatible (49) que delega una tarea para cada comunicación específica. Cada vez que un agente necesita usar un protocolo, la tarea del modelo relacionado puede ser reutilizado fácilmente y sólo la parte del código consagrado al tratamiento de la información requiere modificación.

A continuación se expone parte del código generado, en este caso sólo el correspondiente al Agente_Compra, debido a la longitud de los mismos. El código correspondiente a los restantes agentes se muestra en el Anexo 3.

Código: Agente_Compra

```
public class Agente_Compra extends FIPAOSAgent {
    private static String AGENT_TYPE = Agente_Compra;
    public String version = 1.0;
    public void Agente_Compra(String platform, String name, String
ownership) {
        super(platform,name,ownership);
        setup();}
    public void shutdown() {
        //insert here your code }
    private void setup() {
        //insert here your code }
    private void register_WithAMS() {
        //insert here your code }
    private void register_WithDF() {
        //insert here your code }
    public class Recibir_Solicitud_Compra extends Task {
        public void Recibir_Solicitud_Compra() {
            //insert here your code }
        public void startTask() {
//insert here your code }
        // end of Recibir_Solicitud_Compra class }
    public class Crea_nuava_compra extends Task {
        public void Crea_nuava_compra() {
            //insert here your code }
```

```
public void startTask() {
//insert here your code  }
// end of Crea nuava compra class  }
public class Solicita explorar productos extends Task {
public void Solicita explorar productos() {
//insert here your code  }
public void startTask() {
//insert here your code  }
// end of Solicita explorar productos class  }
public class Recibir seleccionar producto extends Task {
public void Recibir seleccionar producto() {
//insert here your code  }
public void startTask() {
//insert here your code  }
// end of Recibir seleccionar productoclass  }
public class Agregar producto al carro de compras extends Task {
public void Agregar producto al carro de compras() {
//insert here your code  }
public void startTask() {
//insert here your code  }
// end of Agregar producto al carro de compras class  }
public class Solicitar prefactura extends Task {
public void Solicitar prefactura() {
//insert here your code  }
public void startTask() {
//insert here your code  }
// end of Solicitar prefactura class  }
public class Emitir prefactura extends Task {
public void Emitir prefactura() {
//insert here your code  }
public void startTask() {
//insert here your code  }
// end of Emitir prefactura class  }
public class Reservar productos extends Task {
public void Reservar productos() {
//insert here your code  }
public void startTask() {
//insert here your code  }
// end of Reservar productos class  }
public class Cambiar estado extends Task {
public void Cambiar estado() {
//insert here your code  }
public void startTask() {
//insert here your code  }
// end of Cambiar estadoclass  }
public class otificar OK extends Task {
public void otificar OK() {
//insert here your code  }
public void startTask() {
//insert here your code  }
// end of otificar OK class  }
public class Recibir solicitud compra extends Task {
// end of Recibir solicitud compra class  }
```

```
public class Notificar OK extends Task {  
    // end of Notificar OKclass }  
// end ofAgente_Compra class}
```

2.6.2. Perfeccionamiento del Código Básico

Esta fase, más bien convencional, consiste en la obtención del código fuente designado del sistema. El programador refina el código de la aplicación generado de la etapa de diseño y los patrones reutilizados.

Esta etapa de la metodología no se corresponde con los objetivos trazados para este capítulo, pues la implementación del sistema en cuestión no está concebida como parte de este trabajo, es por esta razón que ha sido recomendado desarrollarla en posteriores trabajos.

2.7. Modelo de Despliegue

Este nivel constituye un modelo de la distribución de las partes del sistema, a través de unidades de procesamiento de hardware; y su migración entre las mismas. Incorpora una única fase.

2.7.1. Configuración de Despliegue

Esta etapa describe la asignación de agentes a las unidades físicas de procesamiento disponibles y cualquier restricción en la migración y movilidad mediante el uso de diagramas de despliegue.

Esta fase es la respuesta a la necesidad de detallar la posición de los agentes en sistemas distribuidos o en contextos de agentes móviles. El Diagrama de Configuración de Despliegue (como se muestra en la Figura 35) describe donde se localizan los agentes y qué unidades de procesamiento necesitan para la comunicación entre sí. Las comunicaciones entre ellos son representadas por líneas con el estereotipo de comunicación, como en el Diagrama Descripción de Roles. Para cada comunicación descrita en dicho diagrama (si tiene lugar entre agentes en unidades de procesamiento diferentes) se traza una de las líneas antes

mencionadas. El agente receptor tiene una interfaz para mostrar que es capaz de soportar esa comunicación (es decir, que entiende el protocolo usado).

En este diagrama es posible especificar los dispositivos de hardware usados por los agentes y los modos de comunicación entre agentes en unidades de procesamiento diferentes (por ejemplo, redes por cable o inalámbricas). Si dos agentes en nodos de procesamiento diferentes necesitan comunicarse, un camino de conexión debe de ser proporcionado entre dichos nodos. Estas restricciones sobre las conexiones también podrían ser dinámicas.

En sentido general puede concluirse que la Configuración de Despliegue representa la estructura física de los SMA.

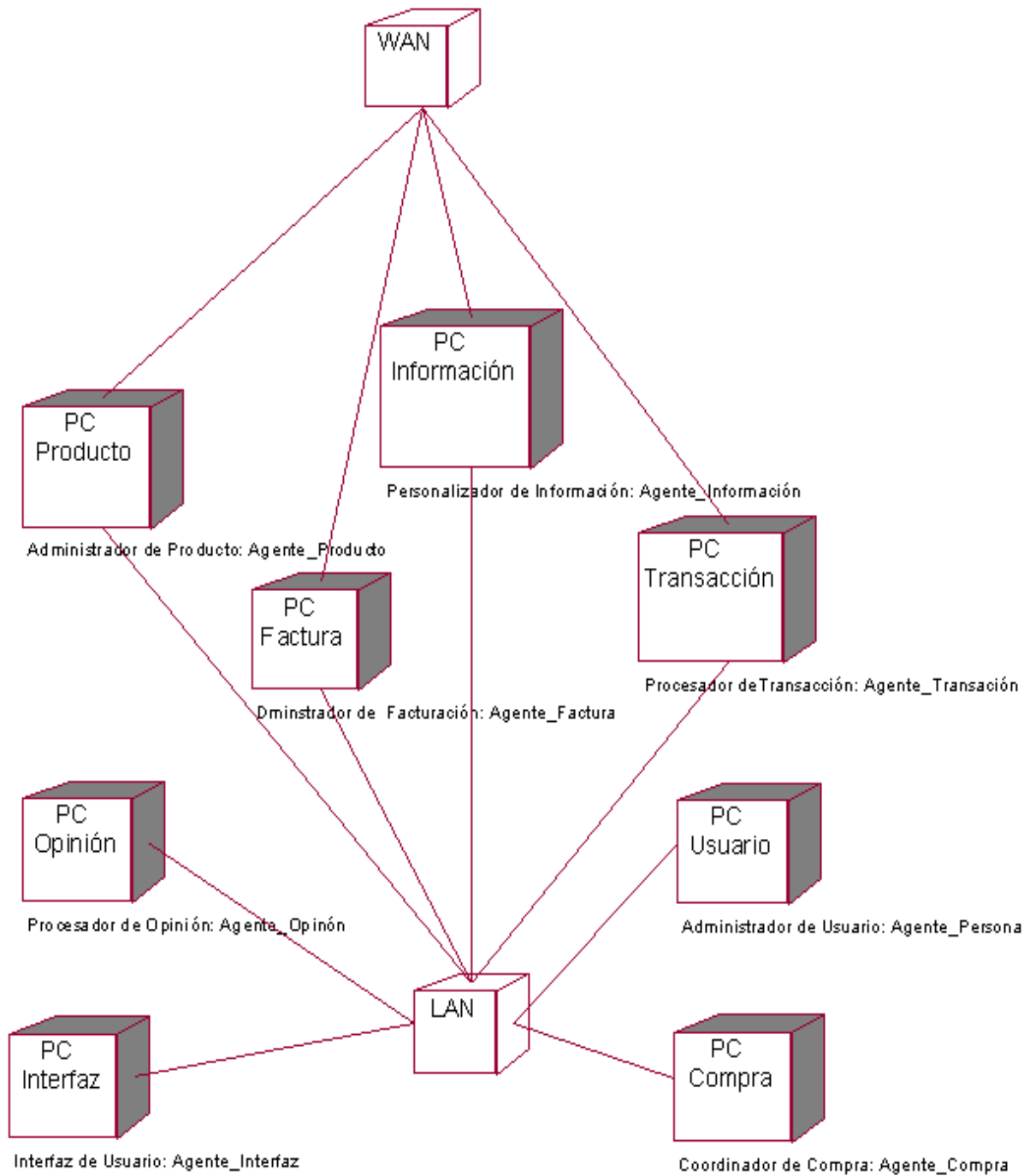


Figura 35: Diagrama de Despliegue.

2.8. Conclusiones

El desarrollo de las etapas que propone la metodología PASSI para la creación de SMA ha permitido obtener una serie de artefactos que facilitan el trabajo del programador y con ello la calidad y rapidez del proceso de construcción del sistema. Además permite poner en práctica lo planteado por los creadores de dicha metodología en un problema real, demostrando las ventajas que esta presenta y las facilidades que brinda a los ingenieros informáticos.

Con el desarrollo de este capítulo se ha podido llegar a una arquitectura que permite la creación de una aplicación destinada al Comercio Electrónico y que brinda personalización a los clientes, para ello han sido de gran utilidad las técnicas del Paradigma Orientado a Agentes.

CAPÍTULO 3: ANÁLISIS DE LOS RESULTADOS OBTENIDOS

3.1. Introducción

En el presente capítulo se realiza un análisis de los resultados obtenidos a través del desarrollo de la metodología PASSI aplicada al proyecto Plataforma de Comercio Electrónico B2B, mediante un estudio detallado de los artefactos generados en cada una de las fases que plantea la metodología. Además se utiliza la técnica de validación de expertos con el objetivo de arribar a conclusiones más acertadas sobre la documentación obtenida en el capítulo anterior.

3.2. Análisis de los resultados obtenidos

Una vez generados todos los artefactos planteados por la metodología PASSI, poniendo en práctica los requisitos que la misma plantea, es de vital importancia realizar un estudio que permita evaluar el nivel de relevancia que representa este trabajo según el problema existente y el aporte que brinda a la solución de este.

Partiendo de la necesidad del empleo de la tecnología de agentes para la obtención de una guía que permita el desarrollo de un sistema de Comercio Electrónico que garantice el trato personalizado a los clientes y el aprovechamiento de las fluctuaciones de precios, teniendo en cuenta los objetivos planteados, se realiza un análisis del presente trabajo subdividiéndolo en las distintas fases de la metodología.

Modelo de Requerimiento del Sistema: Esta fase permitió transformar los requisitos que el sistema debe cumplir en un conjunto de diagramas que constituyen el punto de partida para la descripción funcional del sistema.

- Se seleccionaron los CU necesarios con el fin de brindar personalización a los clientes, como por ejemplo: Sugerir_Productos, Recuperar_Prod_Reservados, Buscar_Inf_Especializada y Procesar_Opinion.

- Los diagramas de Descripción de Entorno y Descripción de Dominio proporcionaron una clara idea de la estructura del sistema en términos del entorno y funcionalidades.
- La identificación de agentes y su respectivo diagrama (el cual fue generado por el PASSI Tool-Kit como parte de las facilidades de la metodología PASSI) permitirán a los desarrolladores enmarcarse en las responsabilidades agrupadas por agentes con el fin de buscar autonomía en el funcionamiento del sistema y delimitar las acciones a realizar por cada agente.
- Los diagramas correspondientes a la identificación de roles y tareas, así como la descripción de los escenarios seleccionados, permitieron detallar cada acción en términos de tareas según los roles correspondientes a cada agente, proporcionando una idea más acertada sobre su especialización y las posibles interacciones con otros agentes.
- Esta fase de forma general proporcionó una vista estructural del sistema y constituyó la base fundamental en el desarrollo de las fases posteriores.

Modelo de Sociedades de Agente: Esta fase permitió a partir e la estructura descrita obtener un conjunto de diagramas que describen las interacciones y dependencias entre los agentes.

- La descripción de la ontología de comunicación permitió dar seguimiento a la estructura de las comunicaciones entre agentes.
- Mediante los diagramas de Ontología de Dominio y de Ontología de Comunicación se representaron las relaciones y dependencias entre agentes dejando definido un lenguaje común y un protocolo de comunicación.
- Se obtuvo un código en lenguaje Java correspondiente a la ontología descrita, lo que constituye un elemento esencial para el futuro trabajo de los programadores.
- El diagrama de Descripción de Roles permitió ver una forma de interacción entre agentes en términos de clases correspondientes a cada rol y especificando además métodos que describen detalladamente las funciones de cada rol.

- Esta fase constituyó el eje central de la comunicación entre agentes y las cooperaciones necesarias para cumplir las metas trazadas.

Modelo de Implementación de Agente: Esta fase estuvo dirigida a conformar el diseño funcional del sistema en término de atributos y métodos.

- El PASSI Tool-Kit permitió la generación de diagramas que muestran la estructura del SMA como un todo y como unidades de agentes. Estos diagramas proporcionaron una visión acertada sobre el diseño del sistema mediante clases relacionadas lógicamente.
- Con el fin de lograr mayor claridad en la secuencia lógica del SMA se conformó el diagrama correspondiente a la conducta del mismo, como complemento de los anteriores modelos.
- Esta etapa permitió obtener una estructura detallada del software listo para ser implementado.

Modelo de Código: Esta fase constituyó un modelo de la solución del problema planteado en forma de código reutilizable.

- Como parte de las facilidades de PASSI mediante la herramienta PASSI Tool-Kit se realizó una generación de código básico que resulta de gran relevancia para el futuro trabajo de los desarrolladores.
- El código obtenido incluyó un conjunto de patrones de agentes y tareas que facilitan el desarrollo del sistema.
- Esta etapa permitió de forma general proporcionar una herramienta de código como base para la implementación total del SMA.

Modelo de Despliegue: Esta fase es de vital importancia debido a que proporcionó una vista lógica de la distribución del SMA en términos de hardware.

- El Diagrama Descripción de Despliegue permitió la ubicación de los agentes en las unidades físicas de procesamiento y las comunicaciones entre cada dispositivo.
- Esta fase es el plano fundamental en el futuro despliegue de la aplicación.

El empleo de la metodología PASSI como parte de la tecnología de agentes permitió obtener de forma eficiente una guía compuesta por diagramas y descripciones que permite implementar un sistema que garantice el trato personalizado y el aprovechamiento de las fluctuaciones de precio beneficiando a los clientes en todo momento. Además el empleo de los SMA para el funcionamiento de la aplicación proporcionó facilidad en su creación, dado que esta tecnología de agentes es muy eficaz en sistemas complejos y autónomos.

3.3. Método de validación de expertos

Con el objetivo de conocer cuan eficiente puede resultar la propuesta de diseño del SMA a desarrollar se hace necesario conocer el criterio de importantes personalidades en el campo de la Inteligencia Artificial y en especial de la tecnología de agentes para lo cual se utiliza uno de los Métodos de Expertos existentes, debido a que estos métodos se encargan de obtener una idea de cuan eficaz ha sido el trabajo realizado a partir de las consultas a personas que tienen grandes conocimientos sobre el entorno objeto de la investigación.

En este caso para realizar la validación mediante expertos se decidió realizar un análisis a partir de la puesta en práctica del Método Delphi, el cual fue desarrollado por Olaf Helmer y otros en la Rand Corporation a mediados de la década de 1960, es considerado un método de investigación sociológica, que independientemente de que pertenece al tipo de entrevista de profundidad en grupo, se aparta de ellas agregando características particulares. Es una técnica grupal de análisis de opinión, parte de un supuesto fundamental y de que el criterio de un individuo particular es menos fiable que el de un grupo de personas en igualdad de condiciones, en general utiliza e investiga la opinión de expertos(50).

Se entiende por experto tanto al individuo u organización, con un elevado nivel de calificación en una esfera, capaz de ofrecer valoraciones conclusivas de un problema en cuestión con un máximo de competencia. De forma general son los mayores conocedores del tema a analizar, pueden ser especialistas internos o externos.

Las principales características del método están dadas por el anonimato de los participantes (excepto el investigador), iteración (manejar tantas rondas como sean

necesarias), retroalimentación (feedback) controlada, sin presiones para la conformidad, respuesta de grupo en forma estadística (el grado de consenso se procesa por medio de técnicas estadísticas) y justificación de respuestas (discrepancias/consenso). Suelen distinguirse tres etapas o fases fundamentales en la aplicación del método.

1. Fase preliminar. Se delimita el contexto, los objetivos, el diseño, los elementos básicos del trabajo y la selección de los expertos.
2. Fase exploratoria. Elaboración y aplicación de los cuestionarios según sucesivas vueltas, de tal forma que con las respuestas más comunes de la primera se confecciona la siguiente.
3. Fase final. Análisis estadísticos y presentación de la información.

Según los autores(50),(51) el método tiene grandes ventajas debido a que, permite obtener información de puntos de vista sobre temas muy amplios y específicos, el horizonte de análisis puede ser variado y permite la participación de un gran número de personas sin dificultades con anonimato y retroalimentación.

Plantean como desventajas del método, que no es totalmente fiable porque incluso los más eminentes especialistas pueden equivocarse, así como su tiempo de ejecución y la masiva participación.

3.4. Validación de expertos mediante el método Delphi aplicado a la modelación del sistema de Comercio Electrónico para la empresa Cubalse

Para la aplicación del método es necesario considerar metodológicamente dos aspectos básicos de su caracterización sobre los cuales se sustenta, que son:

La selección del grupo de expertos a encuestar: personas conocedoras, con reconocida competencia y con experiencia en el tema que garantice la confiabilidad de los resultados, creativos e interesados en participar.

Elaboración de los cuestionarios: tener en cuenta la teoría de la comunicación, con mecanismos que reduzcan los sesgos en las respuestas, preguntas claras, precisas

e independientes. Suelen ser preguntas cuantitativas para calcular medias y rangos, y cualitativas para la justificación de sus opiniones.

Para la aplicación del método a partir de lo expresado anteriormente se tuvieron en cuenta los siguientes pasos lógicos:

1. Planificación del criterio de expertos

- Concepción inicial del problema.
- Selección de los expertos.
- Los componentes o elementos necesarios para llevar a cabo el trabajo.

2. Elaboración y aplicación de las encuestas.

3. Procesamiento y análisis de información.

Para dar cumplimiento a los objetivos trazados en este epígrafe, a continuación se desarrollan los pasos mencionados.

3.4.1. Planificación del criterio de expertos

Concepción inicial del problema: Se partió de la necesidad de la modelación de un SMA para el Comercio Electrónico, deseando obtener una guía que permitiera a los futuros programadores la creación de un sistema de ventas online que garantizara la personalización a los clientes y el aprovechamiento de las fluctuaciones de precio. Para cumplir los objetivos propuesto se utilizó la metodología PASSI.

Selección de los expertos: La selección del grupo de expertos se hizo aplicando la metodología elaborada por el Comité Estatal para la Ciencia y la Técnica de Rusia para seleccionar expertos en un tema, y según lo que plantea la misma se realizaron los siguientes pasos:

- Confección de un listado inicial de personas posibles de cumplir los requisitos para ser expertos en la tecnología de Agentes Inteligentes, previamente consultada su disposición para participar.

- Se les realiza una primera pregunta de autoevaluación en una escala creciente del 1 al 10, valor que se corresponde con el grado de conocimiento en la materia, de donde se obtiene su Coeficiente de Conocimiento.
- Se les realiza una segunda pregunta que permite valorar un grupo de aspectos que influyen sobre el nivel de argumentación o fundamentación del tema a estudiar, tales como: categoría científica, grado de conocimiento en la tecnología de Agentes Inteligentes, su experiencia teórica, su experiencia práctica, la bibliografía nacional consultada, la bibliografía internacional consultada, su conocimiento del estado del problema y su intuición, de donde se obtiene el Coeficiente de Argumentación. Estos aspectos propician obtener resultados con calidad, junto a otras cualidades propias de estos como pueden ser: la seriedad, la honestidad, la sinceridad y la responsabilidad, lo que hace que las opiniones brindadas sean confiables y válidas para el objetivo propuesto.
- Finalmente con el Coeficiente de Conocimiento y el Coeficiente de Argumentación se calcula el Coeficiente de Competencia, el cual puede ser alto medio o bajo. Siempre se escogen aquellos expertos que tengan Coeficiente de Competencia alto o medio, nunca los de coeficiente bajo.

En el Anexo 4a) se muestra la plantilla a llenar por cada persona escogida como posible experto y en el Anexo 4b) se llevan a cabo los pasos correspondientes a la selección de los expertos.

Debido a la limitación de tiempo y a la lejanía de un grupo de expertos se decidió realizar la consulta a un grupo no muy amplio de personalidades, pero si a los mayores conocedores del tema en la región (Cuba) y de algunas ciudades europeas. De forma general los profesionales entrevistados aportaron respuestas con un alto peso, debido a que el nivel de conocimiento del tema es favorable (Valor de 10 puntos) y su experiencia teórico-práctica, así como el nivel de bibliografía nacional e internacional consultada se corresponden con el rango alto en su mayoría y en lo referente al conocimiento del estado del problema es alto para los expertos cubanos y medio para el experto europeo con el que se pudo contactar como representante del grupo de investigación de Inteligencia Artificial de la Universidad Complutense de

Madrid . El aspecto intuición se mantiene con un nivel alto para todos los expertos seleccionado.

Los expertos consultados son Máster o Doctores en Ciencias, con participación en numerosos eventos y la realización de trabajos e investigaciones relacionadas con el tema de la tecnología de agentes. Presentan más de 15 años de experiencia y han sido las personas dedicadas a promover el tema de Agentes Inteligentes.

Los componentes o elementos necesarios para llevar a cabo el trabajo:

- Tener conocimiento de la tecnología de agentes.
- Contar con los elementos necesarios de las metodologías que plantea la ISOA lo que permitió la selección de la metodología PASSI.
- Conocer detalladamente las fases de la metodología PASSI.
- Tener acceso a la herramienta PASSI Tool-Kit y conocer su funcionamiento.

3.4.2. Elaboración y aplicación de las encuestas

Inicialmente se debía mandar un cuestionario a los expertos pidiéndoles que dieran su opinión sobre los temas de interés de la metodología PASSI, este paso fue sustituido por los resultados obtenidos en el Framework de evaluación de la metodología PASSI (avalada por estos expertos en investigaciones antes realizadas), expuestos en el Anexo 5, siendo la aplicación de esta metodología el objeto a validar.

En la elaboración de los cuestionarios se utilizaron preguntas cerradas mediante la votación por una o más alternativas, las cuales abarcaron cada uno de los artefactos que se fueron generando en las diferentes fases de la metodología, y finalmente se utilizaron las preguntas abiertas donde los expertos podían expresar su criterio para justificar estimaciones y factores que pudieran influir en el comportamiento de una determinada cuestión.

A los expertos se les manda un resumen de lo obtenido en el Framework de PASSI mencionado anteriormente, pero en esta ocasión centrado en aspectos específicos

del sistema modelado y se les pide que llenen el cuestionario y den sus razones respecto a las opiniones en que difieren.

La obtención de valoraciones estables por parte de los expertos no resultó prolongada, por lo que no fue necesario continuar con las entrevista.

3.4.3. Procesamiento y análisis de la información.

Las encuestas realizadas a expertos en la tecnología de agentes, constituyeron el punto de partida para la obtención de una categoría en términos de calidad para el trabajo realizado, es por ello que en la confección de los cuestionarios se le dio un peso a cada posible respuesta, en el caso de las opciones: bien, regular o mal, se otorgó 5 puntos a la máxima categoría, 4 a la media y 3 a la mínima, de esta forma se obtiene un termino cuantitativo de indicador a medir. En las restantes preguntas se otorgó un peso entre 0 y 5 puntos que sirvió para realizar un control estadístico de los resultados obtenidos como se muestra en el Anexo 6a). El Anexo 6b) muestra los resultados de la encuesta realizada a uno de los expertos.

El control estadístico se llevó a cabo mediante el cálculo de los valores promedio en cada fase y finalmente un promedio total de calidad por cada experto. La gráfica correspondiente a los resultados obtenidos se muestra en la Figura 36.

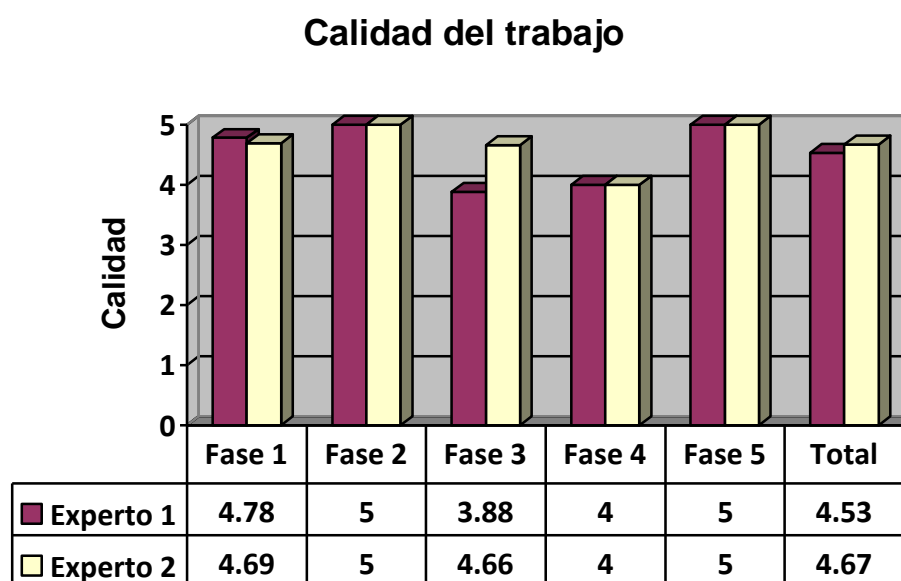


Figura 36: Gráfica del promedio de la calidad por cada experto consultado.

Para confirmar la existencia del consenso de los expertos a través de respuestas coincidentes se realizó un cálculo a partir de la fórmula de Kendall para calcular el Coeficiente de Correlación o Coeficiente de Concordancia como se muestra en el Anexo 7.

Una vez obtenido un valor aceptado de concordancia se concluye que no es necesario realizar más cuestionarios, por tanto se procede al análisis de los resultados obtenidos.

El control estadístico se llevó a cabo mediante un promedio por fases y general según los resultados obtenidos, lo cual permitió obtener un único valor de calidad para cada fase y de forma general, con estos términos se calculó el porcentaje de calidad del trabajo como se muestra en la gráfica de la Figura 37.

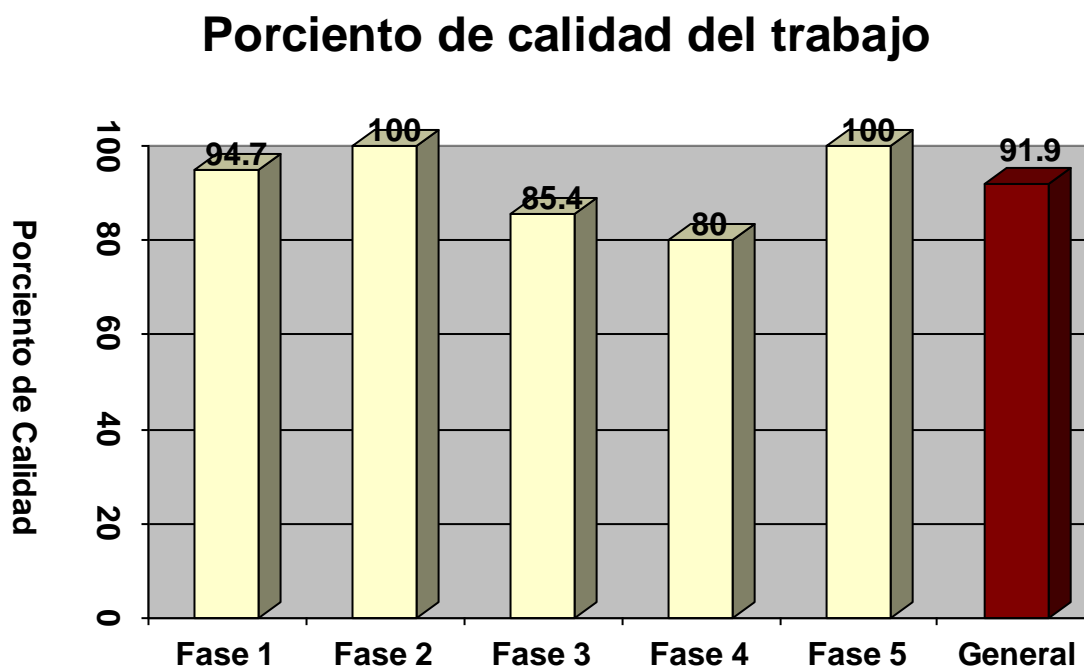


Figura 37: Gráfica del porcentaje de calidad del trabajo.

El trabajo presenta un 91.9 % de calidad según la validación realizada por los expertos y con un valoración de 5 puntos otorgada por los mismos.

Como parte de los resultados que se obtuvieron se demostró que:

- Era necesario el empleo de la tecnología de agentes.

- Se hizo un correcto uso de la metodología PASSI.
- Se generaron todos los artefactos correspondientes a cada fase descrita con un adecuado nivel de calidad llegando a obtener un 100 % en las fases 2 y 5.
- Se realizó un correcto uso de las herramientas y lenguajes de modelado.
- La documentación obtenida satisface las necesidades de la empresa Cubalse.

3.5. Conclusiones

Con el desarrollo de este capítulo se ha podido obtener un resumen significativo de los beneficios que aportaron cada uno de los artefactos generados utilizando la metodología PASSI, garantizando principalmente la personalización al cliente y el aprovechamiento de las fluctuaciones de precios. Se obtuvo también una completa validación por parte de los expertos en el tema de agentes, gracias a la utilización del método Delphi. De esta forma queda completamente validada la guía de trabajo para los futuros programadores obtenida en el Capítulo 2.

CONCLUSIONES

Teniendo en cuenta el objetivo trazado referente a la modelación de un Sistema Multi-Agente(SMA) para la Gestión de Ventas Personalizadas en el proyecto Plataforma de Comercio Electrónico Empresa-Empresa(B2B) para la empresa Cubalse y con relación a los resultados esperados, se puede decir que se cumplió con dicho objetivo de forma satisfactoria. Se llevó a cabo un estudio que permitió conocer y emplear los conceptos y principios del Paradigma Orientado a Agentes, lo que constituyó el eje central en el desarrollo de este trabajo. La revisión de la bibliografía correspondiente a la tecnología de agentes, así como su aplicación en sistemas complejos permitió reafirmar la necesidad del desarrollo de un Sistema Multi-Agente para la creación del sistema de Comercio Electrónico para la Gestión de Ventas Personalizadas correspondiente a la empresa Cubalse. Como parte de los resultados esperados se logró obtener una guía que abarca todos los artefactos generados durante las etapas de análisis, diseño e implementación según lo plantea la metodología de desarrollo de software utilizada (PASSI referente al Proceso para la Especificación e Implementación de Sociedades de Agentes), lo que permitirá a los desarrolladores la implementación del sistema propuesto.

La metodología PASSI, seleccionada debido a las facilidades que presenta, entre ellas la presencia de las fases fundamentales para el desarrollo de una aplicación, su carácter iterativo e incremental, así como la reutilización de código y la comodidad de contar con la herramienta PASSI Tool-Kit permitió obtener una documentación básica para el desarrollo del sistema propuesto. Esta documentación, brinda posibilidades que (al ser implementado el sistema) representa mejoras considerables para el comercio en Cuba y constituye el punto de partida para la revolución del Comercio Electrónico en el país.

RECOMENDACIONES

Se recomienda:

Que a partir de lo desarrollado en el presente trabajo, se proceda con la creación del sistema para la empresa Cubalse según lo expuesto en esta guía, siendo implementado de forma total y con ello lograr promover el Comercio Electrónico en Cuba.

Continuar extendiendo el Comercio Electrónico al Comercio Minorista en Cuba, tomando como base lo expuesto en el presente trabajo.

Profundizar en la tecnología de Agentes Inteligentes, y explotar todas las posibilidades que la misma brinda en el desarrollo de complejos sistemas.

BIBLIOGRAFÍA

1. *Ciber-ventas [En línea]* Disponible en: <http://www.ciber-ventas.com.ar> (9 de Noviembre del 2007).
2. *Alta Voz [En línea]* Disponible en: <http://www.altavoz.net> (9 de Noviembre del 2007).
3. *Smart Sales [En línea]* Disponible en: <http://www.smartsales.com.ar> (9 de Noviembre del 2007).
4. *Mercado Libre [En línea]* Disponible en: <http://mercadolibre.com> (9 de Noviembre del 2007).
5. *El Economista* Disponible en: http://www.economista.cubaweb.cu/2002/nro160/160_308.html (21 de Noviembre del 2007).
6. *Citmatel [En línea]* Disponible en: <http://www.citmatel.cu/linea2.php> (21 de Noviembre del 2007).
7. ALFREDO SIMÓN, R. V., ALEJANDRO ROSETE... *Las metodologías para el desarrollo de Sistemas Multi-Agentes y RUP*. 16 p.
8. V. JULIÁN, V. B. *Agentes Inteligentes: el siguiente paso en la Inteligencia Artificial*. 2000.
9. WOOLDRIDGE M., N. J., KINNY D *The Gaia Methodology for Agent-Oriented Analysis and Design. International Journal of Autonomous Agents and Multi-Agent Systems*. 2000.
10. ---. *Intelligent Agents: Theory and Practice. The Knowledge Engineering Review*. 1995, vol. 10, 115-152 p.
11. *FIPA Methodology: Glossary of Terms*. 2003, Disponible en: <http://www.fipa.org> (12 de Octubre del 2007).
12. ERRECALDE, M. L. *Agentes y Sistemas Multiagente. Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC). En 2006*.
13. LUIS MENGUAL, N. B. Y. J. B. *Arquitectura Multi-Agente sugura basada en un sistema de implementación automática de protocolos de seguridad. Universidad Politécnica de Madrid*.
14. *Interacción y coordinación en Sistemas Multiagente. Departamento de Informática Universidad Nacional de San Luis (UNSL). San Luis. Argentina Septiembre del 2006*.

15. N. R. JENNINGS, K. S., AND M. WOOLDRIDGE. *A roadmap of agent research and development*. *Autonomous Agents and Multi-agent Systems*. 1998, vol. 1, 7-38 p.
16. FERBER, J. *Multi-Agent Systems*. Addison-Wesley, 1999.
17. P., R. S. Y. N. *Artificial Intelligence: A Modern Approach*. Englewood Cliffs, NJ: Prentice-Hall. 1995.
18. P., G. M. R. Y. K. S. *Software agents*. *Communications of the ACM*. 1994, 48-53, 147p.
19. JOSÉ M. MOLINA LÓPEZ, J. G. H. Y. A. M. B. B. *Agentes y Sistemas Multiagentes*. 2004, Disponible en: Departamento de Informática, Univ. Carlos III de Madrid <http://giaa.inf.uc3m.es>
Centro de Difusión de Tecnologías, Univ. Politécnica de Madrid <http://www.ceditec.etsit.upm.es>.
20. GÓMEZ, J. J. *Metodologías para el desarrollo de sistemas multi-agente*. Departamento de Sistemas Informáticos y Programación Facultad de Informática, Universidad Complutense.
21. N. R. JENNINGS, K. S., AND M. WOOLDRIDGE. *A roadmap of agent research and development*. *Autonomous Agents and Multi-agent Systems* 1998, 1:275–306 p.
22. CARLES SIERRA, J. A. R.-A., PABLO NORIEGA. *Ingeniería de Sistemas Multiagente vía Instituciones Electrónicas*.
23. NWANA, H. S., ROSENSCHEIN, J., SANDHOLM. *Agent-Mediated Electronic Commerce: Issues, Challenges and some Viewpoints*. In *Proceedings of the Second International Conference on Autonomous Agents – AGENTS'98*. 1998, 189-196 p.
24. PABLO NORIEGA, J. A. R.-A., CARLES SIERRA. *Aplicaciones de Tecnología de Agentes en Comercio Electrónico*. Instituto Investigación en Inteligencia Artificial (IIA-CSIC). 2005.
25. *Jango* [En línea] Disponible en: <http://www.jango.com/> (15 de Noviembre del 2007).
26. *FireFly* [En línea] Disponible en: <http://www.firefly.com/> (15 de Noviembre del 2007).
27. *Bargain Finder* [En línea] Disponible en: <http://bf.cstar.ac.com/> (15 de Noviembre del 2007).
28. *Kasbah* [En línea] Disponible en: <http://www.kasbah.media.mit.edu> (15 de Noviembre del 2007).
29. CARLA SALAZAR SERRUDO, J. G. V. *Sobre el Comercio Electrónico basado en agentes*. 2001.

30. MAGMA Minnesota Agent Marketplace Architecture [En línea] Disponible en: <http://www.magma.ca/~mrw/agents/> (15 de Noviembre del 2007).
31. MADRUGA, A. *La Inteligencia Artificial desde un enfoque humanista*. [En línea] 2007, Disponible en: <http://alejandro313.blog.com.es/> (15 de Noviembre del 2007).
32. ANA B. GIL, Z. G., FRANCISCO J. GARCÍA. *Recomendadores en un Sistema Multi-agente Adaptativo para el Comercio Electrónico*. 2005, 8 p.
33. CORPORATIONS, I. *Guía de comercio electrónico. Estrategias de éxito para el comercio electrónico*. 2000, 16 p.
34. JENNINGS, N. R. *An agent-based approach for building complex software systems*. *Communications of the ACM*. 2001, vol. 44 (No. 4, April 2001), 35-41 p.
35. WEIß, G. *Multiagent systems: a modern approach to distributed artificial intelligence* The MIT Press, Cambridge. 2000.
36. IGLESIAS C., M. G., GONZÁLEZJ. C., VELASCO J. R. *Analysis and design of multiagent systems using MAS-CommonKADS*, *Intelligent Agents IV LNAI 1998*, 1365p.
37. DELOACH, S. *Analysis and Design using MaSE and agentTool*, *12th Midwest Artificial Intelligenc Cognitive Science Confence MAICS2001, Oxford, Ohio*. 2001.
38. BRESCIANI P., P. A., GIORGINI P., GIUNCHIGLIA F., MYLOPOULOS J. *Tropos: An Agent-Oriented Software Development Methodology Autonomous Agents and Multi-Agent Systems 8*. 2004, 203-236 p.
39. L. PADGHAM, M. W. P. *A Methodology for Developing Inteligent Agents*. *Firts Internacional Joint Conference on Autonomous and Multi-Agent Systems AAMAS, Bologna, Italy*. 2002.
40. CHELLA A., C. M., SABATUCCI L., SEIDITA V. . *From PASSI to Agile PASSI: tailoring a design process to meet new needs*. *2004 IEEE/WIC/ACM International Joint Conference on Intelligent Agent Technology (IAT'04)*. September 20-24, 2004, Beijing (China)
41. *PASSI: a Process for Agents Societies Specification and Implementation* Disponible en: <http://www.mozart.csai.unipa.it> (5 de diciembre de 2007).
42. GÓMEZ SANZ JORGE J., R. F. *The INGENIAS Methodology*. *Fourth Iberoamerican Workshop on Multi-Agent Systems Iberagents 2002*.
43. COSSENTINO M., B. P., LOMBARDO S., SABATUCCI L. *Introducing Pattern Reuse in the Design of Multi-Agent Systems*, *Workshop : Agent Infrastructure, Tools and Applications at NODe 2002*. October 2002, Germany.

-
44. PÉREZ P, R. *Evaluación de la Metodología PASSI para la Especificación e Implementación de Sistemas Multi-Agentes: Universidad de las Ciencias Informáticas, Departamento de la Especialidad Facultad 3 Ciudad de la Habana, Cuba.* 2006.
 45. *Ingenias. Grasia. Grupo de Agentes de Software: Ingeniería y aplicaciones.* Disponible en: <http://grasia.fdi.ucm.es/ingenias/Spain/estado/index.php> (15 de Diciembre del 2007).
 46. JACOBSON I., B. G., RUMBAUGH J. “*El proceso unificado de software*”. Madrid Pearson Educación 2000.
 47. *Tutorial de la herramienta PASSI Tool-Kit ver 1.1.1: Universidad de Palermo. Facultad de Ingeniería Informática*
 48. BORDINI R. H, D. M., DIX J., SEGHTROUCHNI A. “*Multi-Agent Programming. Languages, Platforms and Applications*” New York, United States of America, Springer, 2005
 49. *FIPA Foundation for Intelligent Physical Agents* Disponible en: <http://www.fipa.org/> (5 de Abril del 2008).
 50. RUIZ OLABUÉNAGA, J. E. I., M. A. . *La técnica Delphi. La descodificación de la vida cotidiana. Métodos de investigación cualitativa.* 1989, 171-179 p.
 51. PARISCA, S. El método Delphi. Gestión tecnológica y competitividad. Estrategia y filosofía para alcanzar la calidad total y el éxito en la gestión empresarial. 1995 La Habana: Academia, n p. 129-130.