



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRONICA

ESCUELA DE INGENIERÍA EN SISTEMAS

ANÁLISIS COMPARATIVO DE LA PRODUCTIVIDAD DE METODOLOGIAS COMMONKADS VS BUCHANAN PARA EL DESARROLLO DE UN SISTEMA EXPERTO DE GESTIÓN DE CULTIVOS PARA LA JURECH.

TESIS DE GRADO

Previa a la obtención del título de

INGENIERO EN SISTEMAS INFORMÁTICOS

AUTOR: FAUSTO DANILO LLANGARÍ SILVA

RIOBAMBA-ECUADOR

2016

©2016, Fausto Danilo Llangarí Silva

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
ESCUELA DE INGENIERÍA EN SISTEMAS

El Tribunal de Tesis certifica que: La Tesis de Grado: “ANÁLISIS COMPARATIVO DE LA PRODUCTIVIDAD DE METODOLOGIAS COMMONKADS VS BUCHANAN PARA EL DESARROLLO DE UN SISTEMA EXPERTO DE GESTIÓN DE CULTIVOS PARA LA JURECH.”, de responsabilidad del señor Fausto Danilo Llangarí Silva, ha sido minuciosamente revisada por los Miembros del Tribunal de Tesis, quedando autorizada su presentación.

ING. WASHINGTON LUNA

**DECANO DE LA FACULTAD DE
INFORMÁTICA Y ELECTRÓNICA**

ING. PATRCIO MORENO

**DIRECTOR DE LA ESCUELA DE
INGENIERÍA EN SISTEMAS**

ING. EDWIN MEJÍA

DIRECTOR DE TESIS

ING. IVÁN MENES

MIEMBRO DE TESIS

NOTA:

Yo, Fausto Danilo Llangarí Silva soy responsable de las ideas, doctrinas y resultados expuestos en esta Tesis y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo.

Fausto Danilo Llangarí Silva

DEDICATORIA

La realización de este trabajo va dedicado con mucho amor a mi familia Vane, Alvin y en especial a mi Madre Jaqueline por su apoyo incondicional en todo momento, sus consejos, sus valores y enseñanzas, y sobre todo su infinito amor, que me han permitido cumplir esta meta.

AGRADECIMIENTO

A Dios por bendecirme cada día y haberme brindado el don de la vida, a la Madre dolorosa por acogerme siempre bajo su manto y acompañarme en cada paso que doy.

Al director y miembro de tesis Ing. Fernando Mejía, que con sus conocimientos fueron una guía en el desarrollo de la misma.

A la Escuela Superior Politécnica de Chimborazo
y sus docentes por brindarme
una excelente formación académica.

INDICE DE ABREVIATURAS

AM:	Agent Model
API:	Interfaz de Programación de Aplicaciones.
BD:	Base de datos
CBR:	Case Based Reasoning
CM:	Communication Model
CML:	Conceptual Modeling Language
DAFO:	Debilidades, Amenazas, Fortalezas y Oportunidades
DARTS:	Design Approach for Real-Time Systems
DM:	Design Model
E / S:	Entrada / Salida
ESPOCH:	Escuela Superior Politécnica de Chimborazo
GUI:	Interfaz Gráfica de Usuario
IA:	Inteligencia Artificial
IC:	Ingeniero del Conocimiento
KADS:	Knowledge Acquisition and Documentation Structuring
KM:	Knowledge Model
KRL:	Knowledge Representation Language
KSL:	Knowledge System Language
KSM:	Knowledge Structure Manager
LDP:	Language Design Program
MIKE:	Model-based and Incremental Knowledge Engineering
mm:	milímetro
ms:	milisegundo
MVC:	Model - View - Controller
OM:	Organization Model
OMG:	Object Management Group
OMT:	Object Modeling Technique
PM:	Project Management
PSM:	Problem Solving Method
RAE:	Real Academia Española
ROOM:	Real-Time Object-Oriented Modeling
ROPES:	Rapid Object-Oriented Process for Embedded Systems
RT-UML:	Real-Time - Unified Modeling Language
RTTM:	Real-Time Task Model

SE:	Sistema Experto
SBC:	Sistema Basado en el Conocimiento
STR:	Sistema de Tiempo Real
SBCTR:	Sistema Basado en el Conocimiento de Tiempo Real
TAN:	Tarea de Alto Nivel
TM:	Task Model
TOPKAT:	The Open Practical Knowledge Acquisition Toolkit
TTR:	Tarea de Tiempo Real
UML:	Unified Modeling Language

ÍNDICE GENERAL

DEDICATORIA.....	v
AGRADECIMIENTO	vi
INDICE DE ABREVIATURAS.....	vii
ÍNDICE GENERAL	ix
ÍNDICE DE TABLAS.....	xiv
ÍNDICE DE FIGURAS.....	xvi
RESUMEN	xvii
SUMARY.....	xviii
CAPITULO I.....	- 1 -
MARCO REFERENCIAL	- 1 -
1.1 Antecedentes	- 1 -
1.2 Delimitación	- 2 -
1.3 Formulación del Problema.....	- 2 -
1.3.1 Sistematización del Problema.....	- 2 -
1.4 Justificación del Proyecto de Tesis	- 3 -
1.4.1 Justificación Teórica	- 3 -
1.4.2 Justificación Metodológica	- 3 -
1.4.3 Justificación Práctica	- 3 -
1.4.4 Alineación del Tema.....	- 4 -
1.5 Objetivos	- 4 -
1.5.1 Objetivo General.....	- 4 -
1.5.2 Objetivos Específicos	- 4 -
1.6 Hipótesis	- 5 -
CAPITULO II	- 6 -
MARCO TEÓRICO	- 6 -
2.1 Introducción.....	- 6 -
2.2 Inteligencia Artificial.....	- 6 -
2.3 Sistema Experto.....	- 8 -
2.3.1 Evolución.....	- 10 -
2.3.1.2 Evolución de los Sistemas Expertos en el Tiempo	- 10 -
2.3.2 Tipos de Sistemas Expertos.	- 11 -

2.3.2.1	<i>Basados en Reglas.</i>	- 11 -
2.3.2.2	<i>Basados en Casos (CBR or Case Based Reasoning).</i>	- 11 -
2.3.2.3	<i>Basados en Redes Bayesianas.</i>	- 11 -
2.3.3	Áreas de Aplicación	- 12 -
2.3.4	Experto Humano	- 12 -
2.3.4.1	<i>Atributos de un Experto Humano.</i>	- 12 -
2.3.5	Conocimiento	- 13 -
2.3.6	Adquisición y Representación del Conocimiento.	- 13 -
2.3.6.1	<i>Adquisición del Conocimiento</i>	- 13 -
2.3.6.2	<i>Representación del Conocimiento.</i>	- 13 -
2.3.7	Arquitectura de un Sistema Experto	- 17 -
2.3.8	Base de Conocimiento	- 17 -
2.3.9	Base de Hechos	- 17 -
2.3.10	Base de Datos.	- 17 -
2.3.11	Motor de Inferencia	- 18 -
2.3.12	Interfaz de Usuario	- 18 -
2.3.13	Subsistema de Explicación	- 18 -
2.3.14	El Módulo de Adquisición del Conocimiento.	- 19 -
2.4	Ventajas.	- 19 -
2.5	Limitaciones	- 19 -
2.6	Metodologías	- 20 -
2.6.1	<i>Metodologías para el Desarrollo de un Sistema Experto</i>	- 20 -
2.6.2	<i>Metodología de la Investigación</i>	- 22 -
2.7	Metodología CommonKADS	- 23 -
2.7.1	Perspectiva Histórica	- 23 -
2.7.2	Ciclo de Vida en CommonKADS	- 25 -
2.7.2.1	<i>El Análisis</i>	- 26 -
2.7.2.2	<i>El Diseño</i>	- 26 -
2.7.2.3	<i>Implantación del Sistema</i>	- 26 -
2.7.2.4	<i>Instalación</i>	- 26 -
2.7.2.5	<i>El Uso</i>	- 26 -
2.7.2.6	<i>El mantenimiento y Refinamiento del Conocimiento.</i>	- 26 -
2.7.3	Los Modelos de CommonKADS	- 26 -
2.7.3.1	<i>Modelo de la Organización.</i>	- 27 -
2.7.3.2	<i>Modelo de Tareas</i>	- 30 -

2.7.3.3	<i>Modelo de Agentes</i>	- 32 -
2.7.3.4	<i>Modelo de Conocimientos</i>	- 35 -
2.7.3.5	<i>Modelo de Comunicación</i>	- 36 -
2.7.3.6	<i>Modelo de Diseño</i>	- 39 -
2.7.4	<i>Integración de los Modelos</i>	- 41 -
2.7.5	<i>Ventajas y Desventajas de CommonKADS</i>	- 43 -
2.8	Metodología Buchanan	- 45 -
2.8.1	<i>Perspectiva Histórica</i>	- 45 -
2.8.2	<i>Etapas</i>	- 45 -
2.8.2.1	<i>Identificación</i>	- 45 -
2.8.2.2	<i>Conceptualización</i>	- 46 -
2.8.2.3	<i>Elicitación (Formalización)</i>	- 46 -
2.8.2.4	<i>Implementación</i>	- 46 -
2.8.2.5	<i>Prueba</i>	- 46 -
	CAPITULO III	- 48 -
3.1	Introducción	- 48 -
3.2	Determinación de los Indicadores para la Comparación de Productividad	- 48 -
3.2.1	<i>Disponibilidad de Información</i>	- 50 -
3.2.2	<i>Líneas de Código</i>	- 50 -
3.2.3	<i>Tiempo de Desarrollo</i>	- 51 -
3.2.4	<i>Esfuerzo de Desarrollo</i>	- 52 -
3.2.5	<i>Costo</i>	- 52 -
3.3	Metodologías a evaluar	- 53 -
3.4	Desarrollo de Prototipos	- 53 -
3.4.1	<i>Aplicación Metodología CommonKADS</i>	- 53 -
3.4.1.1	<i>Modelo de la Organización</i>	- 53 -
3.4.1.2	<i>Modelo de Tareas</i>	- 57 -
3.4.1.3	<i>Modelo de Agentes</i>	- 58 -
3.4.1.4	<i>Modelo de Conocimiento</i>	- 60 -
3.4.1.5	<i>Modelo de Comunicación</i>	- 64 -
3.4.1.6	<i>Modelo de Diseño</i>	- 66 -
3.4.2	<i>Aplicación Metodología Buchanan</i>	- 68 -
3.4.2.1	<i>Identificación</i>	- 68 -
3.4.2.2	<i>Conceptualización</i>	- 68 -
3.4.2.3	<i>Formalización</i>	- 69 -

3.4.2.4	<i>Implementación</i>	- 71 -
3.5	Análisis comparativo	- 74 -
3.5.1	<i>Disponibilidad de Información</i>	- 74 -
3.5.2	<i>Líneas de Código</i>	- 77 -
3.5.3	<i>Tiempo de Desarrollo</i>	- 78 -
3.5.4	<i>Esfuerzo de Desarrollo</i>	- 79 -
3.5.5	<i>Costo</i>	- 81 -
3.6	Resultado de la comparación de las Metodologías CommonKADS vs. Buchanan ..	82 -
3.7	Comprobación de la Hipótesis	- 83 -
CAPITULO IV		- 86 -
IMPLEMENTACIÓN DEL SISTEMA EXPERTO		- 86 -
4.1	Junta General de Usuarios de Riego Chambo-Guano JURECH	- 86 -
4.2	Aplicación de CommonKADS para el desarrollo de SYS-Cultivos	- 87 -
4.2.1	<i>Modelo de la Organización</i>	- 87 -
4.2.1.1	<i>Formulario OM - 1</i>	- 87 -
4.2.1.2	<i>Formulario OM - 2</i>	- 88 -
4.2.1.3	<i>Formulario OM - 3</i>	- 90 -
4.2.1.4	<i>Formulario OM - 4</i>	- 91 -
4.2.1.5	<i>Formulario OM - 5</i>	- 92 -
4.2.2	<i>Modelo de Tareas</i>	- 93 -
4.2.2.1	<i>Formulario TM - 1</i>	- 93 -
4.2.2.2	<i>Formulario TM – 2</i>	- 95 -
4.2.3	<i>Modelo de Agentes</i>	- 96 -
4.2.3.1	<i>Formulario AM-1:</i>	- 96 -
4.2.4	<i>Modelo de Conocimiento</i>	- 97 -
4.2.4.1	<i>CM1 (Conocimiento del Dominio)</i>	- 98 -
4.2.4.2	<i>CM2 (Especificación del Esquema del Dominio)</i>	- 98 -
4.2.4.3	<i>CM3 (Base de Conocimiento)</i>	- 103 -
4.2.5	<i>Modelo de Comunicación</i>	- 106 -
4.2.5.1	<i>Modelo de Comunicación. CM-1</i>	- 107 -
4.2.6	<i>Modelo de Diseño</i>	- 107 -
4.2.6.1	<i>Arquitectura de Sistemas. DM-1</i>	- 107 -
4.2.6.2	<i>Especificación de la Plataforma DM2</i>	- 108 -
4.3	Implementación	- 110 -

CONCLUSIONES	- 114 -
RECOMENDACIONES.....	- 116 -
GLOSARIO DE TÉRMINOS	- 117 -
BIBLIOGRAFÍA	- 118 -
ANEXOS.....	- 120 -

ÍNDICE DE TABLAS

Tabla II. I. Evolución de SE	- 10 -
Tabla II. II. Áreas de aplicación	- 12 -
Tabla III. III. Indicadores de evaluación de productividad	- 48 -
Tabla III. IV. Valoración y Porcentajes.....	- 49 -
Tabla III. V. Criterios de evaluación de disponibilidad de información	- 50 -
Tabla III. VI. Criterios de evaluación de líneas de código.....	- 51 -
Tabla III. VII. Criterios de evaluación de tiempo de desarrollo	- 51 -
Tabla III. VIII. Criterios de evaluación de Esfuerzo de desarrollo.	- 52 -
Tabla III. IX. Criterios de evaluación del costo.....	- 52 -
Tabla III. X. Aspectos variables	- 54 -
Tabla III. XI. Atención al Usuario.....	- 56 -
Tabla III. XII. Análisis de las tareas - Sugerencia para cultivo	- 57 -
Tabla III. XIII. Análisis de las tareas-obtener información del Usuario	- 58 -
Tabla III. XIV. Modelo de agentes- Usuario	- 59 -
Tabla III. XV. Modelo de agentes- Agrónomo.....	- 59 -
Tabla III. XVI. Modelo de comunicación	- 65 -
Tabla III. XVII. modelo de diseño	- 66 -
Tabla III. XVIII. Especificación de la plataforma de implantación	- 66 -
Tabla III. XIX. Caso de Uso Registrar	- 71 -
Tabla III. XX. Caso de Uso Consultar Condición	- 72 -
Tabla III. XXI. Caso de Uso Editar	- 73 -
Tabla III. XXII. Caso de Uso Obtener Sugerencia	- 73 -
Tabla III. XXIII. Evaluación del indicador disponibilidad de información	- 75 -
Tabla III. XXIV. Evaluación del indicador líneas de código.....	- 77 -
Tabla III. XXV. Evaluación del indicador tiempo de desarrollo	- 78 -
Tabla III. XXVI. Evaluación del Esfuerzo de desarrollo	- 80 -
Tabla III. XXVII. Evaluación del indicador costo.....	- 81 -
Tabla III. XXVIII. Resultado de la comparación de las metodologías.	- 82 -
Tabla IV. XXIX. Aspectos de variantes	- 88 -
Tabla IV. XXX. Atención al Usuario	- 90 -
Tabla VI. XXXI. Tarjetas sobre Recursos de Conocimiento	- 91 -
Tabla VI. XXXII. Elementos Documento Viabilidad.....	- 92 -
Tabla IV. XXXIII. Análisis de las tareas-obtener información del paciente	- 93 -
Tabla IV. XXXIV. Análisis de las tareas - Sugerencia para cultivo	- 94 -
Tabla IV XXXV. Elemento de conocimiento- Ítem de Conocimiento del Ing. Agrónomo ...	- 95 -
Tabla IV. XXXVI: Ítem de Conocimiento del Ing. Agrónomo	- 95 -
Tabla IV. XXXVII. Modelo de Agentes- Usuario.....	- 96 -
Tabla IV. XXXVIII. Modelo de agentes- Agrónomo.....	- 97 -
Tabla IV. XXXIX: Regla 1	- 104 -
Tabla IV. XL. Regla 2.....	- 104 -
Tabla IV. XLI. Regla 3	- 105 -
Tabla IV. XLII. Regla 4	- 105 -
Tabla IV. XLIII. Regla 5	- 105 -
Tabla IV. XLIV. Modelo de comunicación.....	- 107 -
Tabla IV. XLV. Modelo de diseño	- 107 -

Tabla IV. XLVI. Especificación de la plataforma de implantación	- 108 -
Tabla IV. XLVII. Conclusión.....	- 111 -
Tabla IV. XLVIII. Hechos	- 112 -
Tabla IV. XL IX. Relaciones.....	- 112 -

ÍNDICE DE FIGURAS

Figura I. 1. Sistemas Basados en conocimiento.....	- 7 -
Figura II. 2. Sistema Experto.....	- 8 -
Figura II. 3. Arquitectura de un sistema experto genérico	- 14 -
Figura II. 4. Base de conocimientos del detective	- 16 -
Figura II. 5. Modelos de CommonKADS.....	- 27 -
Figura II. 6. Modelo de la organización de CommonKADS	- 29 -
Figura II. 7. Modelo de tareas de CommonKADS	- 31 -
Figura II. 8. Modelo de agentes de CommonKADS.....	- 34 -
Figura II. 9. Jerarquía del modelo de conocimientos de CommonKADS	- 36 -
Figura II. 10. Modelo de comunicaciones de CommonKADS	- 38 -
Figura II. 11. Pasos del diseño del sistema	- 39 -
Figura II. 12. Documentación del ciclo de la gestión del proyecto en CommonKADS	- 43 -
Figura II. 13. Ciclo de vida buchanan	- 45 -
Figura II. 14. Red completamente comunicada	- 47 -
Figura III. 15. Estructura de la organización	- 55 -
Figura III. 16. Proceso – Proceso de atención al Usuario	- 56 -
Figura III. 17. Diagrama de conceptos	- 60 -
Figura III. 18. Relación de conceptos.....	- 63 -
Figura III. 19. Esquema de nivel del dominio	- 64 -
Figura III. 20. Modelo de comunicación	- 65 -
Figura III. 21. Diagrama de proceso del sistema experto	- 67 -
Figura III. 22. Resultado del indicador disponibilidad de información	- 76 -
Figura III. 23. Resultado del indicador Líneas de código	- 77 -
Figura III. 24. Resultado del indicador tiempo de desarrollo.....	- 79 -
Figura III. 25. Resultado del indicador Esfuerzo de desarrollo	- 80 -
Figura III. 26. Resultado del indicador Costo.....	- 81 -
Figura III. 27. Resultado Total comparación de las metodologías.....	- 83 -
Figura IV. 28. Estructura de la organización	- 89 -
Figura IV. 29. Proceso – Proceso de atención al Usuario	- 90 -
Figura IV. 30. Diagrama de conceptos	- 98 -
Figura IV. 31. Relación de conceptos.....	- 101 -
Figura IV. 32. Esquema de nivel del dominio	- 102 -
Figura IV. 33. Base de conocimiento	- 103 -
Figura IV. 34. Modelo de comunicación	- 106 -
Figura IV. 35. Diagrama de proceso del sistema experto	- 108 -
Figura IV. 36. Pantalla Principal	- 109 -
Figura IV. 37. Pantalla Preguntas	- 109 -
Figura IV. 37. Pantalla Resultado.....	- 110 -
Figura IV. 38. Base de Conocimientos.....	- 111 -

RESUMEN

Análisis comparativo de la productividad de metodologías CommonKADS vs Buchanan para el desarrollo de un sistema experto de gestión de cultivos para la Jurech en el cantón Riobamba, análisis que permitió determinar a CommonKADS como la mejor metodología para el desarrollo del sistema experto misma que proporciona herramientas más adecuadas para sistemas basados en el conocimiento mejorando la productividad en el desarrollo a lo largo de su ciclo de vida con sus etapas Análisis, Diseño, Implantación del sistema, Uso, Mantenimiento y refinamiento del conocimiento, posee modelos que facilitan la adquisición del conocimiento brindado por el experto humano así como la correcta formalización del mismo. Los indicadores que se utilizaron para medir la productividad fueron Disponibilidad de Información, Líneas de código, Tiempo de desarrollo, Esfuerzo de desarrollo y Costo. Para la comparación se desarrollaron dos prototipos los cuales permitieron medir los indicadores mientras que en el desarrollo del sistema SysCultivos las tecnologías utilizadas el gestor de base de datos MySQL y .Net Visual Basic 2012. Los valores obtenidos para Disponibilidad de información son 20% CommonKADS vs 5% Buchanan, Líneas de código 20% CommonKADS vs 10% Buchanan, Tiempo de desarrollo 10% CommonKADS vs 20% Buchanan, Esfuerzo de desarrollo 20% CommonKADS vs 10% Buchanan, Costo 10% CommonKADS vs 20% Buchanan. Se concluye que la metodología CommonKADS provee mejores características en cuanto productividad con un 80% de cumplimiento de los parámetros establecidos ante el 65% de la metodología Buchanan. Se recomienda a los usuarios de la junta de riego Chambo – Guano una vez implementado el Sistema Experto sea utilizado como apoyo y referencia en la selección de cultivos a ser sembrados.

Palabras Claves:

<SISTEMAS INFORMATICOS >, <SISTEMA EXPERTO>, <SISTEMA [SYSCULTIVOS]>, < JUNTA DE RIEGO CHAMBO - GUANO [JURECH]>, < METODOLOGIA COMMONKADS >, < METODOLOGIAS BUCHANAN>, <RIOBAMBA (CANTON)>, <BASES DE DATOS MYSQL>, <VISUAL BASIC 2012>

SUMMARY

Comparative analysis of the productive of CommonKADS vs Buchanan methodologies for the development of an expert system for crop management Jurech in the canton Riobamba, analysis which identified a CommonKADS as best methodology for the development of the expert system itself that provides more adequate tools for knowledge-based improving development productivity throughout their life cycle stages analysis, design, implementation systems system, use, maintenance and refinement of knowledge it has models that facilitate the acquisition provided by the human expert and the correct formation of the same knowledge. The indicators used to measure productivity were availability of information, lines of code, development time, development effort and cost. For comparison two prototypes which can measure the indicators while in the development of technologies used Syscultivos system manager MySQL database and Visual Basic Net 2012. The values obtained for availability of information developed are 20% CommonKADS vs 5% Buchanan, lines Code 20% CommonKADS vs 10% Buchanan, development time 10% CommonKADS vs 20% Buchanan, development effort 20% CommonKADS vs 20% Buchanan. CommonKADS concludes that the methodology provides best characteristics in terms of productivity with a 80% compliance with the parameters set before the 65% methodology Buchanan. It is recommends users of the board of irrigation Chambo - Guano once implemented expert system is used as support and reference in the selection of crops to be planted.

Keywords:

<INFORMATICS SYSTEM>, <EXPERT SYSTEM>, SYSTEM [SYSCULTIVOS], <IRRIGATION BOARD>, <METHODOLOGY COMMONKADS>, <METHODOLOGY BUCHANAN>, <MYSQL DATABASES>, <VISUAL BASIC 2012>

CAPITULO I

MARCO REFERENCIAL

1.1 Antecedentes

Bajo el término de sistemas expertos se encuentran un nuevo tipo de software que imita el comportamiento de un experto humano en la solución de un problema con los conocimientos que el experto humano aplicaría para la solución del mismo.

La elección de una metodología que cumpla las expectativas, requerimientos y que provea de soluciones eficientes a la hora del desarrollo del sistema es un punto de suma importancia que se debe tomar en cuenta en el desarrollo de sistemas expertos.

En la actualidad existen varias metodologías para el desarrollo de sistemas expertos brindando cada una de ellas similares ventajas en ciertos aspectos, dentro de las más usadas se encuentran **COMMONKADS** y **BUCHANAN**.

Metodología es un vocablo generado a partir de tres palabras de origen griego: *metà* (“**más allá**”), *odòs* (“**camino**”) y *logos* (“**estudio**”). El concepto hace referencia al **plan de** investigación que permite cumplir ciertos objetivos en el marco de una **ciencia**.

CommonKADS es una metodología diseñada para el análisis y la construcción de sistemas basados en conocimiento (SBC) de forma análoga a los métodos empleados en ingeniería de software. Fue propuesta y desarrollada por un grupo de investigadores pertenecientes a diversos países de la comunidad Europea, a través del programa ESPRIT para la innovación y la aplicación de Tecnología Informática Avanzada.

Esta metodología a sido usada en la Universidad Católica Santo Toribio de Mogrovejo de Peru para la elaboración de la tesis “Desarrollo de un sistema experto sobre web para un diagnóstico temprano de cáncer de cuello uterino en la Clínica Maternidad Belén, Chiclayo”

Buchanan es uno de los primeros métodos de desarrollo estructurado de sistemas inteligentes fue el propuesto por Buchanan y otros autores en 1983, Buchanan inicialmente pensó su metodología como una metodología de adquisición del conocimiento, e hizo hincapié en las fases que describen

el proceso que sufre la información, desde que fluye del experto, hasta que es finalmente implementada en el sistema. Posteriormente este proceso se tomó como la construcción del sistema inteligente completo.

La incorrecta selección de un cultivo por parte de los productores ocasiona que se genere escases de determinados productos y sobreproducción de otros.

El desarrollo de un sistema experto para la determinación adecuada de los cultivos permitirá resolver este inconveniente antes expuesto, el sistema proporcionará información para tomar la mejor decisión sobre que cultivo es el más idóneo para su siembra.

La Junta General de Usuarios de Riego Chambo-Guano brinda a sus usuarios servicios completos de riego, tecnificación, transferencia de conocimientos, tecnologías nuevas, interrelación entre productores con mercados nacionales y extranjeros, mirando a sus usuarios como agricultores micro empresarios, tiene sus oficinas en la calle Chile y Puruhá.

1.2 Delimitación

El sistema experto de la investigación que se realizará se implementará en la Junta General de Usuarios de Riego Chambo-Guano, con esta investigación se pretende mejorar la gestión de cultivos de los usuarios de esta Junta.

1.3 Formulación del Problema

¿Cuál de las metodologías es la más productiva para el desarrollo del Sistema experto para la gestión de cultivos de la JURECH?

1.3.1 Sistematización del Problema

- ✓ ¿Cuál es el ciclo de vida de las metodologías COMMONKADS VS BUCHANAN?
- ✓ ¿Cuáles serían las mejores herramientas y parámetros para la comparación de metodologías?
- ✓ ¿Cuál es el resultado obtenido por parte del análisis y tabulación de datos comparativos de las metodologías en estudio?
- ✓ ¿Cómo mejorará la utilización de las metodologías para el desarrollo del Sistema Experto?

1.4 Justificación del Proyecto de Tesis

1.4.1 Justificación Teórica

COMMONKADS

Una de las principales cualidades de CommonKADS es el planteamiento del desarrollo de modelos que reflejan diferentes vistas del proyecto.

CommonKADS es importante porque ofrece un marco para la especificación del conocimiento independiente de la implementación, combinando un conjunto de modelos de conocimiento reutilizable para unas tareas que se realizan frecuentemente.

Como por ejemplo el diagnóstico o la planificación entre otras. Además, propone un ciclo de vida en donde se indican las fases, las actividades y los productos más relevantes para un proyecto de desarrollo de un SBC.

BUCHANAN

Buchanan es de las metodologías más importantes para el diseño de una Base de Conocimiento. Este método además de definir una serie de etapas, propone con énfasis una documentación de los procesos: los cuales reemplazan parcialmente al experto y sirven de medio de comunicación y referencia entre los usuarios y los diseñadores.

La característica más importante de esta metodología es la constante relación ente el Ingeniero de conocimiento y el experto del área.

1.4.2 Justificación Metodológica

Para el análisis comparativo de la productividad de metodologías se realizará una comparación basada en parámetros que permitan evaluar la productividad que brinda cada una de las mismas.

1.4.3 Justificación Práctica

El estudio de las Metodologías COMMONKADS vs BUCHANAN nos ayudará a elegir una de ellas para poder aplicarla en el desarrollo del sistema experto de gestión de cultivos de la Junta General de Riego Chambo Guano JURECH analizando cada requerimiento que este necesite de acuerdo a las investigaciones que se desarrollen los usuarios de la junta contarán con un sistema experto para mejorar la gestión de los cultivos y el riego de los mismos así también ayudará a mejorar la producción.

Entre sus funciones la aplicación permitirá asesorar a los usuarios de la junta para escoger el cultivo más idóneo a sembrar, tomando en consideración factores como el ciclo del cultivo, la inversión necesaria, disponibilidad de riego para determinada época, disponibilidad de invernadero, reservorio, tal como lo haría un experto humano.

Para ello se desarrollará dos prototipos uno con la metodología COMMONKADS y otro prototipo con BUCHANAN para determinar cuál brinda mayor productividad.

La metodología que brinde mayor productividad será con la que se desarrolle el sistema experto para la JURECH

1.4.4 Alineación del Tema

Líneas y programa de la Epoch

Tecnología de información comunicación y proceso industriales

Programa para el desarrollo de aplicaciones de software, hardware y telecomunicaciones para planificación territorial.

PNBV

Mejorar la calidad de vida de la población

Política 2.7. Promover el acceso a la información y a las nuevas tecnologías de la información y comunicación para incorporar a la población a la sociedad de la información y fortalecer el ejercicio de la ciudadanía.

1.5 Objetivos

1.5.1 Objetivo General

Analizar la productividad de metodologías COMMONKADS vs BUCHANAN para el desarrollo de un sistema experto de gestión de cultivos para la JURECH.

1.5.2 Objetivos Específicos

-  Investigar y analizar las características de las metodologías COMMONKADS vs BUCHANAN.
-  Definir parámetros de comparación de metodologías
-  Desarrollar prototipos para medir productividad
-  Analizar los resultados obtenidos y seleccionar la metodología más óptima al momento del desarrollo de un Sistema Experto.

- ✚ Implementar un Sistema Experto de Gestión de Cultivos Junta General de Riego Chambo Guano JURECH. (Riobamba – Chimborazo) aplicando la metodología seleccionada.

1.6 Hipótesis

La metodología **COMMONKADS** presenta una mayor productividad frente a la metodología **BUCHANAN** en el desarrollo de un Sistema Experto.

CAPITULO II

MARCO TEÓRICO

2.1 Introducción

Este capítulo abarcará temas relacionados con Inteligencia artificial y sistemas expertos, así como lo referente a su área de aplicación, evolución, arquitectura y metodologías para su desarrollo.

Este capítulo permitirá familiarizarse con los términos manejados en el ámbito de los sistemas expertos, lo cual será la base para un mejor entendimiento de los términos empleados en los capítulos posteriores.

2.2 Inteligencia Artificial.

Dada la indefinición del propio concepto de "inteligencia", prácticamente existe una definición de Inteligencia Artificial por cada autor que escribe sobre el tema. Tal vez una de las definiciones que se puede considerar más ajustada a la realidad es la reflejada en la Encyclopedia Of Artificial Intelligence:

"La IA es un campo de la ciencia y la ingeniería que se ocupa de la comprensión, desde el punto de vista informático, de lo que denomina comúnmente comportamiento inteligente. También se ocupa de la creación de artefactos que exhiben este comportamiento". (Pino , Gómez , & Martínez , 2001)

En cualquier caso, desde distintas perspectivas, en el seno de la IA como ciencia y tecnología se han ido acumulando conocimientos sobre cómo emular las diversas capacidades del ser humano para exhibir comportamientos inteligentes y se han desarrollado sistemas cada vez más perfeccionados que reproducen parcialmente dichas capacidades. Así, el estudio de sensores y mecanismos de intercambio de información con el exterior constituyen, por si mismos, pujantes áreas de investigación y aplicaciones prácticas. También la robótica, como especialidad relacionada con máquinas móviles capaces de interactuar inteligentemente con su entorno y manipular objetos, suele considerarse parcialmente incluida en el ámbito de la IA. Y naturalmente el conjunto da habilidades cognitivas e intelectivas realizadas por el cerebro humano para comprender el mundo que rodea a una persona, interpretar sus señales e informaciones, resolver los constantes problemas de diverso tipo que plantea el quehacer diario, representar y acumular

experiencia y conocimiento, aprender y generar nuevo conocimiento a partir del bagaje disponible mediante diversos modos de razonamiento. etc. (Pino , Gómez , & Martínez , 2001)

Estos componentes del comportamiento inteligente son clasificados por la Encyclopedia Of Artificial Intelligence, en las siguientes disciplinas como subáreas de la Inteligencia Artificial:

Procesamiento de lenguaje natural

Visión artificial

Resolución de problemas

Representación del conocimiento y razonamiento

Aprendizaje

Robótica

La ingeniería asociada a los avances científicos de la IA construye sistemas o máquinas que realizan una o varias de las capacidades estudiadas (especialmente en robótica, que utiliza aspectos de todas ellas), obteniéndose aplicaciones prácticas que pueden ser utilizadas para recibir ayuda en alguna tarea que requiera capacidad intelectual, como es el caso de los sistemas de traducción automática o de reconocimiento de voz disponibles ya en el mercado. Pero, sin ninguna duda, son los Sistemas Expertos (SE), y más genéricamente los Sistemas Basados en el Conocimiento (SBC), el tipo de aplicaciones prácticas de la IA que tienen más difusión e impacto en el mundo empresarial. Estos sistemas, que se comentan ampliamente más adelante, han dado lugar a aplicaciones que han permitido a la IA salir de los laboratorios y hacer sentir su presencia en las plantas industriales y en las oficinas de negocios. Véase Figura I. 1

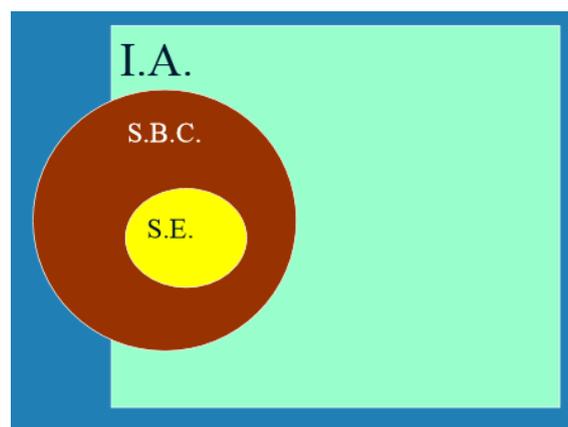


Figura I. 1. Sistemas Basados en conocimiento

Fuente: http://webdelprofesor.ula.ve/economia/guillenr/inteligencia/sist_expert_2.pdf

2.3 Sistema Experto.

Se considera a alguien un experto en un problema cuando esté individuo tiene conocimiento especializado sobre dicho problema. En el área de los (SE) a este tipo de conocimiento se le llama conocimiento sobre el dominio. La palabra dominio se usa para enfatizar que el conocimiento pertenece a un problema específico. (Hernando, 2004)

Antes de la aparición del ordenador, el hombre ya se preguntaba si se le arrebataría el privilegio de razonar y pensar. En la actualidad existe un campo dentro de la inteligencia artificial al que se le atribuye esa facultad: el de los sistemas expertos (SE). Estos sistemas también son conocidos como Sistemas Basados en Conocimiento, los cuales permiten la creación de máquinas que razonan como el hombre, restringiéndose a un espacio de conocimientos limitado. En teoría pueden razonar siguiendo los pasos que seguiría un experto humano (médico, analista, empresario, etc.) para resolver un problema concreto. Este tipo de modelos de conocimiento por ordenador ofrece un extenso campo de posibilidades en resolución de problemas y en aprendizaje. Su uso se extenderá ampliamente en el futuro, debido a su importante impacto sobre los negocios y la industria. (Samper Márquez, 2004)

Un sistema experto es aquel software que emulan el razonamiento de un experto en un dominio concreto, y este nos permite obtener una mejor calidad y rapidez en las respuestas dando así lugar a una mejora de la productividad del experto o la persona que lo utilice, cuenta con un motor de inferencia que actúa sobre una base de conocimiento y una base de hechos como lo muestra la Figura II. 2 (Pino , Gómez , & Martínez , 2001)

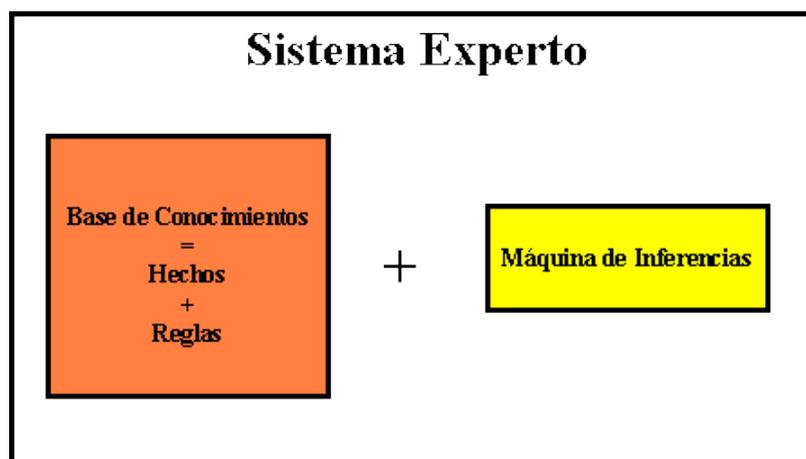


Figura II. 2. Sistema Experto

Fuente: <http://ciberconta.unizar.es/leccion/introduc/483.HTM>

- Según los autores Raúl Pino Diez, Alberto Gómez Gómez y Nicolás de abajo Martínez un Sistema Experto se puede definir como aquel programa de ordenador que contiene la erudición de un especialista humano versado en un determinado campo de aplicación en este sentido los expertos escasean y su contratación supone una gran inversión económica, por lo que se intenta construir un sistema de forma que los conocimientos del experto se representen en una forma que el ordenador pueda procesar. Esto es, un modelo computarizado de las capacidades de razonamiento y habilidades de resolución de problemas del especialista humano.

Según el Autor Lic. Jesús Montes Castro es un software que imita el comportamiento de un experto humano en la solución de un problema. Pueden almacenar conocimientos de expertos para un campo determinado y solucionar un problema mediante deducción lógica de conclusiones.

Son SE aquellos programas que se realizan haciendo explicito el conocimiento en ellos, que tienen información específica de un dominio concreto y que realizan una tarea relativa a este dominio.

Programas que manipulan conocimiento codificado para resolver problemas en un dominio especializado en un dominio que generalmente requiere de experiencia humana. Programas que contienen tanto conocimiento declarativo (hechos a cerca de objetos, eventos y/o situaciones) como conocimiento de control (información a cerca de los cursos de una acción), para emular el proceso de razonamiento de los expertos humanos en un dominio en particular y/o área de experiencia. (Montes Castro, 2004)

Software que incorpora conocimiento de experto sobre un dominio de aplicación dado, de manera que es capaz de resolver problemas de dificultad relativa y apoyar en la toma de decisiones inteligentes en base a un proceso de razonamiento simbólico.

- El tesista considera que los autores Raúl Pino Diez, Alberto Gómez Gómez, Nicolás de abajo Martínez y Lic. Jesús Montes Castro
- Coinciden en que es un sistema que puede almacenar conocimientos de expertos para un campo determinado
- Difieren en que los SE son aquellos programas que se realizan haciendo explicito el conocimiento
- En el presente trabajo un sistema experto se considera según el concepto del autor: Lic. Jesús Montes Castro, Es un software que imita el comportamiento de un experto humano en la

solución de un problema. Pueden almacenar conocimientos de expertos para un campo determinado y solucionar un problema mediante deducción lógica de conclusiones.

2.3.1. Evolución

“A pesar de la evolución de los sistemas expertos y su gran labor en la solución de problemas facilitando la vida del hombre, su inteligencia seguirá siendo artificial, porque no se ha podido construir un sistema experto que tenga las mismas sensaciones, emociones que un ser humano, todos estas estructuras se consideran brutas, porque dependen de la inteligencia humana para poder funcionar”. (Velasco Ortega, 2014)

2.3.1.2 Evolución de los Sistemas Expertos en el Tiempo

Uno de los primeros sistemas expertos se llamó Dendral y era capaz de calcular o descubrir hechos relativos a la estructura molecular a partir de unos datos químicos sin elaborar. Otros sistemas expertos famosos son MYCIN que diagnostica enfermedades de la sangre y que sugiere un tratamiento y PUFF, un sistema similar pero para enfermedades de pulmón. (Biblionetica, 2014) En la siguiente tabla podemos ver los sistemas expertos más representativos.

Tabla II. I. Evolución de SE

Sistema	Fecha	Autor	Aplicación
DENDRAL	1965	Stanford	Deduce información sobre estructuras químicas
Macsyma	1965	MIT	Análisis matemático complejo
HearSay	1965	Carnegie - Mellon	Interpreta en lenguaje natural un subconjunto del idioma
Mycin	1972	Stanford	Diagnóstico de enfermedades de la sangre
Tieresias	1972	Stanford	Herramienta para la transformación de conocimientos
Prospector	1972	Stanford	Exploración mineral y herramientas de identificación
Age	1973	Stanford	Herramienta para generar Sistemas Expertos
OP55	1974	Carnegie – Mellon	Herramientas para desarrollo de Sistemas Expertos
Caduceus	1975	University of Pittsburg	Herramienta de diagnóstico para medicina interna
Rosie	1978	Rand	Herramienta de desarrollo de Sistemas Expertos
R1	1978	Carnegie - Mellon	Configurador de equipos d computación para DEC

Fuente: <http://simulacionymodelos-unefapc.blogspot.com/p/sistemas-expertos.html>

2.3.2 Tipos de Sistemas Expertos.

2.3.2.1 Basados en Reglas.

Trabajan mediante la aplicación de reglas, comparación de resultados y aplicación de las nuevas reglas basadas en situación modificada. También pueden trabajar por inferencia lógica dirigida, bien empezando con una evidencia inicial en una determinada situación y dirigiéndose hacia la obtención de una solución, o bien con hipótesis sobre las posibles soluciones y volviendo hacia atrás para encontrar una evidencia existente (o una deducción de una evidencia existente) que apoye una hipótesis en particular. (Biblionetica, 2014)

2.3.2.2 Basados en Casos (CBR or Case Based Reasoning).

Aplicando el razonamiento basado en casos, donde la solución a un problema similar planteado con anterioridad se adapta al nuevo problema. Es una manera razonar haciendo analogías. Se ha argumentado que más que un método para el razonamiento de computadoras (computer reasoning) es usado por las personas para solucionar problemas cotidianos. (Biblionetica, 2014)

El razonamiento basado en casos se formaliza en cuatro pasos con el propósito de ser utilizado en el razonamiento de computadora:

1. Recordar: Dado un determinado problema, recordar los casos relevantes que pueden solucionarlo.
2. Reutilizar: Adaptar la solución del problema anterior a este nuevo.
3. Revisar: Una vez adaptado el problema probar la solución en el mundo real o en una simulación y si es necesario revisarla.
4. Retener: Después de que la solución ha sido adaptada satisfactoriamente para solucionar el problema dado, almacenar la experiencia resultante como un nuevo caso en la memoria.

2.3.2.3 Basados en Redes Bayesianas.

Es un modelo probabilístico multivariado que relaciona un conjunto de variables aleatorias mediante un grafo dirigido, el cual indica explícitamente influencia causal. Gracias a su motor de actualización de probabilidades, el Teorema de Bayes, las redes bayesianas son una herramienta extremadamente útil en la estimación de probabilidades ante nuevas evidencias.

El Teorema de Bayes es el resultado que da la distribución de probabilidad condicional de una variable aleatoria A dada B en términos de la distribución de probabilidad condicional de la

variable B dada A y la distribución de probabilidad marginal de sólo A. (Naïm, Wuillemin, Leray, Pourret, & Becker, 2004)

2.3.3 Áreas de Aplicación

Los *SE* se aplican a una gran diversidad de campos y/o áreas. A continuación se listan algunas de las principales:

Tabla II. II. Áreas de aplicación

Militar	Informática	Telecomunicaciones
Química	Derecho	Aeronáutica
Geología	Arqueología	Agricultura
Electrónica	Transporte	Educación
Medicina	Industria	Finanzas y Gestión

Realizado por: Fausto Llangarí

2.3.4 Experto Humano

Un experto humano es una persona competente en un área determinada del conocimiento o del saber. Un experto humano es alguien que sabe mucho sobre un tema determinado y que puede dar un consejo adecuado. Esta experiencia sólo se adquiere tras un largo aprendizaje y a base de mucha experiencia en el tema.

2.3.4.1 Atributos de un Experto Humano

Se acepta a un experto humano como tal por una serie de atributos que caracteriza su labor, en la que se le reconoce una verdadera competencia, habilidad y experiencia. En general, el experto humano resuelve problemas complejos, actualiza sus conocimientos, es capaz de justificar sus decisiones respuestas, a las que puede asignar grados de credibilidad.

Un buen experto humano también es capaz de destacar los aspectos más relevantes de un problema en el momento que detecta y reconoce la desaparición de su competencia o pericia cuando el problema en cuestión se acerca a la fronteras del dominio de su experiencia, es decir, es consciente de los límites de su competencia. También es capaz de aprender a razonar, mejor reestructurando sus conocimientos, aplicando reglas o procedimientos para los casos de excepción, y elaborando nuevas reglas o procedimientos para enfrentarse a nuevos problemas.

Además dispone de sentido común para situar su forma de proceder dentro de un ámbito más general. (Barcelo Garcia, 2001)

2.3.5 Conocimiento

Es el conjunto de datos de primer orden, que modelan de forma estructurada la experiencia que se tiene sobre un cierto dominio o que surgen de interpretar los datos básicos. Incluye y requiere del uso de datos e información. Además, combina relaciones, dependencias, y la noción del saber con datos e información. (Rivas Echeverría, 2014)

2.3.6 Adquisición y Representación del Conocimiento.

2.3.6.1 Adquisición del Conocimiento

Es el proceso mediante el cual se captura toda la información necesaria para el desarrollo de la Base de Conocimiento del Sistema Experto.

Existen varias alternativas para acceder al saber necesario para la construcción delegase de Conocimiento. En caso de que la fuente de conocimiento sea un experto humano, se habla de un Sistema Experto.

2.3.6.2 Representación del Conocimiento

Decimos que los expertos humanos resuelven problemas haciendo uso de una gran cantidad de hechos y heurísticos específicos del dominio. A continuación se muestra cómo los ingenieros del conocimiento analizan el conocimiento de un experto y cómo representan ese conocimiento en el software que constituye un sistema experto. El estudio se centra en una de las partes constituyentes de los sistemas de conocimiento. La Figura II. 3 representa la arquitectura de un sistema experto genérico, que puede dividirse en dos partes: una está formada por la base de conocimientos y la memoria de trabajo, y la otra, por el motor de inferencias y los restantes subsistemas e interfaces. (Rivas Echeverría, 2014)

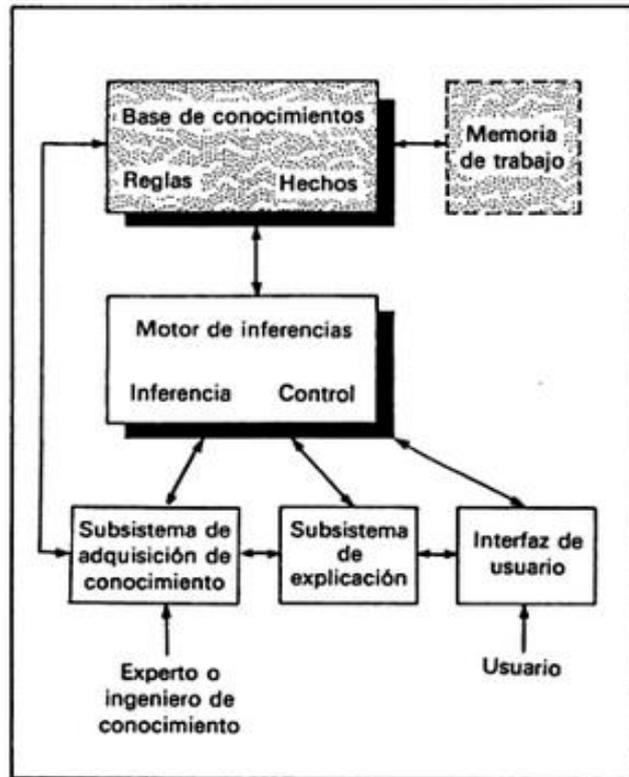


Figura II. 3. Arquitectura de un sistema experto genérico

Fuente: Sistemas expertos: aplicaciones de la inteligencia artificial en la actividad empresarial - David King, Paul Harmon - Google Libros

La base de conocimientos contiene los hechos y las reglas que resumen el conocimiento del experto. A continuación se presenta cómo se pueden representar hechos y reglas en una base de conocimientos. El motor de inferencias incluye las estrategias de inferencia y de control que utiliza un experto al manipular los hechos y las reglas. Los subsistemas y las interfaces que aparecen en la Figura II. 3 varían mucho de un sistema a otro. (Rivas Echeverría, 2014)

2.3.6.2.1 Estrategias para la Representación del Conocimiento

Se considera cinco formas diferentes para codificar los hechos y las relaciones que constituyen el conocimiento. Haciendo uso del conocimiento de Sherlock Holmes para ilustrar cada uno de los cinco enfoques, a saber:

- Redes semánticas
- Triplas objeto-atributo-valor

- Reglas
- Marcos
- Expresiones lógicas.

Cada uno de los métodos tiene ventajas e inconvenientes. En lo que sigue, se dará una panorámica de cada uno de ellos.

Redes Semánticas

El esquema de representación más general, y también uno de los más antiguos en IA, es la red semántica. Una red semántica es una colección de objetos llamados nodos, conectados por arcos o enlaces. Generalmente, tanto los nodos como los enlaces tienen etiquetas asociadas. La Figura II. 4 muestra una parte del conocimiento de Holmes representado como una red semántica. No hay restricciones absolutas sobre los nombres que pueden darse a los nodos y a los enlaces. Sin embargo, algunos convenios típicos son los siguientes:

1. Los nodos se utilizan para representar objetos y descriptores.
 - Los objetos pueden ser objetos físicos visibles o tangibles. Las chaquetas, sombreros y las personas de la base de conocimientos del detective son ejemplos de objetos físicos.
 - Los objetos pueden ser también entes conceptuales, como actos, acontecimientos o categorías abstractas. «China» y «2» son nodos conceptuales en la base de conocimientos del detective: los lugares y los números son más abstractos que los codos y las mangas.
 - Los descriptores proporcionan informaciones adicionales sobre los objetos. «Gastado» y «brillante», por ejemplo, dan informaciones sobre la chaqueta de Wilson.
2. Los enlaces relacionan objetos y descriptores. Un enlace puede representar cualquier relación. Entre los enlaces más comunes están:
 - Los enlaces es-un, que se utilizan a menudo para representar las relaciones clase/ejemplar. En nuestra base de conocimientos del detective, observamos, por ejemplo, que Holmes es-un hombre. Es decir, Holmes es un ejemplar («instance») de una clase, la de las personas.
 - Otra relación muy utilizada es tiene-un. Los enlaces tiene-un identifican nodos que son propiedades de otros nodos. En nuestra base de conocimientos, una chaqueta tiene-un

manga. A diferencia de la relación es-un, una manga no es un ejemplar de chaqueta. Los enlaces tiene-un corresponden a relaciones parte-subparte.

- Algunos enlaces son definitorios. En esta base de conocimientos, el enlace cubre entre mangas y brazos es un buen ejemplo de enlace definitorio, como también lo es el enlace viste entre personas y ropas.
- Otros enlaces reflejan conocimiento heurístico, como «el escribir causa que los puños estén brillantes». Las relaciones como éstas enriquecen la red, al proporcionar nuevos enlaces.

La gran ventaja de este esquema de representación es su flexibilidad. Conforme se van necesitando, pueden irse definiendo nuevos nodos y enlaces. Si, por ejemplo, estuviésemos representando un dispositivo eléctrico, cada nodo podría ser un componente que estaría conectado-con otros. Si la red describiera una jerarquía institucional, los enlaces más útiles entre personas podrían ser informa-a o supervisa-a. (King & Harmon, 1988)

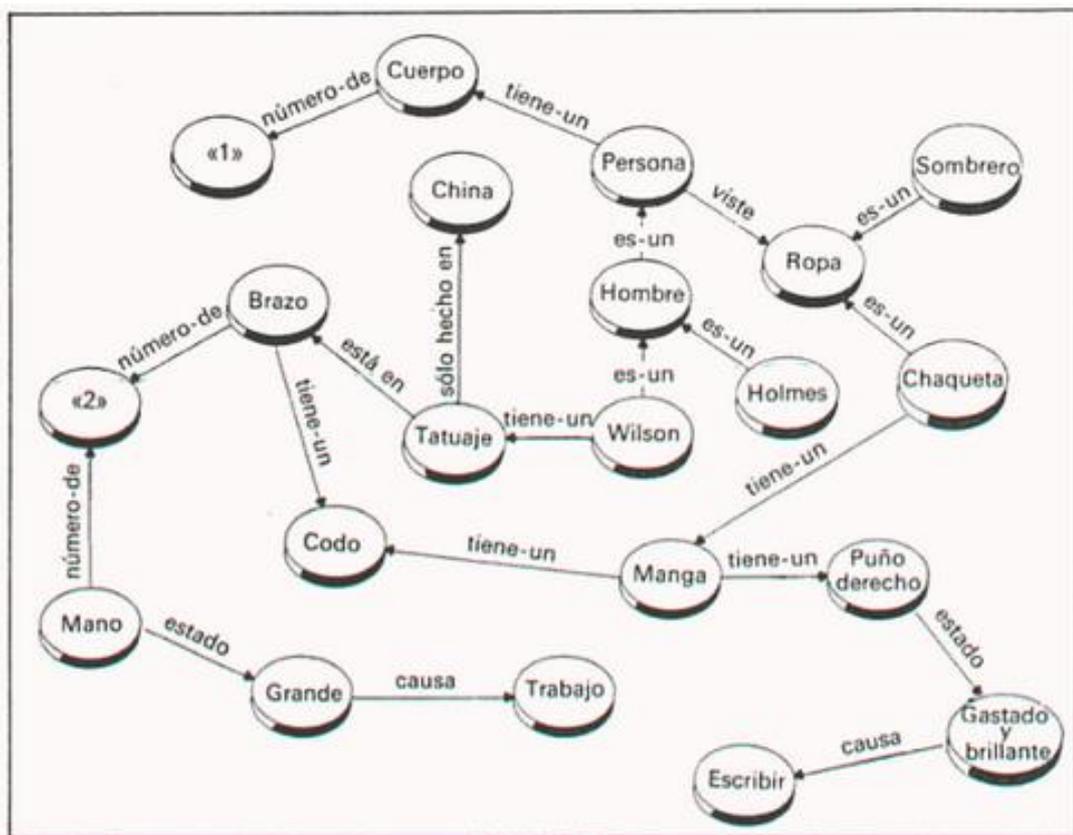


Figura II. 4. Base de conocimientos del detective

fuelle: Sistemas expertos: aplicaciones de la inteligencia artificial en la ... - David King, Paul Harmon - Google Libros

2.3.7 Arquitectura de un Sistema Experto

No existe una estructura de sistema experto común. Sin embargo, la mayoría de los sistemas expertos tienen unos componentes básicos: base de conocimientos, motor de inferencia, base de datos e interfaz con el usuario. Muchos tienen, además, un módulo de explicación y un módulo de adquisición del conocimiento. Véase Figura II. 3 (Samper Márquez, 2004)

2.3.8 Base de Conocimiento

Contiene el conocimiento especializado extraído del experto en el dominio. Es decir, contiene conocimiento general sobre el dominio en el que se trabaja. El método más común la representación del conocimiento es mediante reglas de producción.

Una característica muy importante es que la base de conocimientos es independiente del mecanismo de inferencia que se utiliza para resolver los problemas. De esta forma, cuando los conocimientos almacenados han quedado obsoletos, o cuando se dispone de nuevos conocimientos, es relativamente fácil añadir reglas nuevas, eliminar las antiguas o corregir errores en las existentes. No es necesario reprogramar todo el sistema experto.

Las reglas suelen almacenarse en alguna secuencia jerárquica lógica, pero esto no es estrictamente necesario. Se pueden tener en cualquier secuencia y el motor de inferencia las usará en el orden adecuado que necesite para resolver un problema. (Samper Márquez, 2004)

2.3.9 Base de Hechos

Es una parte de la memoria del ordenador que se utiliza para almacenar los datos recibidos inicialmente para la resolución de un problema. Contiene conocimiento sobre el caso concreto en que se trabaja. También se registrarán en ella las conclusiones intermedias y los datos generados en el proceso de inferencia. Al memorizar todos los resultados intermedios, conserva el vestigio de los razonamientos efectuados; por lo tanto, se puede utilizar para explicar las deducciones y el comportamiento del sistema. (Samper Márquez, 2004)

2.3.10 Base de Datos

Un conjunto de información almacenada en memoria auxiliar que permite acceso directo y un conjunto de programas que manipulan esos datos.

Base de Datos es un conjunto exhaustivo no redundante de datos estructurados organizados independientemente de su utilización y su implementación en máquina accesibles en tiempo real y compatibles con usuarios concurrentes con necesidad de información diferente y no predicable en tiempo. (Samper Márquez, 2004)

2.3.11 Motor de Inferencia

Es un programa que controla el proceso de razonamiento que seguirá el sistema experto. Utilizando los datos que se le suministran, recorre la base de conocimientos para alcanzar una solución. La estrategia de control puede ser de encadenamiento progresivo o de encadenamiento regresivo. En el primer caso se comienza con los hechos disponibles en la base de datos, y se buscan reglas que satisfagan esos datos, es decir, reglas que verifiquen la parte SI. (Samper Márquez, 2004)

2.3.12 Interfaz de Usuario

Permite que el usuario pueda describir el problema al sistema experto. Interpreta sus preguntas, los comandos y la información ofrecida. A la inversa, formula la información generada por el sistema incluyendo respuestas a las preguntas, explicaciones y justificaciones. Es decir, posibilita que la respuesta proporcionada por el sistema sea inteligible para el interesado. También puede solicitar más información si le es necesaria al sistema experto. En algunos sistemas se utilizan técnicas de tratamiento del lenguaje natural para mejorar la comunicación entre el usuario y el sistema experto. (Samper Márquez, 2004)

2.3.13 Subsistema de Explicación

Diseñado para aclarar al usuario la línea de razonamiento seguida en el proceso de inferencia. Si el usuario pregunta al sistema cómo ha alcanzado una conclusión, éste le presentará la secuencia completa de reglas usada. Esta posibilidad de explicación es especialmente valiosa cuando se tiene la necesidad de tomar decisiones importantes amparándose en el consejo del sistema experto. Además, de esta forma, y con el tiempo suficiente, los usuarios pueden convertirse en especialistas en la materia, al asimilar el proceso de razonamiento seguido por el sistema. El subsistema de explicación también puede usarse para depurar el sistema experto durante su desarrollo. (Samper Márquez, 2004)

2.3.14 El Módulo de Adquisición del Conocimiento.

Permite que se puedan añadir, eliminar o modificar elementos de conocimiento (en la mayoría de los casos reglas) en el sistema experto. Si el entorno es dinámico es muy necesario, puesto que, el sistema funcionará correctamente sólo si se mantiene actualizado su conocimiento. El módulo de adquisición permite efectuar ese mantenimiento, anotando en la base de conocimientos los cambios que se producen. (Samper Márquez, 2004)

2.4 Ventajas

Estos programas proporcionan la capacidad de trabajar con grandes cantidades de información, que son uno de los grandes problemas que enfrenta el analista humano que puede afectar negativamente a la toma de decisiones pues el analista humano puede depurar datos que no considere relevantes, mientras un *SE* debido a su gran velocidad de proceso analiza toda la información incluyendo las no útiles para de esta manera aportar una decisión más sólida. (Justo, 2004)

2.5 Limitaciones

Es evidente que para actualizar se necesita de reprogramación de estos (tal vez este sea una de sus limitaciones más acentuadas) otra de sus limitaciones puede ser el elevado costo en dinero y tiempo, además que estos programas son poco flexibles a cambios y de difícil acceso a información no estructurada. (Justo, 2004)

Debido a la escasez de expertos humanos en determinadas áreas, los *SE* pueden almacenar su conocimiento para cuando sea necesario aplicarlo. Así mismo los *SE* pueden ser utilizados por personas no especializadas para resolver problemas. Además si una persona utiliza con frecuencia un *SE* aprenderá de él.

Por otra parte la inteligencia artificial no ha podido desarrollar sistemas que sean capaces de resolver problemas de manera general, de aplicar el sentido común para resolver situaciones complejas ni controlar situaciones ambiguas.

El futuro de los *SE* da vueltas por la cabeza de cada persona, siempre que el campo elegido tenga la necesidad y/o presencia de un experto para la obtención de cualquier tipo de beneficio. (Montes Cerra , 2003)

2.6 Metodologías

2.6.1 Metodologías para el Desarrollo de un Sistema Experto

Al igual que para desarrollar un sistema de información convencional existen varias metodologías de desarrollo como la Ingeniería de la Información, tendencias estructuradas y orientadas a objetos, así existen varias metodologías para desarrollar un sistema experto. Como ya sabemos el área de sistemas expertos es relativamente joven por lo cual no se dispone de una única metodología sino que cada autor propone una de acuerdo a su forma de desarrollo. Sin embargo existen algunas que han tenido éxito mas que otras lo cual ha llevado a su mayor difusión. (DURKIN, 2014)

A continuación se mencionan algunas

- ◆ Metodología de Buchanan
- ◆ Metodología de Grover
- ◆ Metodología de Brule
- ◆ Metodología de Blanque y García Martínez
- ◆ Metodología KADS
- ◆ Ingeniería del Software,..... etc.

Metodología Buchanan

En la adquisición de conocimiento el ingeniero de conocimiento procede a través de una serie de etapas para producir un Sistema Experto.

Se destacan 6 etapas fundamentales:

- Identificación
- Conceptualización
- Formalización
- Implementación
- Testeo
- Revisión del prototipo

La característica más importante de esta metodología es la constante relación entre el Ingeniero de Conocimiento y el Experto de Campo. (Jimenez, 2014)

Metodología de Grover

El método de Grover propone una serie de etapas en el desarrollo del proceso de adquisición del conocimiento, cada una de las cuales va acompañada de una documentación detallada.

Se distinguen tres etapas:

- Definición del dominio
- Formulación del conocimiento fundamental
- Consolidación del conocimiento Basal

La característica más importante es la obtención de documentación que puede reemplazar parcialmente al experto, y servir a los diseñadores y usuarios como medio de documentación y referencia. (Jimenez, 2014)

Metodología de Brulé

Muchos de los trabajos en Sistemas Expertos no son dirigidos correctamente. En la mayoría de los casos el problema se encuentra en la construcción del software y no en la adquisición del conocimiento, la metodología de Brulé pretende dar solución a estos inconvenientes

Se distinguen siete etapas:

- Pre-planeamiento
- Diseño y especificación
- Desarrollo temprano
- Implementación
- Evaluación
- Supervisión
- Mantenimiento

La característica más importante de esta metodología es el desarrollo de un Sistema Experto temprano, que incrementalmente converge al sistema experto final. (Jimenez, 2014)

Javier Blanquet y García Martínez

Se distinguen cinco etapas:

- Adquisición del conocimiento
- Enunciación de conceptos
- Parametrización de conceptos

- Planteo de causalidades
- Verificación

La característica más importante es la etapa de planteo de causalidades, ya que los grafos de causalidades son una excelente herramienta para la representación del conocimiento previo a la formalización de reglas y la verificación, ya que compara el procedimiento que realiza el experto de campo con el que realizará el sistema; pudiendo establecer la performance del sistema. (Jimenez, 2014)

Metodología CommonKADS

El desarrollo de un sistema basado en el conocimiento, desde el punto de vista de CommonKADS, se comprende como la construcción de una serie de modelos de comportamiento de solución de problemas, vistos en su contexto organizacional y de aplicación concreto. En CommonKADS se ofrecen un conjunto de formularios que facilitan la construcción del sistema y permiten obtener las especificaciones y los requerimientos de un problema y su solución. (Rodríguez, 2014)

Ciclo de vida de CommonKADS

Esta metodología está fundamentada en el modelo del ciclo de vida en espiral el cual proporciona una estructura para el desarrollo del sistema computarizado.

- Existe un conjunto de fases con un orden de ejecución predeterminado que constituyen el desarrollo.
- Dentro de cada fase se lleva a cabo un conjunto de actividades distintas.
- Al final de cada fase se produce uno o más productos tangibles, como documentos, informes, diseños, entre otros, que normalmente constituyen entradas a otras fases.

CommonKADS está formada por un conjunto de etapas, cada una con tareas y productos asociados, las cuales son:

- Análisis
- Diseño
- Implantación del sistema
- El uso
- El mantenimiento y refinamiento del conocimiento.

2.6.2 Metodología de la Investigación

Esta metodología tiene como propósito, que el lector tenga una visión general del proceso y al mismo tiempo lo guíe para realizar un proyecto.

La metodología de la investigación supone la realización de una serie de actividades, para alcanzar los objetivos deseados deben preverse y organizarse de una cierta manera. Esta consiste en la previsión de las distintas actividades que deberá realizar el investigador durante el proceso de la investigación y respecto a cada una, es preciso calcular el tiempo que durará su ejecución y los recursos necesarios para su realización.

Es importante considerar que a las actividades investigadoras son modificables, que puede en la mayoría de los casos variarse y perfeccionarse durante el desarrollo del proceso para adaptarla a los requerimientos institucionales. La metodología de la investigación es solo un instrumento de orientación general. (Jimenez, 2014)

2.7 Metodología CommonKADS

2.7.1 Perspectiva Histórica

Es una metodología para la construcción de sistemas basados en el conocimiento, resultado de varios proyectos enmarcados dentro del programa ESPRIT, para la innovación y la aplicación de tecnología informática avanzada en la Unión Europea. Fue desarrollada en la Universidad de Ámsterdam en cooperación con varios socios europeos, como universidades, organizaciones de investigación, casas de software y de consultoría. Con dicha metodología se han desarrollado muchos sistemas de conocimiento y por eso actualmente es considerada por muchas compañías y organizaciones alrededor del mundo como un estándar para la ingeniería del conocimiento y de los SBC.

Inicialmente, se planteó el desarrollo de un método para la adquisición del conocimiento en el proceso de construcción de un sistema basado en el conocimiento al cual se llamó KADS (Knowledge Acquisition Design System).

Posteriormente y dado los buenos resultados obtenidos, se amplió el proyecto a la construcción de una metodología completa para el desarrollo de sistemas basados en el conocimiento, empezando en el análisis mismo de la organización en donde se va a hacer el SBC hasta la gestión del proyecto, pasando por el diseño del software. Es en ese momento cuando se propone y acepta el nombre de CommonKADS.

CommonKADS está fundamentada en los siguientes principios:

La ingeniería del conocimiento hoy en día se enfoca en la realización de actividades de modelado, antes era vista sólo como un proceso de extracción de la pericia del experto para traducirla a una forma computacional.

En CommonKADS un proyecto de conocimiento incluye la construcción de una serie de modelos que constituyen parte del producto entregado. Estos modelos reflejan diferentes puntos de vista del conocimiento inmerso en un problema y en su solución. Cada uno tiene un propósito específico, unos productos asociados y unas estrategias para su desarrollo. Es importante recordar que un modelo es una abstracción útil de alguna parte de la realidad que hace posible focalizar ciertos aspectos e ignorar otros.

El modelado del conocimiento primero se concentra en su estructura conceptual y después en los detalles de la programación, aunque muchos constructores de software tienen la tendencia a tomar el ordenador como el punto de referencia dominante en sus actividades de análisis y diseño. En CommonKADS se sigue el principio que esbozó Alan Newell en 1982 “para que el conocimiento pueda ser modelado en un nivel conceptual debe ser independiente de las construcciones informáticas específicas y de la implantación del software”.

El conocimiento tiene una estructura interna estable en la que aparecen muestras similares, lo que facilita su análisis para obtener tipos, patrones, roles y estructuras del conocimiento específico, y así se modela como un todo funcional bien estructurado, formado por partes que juegan diferentes roles restrictivos y especializados en la solución de problemas.

Un proyecto de conocimiento tiene que ser gestionado como un proyecto de aprendizaje basado en la experiencia, en forma de espiral controlada. CommonKADS de esta forma favorece el enfoque de administración de proyectos ordenable, balanceado y que permite un aprendizaje

estructurado, en donde los resultados o “estados” de los modelos actúan como indicadores de gestión para saber cómo se han realizado las actividades y qué pasos deben seguirse después.

CommonKADS refleja la influencia de paradigmas ampliamente conocidos como es el análisis y el diseño estructurado, la orientación por objetos, la teoría de las organizaciones, la reingeniería de procesos y la gerencia de la calidad. Esto tiene la ventaja que toma en cuenta tanto la experiencia como las estructuras de información existentes en la organización.

Desde el punto de vista de CommonKADS, el SBC es un modelo operacional que exhibe los comportamientos deseados que se han especificado u observado en el mundo real, definen que el desarrollo de un sistema basado en el conocimiento, desde el punto de vista de CommonKADS, es entendido como la construcción de una serie de modelos de comportamiento de solución de problemas, vistos desde su contexto organizacional y de aplicación concreto. En estos modelos se incluyen tanto los conocimientos de los expertos como los de otros sistemas del entorno, tales como la organización, el usuario y la interacción entre éste y el sistema. Un SBC es una realización computacional asociada con estos modelos.

CommonKADS también ofrece una serie de formularios que facilita la labor de construcción del sistema y que permite obtener las especificaciones y los requerimientos de un problema y su solución, desde el punto de vista de su relación con el resto de la organización, de los entes que participan en el problema y del conocimiento que se requiere para llegar al sistema final. Estos formularios son su forma estructural. (Henaó Cálad, 2001)

2.7.2 Ciclo de Vida en CommonKADS

CommonKADS está fundamentada en el modelo del ciclo de vida en espiral que tanto se trabaja en la Ingeniería del Software y que proporciona una estructura para el desarrollo del sistema computarizado:

- El desarrollo se divide en un conjunto de fases con un orden de ejecución predeterminado.
- Dentro de cada fase debe llevarse a cabo un conjunto de actividades distintas.
- Al final de cada fase han de producirse uno o más productos tangibles (por ejemplo, documentos, informes, diseños, programas) normalmente como entradas a otras fases.

La metodología está formada por una serie de etapas, cada una con unas tareas y productos asociados. Brevemente éstas son:

2.7.2.1 El Análisis

Se realiza para comprender el problema desde el punto de vista de la solución que se piensa desarrollar. Está formado por la especificación de los requerimientos externos del sistema basado en el conocimiento y por un análisis del problema específico. Los productos que se deben obtener en esta etapa son: un documento del proyecto, un documento de los requerimientos, un documento del modelo (modelo conceptual), un documento de viabilidad y un documento de apoyo. (Henao Cálad, 2001)

2.7.2.2 El Diseño

En el cual se hace una descripción del comportamiento del sistema (descripción funcional) y una descripción física en la que se especifica detalladamente cada uno de sus componentes. De esta etapa debe salir toda la especificación modular del sistema y la descripción detallada de cómo debe ser, desde el punto de vista computarizado. (Henao Cálad, 2001)

2.7.2.3 Implantación del Sistema

En esta etapa se considera tanto la integración del software desarrollado como su adaptación en la organización. (Henao Cálad, 2001)

2.7.2.4 Instalación

Consiste en la puesta en marcha del sistema con el fin de que comience a operar en la empresa, iniciándose su proceso productivo. (Henao Cálad, 2001)

2.7.2.5 El Uso

Se plantean actividades relacionadas con el manejo mismo del sistema y de las salidas o resultados que éste proporciona. (Henao Cálad, 2001)

2.7.2.6 El mantenimiento y Refinamiento del Conocimiento.

2.7.3 Los Modelos de CommonKADS

Permiten describir el conocimiento de la solución de problemas en un dominio particular usando niveles de abstracción que posibilitan al ingeniero del conocimiento el detallar el proceso de solución en una forma independiente del dominio. La idea central de esta metodología es agrupar los datos relevantes en modelos separados. En la Figura II. 5 se presentan los diferentes modelos que soportan el análisis del conocimiento en CommonKADS y que constituyen su núcleo. (Henao Cálad, 2001)

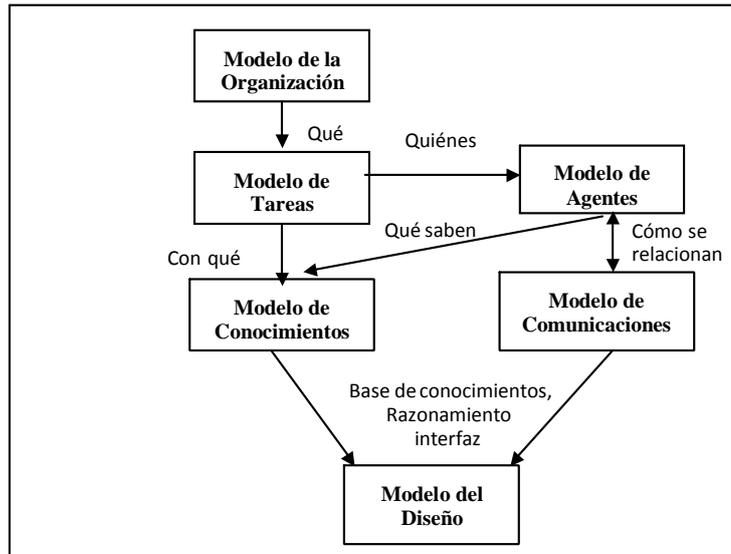


Figura II. 5. Modelos de CommonKADS

fuelle: <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>

2.7.3.1 Modelo de la Organización

Este modelo refleja el análisis de las principales características de una organización con el objetivo de descubrir problemas que pueden ser solucionados por sistemas de conocimiento, establecer su viabilidad y evaluar el impacto que tendría en el entorno donde se implanten. Está formado por una serie de constituyentes o conceptos que reflejan la información y el conocimiento de la organización, sus problemas y sus soluciones, especialmente basadas en el conocimiento. En la Figura II.6 se presenta su estructura, siguiendo la notación de donde:

- El Contexto organizacional se relaciona con las características claves del ambiente de la organización, tales como la misión, la visión, los objetivos de la organización.
- Problemas y oportunidades: Constituyen la lista de los problemas de la organización o las necesidades que son consideradas para ser más o menos urgentes que pueden ser solucionados o mejorados por el SBC.
- Problema actual, referente a un problema o a una oportunidad en la cual la compañía ha decidido hacer esfuerzos y que ha sido seleccionado de la lista de Problemas y Oportunidades.
- Solución: Corresponde a los escenarios que han sido o serán desarrollados para solucionar el problema actual o modificar las necesidades presentes.
- Función: Es un inventario de las funciones que pueden ser distinguidas en una organización particular. Por ejemplo producción, finanzas, relaciones laborales o comerciales.

- Proceso: Se refiere al flujo de trabajo (work-flow) o de control (control-flow) de los procedimientos básicos de la empresa.
- Estructura: Indica la disposición de la organización en función de sus departamentos, grupos, unidades o secciones. También, la descripción del proceso del negocio, entendiendo un proceso como una parte relevante de la cadena de valor que es enfocada.
- Personas o roles que se juegan en una organización y que son fundamentales en los procesos y funciones de la empresa.
- Conocimiento: Representa el conocimiento general y de alto nivel que puede influir en la definición del problema actual o en sus soluciones.
- Recursos computacionales que soportan los procesos de la compañía.
- Otros recursos: Referente a los demás recursos (económicos, de papelería, de propiedades, entre otros) que se requieren en la compañía para realizar las funciones y cumplir con los objetivos de la organización.
- Cultura y Poder: Son las relaciones que existen entre los roles, las formas en que se realizan las actividades y las políticas formales e informales que soportan la organización.

Estos constituyentes están clasificados como variables o invariables para que se puedan tener diferentes instancias del modelo en diferentes soluciones para el mismo problema:

- Dependientes de la solución considerada. En la Figura 6 no están dentro de un marco de líneas punteadas.
- Los invariables son los que en la Figura II. 6 están encerrados en un marco de líneas punteadas. No dependen del tipo de solución en particular porque se asumen como fijos. (Henaó Cálad, 2001)

En CommonKADS, también se han incluido unos formularios que contienen la descripción de los constituyentes y que permiten que el Ingeniero del Conocimiento fácilmente refleje la información relativa a su organización. A continuación se nombran los que pertenecen al Modelo de la Organización (Organization Model - OM). En The CommonKADS Methodology [version 0.5] están detallados:

- OM-1: Identificación en la organización los problemas y oportunidades orientados al conocimiento.
- OM-2: Descripción de los aspectos de la organización que tienen impacto en o están afectados por la solución de conocimiento escogida.

- OM-3: Descripción del proceso desde el punto de vista de las tareas en que está compuesto y sus características principales.
- OM-4: Descripción del componente de conocimiento del modelo de la organización y sus principales características.
- OM-5: Lista de chequeo para el documento de viabilidad de la decisión.

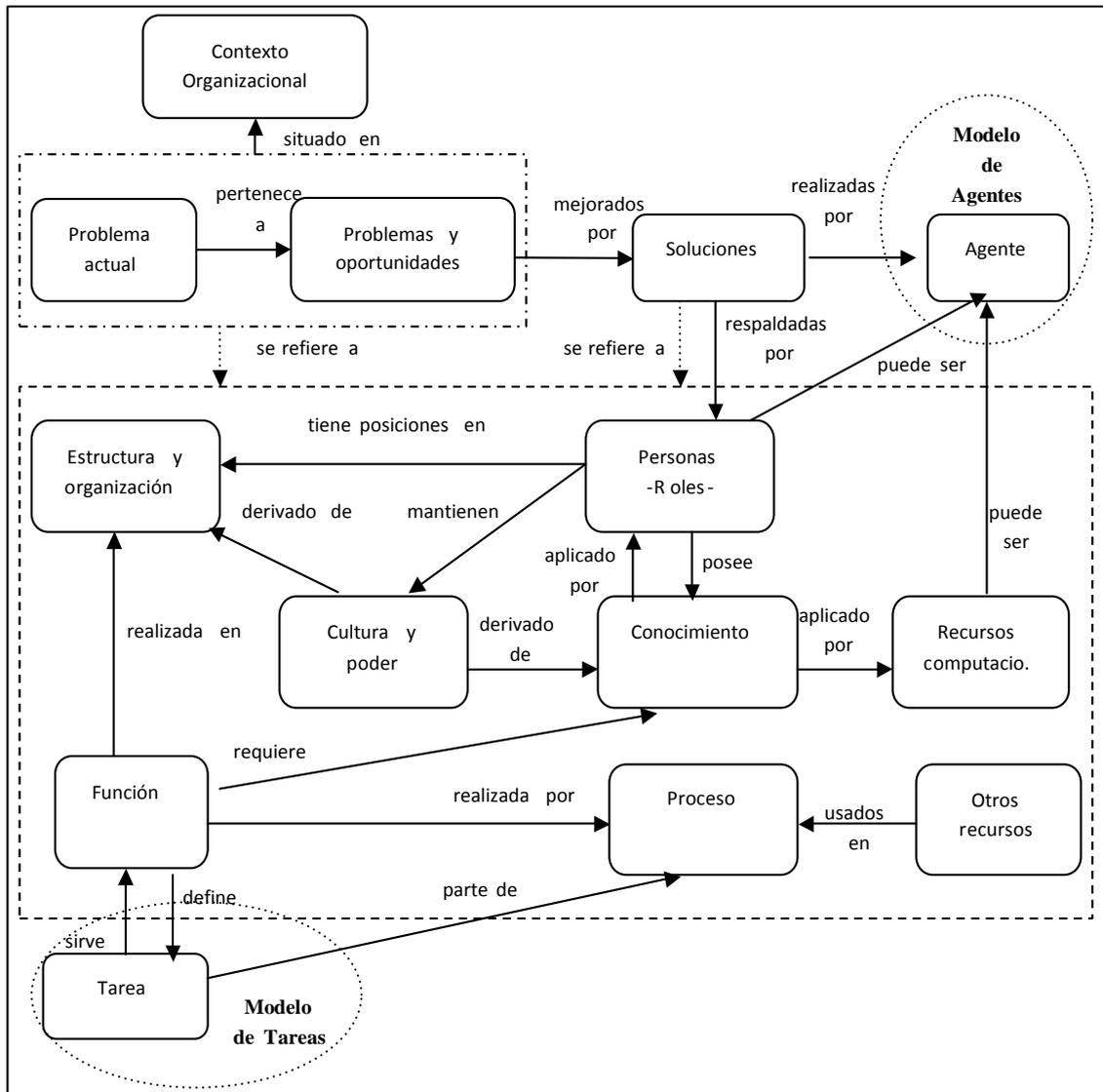


Figura II. 6. Modelo de la organización de CommonKADS

fuelle: <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>

2.7.3.2 Modelo de Tareas

Para CommonKADS una tarea es una parte de un proceso de negocios que representa una serie de actividades orientadas a alcanzar un objetivo, llevada a cabo por unos agentes que siguen unos criterios de rendimiento y calidad. La tarea recibe entradas y entrega salidas deseables en una forma controlada y estructurada, consume recursos y requiere (y provee) conocimiento y otras habilidades. El análisis de tareas le sirve al IC para organizar una vista de las tareas principales que un experto realiza en un área dada y para determinar el alcance del SBC que servirá de soporte en el análisis de viabilidad del proyecto.

En Duursma Task Model Definition and Task Analysis Process se dice "no es posible definir el concepto de tareas en cuanto a las condiciones suficientes y necesarias, pero nosotros somos capaces de describir qué clase de actividades y comportamientos se pueden considerar tareas y que pueden ser aplicados muy útilmente en la metodología". A continuación se presentan estas características:

- Una tarea tiene un comienzo y un final confirmados. Se ejecuta en un período relativamente corto o puede ser dividida en sub-tareas que permiten que se cumpla esto.
- Cada sentencia de la tarea debe ser entendida claramente por quien hace el trabajo. Una sentencia es una serie coherente de actividades.
- La sentencia describe una parte finita e independiente del trabajo. Es decir, tiene significado en el contexto del proceso y sus instrucciones deben dar una descripción completa de la correspondiente tarea.
- Cada tarea debe ser manejable en función del tiempo consumido en su realización.
- Una tarea comienza con una clave observable que permite definir cuando ésta es iniciada. Es importante resaltar que en el caso en que la tarea sea continua, la clave no existe.
- La sentencia de una tarea debe evitar el incluir calificativos.
- La sentencia usa un verbo, excepto cuando varias acciones se ejecutan juntas.
- La tarea se debe poder medir, su resultado o producto puede ser estimado o medido.

Por lo tanto, el modelo de tareas permite reflejar el proceso de análisis de una tarea elegida. El enfoque que presenta CommonKADS para este modelo tiene dos ideas innovadoras: 1) Se pueden tener varias versiones del modelo para modelar las situaciones actuales, intermedias y requeridas; varios modelos de tareas pueden ser instanciados en diferentes estados de un proyecto. 2) El tener el modelo enmarcado en un desarrollo orientado a los riesgos permite que se revisen continuamente. (Henaó Cálad, 2001)

Los constituyentes de este modelo se ven relacionados en la Figura II. 7 y son los siguientes:

- Tarea: Entidad que representa una tarea del proceso y que es la parte central del modelo.
- Característica: Refleja una propiedad de la tarea que la caracteriza en cuanto a un lenguaje abstracto.
- Entorno: Representa las restricciones formales en la ejecución de la tarea, que son establecidas dentro de la organización, a través de las leyes generales, de reglas comerciales o profesionales.
- Ingrediente: Representación de una entrada de la tarea o ingrediente de salida. Un ingrediente describe los contenidos de información que son usados o producidos por la tarea. Esta descripción es orientada hacia la semántica de la información, no a su representación sintáctica.
- Capacidad: Representación de una capacidad necesaria para ejecutar la tarea.

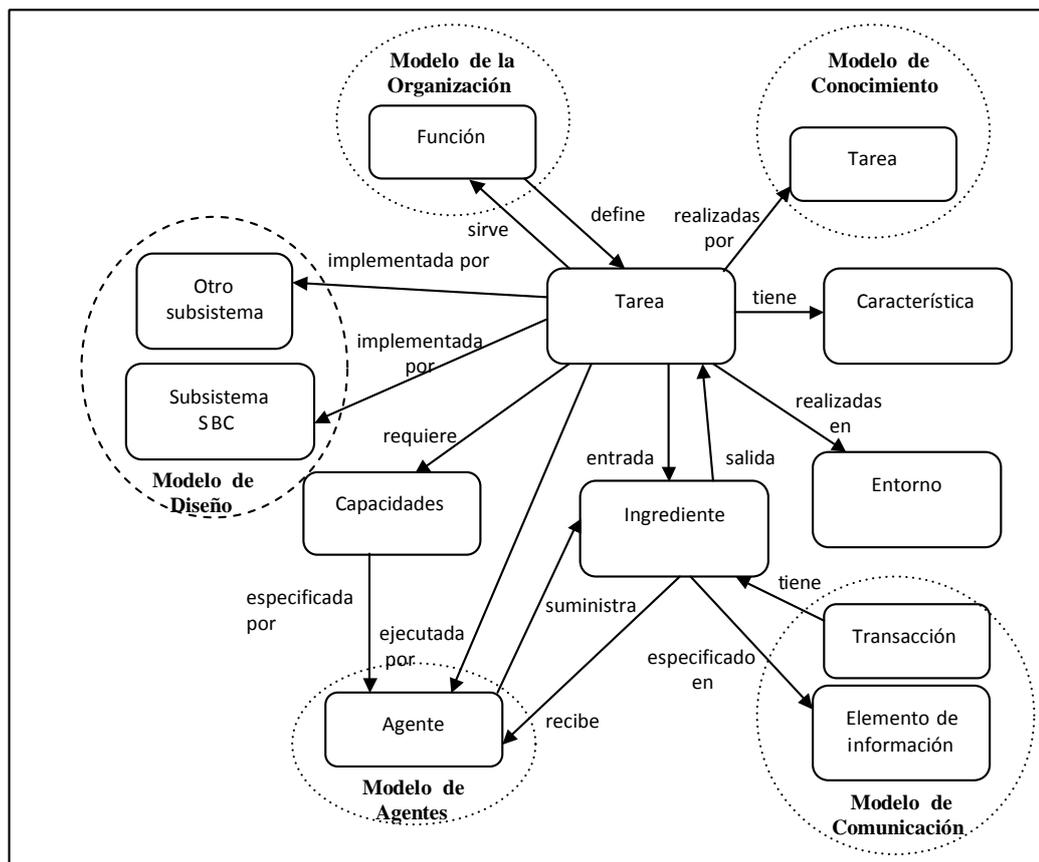


Figura II. 7. Modelo de tareas de CommonKADS

fuentes: <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>

Los formularios que se proponen para este modelo (TM) son:

- TM-1: Descripción refinada de las tareas dentro del proceso objetivo.
- TM-2: Especificación del conocimiento empleado por una tarea y posibles cuellos de botella y áreas para mejorar.

Los resultados que se obtienen al construir las diferentes instancias del modelo de tareas se utilizan en CommonKADS para:

- Documentar el resultado del análisis de las actividades actuales.
- Documentar el resultado del diseño de la tarea y sus actividades propuestas.
- Soportar la administración del cambio organizacional.
- Fijar el alcance de la solución del SBC.
- Soportar la evaluación de la viabilidad del proyecto.
- Identificar las necesidades de capacitación y entrenamiento.
- Evaluar la eficiencia, robustez y calidad del trabajo en la organización.

2.7.3.3 Modelo de Agentes

Para CommonKADS un agente es quien ejecuta una tarea. Puede ser un individuo, un sistema de información o cualquier otra entidad capaz de llevar a cabo dicha ejecución. Incluso el SBC por sí mismo es un agente para CommonKADS, lo mismo que un usuario que va a interactuar con él.

La idea de agente que se maneja en CommonKADS es la de actor, no es exactamente la misma que se trabaja en Agentes Inteligentes o en Sistemas Multiagentes. Para este último, se ha presentado MAS-CommonKADS que es una extensión de CommonKADS que permite modelar sistemas en donde se presentan diversos agentes como sistemas distribuidos.

Este modelo sirve como enlace entre el modelo de tareas, el de comunicación y el de conocimientos para modelar las capacidades y limitaciones que los agentes tienen y que están involucradas en la solución de una tarea. (Henaó Cálad, 2001)

En la Figura II. 8 se muestran los constituyentes del modelo de agentes de CommonKADS:

- Agente: Se desarrolla por cada tipo de agente identificado. Tiene los siguientes atributos:
 - Nombre

- Tipo. Permite identificar si el agente es un humano o un sistema que debe ser desarrollado en el SBC o un sistema que ya existe
- Rol. Este atributo puede ser heredado del modelo de la organización
- Posición. Se refiere al nivel en donde está el agente en la organización

También puede ser heredado del modelo de la organización.

- Capacidades de razonamiento: Comprende todos los requerimientos de pericia del agente, los cuales son importantes o impuestos por la asignación de la tarea y por las necesidades de comunicación. Para los agentes que son componentes del sistema y que son desarrollados dentro del mismo proyecto, este constituyente se usa para comprender una especificación de uno o varios modelos del conocimiento. Para los agentes humanos, es muy difícil hacer una lista completa de estos requerimientos, por lo que sólo se especifican los que son *determinantes para la funcionalidad del sistema o varían entre diferentes usuarios*:
- Capacidades: Este constituyente contiene dos atributos:
 - Habilidades que se tienen para manipular el entorno en diferentes formas y acceso a información sensorial. Esto está relacionado con los dispositivos que pueden imitar los órganos de los sentidos como los brazos robóticos o los sensores en el ordenador.
 - Vocabulario. Describe el lenguaje de comunicación del agente.
- Restricciones: Éste tiene tres atributos, de los cuales los dos primeros solamente se aplican a agentes humanos:
 - Normas que se refieren a lo que el agente considera que es lo más apropiado para hacer en ciertas situaciones.
 - Preferencias de cómo le gustaría al agente realizar una tarea en particular.
 - Permisos que tiene el agente dentro de una tarea.

Las relaciones que hay entre el modelo de agentes y los demás se interpretan de la siguiente forma:

- Relaciones Organización – Agente: Todos los agentes corresponden a personas o recursos en el *modelo de la organización*. La posición y el rol de los agentes son coherentes con sus responsabilidades.

- Relaciones Tarea – Agente: Todas las tareas son asignadas a los agentes que son capaces de ejecutarlas y todos los ingredientes en el *modelo de tareas* son servidos y recibidos por los agentes relevantes.
- Relaciones Conocimiento – Agente: Las capacidades de razonamiento descritas en *el modelo de agentes* son modeladas adecuadamente por (un subconjunto de) el correspondiente modelo de conocimientos.
- Relaciones Comunicación – Agente: Todas las transacciones tienen al menos dos agentes involucrados, uno que posee la información y otro que la recibe para el mismo ingrediente. Al menos dos de los agentes involucrados tienen la capacidad y los permisos requeridos para participar en la transacción.

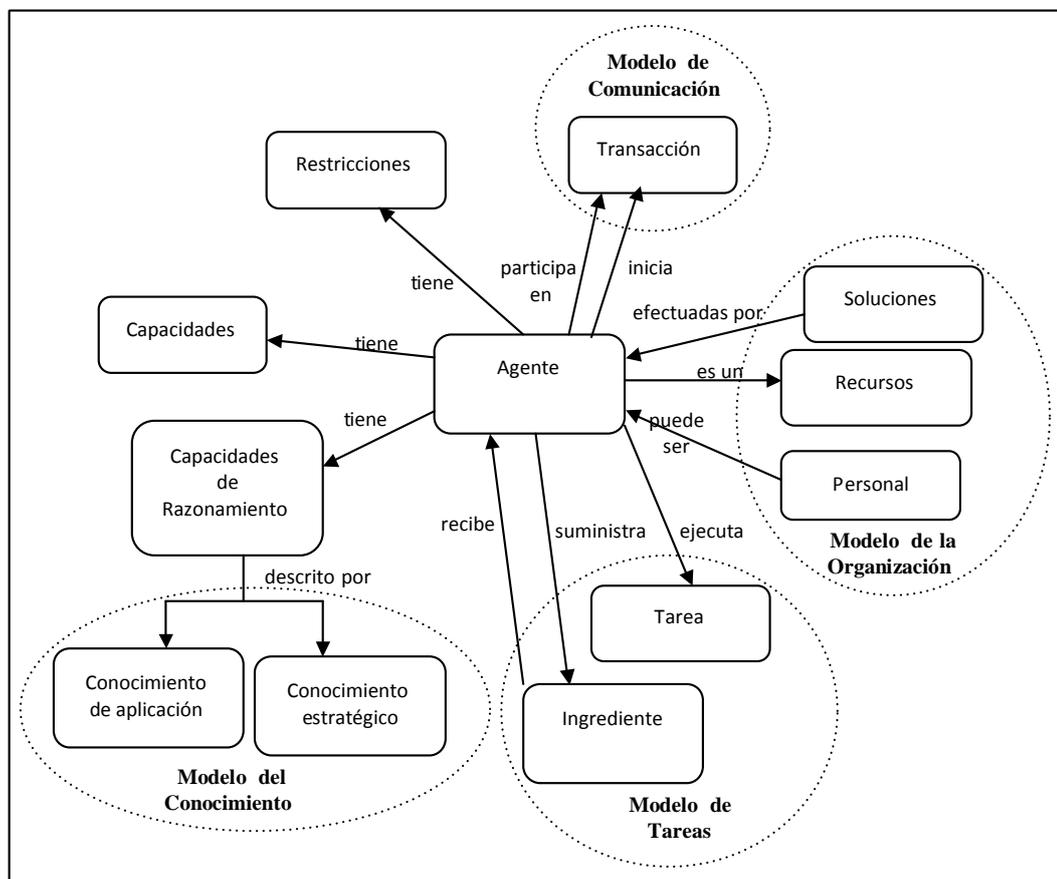


Figura II. 8. Modelo de agentes de CommonKADS

fuelle: <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>

Para este modelo sólo se ha proporcionado el siguiente formulario (AM):

- AM-1: Especificaciones del Agente de acuerdo con el modelo de agentes de CommonKADS

2.7.3.4 Modelo de Conocimientos

Su propósito es explicar en detalle los tipos y estructuras del conocimiento usado en la realización de una tarea. Para dicha definición se ha desarrollado el lenguaje CML2 (CML - Conceptual Modeling Language), que proporciona todas las estructuras necesarias para especificar los datos y el conocimiento del sistema. La definición que se hace en este lenguaje, es independiente de la implementación del mismo. CML2 está basado en el lenguaje (ML)2. El modelo de conocimientos sigue una estructura que determina las diferentes categorías del conocimiento que se maneja en el SBC, como se puede apreciar en la figura II. 8.

En CommonKADS el conocimiento está diferenciado, dependiendo del tipo de conocimiento que se trate (niveles). La importancia de separar el conocimiento del dominio del de control es que permite hacer su reutilización. Así, el conocimiento del dominio puede ser utilizado de nuevo para diferentes tareas y a su vez el de la tarea en diferentes dominios.

- El conocimiento del dominio tiene como propósito definir la conceptualización del dominio y debe ser representado en forma independiente de su uso.
- El conocimiento de inferencia define el primer tipo de conocimiento de control. Específica las derivaciones que constituyen un método de solución del problema, la forma como se usa el conocimiento del dominio en las inferencias y los roles del conocimiento que modelan las premisas y las conclusiones de las deducciones.
- El conocimiento de la tarea representa una estrategia fija para alcanzar las metas de la solución del problema. Por lo tanto es otro tipo de control que tiene como propósito especificar el registro de la ejecución de los pasos de inferencia básicos definidos en el conocimiento de inferencia.

El modelo de interpretación del SBC está formado por el conocimiento de control, en este caso por el de inferencia y el de tarea. Esto también se conoce como un Método de Solución de Problemas que define en términos genéricos un modelo del comportamiento de la capacidad de solución de problemas del sistema. Los PSM forman librería que permiten su reutilización y pueden ser usados como guía en la adquisición del conocimiento del dominio y del conocimiento adicional de la solución de problemas específico del dominio, tal como las heurísticas y las restricciones. (Henaó Cálad, 2001)

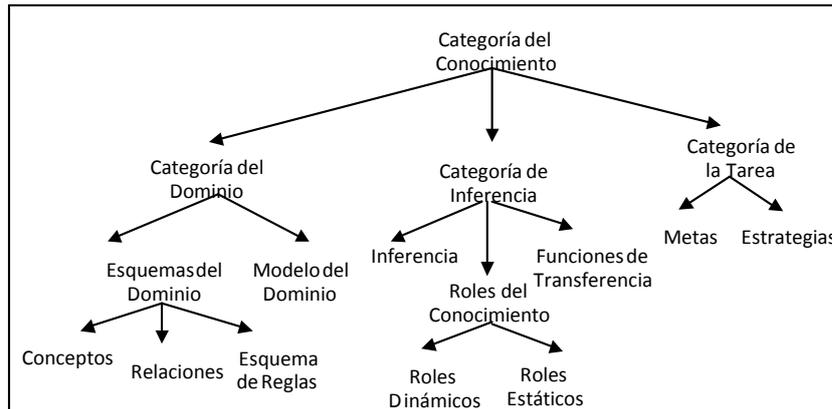


Figura II. 9. Jerarquía del modelo de conocimientos de CommonKADS

fuelle: <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>

Para el modelo del conocimiento sólo se ha planteado el siguiente formulario (Knowledge Model - KM):

- KM-1: Lista de chequeo de la documentación del dominio

Muchos métodos y metodología utilizan como base este modelo del conocimiento, haciendo variaciones en algunos de sus conceptos, pero manteniendo la estructura y la idea fundamental de reutilización.

2.7.3.5 Modelo de Comunicación

El propósito de este modelo es especificar los procedimientos de intercambio de información para realizar la transferencia de conocimiento entre los agentes que participan en la ejecución de una tarea. Al igual que con el modelo anterior, esto es hecho en una forma conceptual e independiente de la implementación.

El componente clave del modelo es la transacción que describe los actos de comunicación entre los diferentes agentes que participan en una tarea en el sistema. El cual dice qué objetos de información son intercambiados entre qué agentes y qué tareas. Como plantea The CommonKADS Methodology [version 0.5] “Las transacciones son bloques de construcción para el diálogo completo entre dos agentes, lo cual es descrito en el plan de comunicación”. Por tanto, la transacción por sí misma consiste de diversos mensajes que son detallados en la especificación

del intercambio de información, basada en tipos y patrones de comunicación. (Henaó Cálad, 2001)

Este modelo se construye desde lo general hasta lo particular, de la siguiente forma:

- Se define el plan completo de comunicación que dirige el diálogo entre los agentes.
- Se determinan las transacciones individuales que relacionan dos tareas, llevadas a cabo por dos agentes diferentes.
- Se especifica el intercambio de información que detalla la estructura interna de los mensajes de una transacción.

En la Figura II.10 se presentan los siguientes constituyentes o conceptos de este modelo:

- Plan de comunicación: Describe los requerimientos en el orden de las transacciones, siguiendo la sintaxis que se presenta en The Common-KADS Communication Model. Technical Report ESPRIT Project P5248 KADSII, KADS-II/M3/SICS/TR/006/V.2.0. Contiene la lista de tareas que son llevadas a cabo por el agente que se está considerando, la descripción de éste, las funciones de transferencia que pertenecen a la estructura de la tarea o de la inferencia que participan en la comunicación, y un diagrama de estados o pseudo – código que refleja la especificación del control sobre las transacciones. Este concepto tiene dos atributos: Preferencias y Requerimientos.
- Transacción: Describe la estructura de las transacciones individuales. Una transacción tiene como propósito básico transferir una serie de ingredientes del dueño de la información hacia un recipiente de información. Cada instancia de la transacción debería realizar al menos una instancia de la tarea de transferencia en el modelo del conocimiento, excepto para la inicial que indica que el usuario necesita usar el sistema para un propósito en particular. Sus atributos son:
 - Nombre, Tipo de comunicación, Ingredientes adicionales, Restricciones.
- Discurso: Define el plan para llevar a cabo la transacción en particular como un conjunto de interacciones únicas o acciones lingüísticas. Se utiliza para expresar cosas como el hecho que hay que suspender un mecanismo o para fraccionar la transferencia de un ingrediente.
- Artículos de información: Deben precisar cómo se expresan las diferentes acciones lingüísticas que ocurren en el discurso. Se utilizan para definir la forma en que son transferidos los

ingredientes. Este concepto está formado por dos atributos: el objeto sintáctico y el medio de salida.

- Capacidades: Por cada parte involucrada en una transacción hay un conjunto de capacidades que el agente tiene que tener para ejecutar dicha transacción. Para esto, se tienen dos atributos: habilidad y conocimiento. El primero sirve para describir el conocimiento relacionado con la tarea de razonamiento del sistema o con el conocimiento del agente. El segundo se refiere a la(s) habilidad(s) que el agente debe tener, aparte de su conocimiento, para participar en la transacción.

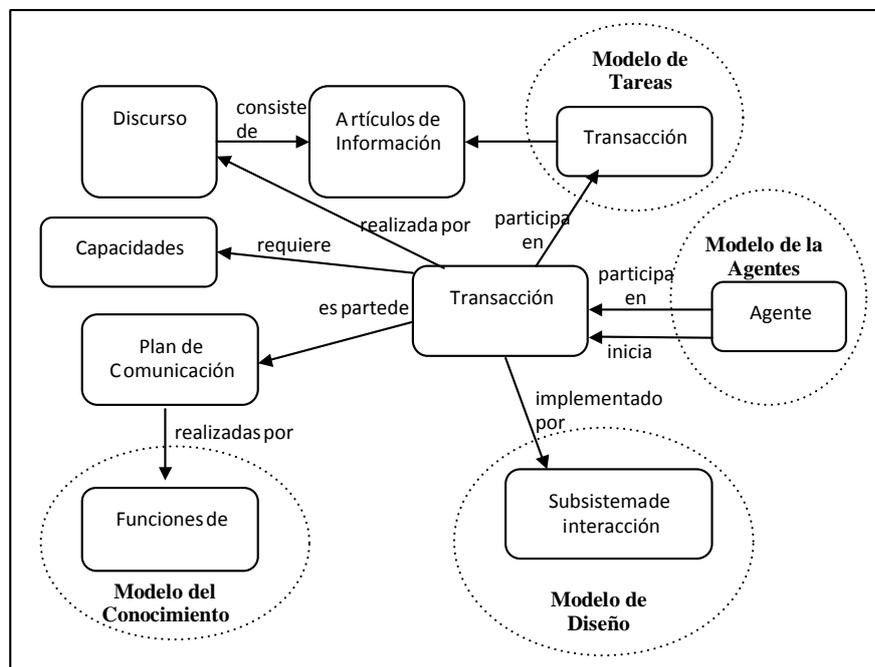


Figura II. 10. Modelo de comunicaciones de CommonKADS

fuelle: <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>

Al igual que los demás modelos, tiene unos formularios (Communication Model CM) que facilitan su construcción:

- CM-1: Especificación de las transacciones que participan en el diálogo entre dos agentes en el *modelo de comunicaciones*.
- CM-2: Especificación de los mensajes y la información que forman una transacción individual dentro del *modelo de comunicaciones*.

2.7.3.6 Modelo de Diseño

Proporciona la especificación técnica del sistema en cuanto a la arquitectura, la plataforma de implementación, los módulos de software, los métodos y mecanismos computables, necesarios para implementar las funciones ofrecidas en los demás modelos.

Este modelo es diferente a los demás porque parte del mundo del software. Es decir, está en el dominio del software del sistema ya que está relacionado con el software y su organización interna. En cambio los demás pertenecen al dominio de la aplicación.

Las entradas a este modelo son: El *modelo de conocimientos* que se puede ver como una especificación de los requerimientos de solución del problema y las manifestaciones de la interacción externa y requerimientos no funcionales definidos en el *modelo de la organización*. Sirve para describir la estructura que tiene el sistema de software que se necesita para construirlo en función de sub-sistemas, módulos de software, mecanismos computarizados y constructores que se requieren para implementar los modelos de *conocimientos* y *de comunicación*. (Henao Cálad, 2001)

El proceso del diseño consiste de cuatro pasos que se pueden apreciar en la figura II.11 y que son los siguientes:

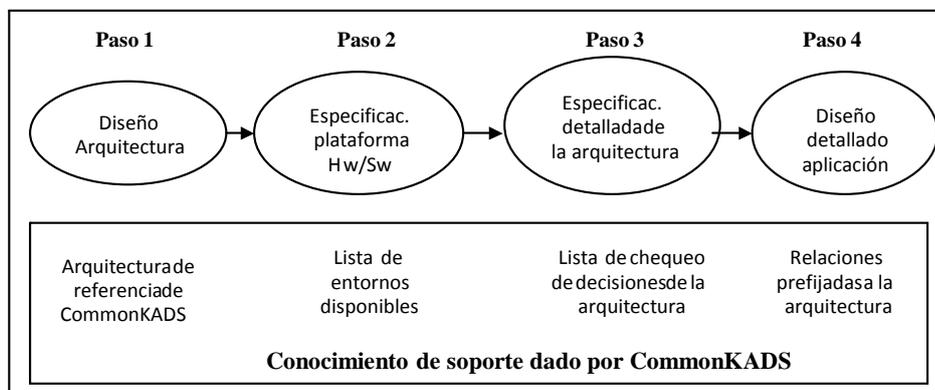


Figura II. 11. Pasos del diseño del sistema

fuelle: <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>

- Paso 1: Diseñar la arquitectura del sistema que define la estructura general del software que se está construyendo y que comprende la descomposición del sistema en sub-sistemas, un régimen de control global y una descomposición de los sub-sistemas en módulos de software.

Para este paso se cuenta con el formulario (Design Model - DM) DM-1: Descripción de la arquitectura del sistema.

- Paso 2: Identificar la plataforma de implementación objetivo, es decir, escoger el hardware y software que debería ser usado en el sistema. Para esto se tienen una serie de características relevantes para considerar en el momento en que se va a elegir el software: disponibilidad de las librerías de los objetos, representación del conocimiento declarativo, interfaces estándar con otro software, flujo de control, soporte de CommonKADS y, se cuenta con el formulario DM-2: Especificación de las facilidades ofrecidas por y en el sistema objetivo que será implementado.
- Paso 3: Especificar los componentes de la arquitectura. En este paso se diseñan en detalle cada uno de los sub-sistemas identificados en el paso 1. Así que se hace el diseño detallado de la representación, el control y las interfaces. Para esto CommonKADS provee una lista de chequeo para facilitar la toma de decisiones. Se cuenta con el formulario DM-3: Lista de chequeo de decisiones en relación con la especificación de la arquitectura.
- Paso 4: Especificar la aplicación dentro de la arquitectura. Se toman los ingredientes de los modelos de análisis (tareas, inferencias, modelos del dominio, transacciones) y se reflejan en la arquitectura. Para el diseño detallado de la aplicación (paso 4) se tiene el formulario DM-4: Decisiones de la aplicación de diseño que especifica las decisiones tomadas para cada uno de los elementos de la arquitectura.

CommonKADS proporciona algunas ayudas para realizar cada uno de los pasos del *modelo de diseño*:

- La arquitectura de referencias está basada en la metáfora Modelo-Vista Controlador (Model View Controller - MVC) que fue desarrollada como un paradigma para diseñar programas en el lenguaje SmallTalk-80. En esta arquitectura se distinguen tres sub-sistemas principales:
 - El modelo de la aplicación que contiene las funciones de razonamiento. Los datos estáticos son las bases de conocimiento y los datos dinámicos manipulados durante el proceso de razonamiento.
 - Las vistas que especifican las visualizaciones de las funciones y datos de la aplicación. Hacen que la información estática y dinámica de la aplicación esté disponible para los agentes externos.
 - El controlador es la unidad de control que maneja los eventos internos y externos.

2.7.4 Integración de los Modelos

Los modelos de CommonKADS están clasificados en tres niveles o vistas que posibilita el tener la información completa para construir el SBC en forma eficiente.

- El nivel del entorno, que relaciona la información del entorno del sistema de conocimientos. Implica tener entendimiento del contexto de la organización, de su ambiente y de los factores críticos de éxito correspondientes al sistema de conocimientos. En este nivel los modelos Organizacional, de Tareas y de Agentes permiten responder a las siguientes preguntas: ¿Por qué un sistema de conocimiento es una ayuda potencial o una buena solución de la situación actual? ¿Cuáles problemas son los que se acoplan más con este tipo de solución? ¿Cuáles son los beneficios, el impacto y los costos organizacionales que tendría esta solución?
- El nivel de conceptos para especificar lo que se quiere modelar. Responde preguntas como: ¿Cuál es la naturaleza y la estructura del conocimiento involucrado en la tarea? ¿Cuál es la naturaleza y estructura de la comunicación correspondiente? ¿Qué conocimiento se requiere para solucionar el problema? Por tanto, es necesario tener modelos que presenten la descripción conceptual del conocimiento aplicado a una tarea y los datos que son manejados y entregados por un sistema de conocimientos. En este nivel se tiene el modelo de conocimientos y de comunicación.
- El nivel del artefacto o componente para identificar los aspectos técnicos de programación y de construcción en el ordenador. Da solución a preguntas del cómo: ¿Cómo tiene que ser implementado el conocimiento en un sistema basado en el ordenador? ¿Cómo debe ser la arquitectura del software? ¿Cómo se debe traducir el conocimiento a un lenguaje comprensible por el ordenador? En este nivel está el modelo de diseño.
- Para ver la integración de los modelos es útil plantear un ejemplo sencillo en el que se haga esto explícito: en la interacción entre un usuario que proporciona datos a un sistema de conocimientos y donde este último razona y le presenta resultados al primero, entonces los modelos contendrían en términos generales lo siguiente: El modelo de agentes la descripción de los agentes involucrados, junto con sus capacidades; el modelo de tareas presenta las tareas, sus entradas y salidas (objetos de información) y su asignación a los diferentes agentes; Si la tarea es intensiva en conocimientos, entonces se describe en el modelo de conocimientos,

incluyendo una función de transferencia que indique que la entrada y la salida del proceso de razonamiento deben ser obtenidas o entregadas a otro agente; En el modelo de comunicación se describe la comunicación entre los agentes que participan en el sistema; y por último, en el modelo del diseño se especifica la arquitectura física de dicho sistema. (Henaó Cálad, 2001)

- Además de todo lo anterior, CommonKADS plantea una serie de consideraciones para la Gestión del Proyecto de Conocimiento (Project Management - PM), formada por cuatro actividades:
- Revisión. Se revisa el estado actual del proyecto y se definen los principales objetivos para el ciclo siguiente. Es el primer estado en la administración del proyecto de cada ciclo.
- Riesgo: Se identifican los obstáculos que pueden presentarse en el desarrollo del proyecto y que pueden impedir que se cumpla con los objetivos definidos. Se cuenta con el formulario PM-1: Identificación y valoración de los riesgos del proyecto.
- Plan. Se hace el plan detallado para el siguiente ciclo, en función del grado de finalización de uno o varios modelos (estados de los modelos). Se tiene el formulario PM-2: Cómo describir el estado del modelo como un objetivo a ser alcanzado en el proyecto.
- Seguimiento, para registrar los cambios y los resultados obtenidos.

Para cada una de estas actividades se ha definido una serie de documentos, reflejados en la Figura II. 12, que deben ser creados al comienzo del proyecto y completados durante el progreso de cada ciclo.



Figura II. 12. Documentación del ciclo de la gestión del proyecto en CommonKADS

fuelle: <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>

2.7.5 Ventajas y Desventajas de CommonKADS

Ventajas o Fortalezas de CommonKADS

Una de las principales cualidades de la metodología CommonKADS es el planteamiento del desarrollo de modelos que reflejan diferentes vistas del proyecto. Entre ellos se resalta el modelo de conocimiento en el que las partes que lo conforman son independientes del dominio, es decir que son genéricas y puede ser usadas en otros problemas o SBC que tengan características o comportamientos similares (los llamados modelos de interpretación).

La gestión del proyecto que se plantea en la metodología CommonKADS es un punto importante a destacar, ya que involucra aspectos administrativos que muchas veces no se toman en cuenta al desarrollar un sistema informático. Esto permite que los productos que resultan al aplicar la metodología, puedan ser valorados e integrados fácilmente en la Gerencia de la organización.

CommonKADS es importante porque ofrece un marco para la especificación del conocimiento independiente de la implementación, combinando un conjunto de modelos de conocimiento reutilizable para unas tareas que se realizan frecuentemente, como por ejemplo el diagnóstico o la planificación, entre otras. Además, propone un ciclo de vida en donde se muestran las fases, las actividades y los productos más relevantes para un proyecto de desarrollo de un SBC.

Adicionalmente, los modelos que propone la metodología permiten reflejar diferentes visiones fundamentales en el SBC, desde el punto de vista de la empresa hasta la forma como éste se diseña. Dentro de estos modelos, el de Agentes y el de Comunicaciones son particulares a CommonKADS, siendo quizá una de las debilidades de las otras metodologías, pues no consideran la comunicación con otros sistemas. Ambos modelos, le permiten al sistema coordinar las actividades con otros agentes y con ello ser más que un sistema basado en el conocimiento para ser un sistema que coopera con otros.

Otro aspecto para resaltar de esta metodología es el hecho de que es una de las más utilizadas para el desarrollo de SBC, tomándose incluso como el estándar europeo. En numerosas universidades y empresas europeas como bancos y compañías entre otras han realizado sus proyectos a través de las técnicas de CommonKADS.

Desventajas o Debilidades de CommonKADS

En general esta metodología cubre todos los aspectos que se necesitan para llevar a buen término un proyecto de desarrollo de un SBC, desde el estudio del problema hasta la implantación del software y su gestión. Los aspectos negativos que se presentan son más de su aplicación que de su conceptualización, porque aplicar lo definido en ella requiere de mucha experiencia y conocimiento de la misma metodología. Esto por varias razones:

- La metodología es muy compleja y amplia.
- Hay mucha información relevante que está en diversos sitios, lo que dificulta su acceso y comprensión.
- No hay una fuente de información que contenga todo lo necesario para su aplicación.
- No hay un ejemplo completo de la aplicación de la metodología que pueda ser utilizado como guía. Hay muchos ejemplos pero parciales.

2.8 Metodología Buchanan

2.8.1 Perspectiva Histórica

Uno de los primeros métodos de desarrollo estructurado de sistemas inteligentes fue el propuesto por Buchanan y otros autores en 1983. Según estos autores la adquisición del conocimiento de un sistema inteligente, y por extensión la construcción de todo el sistema, podía dividirse en las cinco fases de la figura II.13: identificación, conceptualización, formalización, implementación y prueba. (Moret Bonillo, 2005)

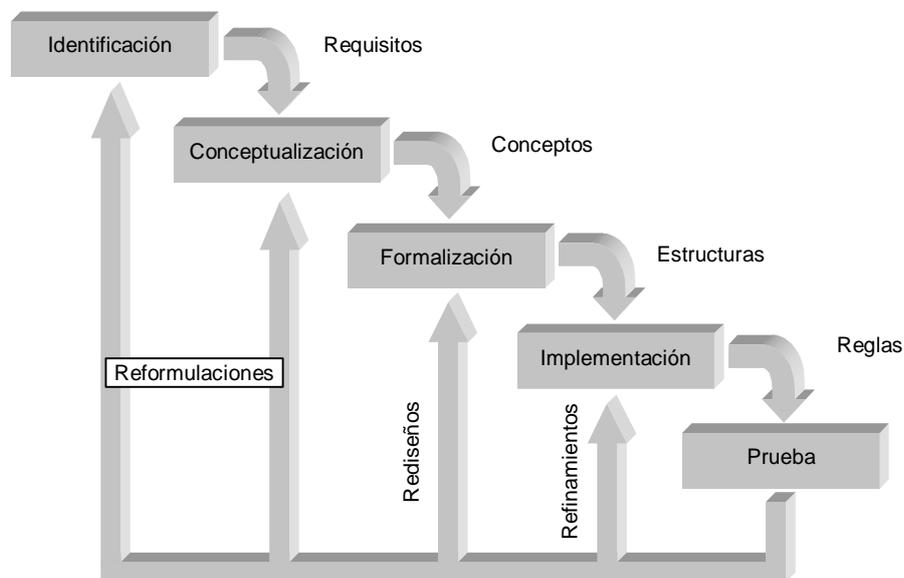


Figura II. 13. Ciclo de vida buchanan

fuelle: <http://www.dc.fi.udc.es/muc/sites/www.dc.fi.udc.es.muc/files/Val-y-U.s.doc>

Sin embargo el proceso real no está muy bien definido como puede sugerir la figura anterior, y más bien representa una aproximación a las distintas y complejas fases que se llevan a cabo a la hora de desarrollar un sistema inteligente, y que pueden variar de una situación a otra.

2.8.2 Etapas

2.8.2.1 Identificación

Fase mediante la que se reconocen aspectos importantes del problema, como son los participantes (expertos del dominio, ingenieros del conocimiento y futuros usuarios), las características del problema (tipo, subareas de que se compone, terminología a utilizar, aspectos fundamentales,

etc.), los recursos disponibles (fuentes de conocimiento, facilidades computacionales, tiempo de desarrollo, financiación, etc.), y las metas a alcanzar (formalizar conocimiento experto, distribuir experiencia, ayudar a la formación de nuevos expertos, etc.). (Moret Bonillo, 2005)

2.8.2.2 Conceptualización

Fase mediante la que se trata de organizar el conocimiento según un esquema conceptual. El experto y el ingeniero del conocimiento tratan de encontrar conceptos que representen el conocimiento del experto, al mismo tiempo que intentan determinar cómo es el flujo de información durante el proceso de resolución de problemas.

Formalización. Esta fase consiste en traducir los conceptos clave, los subproblemas, y las características del flujo de información, identificados durante la fase anterior, en representaciones formales basadas en herramientas o esquemas de la ingeniería del conocimiento. (Moret Bonillo, 2005)

2.8.2.3 Elicitación (Formalización)

Aunque no aparece en el trabajo original de Buchanan, es común incluir una fase de elicitación después de la fase de formalización. En esta fase se lleva a cabo la extracción del conocimiento mediante un soporte físico que es consistente con la información obtenida durante los procesos de identificación y conceptualización. (Moret Bonillo, 2005)

2.8.2.4 Implementación

En esta fase, el ingeniero de conocimiento formula reglas, y estructuras de control, que representan los conceptos y el conocimiento formalizado. El resultado es un programa prototipo que nos permite comprobar si hemos conceptualizado y formalizado bien el conocimiento que el experto tiene sobre el problema. (Moret Bonillo, 2005)

2.8.2.5 Prueba

Esta fase consiste en la evaluación del rendimiento del prototipo construido para encontrar errores o anomalías en la base de conocimientos o en los mecanismos de inferencia.

Buchanan sitúa los lazos de realimentación después de la fase de prueba, pero también indica que el proceso no tiene por qué seguir estrictamente la secuencia representada en la figura anterior.

Autores posteriores, como Mayrhauser, señalan que las retroalimentaciones pueden aparecer entre cualquier par de fases de la metodología. Así, por ejemplo, si el ingeniero del conocimiento no encuentra reglas adecuadas durante la implementación puede requerir una vuelta atrás y una reformulación del problema. La nueva representación del ciclo de vida de los sistemas inteligentes sería tal y como se presenta en la figura II.14 , una red completamente comunicada. (Moret Bonillo, 2005)

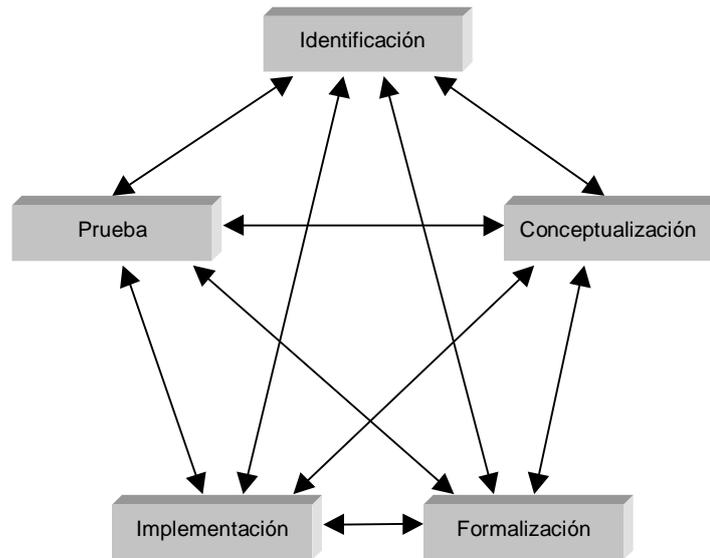


Figura II. 14. Red completamente comunicada

fuelle: <http://www.dc.fi.udc.es/muc/sites/www.dc.fi.udc.es.muc/files/Val-y-US.doc>

Las estructuras de este tipo son muy complejas de controlar y de manejar, ya que el número de iteraciones entre las fases es desconocido, y los objetivos pueden cambiar a medida que avanza el desarrollo. También es difícil llevar a cabo un control de los progresos realizados.

CAPITULO III

ANÁLISIS Y COMPARACIÓN DE METODOLOGÍAS

3.1 Introducción

En este capítulo se detalla el análisis comparativo de productividad de las metodologías CommonKADS y Buchanan utilizando varios indicadores que ayudarán a determinar cuál de las dos metodologías brinda la mejor productividad para el desarrollo del sistema experto, se presenta resultados de las pruebas de medición, y se efectúa un análisis cuantitativo y cualitativo de los resultados con el propósito de asignar un valor de medición a cada uno de los indicadores considerados en las pruebas para conseguir una calificación final de cada metodología.

3.2 Determinación de los Indicadores para la Comparación de Productividad

Los indicadores que se presentan a continuación se basan en los datos del proyecto en estudio y en el registro histórico de proyectos pasados, indicadores que han sido utilizados para el análisis de productividad en varias tesis desarrolladas en la Espoch.

Para determinar la mejor metodología para el desarrollo de un sistema experto se evaluará con los siguientes indicadores de la **Tabla III. III**, y cada indicador será dividido en porcentajes iguales para medir la productividad del sistema, a continuación se detalla.

Tabla III. III. Indicadores de evaluación de productividad

INDICADORES	DESCRIPCIÓN	% de valoración
Disponibilidad de Información	Permite valorar la existencia de información y conocimiento que se puede conseguir de cada metodología por medios accesibles, como Internet, libros y manuales	20%
Líneas de código	Indica el esfuerzo de desarrollo midiendo la cantidad de líneas de código empleadas para el desarrollo de las aplicaciones.	20%
Tiempo de desarrollo	Se refiere al tiempo (horas) requerido para la entrega del sistema, que necesita el programador para su respectiva ejecución.	20%
Esfuerzo de desarrollo	La cantidad de trabajo que el equipo de desarrolladores debe aplicar para el desarrollo del sistema estimado en horas-persona.	20%
Costo	Cantidad de recursos económicos necesarios para el desarrollo del sistema	20%
PRODUCTIVIDAD		100%

Realizado por: Fausto Llangari S.

La metodología que cumpla con la mayoría de los indicadores mencionados en la **Tabla III. III**, será considerada como ganadora. La fórmula propuesta por el investigador para seleccionar la mejor metodología está dado por el 20% de *Disponibilidad de Información*, que permite adquirir información existente en Internet, libros y manuales, más el 20% de *Líneas de código*, si se permite usar menor número de líneas de código para el desarrollo de la aplicación, más el 20% de *Tiempo de desarrollo*, si el tiempo requerido es menor para el desarrollo del sistema, más el 20% de *Esfuerzo de desarrollo* por la cantidad de trabajo estimado en horas-persona sea menor, y más el 20% del *Costo*, cuando los recursos económicos para el desarrollo del sistema sean menores.

$$\text{Productividad} = \text{Disponibilidad de Información} + \text{Líneas de código} + \text{Tiempo de desarrollo} + \text{Esfuerzo de desarrollo} + \text{Costo}$$

En la **TABLA III.IV**, se indica la valoración y los porcentajes que permitirán calificar los criterios de evaluación de cada uno de los indicadores de la **TABLA III.III**.

Tabla III. IV. Valoración y Porcentajes

INDICADOR	VALORACIÓN		(%)
	Cualitativo	Cuantitativo	
• Disponibilidad de Información	Muy bueno	3	20%
• Líneas de código	Bueno	2	10%
• Tiempo de desarrollo	Regular	1	5%
• Esfuerzo de desarrollo	Malo	0	0%
• Costo			

Fuente: Fausto Llangari S.

A continuación se menciona las **características y el criterio de evaluación** de cada uno de los indicadores de productividad.

3.2.1 Disponibilidad de Información

Permite valorar la existencia de información y conocimiento que se puede conseguir de cada metodología por medios accesibles, como Internet, libros y manuales. (JARAMILLO ESTRADA, 2013)

Criterios de evaluación

Este indicador se medirá guardando la cantidad de información disponible en los siguientes medios, como Internet, libros y manuales.

En la Tabla III. V, se establece posibles escalas para determinar la *Disponibilidad de Información*, planteado por el investigador del proyecto de tesis, y el porcentaje de valoración se indicada en la **Tabla III.IV**.

Tabla III. V. Criterios de evaluación de disponibilidad de información

Cuantitativa	0	1	2	3
Rango de información disponible	< 6612	>= 6612 y < 27050	>= 27050 y < 47040	>= 47040
Cualitativa	Malo	Regular	Bueno	Muy bueno
	Muy difícil	Difícil	Fácil	Muy fácil
	Muy poco	Algo	Bastante	Mucho
Porcentajes (%)	0%	5%	10%	20%

Fuente: Fausto Llangari S.

3.2.2 Líneas de Código

Indica el esfuerzo de desarrollo midiendo la cantidad de líneas de código empleadas para el desarrollo de las aplicaciones. (Samaniego Larrea, 2015)

Criterios de evaluación

Este indicador se medirá guardando la cantidad de líneas de código empleadas en el desarrollo de la aplicación.

En la Tabla III. VI, se establece posibles escalas para determinar *Líneas de código*, planteado por el investigador del proyecto de tesis, y el porcentaje de valoración se indicada en la **Tabla III. IV**.

Tabla III. VI. Criterios de evaluación de líneas de código

Cuantitativa	0	1	2	3
Rango de cantidad de líneas de código empleadas.	>3878	>2312 y <=3878	>2231 y <=2312	<=2231
Cualitativa	Malo	Regular	Bueno	Muy bueno
	Mucho	Bastante	Algo	Poco
Porcentajes (%)	0%	5%	10%	20%

Fuente: Fausto Llangari S.

3.2.3 Tiempo de Desarrollo

Este criterio se refiere al tiempo (horas) requerido para la entrega del sistema, que necesita el programador para su respectiva ejecución. (Jara Izurieta, 2014)

Criterios de evaluación

Para medir este indicador se guardara el tiempo en horas empleada para el desarrollo del sistema.

En la Tabla III. VII, se establece posibles escalas para determinar el *Tiempo de desarrollo*, planteado por el investigador del proyecto de tesis, y el porcentaje de valoración se indicada en la **Tabla III. IV**.

Tabla III. VII. Criterios de evaluación de tiempo de desarrollo

Cuantitativa	0	1	2	3
Rango en Tiempo (horas) empleado para el desarrollo.	>300h	>176h y <=300h	>128h y <=176h	<=128h
Cualitativa	Malo	Regular	Bueno	Muy bueno
	Mucho	Bastante	Algo	Poco
Porcentajes (%)	0%	5%	10%	20%

Fuente: Fausto Llangari S.

3.2.4 Esfuerzo de Desarrollo

La cantidad de trabajo que el equipo de desarrolladores debe aplicar para el desarrollo del sistema estimado en horas-persona. (Núñez Zamora & Sánchez Lozada, 2015)

Criterios de evaluación

Para medir este indicador se medirá el esfuerzo estimado en horas-persona para el desarrollo sistema.

En la Tabla III. VIII, se establece posibles escalas para determinar el *Esfuerzo de desarrollo*, planteado por el investigador del proyecto de tesis, y el porcentaje de valoración se indicada en la **Tabla III. IV**.

Tabla III. VIII. Criterios de evaluación de Esfuerzo de desarrollo.

Cuantitativa	0	1	2	3
Rango de Esfuerzo (horas/persona)	>105h	>96h y <=105h	>88h y <=96h	<=88h
Cualitativa	Malo	Regular	Bueno	Muy bueno
	Mucho	Bastante	Algo	Poco
Porcentajes (%)	0%	5%	10%	20%

Fuente: Fausto Llangari S.

3.2.5 Costo

Cantidad de recursos económicos necesarios para el desarrollo del sistema. (Hidalgo & Casigña, 2014)

Criterios de evaluación

Para medir este indicador se conocerá la cantidad de recursos económicos necesarios para el desarrollo del sistema.

En la Tabla III. IX, se establece posibles escalas para determinar el *Costo*, planteado por el investigador del proyecto de tesis, y el porcentaje de valoración se indicada en la **Tabla III. IV**.

Tabla III. IX. Criterios de evaluación del costo

Cuantitativa	0	1	2	3
Rango de cantidad de recursos económicos(\$)necesarios	>\$300	>\$228 y <=\$300	>\$180h y <=\$228	<=\$180
Cualitativa	Malo	Regular	Bueno	Muy bueno
	Mucho	Bastante	Poco	Muy Poco
Porcentajes (%)	0%	5%	10%	20%

Fuente: Fausto Llangari S.

3.3 Metodologías a evaluar

Las metodologías de desarrollo de software son un conjunto de procedimientos, técnicas y ayudas a la documentación para el desarrollo de productos software, las metodologías seleccionadas para ser evaluadas en la presente investigación son:

- Metodología CommonKADS
- Metodología Buchanan

3.4 Desarrollo de Prototipos

3.4.1 Aplicación Metodología CommonKADS

3.4.1.1 Modelo de la Organización

En este primer modelo OM-1 se determinan las principales características de la organización, se descubren problemas y oportunidades del sistema basado en conocimiento.

Formulario OM - 1

Modelo de la Organización

Problemas

- La determinación de un cultivo a ser sembrado se la realiza en forma empírica.
- Carencia de personal capacitado para brindar asesoramiento
- Sobreproducción de productos
- Escases de productos
- Falta de planificación para establecer un calendario de siembra durante el año

- Alto riesgo de temporal de cultivo y procesos de mercado
- Baja tecnificación
- Falta de asesoría técnica dedicada a la elección de cultivo

Oportunidades

- Demanda de asesoramiento de ing. agrónomo o agropecuario.
- Aprovechamiento de los recursos agrícolas del sector
- Gran cantidad de agricultores que quieren hacer un uso correcto de sus recursos agrícolas
- Deseo de la organización por mejorar la calidad de atención al usuario.

Solución 1:

Desarrollo de un Sistema Experto que dará soporte en la selección mas idónea de un cultivo a ser sembrado basado en el ciclo de producción capacidad de inversión y características del lugar a ser sembrado.

Solución 2:

Contratar un experto agrónomo.

Formulario OM - 2

Aspectos variables

Tabla III. X. Aspectos variables

OM-2	Aspectos Variables
Estructura	Organigrama figura No III. 15
Procesos	(A) Atención al Usuario
Personas	(1) Ing Agronomo (2) Usuario
Recursos	Materiales: GPS Barreno Termómetro Pluviómetro

	Cinta
Conocimiento	Conocimiento del Agrónomo Conocimiento de Riego Conocimiento de Fertilización Conocimiento Agro meteorológico Conocimiento Fitosanitario del Cultivo

Realizado por: Fausto Llangari

Estructura de la organización

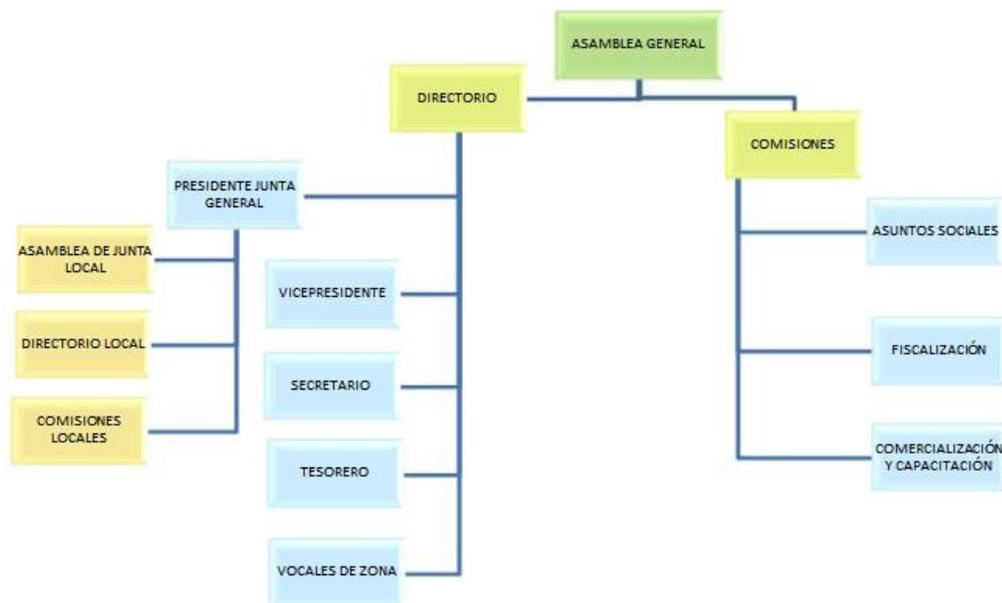


Figura III. 15. Estructura de la organización

fuelle:Jurech

La Junta de riego Chambo – Guano, orientada a la administración del riego para las comunidades de los cantones Chambo, Guano y sus alrededores está representada a cargo de un Presidente de la Junta; y está conformado por diferentes áreas como Directorio , Comisiones , Asambleas; se sitúan en la atención a los Usuarios.

Procesos

En la figura III.16 se muestra los procesos

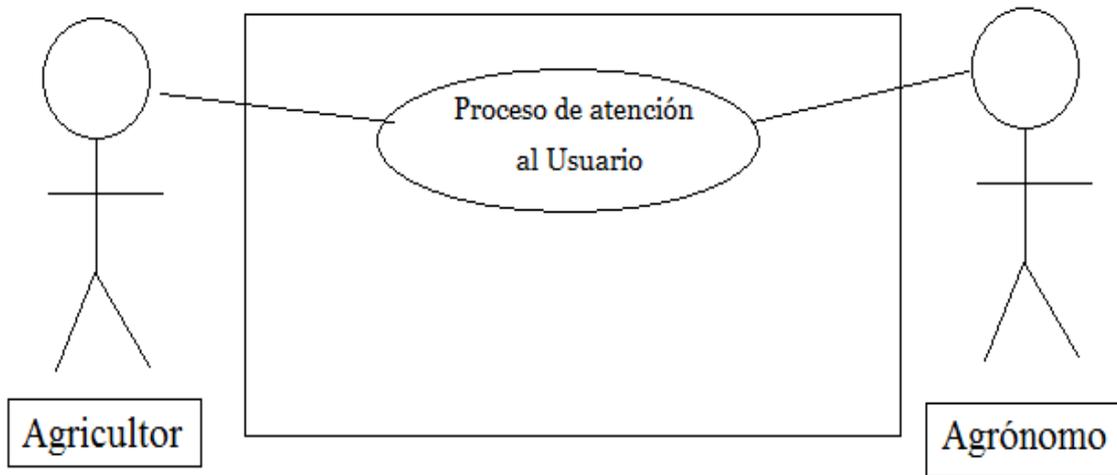


Figura III. 16. Proceso – Proceso de atención al Usuario

Fuente: Fausto Llangari

Formulario OM - 3

DESCOMPOSICION DE PROCESOS

Tabla III. XI. Atención al Usuario

Modelo Organizacional		Atención al Usuario				
Nº	Tarea	Agente	donde	Recursos de conocimiento	Intensiv o	Importan cia
1.1	Obtener la petición del Usuario:	Agrónomo - Agricultor	Oficina Jurech	si	Alta
	Sugerencia para cultivo	Agricultor - agrónomo	oficina	-Conocimientos del agrónomo -conocimientos de mercado	Si	Alta

Realizado por: Fausto Llangari

3.4.1.2 Modelo de Tareas

Describe las tareas que realizan los agentes de la Organización

Formulario TM - 1

Análisis y descripción de las tareas dentro del proceso

Tabla III. XII. Análisis de las tareas - Sugerencia para cultivo

Tarea	1.1 Obtener petición del Usuario
Organización	Comisiones Locales
Objetivos y Valores	Conocer la situación del usuario y su petición.
Objetos Manejados	Entrada: petición del Usuario (sacar ticket) Salida : obtener la información del Usuario
Temporización y Control	Frecuencia : Uno por cada Usuario. Duración : Poca Precondición : Estar disponible Agrónomo Pos-condición: la petición ha sido identificada y almacenada.
Agentes	Usuario Agrónomo
Conocimientos y Competencias	Conocimiento de la secretaria de la junta
Recursos	Agrónomo utiliza: Una Computadora, papel.
Plantilla TM-1 Análisis de Tareas	

Realizado por: Fausto Llangarí

Tabla III. XIII. Análisis de las tareas-obtener información del Usuario

Tarea	1.2 Sugerencia para cultivo
Organización	Comisiones Locales
Objetivos y Valores	Realizar la sugerencia del cultivo más idóneo de acuerdo a los requerimientos del Usuario
Objetos Manejados	Entrada : Petición de Usuario Salida : Cultivo recomendado
Temporización y Control	Frecuencia: Uno por Usuario. Duración : Media. Precondición: que el Agrónomo disponga de la información del Usuario Pos-condición: que se registren toda la información del Usuario.
Agentes	Usuario, Agrónomo
Conocimientos y Competencias	Conocimiento del Agrónomo Conocimiento de Riego Conocimiento de Fertilización Conocimiento Agro meteorológico Conocimiento Fitosanitario del Cultivo.
Recursos	GPS, Barreno, Termómetro, Pluviómetro, Cinta, computadora
Calidad y Rendimiento	Que el Ing Agrónomo reciba correctamente la información y realice el análisis adecuado
Plantilla TM-1 Análisis de Tareas	

Realizado por: Fausto Llangarí

3.4.1.3 Modelo de Agentes

Los Agentes son los ejecutores de una tarea. El modelo de agente describe las características de los agentes, sus competencias, autoridad y restricciones.

Formulario AM-1:

Modelo de Agentes

Tabla III. XIV. Modelo de agentes- Usuario

Modelo de Agentes	Usuario
Nombre	Usuario
Organización	Tipo : Humano.
Implicado en	Realizar la consulta al agrónomo 1.2. Proporcionar datos sobre su situación
Se comunica con	Agrónomo
Plantilla AM-1 Especificación de Agentes	

Realizado por: Fausto Llangarí

Tabla III. XV. Modelo de agentes- Agrónomo

Modelo de Agentes	Agente
Nombre	Agrónomo
Organización	Tipo : Humano. Estructura : Comisiones locales
Implicado en	1.1. Obtener petición del Usuario
Se comunica con	Usuario
Conocimiento	Conocimiento del Agrónomo Conocimiento de Riego Conocimiento de Fertilización Conocimiento Agro meteorológico Conocimiento Fitosanitario del Cultivo
Otras competencias	
Responsabilidades y Restricciones	Responsabilidades: Analizar la situación actual del terreno y su propietario así como sus requerimientos.
Plantilla AM-1 Especificación de Agentes	

Realizado por: Fausto Llangarí

3.4.1.4 Modelo de Conocimiento

En este modelo se desarrollan las tres principales categorías del conocimiento. El conocimiento del dominio, el conocimiento de inferencias y el conocimiento de tareas.

CM1 (Conocimiento del dominio)

Se representa el conocimiento relevante del sistema, sobre el que se desarrolla el proceso de razonamiento objeto del modelado. Una explicación básica de los elementos que componen el dominio del sistema fueron desarrollados.

CM2 (Especificación del esquema del dominio)

En esta fase se aborda la adquisición del conocimiento específicos del dominio concreto de la aplicación. A partir de las entrevistas iniciales con el experto, se llegó a la conclusión de que la manera más eficiente de adquirir el conocimiento era centrar dichas entrevistas en torno a formularios usadas por el especialista.

Diagrama de conceptos: Un concepto se utiliza para definir un conjunto de objetos o instancias utilizado en el dominio de la aplicación.

DIAGRAMA

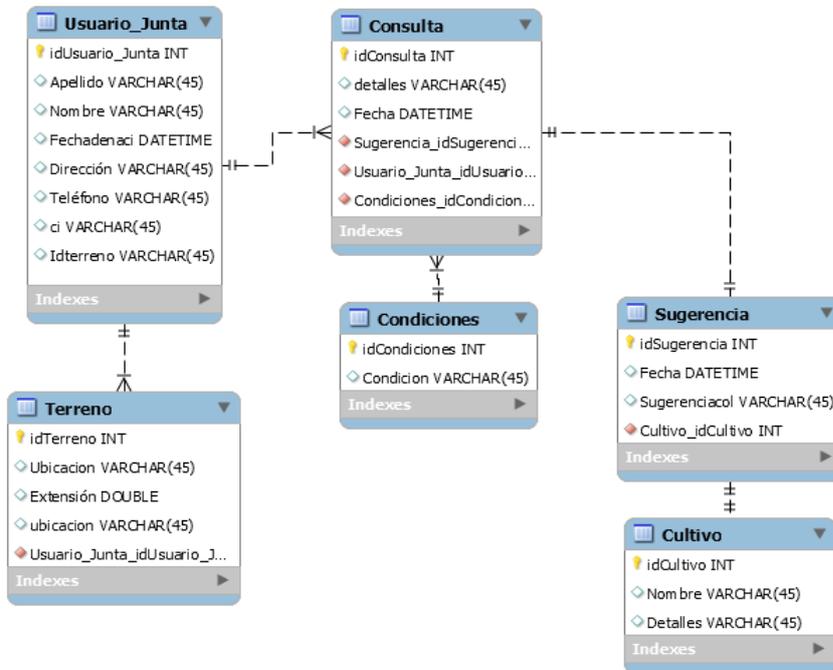


Figura III. 17. Diagrama de conceptos

Fuente: Fausto Llangari

Conceptos

Usuario_Junta: Es la clase que contiene los atributos principales del Usuario.

Terreno: Es la clase que contiene los atributos principales del Terreno.

Sugerencia: Contiene la sugerencia realizada por el Agrónomo

Cultivo: Contiene los atributos de un cultivo.

Consulta: Es la clase que contiene la fecha de atención y detalles de consulta

Condiciones: Es la clase que contiene como atributos las condiciones que determinan a la sugerencia del experto.

Representación de los conceptos

CONCEPT Usuario_Junta:

ATTRIBUTES:

Idusu:int;

Apellido: varchar;

Nombre: varchar;

FechaNaci:integer;

Dirección: varchar;

Teléfono: varchar;

ci: varchar;

Idterreno:int;

END CONCEPT Usuario_Junta;

CONCEPT Terreno;

ATTRIBUTES:

Idterreno:int;

Ubicacion:varchar;

Extensión:doublé;

Idusuario:int;

END CONCEPT Terreno;

CONCEPT Sugerencia;

ATTRIBUTES:

Idsug:int;

Cultivo:varchar;

Fecha:datetime;

END CONCEPT Sugerencia;

CONCEPT Cultivo;

ATTRIBUTES:

Idcul:int;

Nombre:varchar;

detalles:varchar;

END CONCEPT Cultivo;

CONCEPT Consulta;

ATTRIBUTES:

Idcons:int;

detalles:varchar;

Fecha:datetime;

END CONCEPT Consulta;

CONCEPT Condiciones;

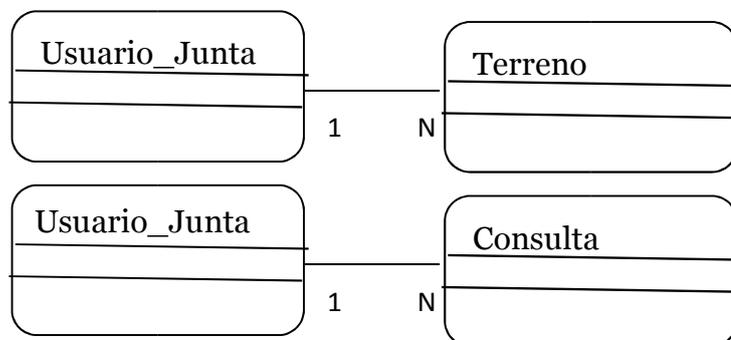
ATTRIBUTES:

Idcond:int;

:varchar;

END CONCEPT Condiciones;

Relaciones: Mediante este tipo de construcción podemos definir relaciones entre conceptos definidos en un esquema del dominio determinado.



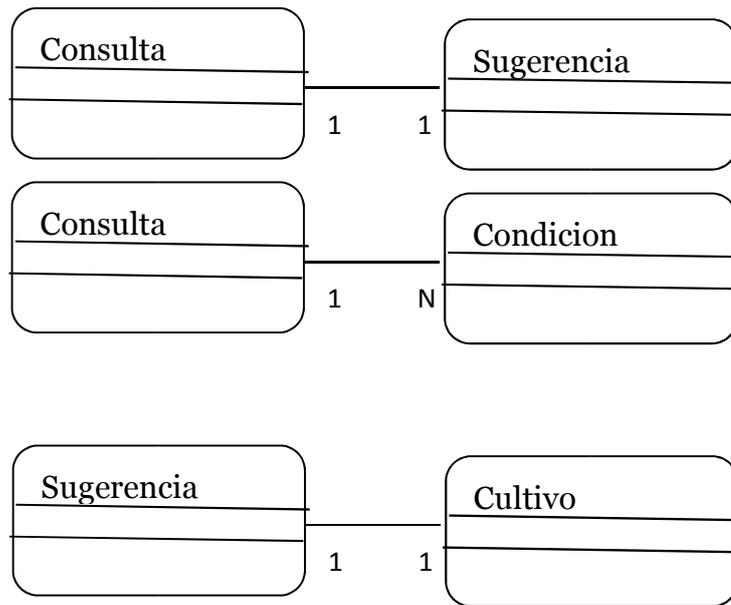


Figura III. 18. Relación de conceptos

Fuente: Fausto Llangari

Tipos generadores de reglas: Nos permite definir relaciones entre dos expresiones lógicas definidas en función de un valor de un atributo de un concepto, en el que en vez de tomar como argumentos determinados conceptos del esquema los argumentos son expresiones sobre conceptos. Para definir un tipo de reglas necesitamos un antecedente y un consecuente en los que indicamos los conceptos sobre cuyas instancias se definen las expresiones lógicas. Para la formulación de las reglas se toma en cuenta la colaboración de un experto con conocimientos amplios sobre cultivos. Por esta razón el sistema diseñado debe ser capaz de ofrecer como salida una sugerencia que corresponda con los valores de las variables de entrada y el criterio del experto.

Nivel Dominio

Para representar el conocimiento abstracto se hará uso del nivel dominio mediante el cual se obtienen las reglas. La figura III.19 muestra una parte del razonamiento que se sigue para determinar la sugerencia más adecuada para un cultivo.

Esquema de nivel de dominio

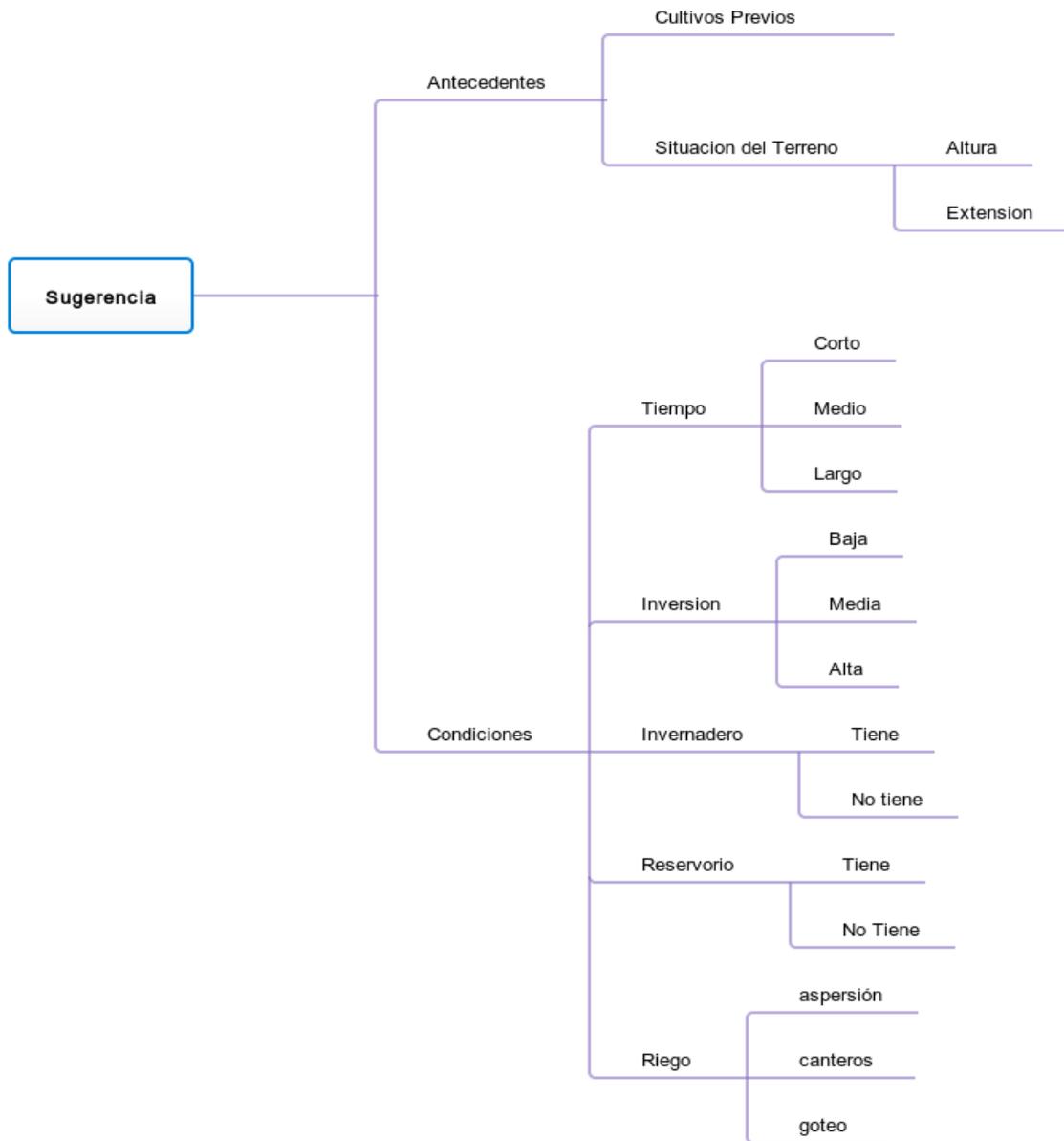


Figura III. 19. Esquema de nivel del dominio

Fuente: Fausto Llangari

3.4.1.5 Modelo de Comunicación

En este modelo se representan los resultados de la ejecución de una tarea por parte de un agente como objetos de información los cuales debe suministrar a otros agentes.

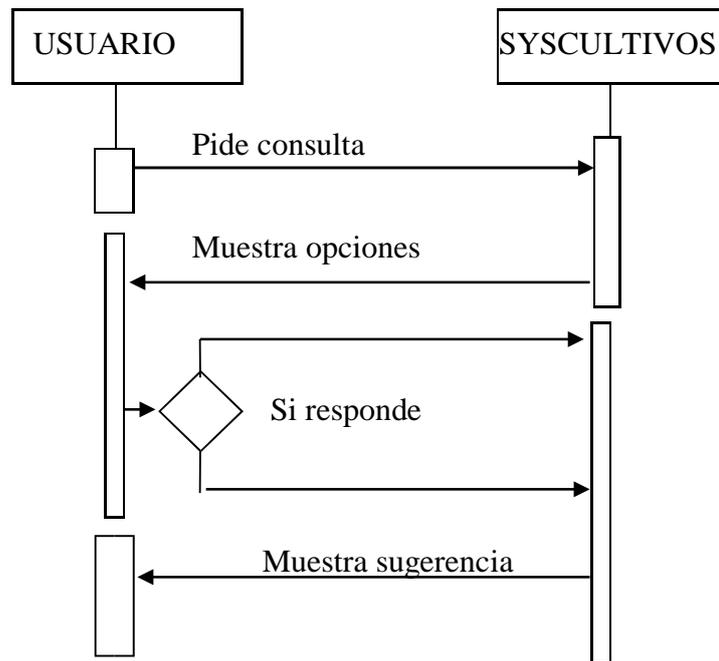


Figura III. 20. Modelo de comunicación

Fuente: Fausto Llangari

En este modelo se representan los resultados de la ejecución de una tarea por parte de un agente como objetos de información los cuales debe suministrar a otros agentes.

Modelo de Comunicación. CM-1

Tabla III. XVI. Modelo de comunicación

Modelo de comunicación	Hoja de trabajo CM-1
Nombre	Pide consulta
Información	Opciones seleccionadas
Involucrados	Usuario - Agente experto
Restricciones	El usuario debe seleccionar alguna de las opciones asociados a su situación actual para que el sistema experto actuara.

Especificación Del intercambio de información	Esta transición está compuesta de un mensaje lo cual es del tipo seleccionar opciones presentadas y lo envía al sistema Experto para que realice una sugerencia de cultivo.
---	---

Realizado por: Fausto Llangarí

3.4.1.6 Modelo de Diseño

En este modelo especificamos la arquitectura del sistema, la plataforma de implementación y los requerimientos para su construcción.

Arquitectura de Sistemas. DM-1

Tabla III. XVII. modelo de diseño

Modelo de diseño	Arquitectura del Sistema : DM-1
Modelo de control	El agente humano (usuario) selecciona las opciones para el cultivo donde el sistema Experto empieza su intervención para la sugerencia del cultivo.
Descomposición de subsistemas	No tiene

Realizado por: Fausto Llangarí

A continuación mostramos la plataforma de implementación y los requerimientos para la construcción del prototipo

Tabla III. XVIII. Especificación de la plataforma de implantación

Modelo de diseño	Hoja de trabajo. Especificación de la Plataforma de Implantación
Software	Microsoft Visual Estudio WAMP SERVER V2.0 SqlYog
Hardware	Computadora Personal
Lenguajes de programación	.Net, MYSQL
Representación del conocimiento	En base a reglas

Realizado por: Fausto Llangarí

PROTOTIPO DEL AGENTE EXPERTO

Diagrama de Proceso del Sistema experto

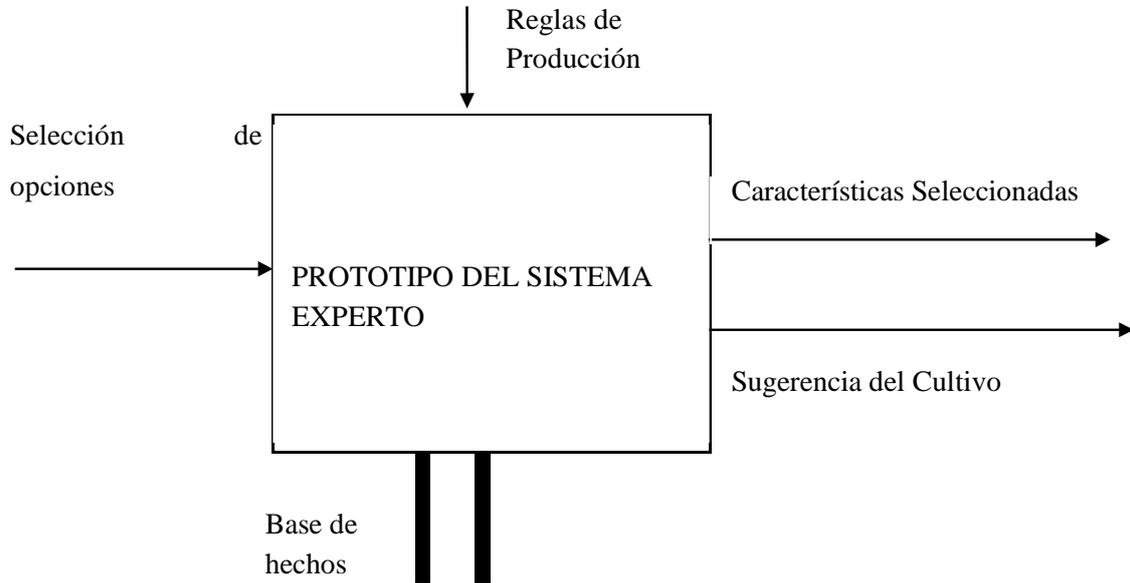


Figura III. 21. Diagrama de proceso del sistema experto

Fuente: Fausto Llangari

3.4.2 Aplicación Metodología Buchanan

3.4.2.1 Identificación

Problema

No se cuenta con un experto en cultivos para asesorar a los usuarios de la junta de riego CHAMBO-GUANO en la correcta selección del cultivo a ser sembrado.

Solución

Se propone desarrollar un Sistema Experto Probabilístico que sirva de apoyo a dichas selecciones con el fin de lograr una mejor producción.

Familiarización con el dominio

Para familiarizarse con el problema y el dominio se realizaron entrevistas al experto para validar la información encontrada y enriquecer la misma. Una vez identificados el problema y el dominio se pasa a identificar las tareas del Sistema Experto.

Tareas del Sistema Experto:

- Permitir ingresar los antecedentes para crear la Base de Conocimiento.
- Brindar una opción del cultivo más óptimo expresado en porcentaje.
- Permitir el ingreso de nuevo conocimiento a la Base de Conocimientos inicial.
- Permitir editar el conocimiento almacenado en la Base de Conocimientos.
- Una vez identificadas las tareas que realizará el Sistema Experto, se pasa a delimitar el sistema.

3.4.2.2 Conceptualización

En esta fase se procederá a la adquisición del conocimiento del sistema experto; en este caso se necesita obtener la información cualitativa y cuantitativa del modelo probabilístico que utilizaremos.

Información cualitativa del modelo (Variables)

Variables objetivo:

- Cultivo sugerido

Variables de observación:

- Propietario
- Ciclo del cultivo
- Capacidad de inversión
- Invernadero
- Reservorio
- Tipo de riego

3.4.2.3 Formalización

- R1: Si el usuario requiere un cultivo de ciclo corto, entonces puede sembrar Rábano, Cilantro, Fréjol, Sandía, Brócoli, Melón, Lechuga, Cebolla Puerro, Zuquini, Sambo, Col, Coliflor, Papa, Pimiento o Tomate Riñón
- R2: Si el usuario requiere un cultivo de ciclo medio, entonces puede sembrar Pastos, Quinoa, Zanahoria, Remolacha, Arveja, Avena, Alfalfa, Cebolla colorada, Col morada, Ajo, Fresa, Babaco o Maíz
- R3: Si el usuario requiere un cultivo de ciclo largo, entonces puede sembrar Mora de Castilla o Tomate de Árbol
- R4: Si el usuario desea realizar una inversión baja, entonces puede sembrar sembrar Rábano, Quinoa, Pastos, Zanahoria, Cilantro, Fréjol, Zuquini, Sambo, Remolacha o Arveja
- R5: Si el usuario desea realizar una inversión media, entonces puede sembrar Col, Avena, Sandía, Brócoli, Alfalfa, Coliflor, Melón, Cebolla colorada, Maíz, Lechuga, Papa, Col morada o Tomate de Árbol
- R6: Si el usuario desea realizar una inversión alta, entonces puede sembrar Mora de Castilla, Ajo, Cebolla Puerro, Fresa, Babaco, Pimiento o Tomate Riñón
- R7: Si el usuario tiene invernadero, entonces puede sembrar Sandía, Melón, Pimiento o Tomate Riñón
- R8: Si el usuario tiene reservorio, entonces puede sembrar Fresa o Tomate Riñón
- R9: Si el usuario tiene riego por aspersión, entonces puede sembrar Quinoa, Pastos, Cilantro Avena Alfalfa Fréjol Zuquini o Sambo

- R10: Si el usuario tiene riego por canteros,
entonces puede sembrar Cilantro, Avena, Alfalfa, Fréjol, Zuquini, Sambo, Rábano, Zanahoria,
Remolacha, Arveja, Col, Sandia, Brócoli, Coliflor, Melón, Cebolla colorada, Maíz, Lechuga,
Papa, Col morada, Tomate de Árbol, Mora de Castilla, Ajo, Cebolla Puerro, Babaco, Pimiento o
Tomate Riñon

- R11: Si el usuario tiene riego por goteo,
entonces puede sembrar Fréjol, Zuquini, Sambo, Zanahoria, Remolacha, Arveja, Col, Sandia,
Brócoli, Coliflor, Melón, Cebolla colorada Maíz, Lechuga, Papa, Col morada, Tomate de Árbol,
Mora de Castilla, Ajo, Cebolla Puerro, Babaco, Pimiento, Tomate Riñon o Fresa

Base de Conocimiento:

R1: [cc] \Rightarrow (Ra v Ci V Fr V Sa V Br V Me V Le V Cp V Zu V Sb V Co V Cf V Pa V Pi V Tr)

R2: [cm] \Rightarrow (Pt V Qu V Za V Re V Ar V Av V Al V Ce V Cm V Aj V Fr V Ba V Ma)

R3: [cl] \Rightarrow (Mc V Ta)

R4: [ib] \Rightarrow (Co V Av V Sa V Br V Al V Cf V Me V Ce V Ma V Le V Pa V Cm v Ta)

R5: [im] \Rightarrow (Co v Av v Sa v Br v Al v Cf v Me v Ce v Ma v Le v Pa v Cm v Ta)

R6: [ia] \Rightarrow (Mc v Aj v Cp v Fr v Ba v Pi v Tr)

R7: [in] \Rightarrow (Sa v Me v Pi v Tr)

R8: [re] \Rightarrow (Fr v Tr)

R9: [ra] \Rightarrow (Qu v Pa v Ci v Av v Al v Fj v Zu v Sb)

R10: [rc] \Rightarrow (Ci v Av v Al v Fj v Zu v Sb v Ra v Za v Re v Ar v Co v Sa v Br v Cf v Me v Ce v
Ma v Le v Pa v Cm v Ta v Mc v Aj v Cp v Ba v Pi v Tr)

R11: [rg] \Rightarrow (Fj v Zu v Sb v Za v Re v Ar v Co v Sa v Br v Cf v Me v Ce v Ma v Le v Pa v Cm v
Ta v Mc v Aj v Cp v Ba v Pi v Tr v Fr)

Haciendo uso de lógica de predicados se hará la representación de la base de conocimiento.

La inferencia se llevará a cabo haciendo uso de El Modus Ponens.

Motor de Inferencia:

MODUS PONENS:

El Modus Ponens es quizás la regla de inferencia más comúnmente utilizada. Se utiliza para obtener conclusiones simples.

En ella, se examina la premisa de la regla, y si es cierta, la conclusión pasa a formar parte del conocimiento, supóngase que se tiene la regla.?

SI A es cierto, ENTONCES? B es cierto"

y que se sabe además que ?A es cierto".

Se concluye que? B es cierto?

INFERENCIA POR ENCADENAMIENTO DE REGLAS:

Esta estrategia de inferencia puede utilizarse cuando las premisas de ciertas reglas coinciden con las conclusiones de otras. Cuando se encadenan las reglas, los hechos pueden utilizarse para dar lugar a nuevos hechos. Esto se repite sucesivamente hasta que no pueden obtenerse más conclusiones.

El tiempo que consume este proceso hasta su terminación depende, por una parte, de los hechos conocidos, y por otra, de las reglas que se activan.

Este algoritmo puede ser implementado de muchas formas. Una de ellas comienza con las reglas cuyas premisas tienen valores conocidos. Estas reglas deben concluir y sus conclusiones dan lugar a nuevos hechos. Estos nuevos hechos se añaden al conjunto de hechos conocidos (Base de Conocimiento), y el proceso continúa hasta que no pueden obtenerse nuevos hechos.

3.4.2.4 Implementación

Especificación de los Casos de Uso del Sistema Experto

1. Registrar

Tabla III. XIX. Caso de Uso Registrar

Caso de uso	Registrar
Código	CU01
Objetivo	• Permite al usuario registrar nuevos (conocimiento) al sistema para mejorar la probabilidad de sugerencia del mejor cultivo
Precondiciones	El usuario debe de estar logueado.
Post condiciones	• Los datos quedan registrados en el sistema.

Actores	Usuario	
Flujo principal	Pasos	
	1.	El usuario hace click en el menú Registrar condición.
	2.	El sistema muestra el formulario de registro.
	3.	El usuario ingresa la descripción, abreviación, probabilidades y luego hace click en botón Guardar.
4.	El sistema guarda la condición y muestra el mensaje: “La condición ha sido creada con éxito”.	
Extensiones	Acción	
	4.	Condición ya existe: El sistema valida si la condición (mediante la abreviación) ya fue registrada anteriormente.
Frecuencia	Poco Frecuente	
Performance	Alta	
Prioridad	Alta	

Realizado por: Fausto Llangarí

2. Consultar Condición

Tabla III. XX. Caso de Uso Consultar Condición

Caso de uso	Consultar Condición	
Código	CU02	
Objetivo	<ul style="list-style-type: none"> Permite al usuario conocer las condiciones que han sido agregados en el sistema con sus respectivos atributos. 	
Precondiciones	<p>El usuario debe de estar logueado.</p> <p>El usuario accede a las opciones del Sistema de acuerdo al perfil del usuario.</p>	
Post condiciones	No aplica	
Actores	Usuario	
Flujo principal	Pasos	
	1.	El usuario hace click en el menú Consultar condición
	2.	El sistema muestra una todas las condiciones ingresados hasta ahora.

Extensiones	Acción	No aplica
Frecuencia	Frecuente	
Performance	Alta	
Prioridad	Alta	

Realizado por: Fausto Llangari

3. Editar

Tabla III. XXI. Caso de Uso Editar

Caso de uso	Editar Condición	
Código	CU03	
Objetivo	Permite al usuario editar las probabilidades de las condiciones (conocimiento), con el fin de mejorar el resultado de la sugerencia.	
Precondiciones	<ul style="list-style-type: none"> Realizado el CU04 	
Post condiciones	<ul style="list-style-type: none"> Los datos de la condición quedan actualizados en el sistema 	
Actores	<ul style="list-style-type: none"> Usuario 	
Flujo principal	Pasos	
	1.	El usuario hace click en el botón Editar.
	2.	El sistema muestra el formulario para editar.
	3.	El usuario Edita los campos deseados y da click en el botón actualizar
	4.	El sistema guarda la condición y muestra el mensaje: "La condición ha sido actualizada con éxito"
Extensiones	Acción	No aplica
	4.	
Frecuencia	Poco Frecuente	
Performance	Alta	
Prioridad	Alta	

4. Obtener Sugerencia

Tabla III. XXII. Caso de Uso Obtener Sugerencia

Caso de uso	Obtener Sugerencia
-------------	--------------------

Código	CU04	
Objetivo	Permite realizar sugerencia del cultivo al usuario, mediante la elección de las condiciones seleccionadas.	
Precondiciones	El usuario debe estar logueado. El usuario accede a las opciones del Sistema de acuerdo al perfil del usuario. Debe existir por lo menos una condición registrada.	
Post condiciones	No aplica	
Actores	• Usuario	
Flujo principal	Pasos	
	1.	El usuario hace click en el botón Editar.
	2.	El sistema muestra el formulario de selección, con la lista de preguntas.
	3.	El usuario elije el/los condiciones
	4.	El sistema muestra la sugerencia de cultivo, mediante un mensaje.
Extensiones	Acción	
	2.	No existen condiciones: El sistema no retorna nada, ya que no existe ninguna condición registrada.
Frecuencia	Frecuente	
Performance	Alta	
Prioridad	Alta	

3.5 Análisis comparativo

3.5.1 Disponibilidad de Información

Las pruebas en este indicador se realizaron basándose en la información existente en los medios como: Libros, Internet, manuales, Tesis, documentos y documentación oficial.

Internet: La búsqueda de información disponible sobre conceptos de CommonKADS y Buchanan se obtuvo mediante el buscador Web más utilizado, Google. Se realizó la búsqueda en la web, tomando como referencia la suma de todos los promedios para la obtención de los resultados. El máximo valor es 39.800 de información disponible en Internet de CommonKADS, y 6.210 de Buchanan (ver Anexo A – Sección 1).

Libros, Manuales y documentos: Se efectuó una búsqueda de libros y temas afines en Google books, (ver Anexo A – Sección 1). El máximo valor es 7.230 de manuales y documentos de CommonKADS, y 402 de Buchanan (ver Anexo A – Sección 1).

Documentación oficial: La documentación oficial se ha analizado obteniendo un total de los resultados encontrados en los enlaces oficiales de cada metodología. El máximo valor es 10 que pertenece a CommonKADS, Tomando en cuenta todos los valores obtenidos (ver Anexo A – Sección 1), el resultado final de documentación disponible:

- En la metodología CommonKADS, el promedio de información disponible como máximo valor es 47040 links de información (ver Anexo A – Sección 1).
- En la metodología Buchanan, el promedio de información disponible como máximo valor es 6612 links de información (ver Anexo A – Sección 1).

Los resultados de cada metodología (*Commonkads* y *Buchanan*) serán valorados de acuerdo al criterio de evaluación de disponibilidad de información planteado en la **Tabla III.XXIII**. A continuación en **Tabla III. XXIII** se detalla los resultados obtenidos.

Tabla III. XXIII. Evaluación del indicador disponibilidad de información

INDICADOR	METODOLOGÍAS			
	COMMONKADS		BUCHANAN	
	Valor	Equivalencia sobre 20%	Valor	Equivalencia sobre 20%
Disponibilidad de información	3	20%	1	5%
Total de Disponibilidad de información sobre 20%	3	20%	1	5%

Fuente: Fausto Llangari S.

A continuación los resultados de la **Tabla III. XXIII** se explican en la **Figura III.22**.



Figura III. 22. Resultado del indicador disponibilidad de información

Fuente: Fausto Llangari S.

Como se observa en la Figura III.22, la metodología *Commonkads* tiene un porcentaje de 20% de disponibilidad de información sobre el 20% del indicador (ver la Tabla III. III), Por la información encontrada en los libros, Internet y manuales disponibles de una manera numerosa, es decir el promedio máximo fue 47040 links de información, por lo tanto tiene una calificación cualitativa Muy bueno, mientras que la metodología *Buchanan* tiene el 5% debido a que la información encontrada fue menor, es decir apenas 6612 links de información, por lo tanto tiene una calificación cualitativa Regular.

Se concluye que la metodología *Commonkads* tiene mayor disponibilidad de información que la metodología *Buchanan*, como se indica en la Figura III.22.

3.5.2 Líneas de Código

Para las pruebas de este indicador se utilizó el programa Visual Estudio 2012, y gracias a la herramienta proporcionada en el mismo programa en el Menú “Analizar” con la opción “Calcular métricas de código para la solución” se obtuvo la cantidad de líneas de código empleadas en el desarrollo de la aplicación.

- El prototipo desarrollado con *CommonKADS*, requirió un total de 2231 líneas.
- El prototipo desarrollado con *Buchanan*, requirió de 2312 líneas.

Los resultados de cada metodología (*Commonkads* y *Buchanan*) serán valorados de acuerdo al criterio de evaluación de líneas de código planteado en la Tabla III. VI. A continuación en Tabla III. XXIV se detalla los resultados obtenidos.

Tabla III. XXIV. Evaluación del indicador líneas de código

INDICADOR	METODOLOGÍAS			
	COMMONKADS		BUCHANAN	
	Valor	Equivalencia sobre 20%	Valor	Equivalencia sobre 20%
Líneas de código	3	20%	2	10%
Total de líneas de código sobre 20%	3	20%	2	10%

Fuente: Fausto Llangari S.

A continuación los resultados de la Tabla III. XXIV se explican en la Figura III.23.



Figura III. 23. Resultado del indicador Líneas de código

Fuente: Fausto Llangari S.

Como se observa en la Figura III.23, la metodología *Commonkads* tiene un porcentaje de 20% de Líneas de código sobre el 20% del indicador (ver la Tabla III. III), debido a la cantidad de líneas de código empleadas en el desarrollo de la aplicación fue de 2231, por lo tanto tiene una calificación cualitativa Muy bueno, mientras que la metodología *Buchanan* tiene el 10% debido a la cantidad de líneas de código empleadas en el desarrollo de la aplicación fue mayor, con 2312 líneas, por lo tanto tiene una calificación cualitativa Bueno.

Se concluye que la aplicación desarrollada con la metodología *Commonkads* tiene menor Líneas de código que la metodología *Buchanan*, como se indica en la Figura III.23.

3.5.3 Tiempo de Desarrollo

Para las pruebas en este indicador se contabilizó el tiempo en *horas* empleadas para la entrega del sistema.

- Con la metodología *CommonKADS*, el tiempo empleado para el desarrollo del prototipo fue de 22 días, 176 horas laborables.
- Con la metodología *Buchanan*, se empleó 16 días, 128 horas para el desarrollo del prototipo.

Los resultados de cada metodología (*Commonkads* y *Buchanan*) serán valorados de acuerdo al criterio de evaluación de tiempo de desarrollo planteado en la Tabla III. VII. A continuación en Tabla III. XXV se detalla los resultados obtenidos.

Tabla III. XXV. Evaluación del indicador tiempo de desarrollo

INDICADOR	METODOLOGÍAS			
	COMMONKADS		BUCHANAN	
	Valor	Equivalencia sobre 20%	Valor	Equivalencia sobre 20%
Tiempo de desarrollo	2	10%	3	20%
Total de tiempo de desarrollo sobre 20%	2	10%	3	20%

Fuente: Fausto Llangari S.

A continuación los resultados de la **Tabla III. XXV** se explican en la **Figura III.24**.



Figura III. 24. Resultado del indicador tiempo de desarrollo

Fuente: Fausto Llangari S.

Como se observa en la **Figura III.24**, la metodología *Commonkads* tiene un porcentaje de 10% de tiempo de desarrollo sobre el 20% del indicador (ver la **Tabla III. III**), dado que el tiempo empleado para el desarrollo del sistema fue 176 horas, por lo tanto tiene una calificación cualitativa **Bueno**, mientras que la metodología *Buchanan* tiene el 20% ya que el tiempo empleado para el desarrollo del sistema fue menor, con 128 horas por lo tanto tiene una calificación cualitativa **Muy bueno**.

Se concluye que el tiempo empleado para el desarrollo del sistema con la metodología *Commonkads* es mayor, que con la metodología *Buchanan*, como se indica en la Figura III.24.

3.5.4 Esfuerzo de Desarrollo

Para las pruebas en este indicador analizara el esfuerzo estimado en horas-persona para el desarrollo del sistema.

- Con la metodología *CommonKADS*, la cantidad de esfuerzo estimado en horas-personas necesarias es: **88** horas/persona (ver Anexo A – Sección 3).
- Con la metodología *Buchanan*, la cantidad de esfuerzo estimado en horas-personas necesarias es: **96** horas/persona (ver Anexo A – Sección 3).

Los resultados de cada metodología (*Commonkads* y *Buchanan*) serán valorados de acuerdo al criterio de evaluación del Esfuerzo de desarrollo planteado en la **Tabla III. VIII**. A continuación en **Tabla III. XXVI** se detalla los resultados obtenidos.

Tabla III. XXVI. Evaluación del Esfuerzo de desarrollo

INDICADOR	METODOLOGÍAS			
	COMMONKADS		BUCHANAN	
	Valor	Equivalencia sobre 20%	Valor	Equivalencia sobre 20%
Costo	3	20%	2	10%
Total del Esfuerzo de desarrollo sobre 20%	3	20%	2	10%

Fuente: Fausto Llangari S.

A continuación los resultados de la **Tabla III. XXVI** se explican en la **Figura III.25**.



Figura III. 25. Resultado del indicador Esfuerzo de desarrollo

Fuente: Fausto Llangari S.

Como se observa en la **Figura III.25**, la metodología *Commonkads* tiene un porcentaje de 20% de Esfuerzo de desarrollo, debido a que el esfuerzo estimado fue 88 horas/persona, por lo tanto tiene una calificación cualitativa **Muy bueno**, mientras que la metodología *Buchanan* tiene el 10% ya que el esfuerzo estimado fue 96 horas/persona, por lo tanto tiene una calificación cualitativa **Bueno**.

Se concluye que el Esfuerzo de desarrollo para el desarrollo del sistema con la metodología *Commonkads* es mejor que la metodología *Buchanan*, como se indica en la **Figura III.25**.

3.5.5 Costo

Para las pruebas en este indicador se verá la cantidad de recursos económicos necesarios para el desarrollo del sistema.

- Con la metodología *CommonKADS*, la cantidad de recursos económicos necesarios es: \$228 (ver Anexo A – Sección 4).
- Con la metodología *Buchanan*, la cantidad de recursos económicos necesarios es: \$180 (ver Anexo A – Sección 4).

Los resultados de cada metodología (*Commonkads* y *Buchanan*) serán valorados de acuerdo al criterio de evaluación del costo planteado en la **Tabla III. IX**. A continuación en **Tabla III. XXVII** se detalla los resultados obtenidos.

Tabla III. XXVII. Evaluación del indicador costo

INDICADOR	METODOLOGÍAS			
	COMMONKADS		BUCHANAN	
	Valor	Equivalencia sobre 20%	Valor	Equivalencia sobre 20%
Costo	2	10%	3	20%
Total del costo sobre 20%	2	10%	3	20%

Fuente: Fausto Llangari S.

A continuación los resultados de la **Tabla III. XXVII** se explican en la **Figura III.26**.



Figura III. 26. Resultado del indicador Costo

Fuente: Fausto Llangari S.

Como se observa en la **Figura III.26**, la metodología *Commonkads* tiene un porcentaje de 10% de Costo, debido a que la cantidad de recursos económicos necesarios para el desarrollo del sistema fue de \$228, por lo tanto tiene una calificación cualitativa **Poco**, mientras que con la metodología *Buchanan* tiene el 20% porque la cantidad de recursos económicos necesarios para el desarrollo del sistema fue menor, es decir de \$180, por lo tanto tiene una calificación cualitativa **Muy Poco**.

Se concluye que la cantidad de recursos económicos necesarios para el desarrollo del sistema con la metodología *Commonkads* es mayor a la cantidad de recursos económicos con la metodología *Buchanan*, como se indica en la **Figura III.26**.

3.6 Resultado de la comparación de las Metodologías CommonKADS vs. Buchanan

En la **Tabla III. XXVIII**, se presenta los resultados adquiridos de la comparación de las metodologías Commonkads y Buchanan, en cada uno de los indicadores.

Tabla III. XXVIII. Resultado de la comparación de las metodologías.

No	INDICADORES	METODOLOGÍAS	
		COMMONKADS	BUCHANAN
		%	%
1	Disponibilidad de información sobre 20%	20%	5%
2	Líneas de código sobre 20%	20%	10%
3	Tiempo de desarrollo sobre 20%	10%	20%
4	Esfuerzo de desarrollo sobre 20%	20%	10%
5	Costo sobre 20%	10%	20%
Total Productividad sobre 100%		80%	65%

Fuente: Fausto Llangari S.

Los resultados de la **Tabla III.18**, se representan en la fórmula planteada anteriormente con los indicadores:

Productividad = Disponibilidad de Información + Líneas de código + Tiempo de desarrollo + Esfuerzo de desarrollo + Costo

- **Commonkads**

Productividad = 20% + 20% + 10% + 20% + 10% = **80%**

- **Buchanan**

Productividad = 5% + 10% + 20% + 10% + 20% = **65%**

A continuación el resultado total de la **Tabla III.18** se explican en la **Figura III.27**.

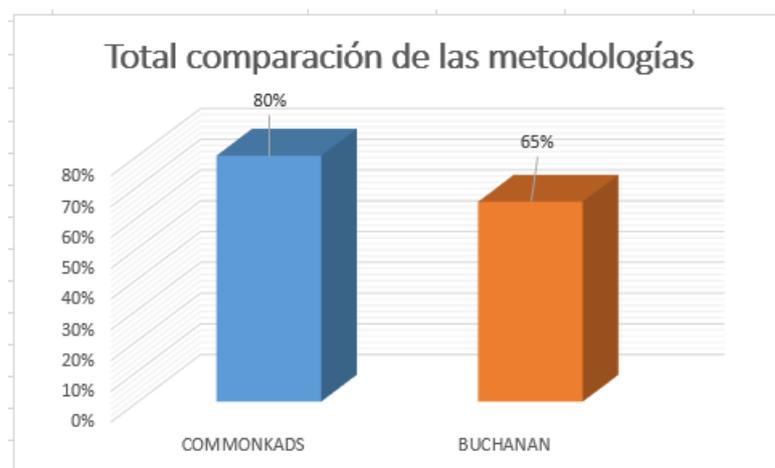


Figura III. 27. Resultado Total comparación de las metodologías

Fuente: Fausto Llangari S.

Se observa claramente que la metodología *Commonkads* es superior con un porcentaje de 80% frente a la metodología *Buchanan* que tiene 65%.

La metodología *Commonkads* es superior con una diferencia de 15% frente a la metodología *Buchanan*, porque es más óptima para desarrollar un Sistema Experto, dicho resultado se explica en la **Figura III.27**.

3.7 Comprobación de la Hipótesis

La comprobación de la hipótesis se realiza en base a las pruebas y recolección de datos aquiridas en la **Tabla III. XXVIII**. Para lo cual se utiliza el Método Estadístico Descriptivo.

Hipótesis: “La metodología COMMONKADS presenta una mayor productividad frente a la metodología BUCHANAN en el desarrollo de un Sistema Experto”.

- **Hipótesis Nula H0** = La metodología COMMONKADS no presentará una mayor productividad frente a la metodología BUCHANAN en el desarrollo de un Sistema Experto para la gestión de cultivos de la Junta General de Riego Chambo Guano.
- **Hipótesis Alternativa H1** = La metodología COMMONKADS presentará una mayor productividad frente a la metodología BUCHANAN en el desarrollo de un Sistema Experto para la gestión de cultivos de la Junta General de Riego Chambo Guano.

$$H_0 = \mu_{SCOMMONKADS} < \mu_{SBUCHANAN}$$

$$H_1 = \mu_{SCOMMONKADS} \geq \mu_{SBUCHANAN}$$

Una buena elección de la metodología que presenta una mayor productividad para el desarrollo de un Sistema Experto es la METODOLOGÍA COMMONKADS con una calificación de 80%, para la selección de esta metodología dependió de los resultados de los siguientes indicadores, *Disponibilidad de información* tiene un porcentaje de 20% debido a que la información buscada en los *libros, Internet y manuales* disponibles fue numerosa, es decir el promedio máximo fue 47040 links de información, *Líneas de código* tiene un porcentaje de 20% dado que la cantidad de líneas de código empleadas en el desarrollo de la aplicación fue de 2231, *Tiempo de desarrollo* tiene un porcentaje de 10% ya que el tiempo empleado para el desarrollo del sistema fue de 176 horas, *Esfuerzo de desarrollo* tiene un porcentaje de 20% puesto que la cantidad de esfuerzo estimado en horas-personas necesarias fue 88 horas/persona, y el indicador *Costo* tiene un porcentaje de 10% por que la cantidad de recursos económicos necesarios para el desarrollo del sistema fue de \$228.

De acuerdo al análisis realizado y los cálculos obtenidos en la **Tabla III. XXVIII** se puede decir que la *metodología Buchanan* no presenta una productividad adecuada, por lo tanto la que presenta mayor productividad es la *metodología Commonkads*.

A partir de ello se puede decir que **Commonkads** será la metodología usada para desarrollar el Sistema Experto para la Junta General de Riego Chambo Guano.

Finalmente queda demostrado que la metodología Commonkads presenta una mayor productividad con un 80% de cumplimiento y con una diferencia de 15%, frente a la metodología Buchanan que brinda un 65%. Por lo cual se acepta la hipótesis (H1) alternativa.

CAPITULO IV

IMPLEMENTACIÓN DEL SISTEMA EXPERTO

4.1 Junta General de Usuarios de Riego Chambo-Guano JURECH

La Junta General de Usuarios de Riego Chambo-Guano brinda a sus usuarios servicios completos de riego, tecnificación, transferencia de conocimientos, tecnologías nuevas, interrelación entre productores con mercados nacionales y extranjeros, mirando a sus usuarios como agricultores micro empresarios, tiene sus oficinas en la calle Chile y Puruhá.

Misión

Brindar a nuestros usuarios servicios completos de riego, tecnificación, transferencia de conocimientos, tecnologías nuevas, interrelación entre productores con mercados nacionales y extranjeros, mirando a nuestros usuarios como agricultores micro empresarios.

Visión

Ser una organización con capacidad autosuficiente, para administrar, operar y mantener el Sistema de Riego, además de vigilar y cumplir con los objetivos mediáticos y a largo plazo.

Objetivos:

Son objetivos de la Junta General de Usuarios del Sistema de Riego Chambo Guano:

- a) Distribuir equitativamente el riego en toda el área de influencia del sistema.
- b) Responsabilizarse por la congestión del Sistema de Riego Chambo Guano, en lo referente a los aspectos técnicos, en el desarrollo financiero y de la infraestructura del sistema, enfocando su gestión colectiva y comunitaria.
- c) Apoyar las iniciativas de gestión integrada de la cuenca del río Chambo, fomentando y desarrollando de modo especial actividades que contribuyan a la conservación del ecosistema páramo; y a aquellas actividades orientadas a evitar la contaminación del agua.
- d) Incentivar el desarrollo de programas de tecnificación del riego, que contribuyan a optimizar el uso del agua, conservar los suelos y garantizar eficiencia en la aplicación del riego a nivel de parcelas.
- e) Fomentar y desarrollar capacidades en los usuarios del Sistema de riego, para el incremento de la producción agroecológica y, el acceso a mercados convencionales y alternativos.
- f) Poner a disposición de los usuarios del Sistema de riego, la prestación de servicios y facilidades para sus actividades productivas, tales como préstamos, alquiler de maquinaria y de limpieza de canales, etc.

4.2 Aplicación de CommonKADS para el desarrollo de SYS-Cultivos

4.2.1 Modelo de la Organización

En este primer modelo OM-1 se determinan las principales características de la organización, se descubren problemas y oportunidades del sistema basado en conocimiento.

4.2.1.1 Formulario OM - 1

Modelo de la Organización

Problemas

- La determinación de un cultivo a ser sembrado se la realiza en forma empírica.
- Carencia de personal capacitado para brindar asesoramiento
- Sobreproducción de productos
- Escases de productos
- Falta de planificación para establecer un calendario de siembra durante el año
- Alto riesgo de temporal de cultivo y procesos de mercado
- Baja tecnificación
- Falta de asesoría técnica dedicada a la elección de cultivo

Oportunidades

- Demanda de asesoramiento de ing. agrónomo o agropecuario.
- Aprovechamiento de los recursos agrícolas del sector
- Gran cantidad de agricultores que quieren hacer un uso correcto de sus recursos agrícolas
- Deseo de la organización por mejorar la calidad de atención al usuario.

Solución 1:

Desarrollo de un Sistema Experto que dará soporte en la selección mas idónea de un cultivo a ser sembrado basado en el ciclo de producción capacidad de inversión y características del lugar a ser sembrado.

Solución 2:

Contratar un experto agrónomo.

4.2.1.2 Formulario OM - 2**Aspectos variables**

Tabla IV. XXIX. Aspectos de variantes

OM-2	Aspectos Variables
Estructura	Organigrama figura No IV. 28.
Procesos	(A) Atención al Usuario
Personas	(1) Ing Agrónomo (2) Usuario
Recursos	Materiales: GPS Barreno Termómetro Pluviómetro Cinta
Conocimiento	Conocimiento del Agrónomo Conocimiento de Riego Conocimiento de Fertilización Conocimiento Agro meteorológico Conocimiento Fitosanitario del Cultivo

Realizado por: Fausto Llangari

Estructura de la organización

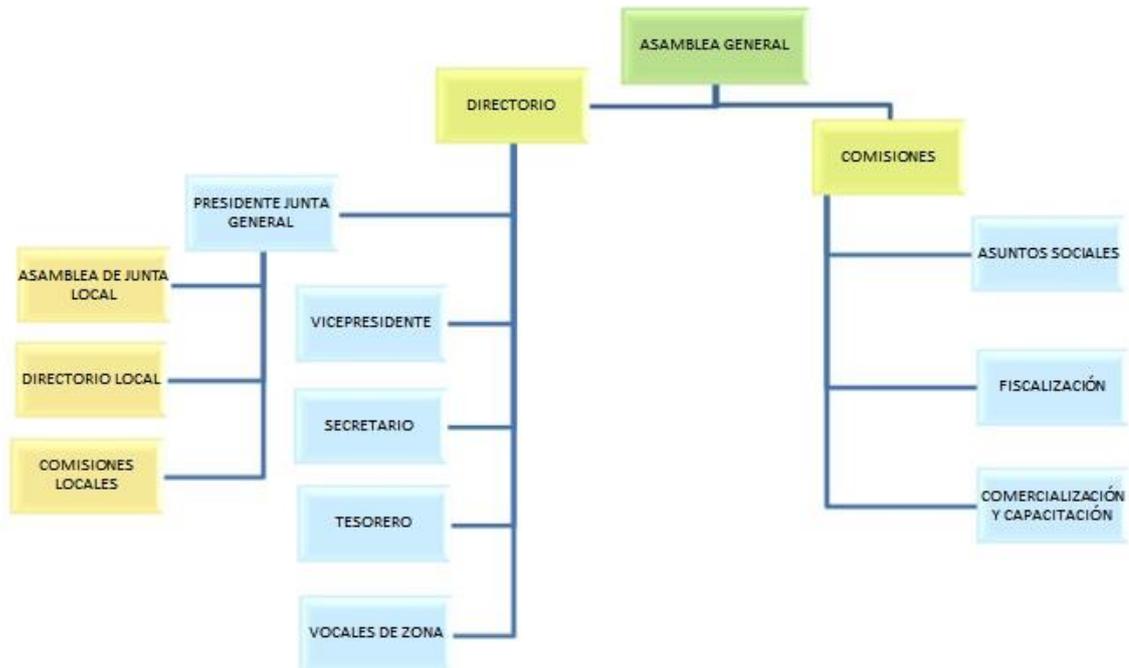


Figura IV. 28. Estructura de la organización

Fuente: Jurech

La Junta de riego Chambo – Guano, orientada a la administración del riego para las comunidades de los cantones Chambo, Guano y sus alrededores está representada a cargo de un Presidente de la Junta; y está conformado por diferentes áreas como Directorio, Comisiones, Asambleas; se sitúan en la atención a los Usuarios.

Procesos

En la figura IV. 29 se muestra los procesos

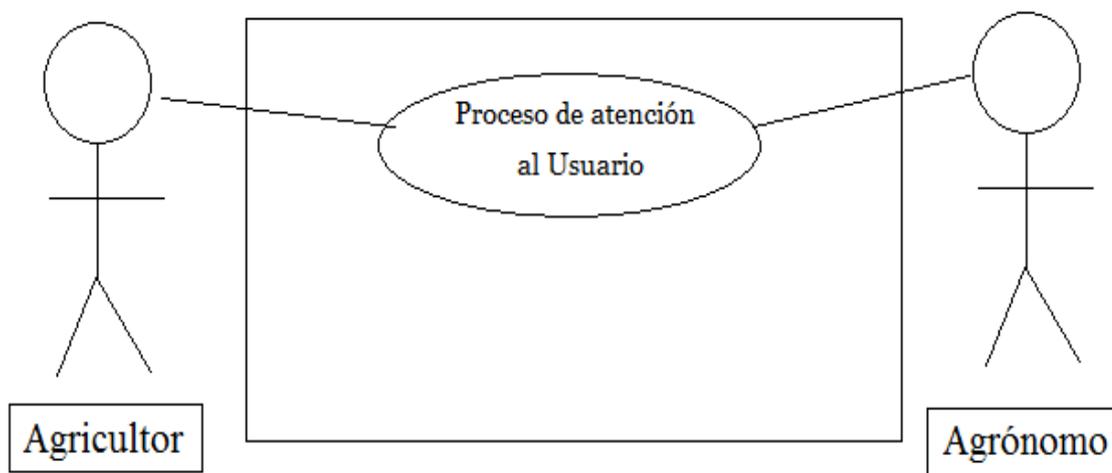


Figura IV. 29. Proceso – Proceso de atención al Usuario

Fuente: Fausto llangari

4.2.1.3 Formulario OM - 3

DESCOMPOSICION DE PROCESOS

Tabla IV. XXX. Atención al Usuario

Modelo Organizacional		Atención al Usuario				
Nº	Tarea	Agente	donde	Recursos de conocimiento	Intensiv o	Importan cia
1.1	Obtener la petición del Usuario:	Agrónomo - Agricultor	Oficina Jurech	si	Alta
	Sugerencia para cultivo	Agricultor - agronomo	oficina	-Conocimientos del agronomo -conocimientos de mercado	Si	Alta

Realizado por: Fausto Llangari

4.2.1.4 Formulario OM - 4

Activos de Conocimiento

Tabla VI. XXXI. Tarjetas sobre Recursos de Conocimiento

Modelo organizacional	OM-4. Tarjetas sobre Recursos de Conocimiento					
	Recurso de conocimiento	Agente	Usado por	¿Forma correcta?	¿Lugar correcto?	¿Tiempo correcto?
Conocimiento del agrónomo	Agrónomo – Agricultor	1.1	si	si	Si	si
Conocimiento de Riego	Agrónomo	1.2 1.3 1.4 1.5	si	si	Si	si
Conocimiento de Fertilización	Agrónomo	1.2 1.3 1.4 1.5	si	si	Si	si
Conocimiento Agro meteorológico	Agrónomo	1.2 1.3 1.4 1.5	si	si	Si	si
Conocimiento Fitosanitario del Cultivo	Agrónomo	1.2 1.3 1.4 1.5	si	Si	Si	Si

Realizado por: Fausto Llangari

4.2.1.5 Formulario OM - 5

Elementos Documento Viabilidad

Tabla VI.XXXII. Elementos Documento Viabilidad

		Valor
Viabilidad del Negocio	Se espera una alta tasa de recuperación de la inversión	9
	Resuelve una tarea útil y necesaria	9
	Los expertos no se sienten amenazados por el proyecto, sino todo lo contrario	9
	Los expertos son capaces de sentirse intelectualmente unidos al proyecto	7
	Mentalización de los directivos y usuarios con expectativas realistas en el alcance y en las limitaciones	7
	Los directivos y usuarios no rechazan de plano esta tecnología	6
	Los Expertos están comprometidos durante toda la duración del proyecto	8
	Puntaje	55
Viabilidad Técnica	Existen Expertos	9
	El experto es cooperativo	9
	Hay escasez de experiencia humana	7
	Hay cambios mínimos en los procedimientos habituales de resolución de la tarea	7
	Las soluciones son explicables e interactivas	5
	La experiencia usada por el experto está más o menos organizada	5
	La tarea es básicamente de tipo heurístico	6
	Puntaje	48
Viabilidad del Proyecto	El sistema puede ser introducido fácilmente	7
	El sistema puede ser mantenido	9
	El sistema puede ser integrado con recursos existentes	8

	El problema es solucionable	9
	Los directivos están verdaderamente comprometidos en el proyecto	8
	Los expertos tienen un brillante historial en la realización de esa tarea	7
	La única justificación para dar un paso en la solución es la calidad de la solución final	7
	La tarea está identificada como un problema en el área	9
	La tarea sirve a necesidades a largo plazo	8
	Existen ya Sistemas Expertos que resuelven esa o parecidas tareas	0
	Puntaje	72
Puntaje total		175

Realizado por: Fausto Llangarí

4.2.2 Modelo de Tareas

Describe las tareas que realizan los agentes de la Organización

4.2.2.1 Formulario TM - 1

Análisis y descripción de las tareas dentro del proceso

Tabla IV. XXXIII. Análisis de las tareas-obtener información del paciente

Tarea	1.1 Obtener petición del Usuario
Organización	Comisiones Locales
Objetivos y Valores	Conocer la situación del usuario y su petición.
Objetos Manejados	Entrada: petición del Usuario (sacar ticket) Salida : obtener la información del Usuario

Temporización y Control	Frecuencia: Uno por cada Usuario. Duración : Poca Precondición : Estar disponible Agrónomo Pos-condición: la petición ha sido identificada y almacenada.
Agentes	Usuario Agrónomo
Conocimientos y Competencias	Conocimiento de la secretaria de la junta
Recursos	Agrónomo utiliza: Utiliza una Computadora, papel.
Plantilla TM-1 Análisis de Tareas	

Realizado por: Fausto Llangarí

Tabla IV. XXXIV. Análisis de las tareas - Sugerencia para cultivo

Tarea	1.2 Sugerencia para cultivo
Organización	Comisiones Locales
Objetivos y Valores	Realizar la sugerencia del cultivo más idóneo de acuerdo a los requerimientos del Usuario
Objetos Manejados	Entrada : Petición de Usuario Salida : Cultivo recomendado
Temporización y Control	Frecuencia: Uno por Usuario. Duración : Media. Precondición: que el Agrónomo disponga de la información del Usuario Pos-condición: que se registren toda la información del Usuario.
Agentes	Usuario, Agrónomo
Conocimientos y Competencias	Conocimiento del Agrónomo Conocimiento de Riego Conocimiento de Fertilización Conocimiento Agro meteorológico Conocimiento Fitosanitario del Cultivo.

Recursos	GPS, Barreno, Termómetro, Pluviómetro, Cinta, computadora
Calidad y Rendimiento	Que el Ing Agrónomo reciba correctamente la información y realice el análisis adecuado
Plantilla TM-1 Análisis de Tareas	

Realizado por: Fausto Llangari

4.2.2.2 Formulario TM – 2

Elemento de conocimiento

Describimos los ítems de conocimiento aludidos en las tablas TM-1

Con las siguientes tablas.

Tabla IV XXXV. Elemento de conocimiento- Ítem de Conocimiento del Ing. Agrónomo

Modelo de Tareas	Ítem de Conocimiento Del Ing. Agrónomo
Nombre	Conocimiento del Ing. Agrónomo
Poseído por	Agrónomo
Usado en	1.1 Obtener petición del Usuario
Dominio	Comisiones Locales

Realizado por: Fausto Llangari

Tabla IV. XXXVI: Ítem de Conocimiento del Ing. Agrónomo

Naturaleza del conocimiento	(Si/no)	¿Cuello de botella/ debe ser mejorado?
Formal, riguroso	Si	
Empírico, cuantitativo		
Heurístico, sentido común	Si	
Altamente especializado	Si	
Basado en experiencia	Si	

Basado en acciones		
Incompleto		
Difícil de verificar		
Tácito, difícil de transferir	No	si
Forma del Conocimiento		
Mental		
papel	Si	si
electrónica	Si	
Habilidades		
Otros		
Disponibilidad del Conocimiento		
Limitaciones en tiempo	No	Si
Limitaciones en espacio	No	Si
Limitaciones de acceso		
Limitaciones de calidad		
Limitaciones de forma	Si	Si

Realizado por: Fausto Llangari

4.2.3 Modelo de Agentes

Los Agentes son los ejecutores de una tarea. El modelo de agente describe las características de los agentes, sus competencias, autoridad y restricciones.

4.2.3.1 Formulario AM-1:

Modelo de Agentes

Tabla IV. XXXVII. Modelo de Agentes- Usuario

Modelo de Agentes	Usuario
Nombre	Usuario
Organización	Tipo : Humano.
Implicado en	Realizar la consulta al agrónomo 1.2. Proporcionar datos sobre su situación

Se comunica con	Agrónomo
Plantilla AM-1 Especificación de Agentes	

Realizado por: Fausto Llangari

Tabla IV. XXXVIII. Modelo de agentes- Agrónomo

Modelo de Agentes	Agente
Nombre	Agrónomo
Organización	Tipo : Humano. Estructura : Comisiones locales
Implicado en	1.1. Obtener petición del Usuario
Se comunica con	Usuario
Conocimiento	Conocimiento del Agrónomo Conocimiento de Riego Conocimiento de Fertilización Conocimiento Agro meteorológico Conocimiento Fitosanitario del Cultivo
Otras competencias	
Responsabilidades y Restricciones	Responsabilidades: Analizar la situación actual del terreno y su propietario así como sus requerimientos.
Plantilla AM-1 Especificación de Agentes	

Realizado por: Fausto Llangari

4.2.4 Modelo de Conocimiento

En este modelo se desarrollan las tres principales categorías del conocimiento. El conocimiento del dominio, el conocimiento de inferencias y el conocimiento de tareas.

4.2.4.1 CM1 (Conocimiento del Dominio)

Se representa el conocimiento relevante del sistema, sobre el que se desarrolla el proceso de razonamiento objeto del modelado. Una explicación básica de los elementos que componen el dominio del sistema fueron desarrollados.

4.2.4.2 CM2 (Especificación del Esquema del Dominio)

En esta fase abordamos la adquisición del conocimiento específicos del dominio concreto de la aplicación. A partir de las entrevistas iniciales con el experto, se llegó a la conclusión de que la manera más eficiente de adquirir el conocimiento era centrar dichas entrevistas en torno a formularios usadas por el especialista.

Esquema de conceptos: Un concepto se utiliza para definir un conjunto de objetos o instancias utilizado en el dominio de la aplicación.

DIAGRAMA

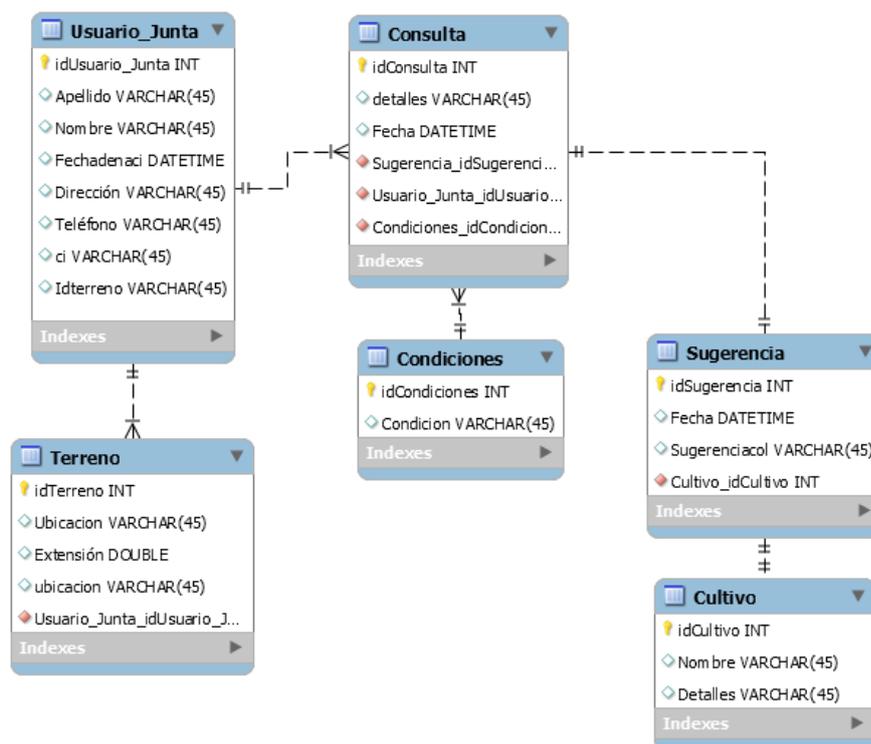


Figura IV. 30. Diagrama de conceptos

Fuente: Fausto Llangarí

Conceptos

Usuario_Junta: Es la clase que contiene los atributos principales del Usuario.

Terreno: Es la clase que contiene los atributos principales del Terreno.

Sugerencia: Contiene la sugerencia realizada por el Agrónomo

Cultivo: Contiene los atributos de un cultivo.

Consulta: Es la clase que contiene la fecha de atención y detalles de consulta

Condiciones: Es la clase que contiene como atributos las condiciones que determinan a la sugerencia del experto.

Representación de los conceptos

CONCEPT Usuario_Junta:

ATTRIBUTES:

Idusu:int;

Apellido: varchar;

Nombre: varchar;

FechaNaci:integer;

Dirección: varchar;

Teléfono: varchar;

ci: varchar;

Idterreno:int;

END CONCEPT Usuario_Junta;

CONCEPT Terreno;

ATTRIBUTES:

Idterreno:int;

Ubicacion:varchar;

Extensión:doublé;

Idusuario:int;

END CONCEPT Terreno;

CONCEPT Sugerencia;

ATTRIBUTES:

Idsug:int;

Cultivo:varchar;

Fecha:datetime;

END CONCEPT Sugerencia;

CONCEPT Cultivo;

ATTRIBUTES:

Idcul:int;

Nombre:varchar;

detalles:varchar;

END CONCEPT Cultivo;

CONCEPT Consulta;

ATTRIBUTES:

Idcons:int;

detalles:varchar;

Fecha:datetime;

END CONCEPT Consulta;

CONCEPT Condiciones;

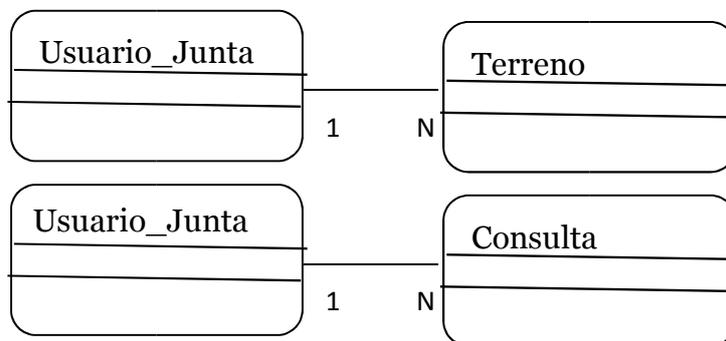
ATTRIBUTES:

Idcond:int;

:varchar;

END CONCEPT Condiciones;

Relaciones: Mediante este tipo de construcción podemos definir relaciones entre conceptos definidos en un esquema del dominio determinado.



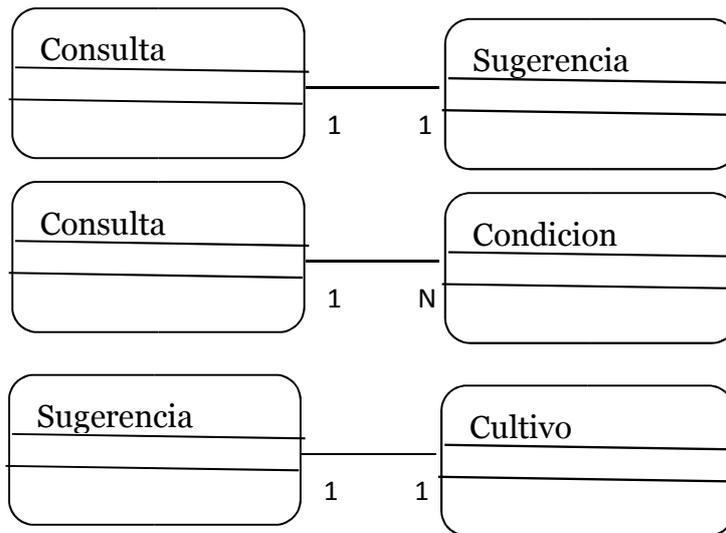


Figura IV. 31. Relación de conceptos

Fuente: Fausto Llangari

Tipos generadores de reglas: Nos permite definir relaciones entre dos expresiones lógicas definidas en función de un valor de un atributo de un concepto, en el que en vez de tomar como argumentos determinados conceptos del esquema los argumentos son expresiones sobre conceptos. Para definir un tipo de reglas necesitamos un antecedente y un consecuente en los que indicamos los conceptos sobre cuyas instancias se definen las expresiones lógicas. Para la formulación de las reglas se toma en cuenta la colaboración de un experto con conocimientos amplios sobre cultivos. Por esta razón el sistema diseñado debe ser capaz de ofrecer como salida una sugerencia que corresponda con los valores de las variables de entrada y el criterio del experto.

Nivel Dominio

Para representar el conocimiento abstracto se hará uso del nivel dominio mediante el cual se obtienen las reglas. La figura IV. 32 muestra una parte del razonamiento que se sigue para determinar la sugerencia más adecuada para un cultivo.

Esquema de nivel de dominio

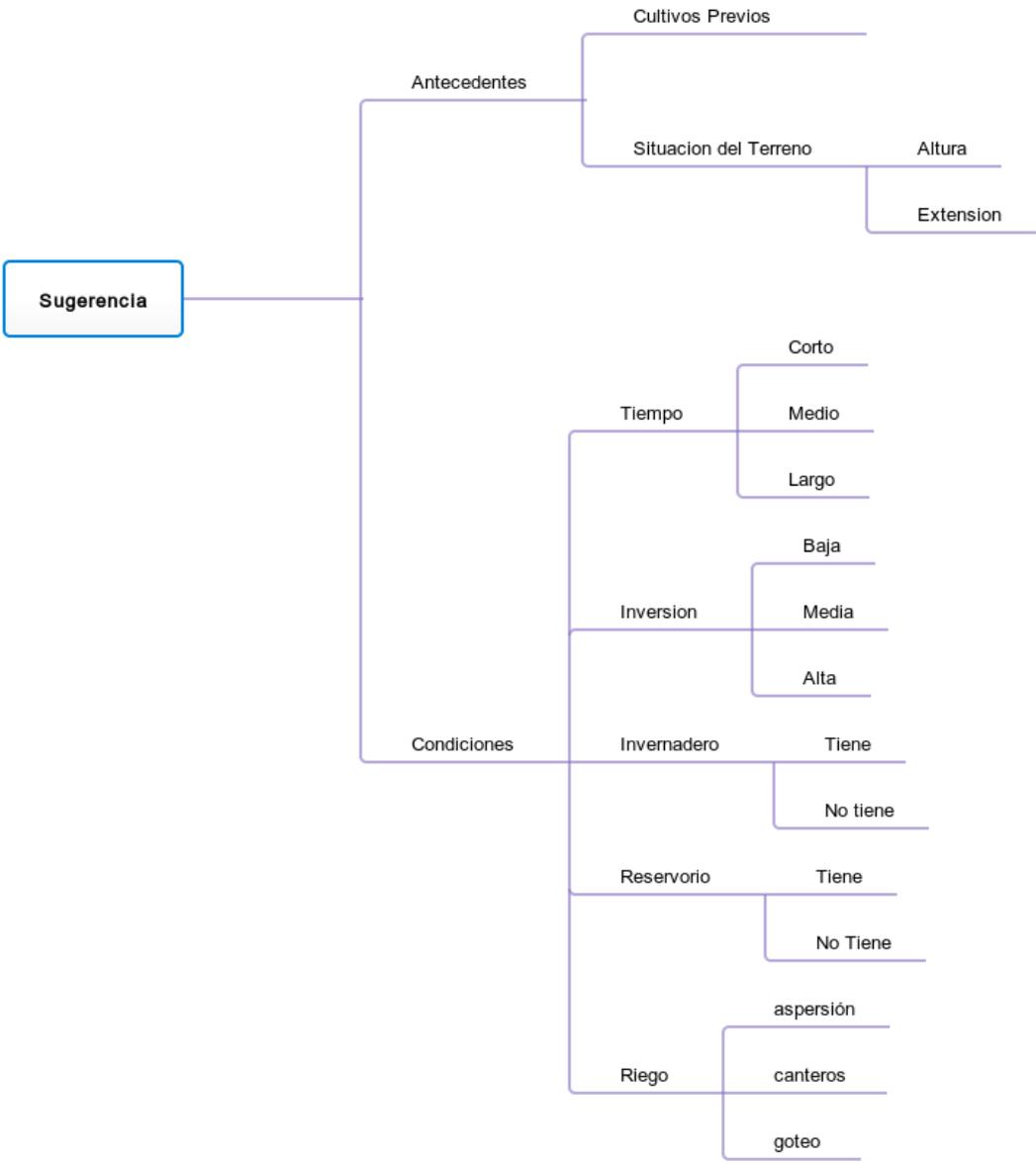


Figura IV. 32. Esquema de nivel del dominio

Fuente: Fausto Llangarí

4.2.4.3 CM3 (Base de Conocimiento)

Contiene el conocimiento especializado extraído a través de la entrevista con el experto, es decir contiene conocimiento general, todos los hechos y reglas del dominio de aplicación. La forma de representación más usada es por reglas de producción, también llamadas reglas de inferencia.

Componentes de una base de conocimientos.

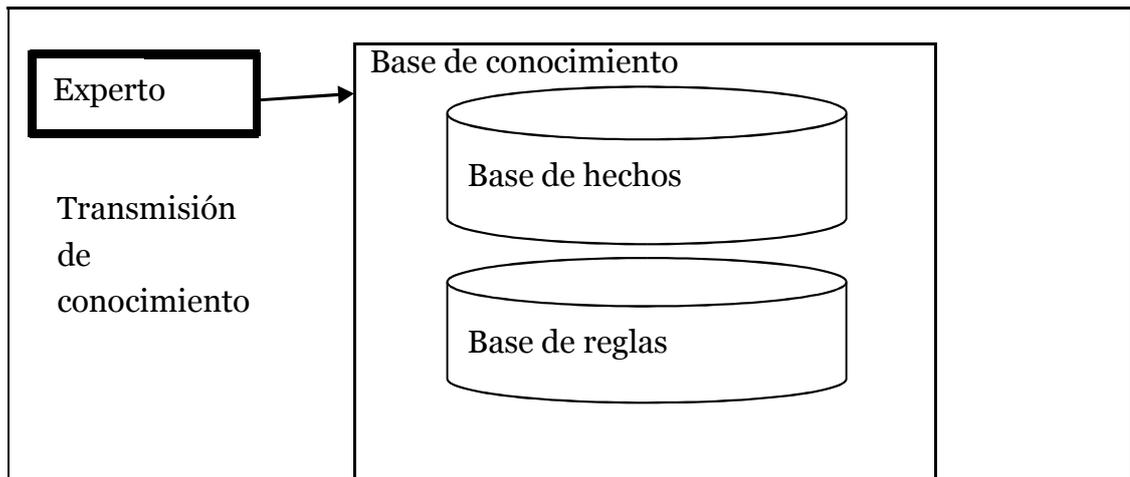


Figura IV. 33. Base de conocimiento

Fuente: Fausto Ilangari

Atributos

Cuyos valores serán pedidos a la Usuario (selección)

H1. Tiempo de producción (ciclo de cultivo)

H2: Capacidad de Inversión por Hectárea

H3: Disponibilidad de Invernadero

H4: Disponibilidad de Reservorio

H5: Disponibilidad de Riego

B) Base de Reglas

Tabla IV. XXXIX: Regla 1

ESTADO DE LA REGLA	TEXTO DE LA REGLA
Palabras del Experto	Existe cultivos de ciclo corto, Medio y largo plazo o perennes
Formulación externa de la regla	SI $\langle H1 \leq 4 \rangle$ ENTONCES $\langle \text{ciclo} = \text{ciclo corto} - \text{sembrar Rábano, Cilantro, Fréjol, Sandía, Brócoli, Melón, Lechuga, Cebolla Puerro, Zucchini, Sambo, Col, Coliflor, Papa, Pimiento o Tomate Riñón} \rangle$ SI $\langle H1 > 4 \rangle \wedge Y \langle H1 \leq 8 \rangle$ si ENTONCES $\langle \text{ciclo} = \text{ciclo medio} - \text{sembrar Pastos, Quinoa, Zanahoria, Remolacha, Arveja, Avena, Alfalfa, Cebolla colorada, Col morada, Ajo, Fresa, Babaco o Maíz} \rangle$ SI $\langle H1 > 9 \rangle$ ENTONCES $\langle \text{ciclo} = \text{ciclo largo} - \text{sembrar Mora de Castilla o Tomate de Árbol} \rangle$
Nombre de la Regla 1	Ciclo de cultivo

Realizado por: Fausto Llangari

Tabla IV. XL. Regla 2

ESTADO DE LA REGLA	TEXTO DE LA REGLA
Palabras del Experto	Según el Ing. Agrónomo el costo por hectárea al sembrar determinado cultivo puede dividirse en baja, media o alta inversión
Formulación externa de la regla	SI $\langle H2 \leq 1000 \rangle$ ENTONCES $\langle \text{inversión} = \text{baja} - \text{sembrar Rábano, Quinoa, Pastos, Zanahoria, Cilantro, Fréjol, Zucchini, Sambo, Remolacha o Arveja} \rangle$ SI $\langle H2 \Rightarrow 5000 \rangle \wedge Y \langle H2 \leq 10000 \rangle$ si ENTONCES $\langle \text{inversión} = \text{media} - \text{sembrar Col, Avena, Sandía, Brócoli, Alfalfa, Coliflor, Melón, Cebolla colorada, Maíz, Lechuga, Papa, Col morada o Tomate de Árbol} \rangle$ SI $\langle H1 > 10000 \rangle$ ENTONCES $\langle \text{inversión} = \text{alta} - \text{sembrar Mora de Castilla, Ajo, Cebolla, Puerro, Fresa, Babaco, Pimiento o Tomate Riñón} \rangle$.
Nombre de la Regla 2	Capacidad de Inversión

Realizado por: Fausto Llangari

Tabla IV. XLI. Regla 3

ESTADO DE LA REGLA	TEXTO DE LA REGLA
Palabras del Experto	Según el Ing. Agrónomo la disponibilidad o no de un invernadero permite el cultivo o no de determinado producto.
Formulación externa de la regla	SI <H3= Si> ENTONCES <invernadero= disponible - sembrar Sandia, Melón, Pimiento o Tomate Riñon >
Nombre de la Regla 3	Disponibilidad de Invernadero

Realizado por: Fausto Llangari

Tabla IV. XLII. Regla 4

ESTADO DE LA REGLA	TEXTO DE LA REGLA
Palabras del Experto	Según el Ing. Agrónomo la disponibilidad de reservorio permite el cultivo de determinado producto.
Formulación externa de la regla	SI <H4 = si ENTONCES <reservorio = disponible - sembrar Fresa o Tomate Riñon >
Nombre de la Regla 4	Disponibilidad de Reservorio

Realizado por: Fausto Llangari

Tabla IV. XLIII. Regla 5

ESTADO DE LA REGLA	TEXTO DE LA REGLA
Palabras del Experto	Según el Ing. Agrónomo la forma de riego facilita el cultivo determinado producto pudiendo ser las más comunes por aspersión, canteros y goteo.
Formulación externa de la regla	SI < H5 = aspersión> ENTONCES <riego= a - sembrar Quinoa, Pastos, Cilantro Avena Alfalfa Fréjol Zuquini o Sambo >. SI < H5 = canteros > ENTONCES <riego= c - sembrar Cilantro, Avena, Alfalfa, Fréjol, Zuquini, Sambo, Rábano, Zanahoria, Remolacha, Arveja,

	Col, Sandía, Brócoli, Coliflor, Melón, Cebolla colorada, Maíz, Lechuga, Papa, Col morada, Tomate de Árbol, Mora de Castilla, Ajo, Cebolla Puerro, Babaco, Pimiento o Tomate Riñon>. SI < H5 = goteo > ENTONCES <riego= g - sembrar Fréjol, Zuquini, Sambo, Zanahoria, Remolacha, Arveja, Col, Sandía, Brócoli, Coliflor, Melón, Cebolla colorada Maíz, Lechuga, Papa, Col morada, Tomate de Árbol, Mora de Castilla, Ajo, Cebolla Puerro, Babaco, Pimiento, Tomate Riñon o Fresa>.
Nombre de la Regla 5	Disponibilidad de Riego

Realizado por: Fausto Llangari

4.2.5 Modelo de Comunicación

En este modelo se representan los resultados de la ejecución de una tarea por parte de un agente como objetos de información los cuales debe suministrar a otros agentes.

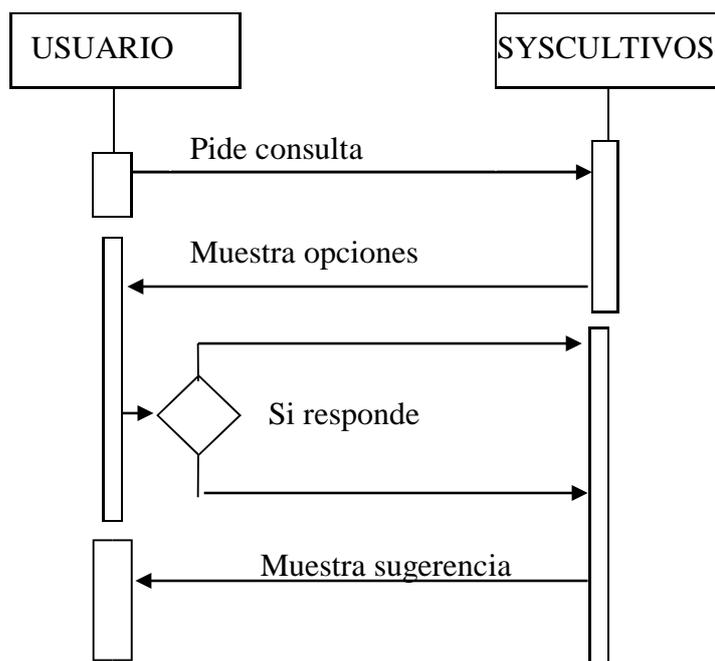


Figura IV. 34. Modelo de comunicación

Fuente: Fausto Llangari

En este modelo se representan los resultados de la ejecución de una tarea por parte de un agente como objetos de información los cuales debe suministrar a otros agentes.

4.2.5.1 Modelo de Comunicación. CM-1

Tabla IV. XLIV. Modelo de comunicación

Modelo de comunicación	Hoja de trabajo CM-1
Nombre	Pide consulta
Información	Opciones seleccionados
Involucrados	Usuario - Agente experto
Restricciones	El usuario debe seleccionar alguna de las opciones asociados a su situación actual para que el sistema experto actúe.
Especificación Del intercambio de información	Esta transición está compuesta de un mensaje lo cual es del tipo seleccionar opciones presentadas y lo envía al sistema Experto para que realice una sugerencia de cultivo.

Realizado por: Fausto Llangari

4.2.6 Modelo de Diseño

En este modelo especificamos la arquitectura del sistema, la plataforma de implementación y los requerimientos para su construcción.

4.2.6.1 Arquitectura de Sistemas. DM-1

Tabla IV. XLV. Modelo de diseño

Modelo de diseño	Arquitectura del Sistema : DM-1
Modelo de control	El agente humano (usuario) selecciona las opciones para el cultivo donde el sistema Experto empieza su intervención para la sugerencia del cultivo.
Descomposición de subsistemas	No tiene

Realizado por: Fausto Llangari

A continuación mostramos la plataforma de implementación y los requerimientos para la construcción del Sistema

4.2.6.2 Especificación de la Plataforma DM2

Tabla IV. XLVI. Especificación de la plataforma de implantación

Modelo de diseño	Hoja de trabajo. Especificación de la Plataforma de Implantación
Software	Microsoft Visual Estudio WAMPSEVER V2.0 SqlYog
Hardware	Computadora Personal Servidor
Lenguajes de programación	.Net, MYSQL
Representación del conocimiento	Es en base de reglas

Realizado por: Fausto Llangari

PROTOTIPO DEL AGENTE EXPERTO

Diagrama de Proceso del Sistema experto

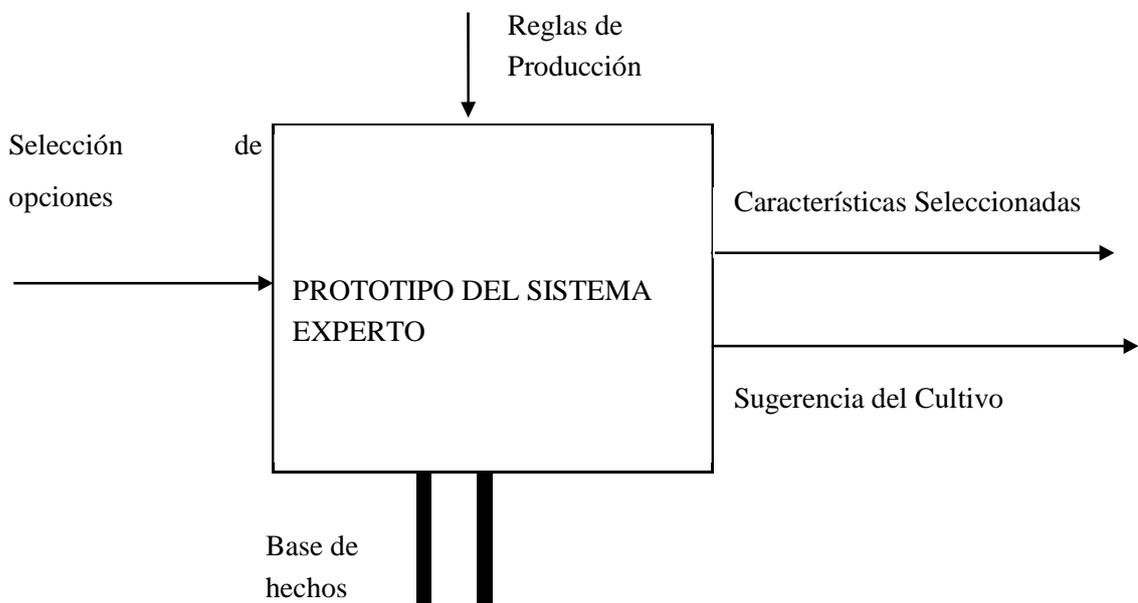


Figura IV. 35. Diagrama de proceso del sistema experto

Fuente: Fausto Llangari

Pantallas

A continuación se presentan las pantallas de ejecución del sistema

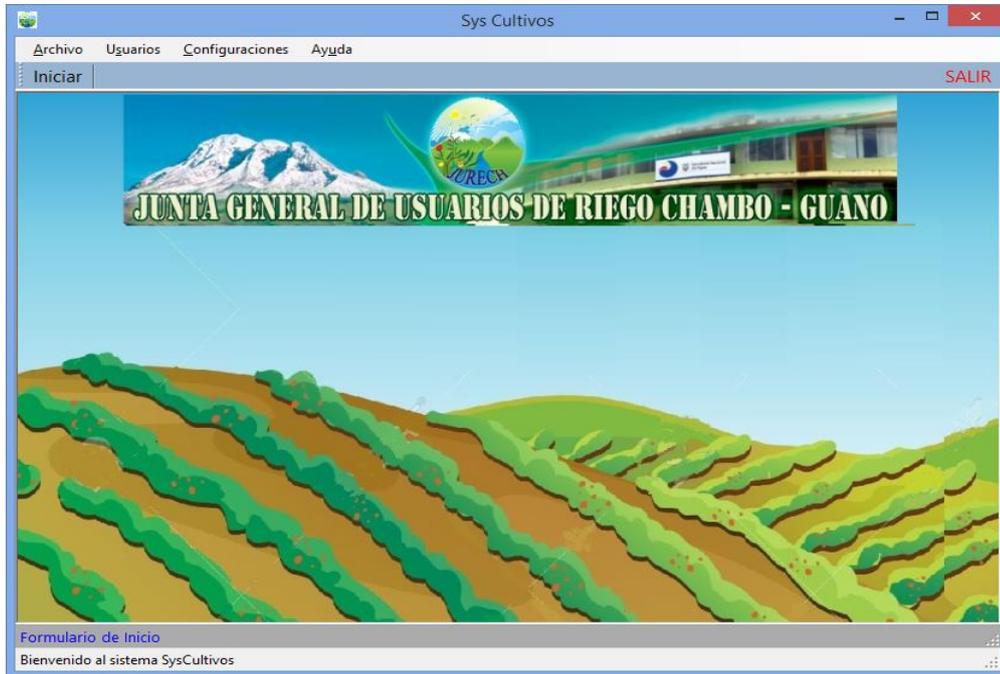


Figura IV. 36. Pantalla Principal

Fuente: Fausto Llangarí



Figura IV. 37. Pantalla Preguntas

Fuente: Fausto Llangarí

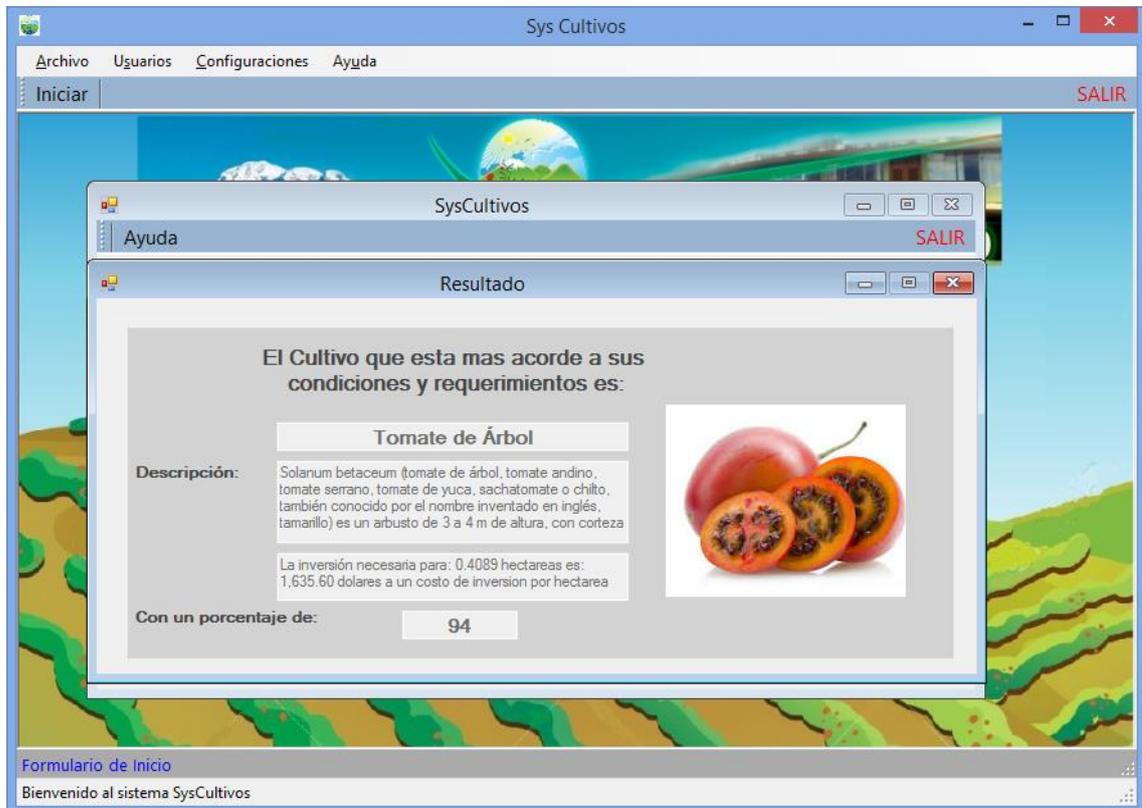


Figura IV. 37. Pantalla Resultado

Fuente: Fausto Ilangari

4.3 Implementación

El Sistema Experto Sys Cultivos que proporcionara un apoyo en la toma de decisión del cultivo a ser sembrado está desarrollado en la plataforma .Net Visual estudio y su base de datos en MySQL.

Mediante la formulación de determinadas preguntas al Usuario, determinara el cultivo que más se ajuste a sus requerimientos y condiciones, mismo que lo mostrara como sugerencia.

Base de Conocimientos

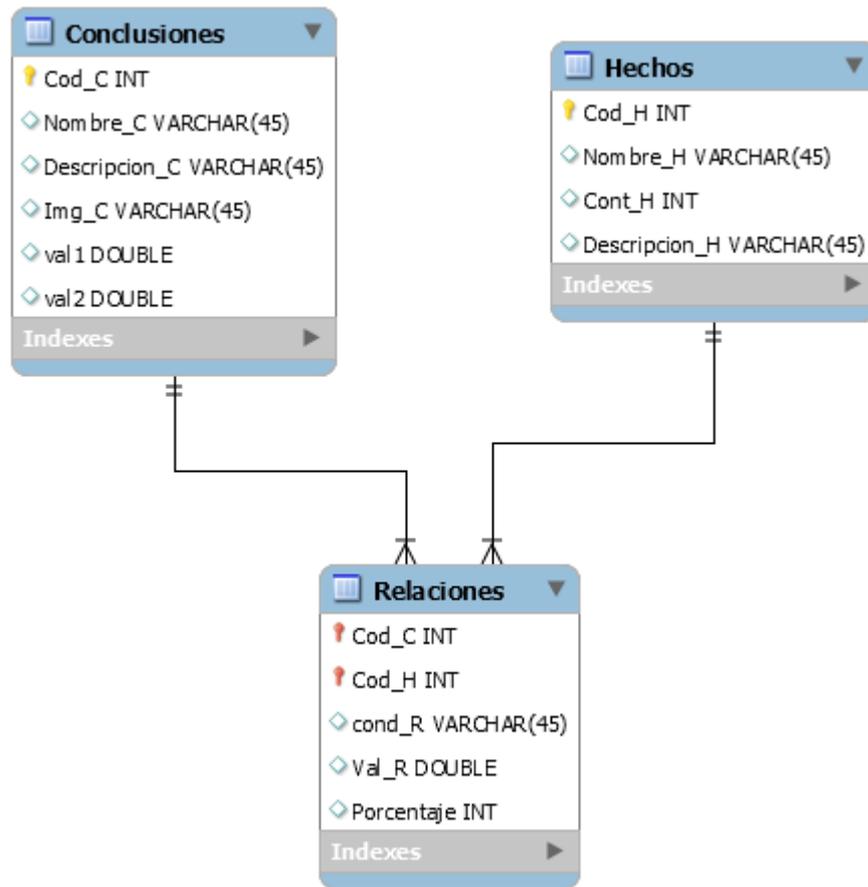


Figura IV. 38. Base de Conocimientos

Fuente: Fausto Llangari

Conclusiones

Tabla IV. XLVII. Conclusión

Campo	Descripción
Cod_C	Identificador de conclusión
Nombre_C	Nombre de la conclusión
Descripcion_C	Descripción de la conclusión
Img_C	Enlace a la imagen de la conclusión
Val1	valores auxiliares conclusión
Val2	valores auxiliares conclusión

Realizado por: Fausto Llangari

Hechos

Tabla IV. XLVIII. Hechos

Campo	Descripción
Cod_H	Identificador del hecho
Nombre_H	Nombre del hecho
Cont_H	Tipo de respuesta
Descripcion_H	Descripción del hecho

Realizado por: Fausto Llangari

Relaciones

Tabla IV. XL IX. Relaciones

Campo	Descripción
Cod_C	Identificador de conclusión
Cod_H	Identificador del hecho
Cond_R	Respuesta de la relación
Val_R	Valor de la relación
Porcentaje	Porcentaje a sumar

Realizado por: Fausto Llangari

Reglas

Las reglas que se han implementado en base al conocimiento proporcionado por el experto:

SI $H1 \leq 4$ ENTONCES \langle ciclo = ciclo corto - sembrar Rábano, Cilantro, Fréjol, Sandía, Brócoli, Melón, Lechuga, Cebolla Puerro, Zuquini, Sambo, Col, Coliflor, Papa, Pimiento o Tomate Riñón \rangle

H1=Tiempo de Producción

R1=Corto

Cx= Rábano

Cod_C	Cod_H	Cond_R
Cx	H1	Corto

Motor de inferencia

El motor de inferencia consta en uno de sus componentes por:

Módulo de almacenamiento de la información

```
Private Sub GuardarRespuesta()  
    Dim MS As String  
    MS = "default"  
    Try  
        connmy.Open()  
        Dim sqlIns As String  
        sqlIns = "GUARDAR_RESP"  
        Dim cmd As New MySqlCommand(sqlIns, connmy)  
        cmd.CommandType = Data.CommandType.StoredProcedure  
        cmd.Parameters.Add("@pCOD_H", MySqlDbType.VarChar)  
        cmd.Parameters.Add("@pRESP", MySqlDbType.VarChar)  
        cmd.Parameters.Add("@pPOR", MySqlDbType.VarChar)  
        cmd.Parameters(0).Value = Preg  
        cmd.Parameters(1).Value = ValResp()  
        cmd.Parameters(2).Value = PP  
        cmd.ExecuteNonQuery()  
    Catch exe As Exception  
        MsgBox(exe.Message, MsgBoxStyle.Critical, "Error")  
    Finally  
        connmy.Close()  
    End Try  
End Sub
```

Y la presentación de la respuesta

```
Private Sub mostrarresultados()  
    Try  
        connmy.Open()  
        Dim sqlCon As String  
        sqlCon = "VER_RESULTADO"  
        Dim cmd As New MySqlCommand(sqlCon, connmy)  
        cmd.CommandType = Data.CommandType.StoredProcedure  
        Dim reader As MySqlDataReader = cmd.ExecuteReader()  
        If reader.Read() Then  
            Cod_Cultivo = CStr(reader("Cod_C"))  
            Porcent = CStr(reader("SUM(Porcentaje)"))  
        End If  
    Catch exe As Exception  
        MsgBox(exe.Message, MsgBoxStyle.Critical, "Error")  
    Finally  
        connmy.Close()  
    End Try
```

CONCLUSIONES

1. El análisis comparativo permitió determinar a CommonKADS como la mejor metodología para el desarrollo del sistema experto ya que proporciona herramientas más adecuadas para el desarrollo de sistemas basados en el conocimiento de esta manera mejorando la productividad.
2. CommonKADS provee mejores características en cuanto productividad con un 80% de cumplimiento de los parámetros establecidos ante el 65% de Buchanan.
3. Al realizar un análisis de la productividad en el desarrollo de un sistema experto con cierta metodología permite determinar con claridad las prestaciones que nos brinda dicha metodología, y debido a su relevancia se ha considerado los factores más importantes entre los cuales se destaca: Disponibilidad de información, Líneas de código, Tiempo de desarrollo, Esfuerzo de desarrollo y Costo.
4. En el caso de estas metodologías para desarrollo de sistemas expertos la disponibilidad de información es el aspecto más importante para la aplicación de determinada metodología.
5. Los prototipos son herramientas útiles para el análisis de los factores de productividad de casos pequeños de estudio que facilitan la comparación para el desarrollo de un proyecto.
6. Se desarrolló prototipos, en los cuales se escogió los procesos más importantes para realizar el análisis de Productividad.
7. El uso de la metodología CommonKADS para analizar y desarrollar el sistema experto, facilita el modelado del conocimiento. La utilización de la metodología permite capturar los principales rasgos del sistema dentro del entorno de aplicación, además de la facilidad que brinda en la estructuración y gestión del mismo.
8. La colaboración del experto humano se facilita al brindarle una correcta información de los objetivos del sistema y la funcionalidad del mismo.
9. El Sistema Experto es utilizado como apoyo y referencia en la selección de cultivos a ser sembrados por usuarios con o sin experiencia.

10. El sistema experto desarrollado, tiene limitaciones porque la información obtenida para construir la base del conocimiento y su validación fue proporcionada por un solo experto y además las restricciones son específicas: inversión, ciclo de cultivos, tipo de riego etc.

RECOMENDACIONES

1. Al realizar un análisis comparativo de productividad entre metodología, se debe escoger muy bien los parámetros de comparación, siendo estos de fácil medición.
2. El sistema solo encierra el cuidado 5 factores de decisión, pero a futuro se puede agregar otros factores como historial de siembra, producción de la zona, etc.
3. Es recomendable guiarse en estudios realizados de las diferentes metodologías, que nos ayuden a familiarizarnos con el desarrollo del sistema usando dicha metodología.
4. Para realizar una evaluación apropiada es recomendable analizar los datos sistemáticamente y de acuerdo a los indicadores propuestos, mediante la utilización de métodos matemáticos y estadísticos.

GLOSARIO DE TÉRMINOS

Experto: es una persona con un conocimiento amplio o aptitud en un área particular del conocimiento.

Inferencia: Es una evaluación que realiza la mente entre expresiones bien formadas de un lenguaje.

Conocimiento: Hechos, o datos de información adquiridos por una persona a través de la experiencia o la educación, la comprensión teórica o práctica de un tema u objeto de la realidad.

Sistema Experto: Es una tecnología diseñada para extraer los conocimientos de un experto y con ellos elaborar un programa para la computadora en forma de reglas del tipo IF-THEM.

MYSQL: Sistema de gestión de bases de datos SQL.

SQL Structured Query Language; lenguaje de consulta estructurada, es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones en ellas.

BIBLIOGRAFÍA

- Barcelo Garcia, M. (2001). *Inteligencia artificial*. Catalunya: UOC.
- Biblionetica*. (26 de 1 de 2014). Obtenido de http://187.174.253.10/Biblionetica/diccionario/tecnologia/experto/tipos_se.htm
- DURKIN, J. (2014). *INGENIERIA DEL CONOCIMIENTO SEGÚN JHON DURKIN*. Obtenido de profesores.fi-.unam.mx/jareyc/SistExp/MetodologiadesarrolloSE.doc
- Henaó Cálad, M. (2001). *Universitat Politècnica de València*. Obtenido de <http://users.dsic.upv.es/grupos/ia/sma/thesis/pdf/TesisMonicaH.pdf>
- Hernando, D. (6 de 4 de 2004). *divulga-ia*. Obtenido de <http://www.divulga-ia.com/cursos/cursos.xml?numero=2&nombre=2003-9-26a&numLecc=1>
- Hidalgo, V., & Casigña, R. (2014). *Dspace ESPOCH*. Obtenido de <http://dspace.esPOCH.edu.ec/handle/123456789/3613>
- Jara Izurieta, E. (2014). *Dspace Epoch*. Obtenido de <http://dspace.esPOCH.edu.ec/handle/123456789/3554>
- JARAMILLO ESTRADA, M. (2013). *Dspace Epoch*. Obtenido de <http://dspace.esPOCH.edu.ec/bitstream/123456789/2530/1/18T00503.pdf>
- Jimenez, J. (29 de 1 de 2014). *joserafaeljimenez*. Obtenido de http://joserafaeljimenez.com/portfolio/portfolio/usm/J.R.Jimenez_USM.pdf
- Justo, F. (6 de 4 de 2004). *Aplicaciones, ventajas y limitaciones de los sistemas expertos*. Obtenido de <http://efelix.iespana.es/efelix/expertaplicaciones.htm>
- King, D., & Harmon, P. (1988). *Sistemas expertos: aplicaciones de la inteligencia artificial en la actividad empresarial*. Madrid: Diaz de Santos SA.
- Montes Castro, J. (6 de 2004). *Sistemas expertos (SE) | GestioPolis*. Obtenido de <http://www.gestiopolis.com/canales2/gerencia/1/sisexp.htm>
- Montes Cerra, M. C. (2003). *eafit*. Obtenido de http://dis.eafit.edu.co/labs/labgic/ARTICULOS_%20PUBLICAR/Sistemas%20expertos.doc
- Moret Bonillo, V. (2005). *Universidad de Coruña*. Obtenido de <http://www.dc.fi.udc.es/muc/sites/www.dc.fi.udc.es.muc/files/Val-y-Us.doc>
- Naïm, P., Willemin, P.-H., Leray, P., Pourret, O., & Becker, A. (2004). *Réseaux bayésiens*. Eyrolles.
- Núñez Zamora, W. R., & Sánchez Lozada, L. R. (2015). *Dspace ESPOCH*. Obtenido de <http://dspace.esPOCH.edu.ec/handle/123456789/4379#sthash.4GuLACfq.dpuf>
- Pino, R., Gómez, A., & Martínez, N. (2001). *Introducción a la inteligencia artificial*. Oviedo: Servicio de publicaciones Universidad de Oviedo. Obtenido de

http://books.google.es/books?id=RKqLMCw3IUkC&printsec=frontcover&dq=sistemas+expertos&hl=es&sa=X&ei=z2qaUfe_FoXJ4AOuhIGADQ&ved=0CDwQ6AEwAg#v=onepage&q&f=false

Rivas Echeverría, F. (26 de 1 de 2014). *webdelprofesor*. Obtenido de http://webdelprofesor.ula.ve/economia/guillenr/inteligencia/sist_expert_2.pdf

Rodríguez, J. (2014). *Serie Científica*. Obtenido de <http://publicaciones.uci.cu/index.php/SC/article/view/868/583>

Samaniego Larrea, M. (2015). *DSpace ESPOCH*. Obtenido de <http://dspace.esPOCH.edu.ec/handle/123456789/4376>

Samper Márquez, J. J. (6 de 3 de 2004). *redcientifica*. Obtenido de <http://www.redcientifica.com/doc/doc199908210001.html>

Velasco Ortega, A. (26 de 1 de 2014). *Evolución de los Sistemas Expertos*. Obtenido de <http://sistemasexpertosevolucion.blogspot.com>

ANEXOS