



**SEP**

SECRETARÍA DE  
EDUCACIÓN PÚBLICA

**TECNOLÓGICO NACIONAL DE MÉXICO**



**INSTITUTO TECNOLÓGICO  
DE CIUDAD MADERO**

**DIVISIÓN DE ESTUDIOS DE POSGRADO E  
INVESTIGACIÓN**

TESIS

**DESARROLLO DE ESQUEMAS DE ARGUMENTACIÓN PARA  
SISTEMAS DE RECOMENDACIÓN EN TOMA DE DECISIONES**

PARA OBTENER EL GRADO DE  
**MAESTRÍA EN CIENCIAS DE LA COMPUTACIÓN**

PRESENTA

**I.S.C. TEODORO EDUARDO MACIAS ESCOBAR**

**G08070685**

DIRECTOR DE TESIS

**DRA. LAURA CRUZ REYES**

CO-DIRECTOR DE TESIS

**DRA. MARÍA LUCILA MORALES RODRÍGUEZ**

CD. MADERO, TAMPS., MÉXICO

OCTUBRE 2016



Cd. Madero, Tamps; a 29 de Septiembre de 2016.

OFICIO No.: U5.180/16  
AREA: DIVISIÓN DE ESTUDIOS  
DE POSGRADO E INVESTIGACIÓN  
ASUNTO: AUTORIZACIÓN DE IMPRESIÓN DE TESIS

**ING. TEODORO EDUARDO MACIAS ESCOBAR**  
**NO. DE CONTROL G08070685**  
**P R E S E N T E**

Me es grato comunicarle que después de la revisión realizada por el Jurado designado para su examen de grado de Maestría en Ciencias de la Computación, el cual está integrado por los siguientes catedráticos:

PRESIDENTE :	DRA. CLAUDIA GUADALUPE GÓMEZ SANTILLÁN
SECRETARIO :	DRA. GUADALUPE CASTILLA VALDEZ
VOCAL :	DRA. LAURA CRUZ REYES
SUPLENTE	DRA. MARÍA LUCILA MORALES RODRÍGUEZ
DIRECTORA DE TESIS:	DRA. LAURA CRUZ REYES
CO-DIRECTORA DE TESIS:	DRA. MARÍA LUCILA MORALES RODRÍGUEZ

Se acordó autorizar la impresión de su tesis titulada:

**"DESARROLLO DE ESQUEMAS DE ARGUMENTACIÓN PARA SISTEMAS DE RECOMENDACIÓN EN TOMA DE DECISIONES"**

Es muy satisfactorio para la División de Estudios de Posgrado e Investigación compartir con Usted el logro de esta meta.

Espero que continúe con éxito su desarrollo profesional y dedique su experiencia e inteligencia en beneficio de México.

**ATENTAMENTE**  
"POR MI PATRIA Y POR MI BIEN"®

**DRA. ADRIANA ISABEL REYES DE LA TORRE**  
**JEFA DE LA DIVISIÓN DE ESTUDIOS**  
**DE POSGRADO E INVESTIGACIÓN**



c.c.p.- Archivo  
Minuta

AIRT'NLCO ' amhl



Ave. 1° de Mayo y Sor Juana I. de la Cruz Col. Los Mangos, C.P. 89440 Cd. Madero, Tam.  
Tel. (833) 357 48 20. e-mail: itcm@itcm.edu.mx  
www.itcm.edu.mx



# DECLARACIÓN DE ORIGINALIDAD

Declaro que este documento de tesis es un trabajo original, indicando de manera explícita en las referencias bibliográficas los datos de las publicaciones, así como sus autores, de aquellos trabajos a los cuales se hizo cita y referencia.

Acepto total responsabilidad en caso de infringir con las leyes de derechos de terceros, así como cualquier reclamación derivada de este documento de tesis que este en relación con los derechos de propiedad intelectual, exonerando de cualquier responsabilidad tanto a mi director de tesis como al Instituto Tecnológico de Ciudad Madero.



---

I.S.C. Teodoro Eduardo Macias Escobar

21 de Octubre de 2016, Cd. Madero, Tamps.

## DEDICATORIA

*Dedico este trabajo a mi familia*

*A mi padre, Teodoro Eduardo Macias Fernández, por ser un ejemplo de responsabilidad, dedicación y humildad y por enseñarme que cualquier objetivo se puede lograr si uno se esfuerza lo suficiente.*

*A mi madre, Norma Leticia Escobar Gutiérrez, por apoyarme siempre cuando más necesitaba ayuda y ser siempre un ejemplo de amor interminable, fortaleza, alegría y el enseñarme a nunca rendirme.*

*A mi hermana, Mariana Macias Escobar, por siempre estar a mi lado apoyándome sin importar a todos los problemas que la vida nos hace enfrentar.*

## AGRADECIMIENTOS

*Agradezco al Consejo Nacional de Ciencia y Tecnología (CONACYT) y a la Dirección General de Educación Tecnológica (DGEST) por su apoyo durante todo el proceso que llevó a la conclusión de este documento.*

*También agradezco al Instituto Tecnológico de Ciudad Madero (ITCM), por todas las disposiciones, facilidades y recursos proporcionados durante el periodo en el cual ejercí mis estudios. De igual manera agradezco a todos los profesores que me apoyaron durante este proceso, los cuales contribuyeron a mi formación académica.*

*Igualmente estoy agradecido con los profesores que conforman el comité tutorial de esta tesis: Dra. Laura Cruz Reyes, Dra. María Lucila Morales Rodríguez, Dr. Nelson Rangel Valdez, Dra. Claudia Guadalupe Gómez Santillán y Dra. Guadalupe Castilla Valdez.*

*Estoy profundamente agradecido con mi directora de tesis, Dra. Laura Cruz Reyes, la cual siempre me apoyo y estuvo al pendiente de mi desarrollo como estudiante y persona, tiene todo mi respeto y admiración por su interminable deseo de avanzar y mejorar siempre. También quiero agradecer de igual manera a mi co-directora de tesis, Dra. María Lucila Morales Rodríguez, por el tiempo y esfuerzo que dedicó durante el desarrollo de este trabajo.*

*Muchas gracias a todas las personas que colaboraron en el desarrollo de este trabajo, en particular a mis compañeros de generación César Alejandro Guerrero Nava, Germán Castillo Guillen, Luis Norberto Sánchez De La Paz y Lucia Janeth Hernández González.*

*Finalmente quiero agradecer a mi familia, que siempre me ha apoyado cuando más la necesito, que ha soportado mis errores y defectos y que siempre ha estado a mi lado dándome ánimos y consejos.*

# Resumen

La toma de decisiones es un proceso complicado y muchas veces menospreciado, en el cual se corre el riesgo de enfrentar gastos innecesarios y obtener resultados de pobre calidad. En muchos de estos casos, el problema a enfrentar no puede ser resuelto en un tiempo razonable utilizando únicamente la capacidad humana, esto debido al que el tipo de problema es demasiado complejo y existe una enorme cantidad de alternativas a revisar. Por esta razón es necesario el apoyo de sistemas inteligentes.

Este trabajo tiene el objetivo de explicar el desarrollo e implementación de un prototipo de un sistema de recomendación para la toma de decisiones, el cual se basa en la información ofrecida por el tomador de decisiones para generar una recomendación a partir de las soluciones predefinidas mediante un método de apoyo a la decisión multicriterio. Dicho sistema también se apoya de la teoría de argumentación mediante el uso de múltiples esquemas de argumentación y un juego de diálogo para interactuar con el usuario, ya sea para defender dicha recomendación u obtener nueva información que pueda llevar a la generación de otra. De esta forma el sistema no solamente es capaz de generar recomendaciones, sino también puede interactuar con el tomador de decisiones para llegar a una solución satisfactoria.

Para comprobar lo anterior se decidió realizar una evaluación de la usabilidad del prototipo en dos grupos diferentes de personas cuyo conocimiento dentro del campo era variado. A cada grupo se le pidió introducirse al prototipo desarrollado y realizar dos problemas (uno introductorio general y otro específico para cada grupo), finalmente se les pidió calificar el sistema en diversos aspectos tales como diseño, satisfacción, usabilidad, entre otros.

El resultado de dicha evaluación generó resultados mayoritariamente positivos, recibiendo una calificación satisfactoria y una aceptación de la mayoría de los evaluadores. Sin embargo fue claro que el prototipo, aunque funcional, aún tenía varios aspectos que se podrían mejorar. Debido a esta razón se establecieron algunas recomendaciones a considerar para los trabajos futuros.

# Summary

The decision making process is very complex and usually underestimated, which brings the risk of unnecessary budget spending and poor quality results. In many cases the problem in question cannot be resolved on a reasonable time relying only on the human capacity, this is because the problem type is too complex and the number of alternatives in said problem is too much to handle. Therefore it is necessary the aiding of intelligent systems.

This work has the objective of explaining the design and implementation of a prototype for a decision-aiding recommender system, which is based on the information provided by the decision maker in order to generate a recommendation from a predefined group of solutions using a multicriteria decision aiding method. Said system is also supported by the argumentation theory using several argumentation schemes and a dialogue game to interact with the user, whether to defend said recommendation or to obtain new information that could lead to the generation of a new one. With this, the system is not only capable of creating recommendations, but it also can interact with the decision maker to reach a satisfying solution.

To substantiate what was previously mentioned it was decided to perform a usability evaluation of the prototype in two different groups of people whose knowledge of the topic is varied. Each group was asked to introduce themselves in the use of the developed prototype and solve two problems (a general introduction problem and a group specific problem); finally the evaluators were asked to grade the system in different aspects such as design, satisfaction, usability and more.

The result of said evaluation generated mostly positive results, receiving a satisfying score and an acceptance from the majority of the evaluators. However it was clear that the prototype, while functional, still had several aspects that could be improved. For that reason, some recommendations for future work were made.

# Tabla de contenido

<b>Capítulo 1. Introducción</b> .....	1
1.1 Definición del problema.....	2
1.2 Justificación y beneficios .....	2
1.3 Objetivo.....	4
1.3.1 Objetivo general.....	4
1.3.2 Objetivos específicos .....	5
1.4 Hipótesis.....	5
1.5 Alcances y limitaciones.....	5
1.6 Organización del documento .....	6
<b>Capítulo 2. Marco teórico</b> .....	7
2.1 Toma de decisiones.....	7
2.1.1 El problema de toma de decisiones.....	7
2.1.2 Sistemas de apoyo a la toma de decisión multicriterio .....	8
2.1.3 Métodos de análisis de decisión multicriterio .....	9
2.1.4 Modelado de preferencias .....	9
2.2 Conceptos de optimización multiobjetivo .....	11
2.3 Problema de cartera de proyectos públicos .....	14
2.3.1 Proyecto.....	14
2.3.2 Características de un problema de cartera de proyectos.....	14
2.4 Teoría de argumentación en la toma de decisiones.....	16
2.4.1 Sistema de recomendación .....	16
2.4.2 Teoría de la argumentación .....	17
2.5 Juegos de diálogo.....	20
2.5.1 Reglas del juego de diálogo.....	20
2.5.1.1 Reglas de locución .....	20
2.5.1.2 Reglas de compromiso .....	21
2.5.1.3 Reglas de diálogo .....	21
2.5.2 Diagrama de transición de estados.....	21
2.6 Estándares de prueba.....	22
2.6.1 Mayoría simple.....	23
2.6.2 Mayoría ponderada .....	23
2.6.3 Orden lexicográfico .....	23
2.6.4 Suma de pesos .....	24
2.6.5 TOPSIS .....	24



2.7 Esquemas de argumentación .....	27
2.7.1 Razonamiento abductivo .....	28
2.7.2 Desde una posición para saber.....	29
2.7.3 Desde una opinión experta .....	29
2.7.4 Evaluación muticriterio de parejas.....	30
2.7.5 Ad ignorantiam .....	30
2.7.6 Causa efecto.....	31
2.7.7 Desde la parcialidad .....	31
<b>Capítulo 3. Estado del arte .....</b>	<b>33</b>
3.1 “Argumentation of the decision made by several aggregation operators based on weights” .	33
3.2 “Multiple Criteria Decision Aiding: a Dialectical Perspective” .....	37
3.3 “A Decision Support System Framework for Public Project Portfolio Selection with Argumentation Theory” .....	43
3.4 Análisis comparativo.....	45
<b>Capítulo 4. Propuesta de solución.....</b>	<b>47</b>
4.1 Propuesta de caracterización de tareas cognitivas en la ayuda a la decisión .....	47
4.1.1 Modelo de evaluación de alternativas .....	47
4.1.2 Construcción de argumentos para la recomendación final .....	51
4.1.2.1 Reglas de juego de diálogo.....	52
4.1.2.2 Diagrama de transición de estados .....	53
4.1.2.3 Esquemas de argumentación.....	56
4.2 Propuesta de arquitectura .....	57
4.2.1 Diagrama modular – primera capa.....	58
4.2.2 Diagrama modular – segunda capa .....	60
4.2.2.1 Módulo de configuración .....	60
4.2.2.2 Módulo de diálogo .....	62
<b>Capítulo 5. Desarrollo del prototipo .....</b>	<b>64</b>
5.1 Configuración de una instancia de selección de cartera .....	65
5.2 Interfaz gráfica de usuario y mapa de navegación .....	66
5.3 Proceso de generación de la recomendación inicial.....	68
5.3.1 Lectura de la instancia.....	68
5.3.2 Definición del estándar de prueba a utilizar .....	70
5.3.3 Selección de la recomendación.....	72
5.3.4 Definición de las reglas del sistema de diálogo .....	73
5.3.5 Generación del diagrama de transición de estados .....	73
5.4 Juego de diálogo .....	74
5.4.1 Acciones del tomador de decisiones .....	75

5.4.2 Aplicación de los esquemas de argumentación .....	82
5.4.3 Generación y respuesta de las preguntas críticas .....	86
5.4.4 Actualización del estándar de prueba utilizado .....	87
5.4.5 Cambio de recomendación .....	88
5.5 Visualización de apoyo a la decisión .....	90
5.5.1 Características de las carteras .....	91
5.5.2 Condiciones actuales del juego de diálogo .....	92
5.5.3 Ayuda temática .....	92
<b>Capítulo 6. Experimentación y resultados</b> .....	94
6.1 Análisis de métodos de evaluación usados en literatura relacionada .....	94
6.2 Método de evaluación propuesto para recomendación de carteras .....	98
6.2.1 Primera etapa de evaluación y resultados .....	98
6.2.2 Segunda etapa de evaluación y resultados .....	100
<b>Capítulo 7. Conclusiones y trabajos futuros</b> .....	108
Conclusión y trabajos futuros .....	108
7.1 Conclusiones .....	108
7.2 Aportaciones .....	109
7.3 Trabajos futuros .....	110
<b>Referencias</b> .....	112
<b>Anexo A</b> Esquemas de argumentación de los estándares de prueba .....	115
<b>Anexo B</b> Ejemplo de aplicación de un esquema de argumentación en el prototipo .....	117
<b>Anexo C</b> Ejemplo de actualización de los valores de los datos de una instancia .....	119
<b>Anexo D</b> Ejemplo del desarrollo de un juego de diálogo en el prototipo .....	123
<b>Anexo E</b> Problema de introducción de la segunda etapa de evaluación .....	127
<b>Anexo F</b> Segundo problema de la segunda etapa de evaluación .....	131
<b>Anexo G</b> Instancias utilizadas en el juego de diálogo de ejemplo y en la segunda etapa de evaluación .....	134

# Listado de figuras

<b>Figura 1.1:</b> Arquitectura del macro-proyecto .....	3
<b>Figura 2.1:</b> Esquema de Soporte para ayuda a la Toma de Decisiones.....	16
<b>Figura 2.2:</b> Diagrama de Transición de Estados, traducido de [Ouerdane, 2009] .....	22
<b>Figura 3.1:</b> Casos considerados por Labreuche [Labreuche, 2006].....	34
<b>Figura 3.2:</b> Ejemplo de un argumento para la evaluación de parejas, traducido de [Ouerdane, 2009] .....	38
<b>Figura 3.3:</b> Diferentes clases de esquemas de argumentación, traducido de [Ouerdane, 2009].....	38
<b>Figura 3.4:</b> Interfaz gráfica del sistema ArgDec [Ouerdane, 2009].....	43
<b>Figura 3.5:</b> Marco de referencia propuesto, traducido de [Cruz-Reyes et al., 2014] .....	44
<b>Figura 4.1:</b> Esquema general de un framework de apoyo a la decisión. ....	47
<b>Figura 4.2:</b> Diagrama de Transición de Estados 1 (DTE1). ....	54
<b>Figura 4.3:</b> Diagrama de Flujo de DTE1.....	54
<b>Figura 4.4:</b> Diagrama de Transición de Estados 2 (DTE2). ....	55
<b>Figura 4.5:</b> Diagrama de Flujo de DTE2.....	56
<b>Figura 4.6:</b> Propuesta de solución.....	58
<b>Figura 4.7:</b> Diagrama Modular – Primera Capa .....	59
<b>Figura 4.8:</b> Diagrama Modular – Segunda Capa .....	60
<b>Figura 4.9:</b> Definición de las Reglas de Locución.....	61
<b>Figura 4.10:</b> Generar Diagramas de Transición de Estados .....	61
<b>Figura 4.11:</b> Definir Estándar de Prueba.....	62
<b>Figura 4.12:</b> Recomendar Solución .....	62
<b>Figura 4.13:</b> Estructura del juego de diálogo .....	64
<b>Figura 4.14:</b> Estructura básica de un esquemas de argumentación.....	64
<b>Figura 5.1:</b> Ventana principal.....	66
<b>Figura 5.2:</b> Mapa de navegación.....	68
<b>Figura 5.3:</b> Inicio de ejecución del sistema .....	69
<b>Figura 5.4:</b> Mensaje del sistema después de la lectura de la instancia.....	70
<b>Figura 5.5:</b> Generación de la recomendación inicial. ....	75
<b>Figura 5.6:</b> Opciones de interacción. ....	75
<b>Figura 5.7:</b> Actualización del estándar de prueba.....	88
<b>Figura 5.8:</b> Actualización de la tabla de desempeño.....	90

<b>Figura 5.9:</b> Visualización de carteras.....	91
<b>Figura 5.10:</b> Visualización de los datos de la conversación.....	93
<b>Figura 5.11:</b> Visualización de la ventana de ayuda. ....	93
<b>Figura 6.1:</b> Promedio por pregunta del primer grupo. ....	103
<b>Figura 6.2:</b> Promedio por pregunta del segundo grupo.....	104
<b>Figura B.1:</b> Estándar de prueba actual y propiedades activas. ....	117
<b>Figura B.2:</b> Estándar de prueba actual y propiedades activas después de la modificación. ....	118
<b>Figura C.1:</b> Recomendación actual durante un juego de diálogo en el prototipo.....	119
<b>Figura C.2:</b> Datos actuales de la conversación durante un juego de diálogo en el prototipo. ....	120
<b>Figura C.3:</b> Actualización de los valores en la tabla de desempeño.....	120
<b>Figura C.4:</b> Nueva recomendación a partir de los valores actualizados. ....	121
<b>Figura C.5:</b> Datos actuales de la conversación después de la actualización de valores.....	122
<b>Figura D.1:</b> Inicio y paso 1 del juego de diálogo.....	124
<b>Figura D.2:</b> Pasos 2 a 7 del juego de diálogo. ....	125
<b>Figura D.3:</b> Pasos 8 a 11 del juego de diálogo. ....	125
<b>Figura D.4:</b> Pasos 12 a 16 del juego de diálogo. ....	126
<b>Figura D.5:</b> Pasos 17 hasta el final del juego de diálogo. ....	126

# Listado de tablas

<b>Tabla 2.1:</b> Esquema de Argumentación del razonamiento abductivo .....	28
<b>Tabla 2.2:</b> Esquema de Argumentación de un argumento desde una posición para saber .....	29
<b>Tabla 2.3:</b> Esquema de Argumentación de un argumento desde una opinión experta.....	30
<b>Tabla 2.4:</b> Esquema de Argumentación evaluación multicriterio de parejas .....	30
<b>Tabla 2.5:</b> Esquema de Argumentación de un argumento Ad Ignorantiam .....	31
<b>Tabla 2.6:</b> Esquema de Argumentación de un argumento causa-efecto.....	31
<b>Tabla 2.7:</b> Esquemas de Argumentación de un argumento desde la parcialidad.....	32
<b>Tabla 3.1:</b> Reglas de locución definidas en [Ouerdane, 2009] .....	40
<b>Tabla 3.2:</b> Análisis del Estado del Arte.....	46
<b>Tabla 4.1:</b> Estándares de prueba y sus propiedades .....	50
<b>Tabla 4.2:</b> Reglas de locución para el juego de diálogo.....	52
<b>Tabla 4.3:</b> Esquemas de argumentación.....	57
<b>Tabla 5.1:</b> Variables de la función <i>read</i> .....	69
<b>Tabla 5.2:</b> Condicionales de activación de las propiedades de los estándares de prueba.....	71
<b>Tabla 5.3:</b> Funciones de los estándares de prueba. ....	72
<b>Tabla 5.4:</b> Elementos de <i>Interaction Options</i> .....	76
<b>Tabla 5.5:</b> Acciones disponibles en <i>DM's Decision</i> .....	76
<b>Tabla 6.1:</b> Resultados de la investigación de evaluaciones previas.....	96
<b>Tabla 6.2:</b> Promedio por sección del primer grupo.....	104
<b>Tabla 6.3:</b> Promedio por sección del segundo grupo. ....	105
<b>Tabla 6.4:</b> Diferencia entre los promedios por pregunta de ambos grupos. ....	106
<b>Tabla 6.5:</b> Diferencia entre los promedios por sección de ambos grupos. ....	107
<b>Tabla A.1:</b> Esquema para un argumento desde mayoría simple [Ouerdane, 2009]. ....	115
<b>Tabla A.2:</b> Esquema para un argumento desde el método lexicográfico [Ouerdane, 2009]. ....	115
<b>Tabla A.3:</b> Esquema para un argumento desde mayoría ponderada [Ouerdane, 2009]. ....	115
<b>Tabla A.4:</b> Esquema para un argumento desde suma de pesos [Ouerdane, 2009].....	115
<b>Tabla A.5:</b> Esquema para un argumento desde TOPSIS.....	116

# Capítulo 1

## Introducción

---

---

Generalmente, las instituciones, empresas y organizaciones tienen un conjunto de acciones que buscan llevar a cabo, sin embargo, sólo existe una cantidad limitada de recursos que se pueden invertir para dichas acciones, lo que quiere decir que no se pueden apoyar todas éstas al mismo tiempo, sino que es necesario elegir un conjunto al cual se le va a dedicar dichos recursos. Esta situación recae en el tomador de decisiones, una persona o grupo encargado de decidir qué acciones deben ser apoyadas y cuáles no. Sin embargo, esta actividad no es tarea fácil ya que usualmente existe una cantidad tan grande de opciones que el razonamiento humano no es suficiente para analizarlas todas, lo cual conlleva a una mala decisión, lo cual a su vez puede tener consecuencias tales como:

- Desperdicio de recursos, los recursos son mal repartidos en las acciones incorrectas.
- Pérdida de ganancias, la empresa pierde las ganancias que les hubiera dado una elección correcta de acciones a invertir.

Este trabajo buscó contribuir al desarrollo de los sistemas de apoyo a la toma de decisiones y los sistemas de recomendación mediante la caracterización de tareas cognitivas involucradas en el proceso de ayuda a la decisión dentro de un sistema. Dichos sistemas deben permitir destinar los recursos disponibles a las acciones previstas, analizar las posibles soluciones de una manera sencilla, ofrecer soluciones que optimicen resultados y ganancias y al mismo tiempo satisfacer cualquier restricción o preferencia establecida por el tomador de decisión.

## 1.1 Definición del problema

Dado un Tomador de Decisiones  $DM$ ; un Problema de Optimización Multiobjetivo  $P$ ; y un Sistema de Recomendación  $R$ ,

¿Es posible desarrollar una estructura de argumentos que permita a un  $R$  proporcionar recomendaciones y explicaciones para un  $DM$  durante el proceso de solución de  $P$ ?

## 1.2 Justificación y beneficios

La evaluación y selección de un conjunto de proyectos con impacto social es una de las principales tareas dentro de las organizaciones gubernamentales, fundaciones, grupos enfocados a la investigación y desarrollo, entre otras. Generalmente existe una cantidad escasa de recursos que deben ser asignados a dichos proyectos de tal manera que se busca satisfacer la demanda y optimizar los beneficios.

Comúnmente, esto lleva a un proceso de toma de decisión, el cual cae en manos de la entidad decisora, quien decidirá la cantidad de recursos a asignar a cada uno de los proyectos posibles. Usualmente cuando este paso es subestimado, se toman decisiones de pobre calidad que llevan a consecuencias graves, por lo que es necesario llevar a cabo un estudio exhaustivo de los proyectos en cuestión. A pesar de la importancia económica de este problema, debido a su complejidad sólo se han elaborado formulaciones parciales e independientes del mismo.

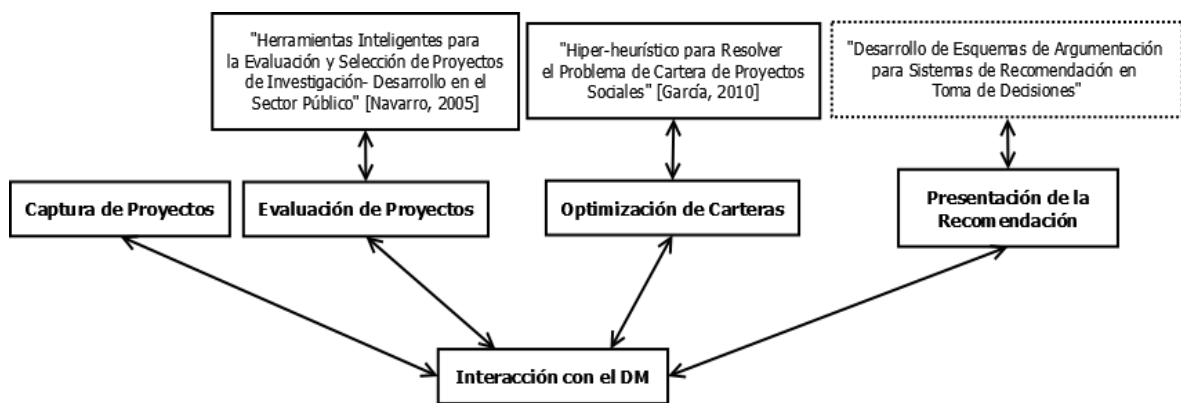
Por esta razón se han generado sistemas de apoyo a la decisión basados en métodos formales que permitan a las personas encargadas de tomar las decisiones tener herramientas confiables, efectivas y robustas que le permitan distribuir los recursos de la mejor manera posible.

Sin embargo, muchos de esos sistemas solo ofrecen una solución al usuario sin justificar la razón por la cual se ha tomado dicha decisión. En este proyecto se buscó desarrollar un sistema que no solo fuera capaz de generar una solución de alta calidad, sino también pudiera establecer un diálogo con el usuario tanto para defender dicha decisión como para obtener nueva información que le permitiera seleccionar aquella solución que se adecue más a las preferencias de éste.

Además, cabe mencionar que este trabajo está enfocado al problema de cartera de proyectos públicos (SPP por sus siglas en inglés), utilizando métodos de análisis de decisión multicriterio (MCDA por sus siglas en inglés) como apoyo. Dicho problema ha sido poco abordado debido a su dificultad, ya que un aspecto importante del SPP es que no solamente se requiere manipular datos objetivos, sino también datos subjetivos dados por el DM, lo cual hace más difícil la comparación entre proyectos. También la cantidad de proyectos y objetivos presentes en los problemas reales son usualmente de un tamaño más elevado de lo que puede manipular la gran mayoría de los sistemas de recomendación en la actualidad.

De igual forma se buscó apoyar a la consolidación del macro-proyecto formulado por el grupo de trabajo al que se pertenece con una herramienta que sea capaz de utilizarse en casos reales, por lo que es necesario abordar el problema con las variantes y dimensiones que esto implica.

En trabajos previos del grupo se han desarrollado métodos y modelos matemáticos que trabajan en condiciones y eventos artificiales, tal como se puede ver en la Figura 1.1, se planeó facilitar la interacción con usuarios reales apoyándose de éstos métodos, particularmente para la etapa de generación de recomendaciones.



**Figura 1.1:** Arquitectura del macro-proyecto

Con esta planeación, las organizaciones públicas serán capaces de repartir los recursos actuales dentro del entorno de trabajo, de tal forma que se optimicen las ganancias, satisfaciendo cada uno de los objetivos deseados, lo cual llevará a un mayor beneficio para



dichas organizaciones y para la sociedad. Además de poder entender las razones detrás de dicha elección al haber establecido un diálogo previo con el sistema de apoyo a la decisión.

Se planeó abordar el desarrollo de un sistema de apoyo a la decisión enfocada a la recomendación de soluciones para el SPP, buscando integrar aspectos teóricos con tecnologías emergentes, creando una herramienta robusta y eficiente que utilice métodos formales para la solución de un problema real. Normalmente los trabajos que se orientan en conjuntar la teoría con la práctica suelen abarcar sólo aspectos particulares del proceso general de la toma de decisiones, mientras que son pocos los trabajos que siguen un enfoque integral.

No se puede determinar un enfoque único para resolver todos los problemas, ya que debido a la variedad de éstos, ciertos enfoques funcionan mejor que otros en problemas particulares. Es importante elegir el enfoque adecuado para resolver diferentes tipos de problemas, con la ayuda de un sistema de apoyo a la decisión se puede analizar el problema para facilitar la elección del enfoque que más convenga a dicha situación, sin embargo, elegir el enfoque correcto es complejo debido a la naturaleza subjetiva del problema y a las preferencias establecidas por los tomadores de decisión. De igual manera, para el desarrollo de un subsistema de recomendación que forme parte de un sistema de apoyo a la decisión también se debe considerar que diferentes problemas requieren diferentes enfoques que soporten la recomendación. En este trabajo se propuso la construcción de esquemas de argumentación para diferentes tipos de problema.

## **1.3 Objetivo**

A continuación, se hará mención del objetivo general de este proyecto, así como de los objetivos específicos que se han planteado.

### **1.3.1 Objetivo general**

Caracterizar tareas cognitivas involucradas en el proceso de ayuda a la decisión para guiar la interacción con usuarios en un Sistema de Recomendación.

### **1.3.2 Objetivos específicos**

- 1) Analizar modelos de caracterización de tareas cognitivas involucradas en el proceso de ayuda a la decisión (MCTC-AD).
- 2) Desarrollar una estrategia MCTC-AD basadas en argumentación que guie la interacción con usuarios en un sistema de recomendación para la toma de decisión en problemas de optimización.
- 3) Integrar en un prototipo de sistema de recomendación el MCTC-AD propuesto.

### **1.4 Hipótesis**

El prototipo desarrollado en este trabajo, el cual es un sistema de recomendación para la toma de decisiones apoyado en los esquemas de argumentación, le permite al usuario comprender a mayor profundidad y de manera más fácil el problema en cuestión, así como identificar más rápidamente dentro de un conjunto de soluciones aquella que se acopla mejor a sus preferencias y restricciones.

La aplicación de la teoría de la argumentación dentro de este prototipo mediante el uso de un juego de diálogo le permite al usuario no solo saber cuál de las opciones dentro del conjunto de soluciones es la mejor, sino también el comprender las razones de esto.

### **1.5 Alcances y limitaciones**

Este trabajo contempla como objetos de estudio al menos:

- Un problema de optimización
- Un esquema de caracterización de tareas cognitivas (Esquemas de Argumentación)
- Un sistema de recomendación

El sistema de recomendación implementado está acotado por los siguientes aspectos:

- Sólo considerará instancias pequeñas del problema de optimización con un máximo de 4 objetivos y 25 proyectos;
- La interacción estará restringida a un conjunto de diálogos predefinidos.
- La interfaz de usuario se implementará en un ambiente local;
- El uso del procesamiento de lenguaje natural no será considerado; y

- No incluirá ningún esquema de reducción de argumentos.

## **1.6 Organización del documento**

A continuación, se hará una descripción de la forma en la que está organizado este documento, así como una breve explicación de cada uno de los capítulos que lo componen.

El capítulo 2 contiene el marco teórico, dentro del cual se revisan aspectos de suma importancia para este trabajo tales como la toma de decisiones, el problema de cartera de proyectos públicos, la teoría de la argumentación, juegos de diálogo, estándares de prueba y esquemas de argumentación.

En el capítulo 3 se analiza el estado del arte, donde se hace enfoque en los trabajos relacionados a este proyecto.

Por otro lado, el capítulo 4 describe la arquitectura propuesta para el sistema de recomendación para la toma de decisiones en el que se detallan los diferentes elementos y módulos a utilizar. El capítulo 5 describe la implementación del prototipo diseñado, así como el funcionamiento de sus diferentes módulos y componentes.

Dentro del capítulo 6 se incluye todo lo relacionado al proceso de evaluación realizado para este proyecto, incluyendo tanto la investigación previa realizada con el objetivo de definir la forma de llevar a cabo este proceso, así como la realización del mismo, mostrando finalmente los resultados obtenidos.

Finalmente, en el capítulo 7 se establecen las conclusiones de este trabajo, así como las aportaciones generadas y el trabajo futuro para próximos proyectos que deseen abordar el tema al que está enfocado este documento.

# Capítulo 2

## Marco Teórico

---

---

Antes de introducirnos en el trabajo desarrollado en este proyecto, es necesario entender diversos conceptos relacionados a temas tales como la toma de decisiones, optimización multiobjetivo, estándares de prueba y los esquemas de argumentación. Esto con el objetivo de facilitar el entendimiento de las diversas acciones realizadas durante el proceso de diseño e implementación del prototipo de sistema de recomendación desarrollado en esta tesis. En este capítulo se mostrarán los conceptos necesarios para completar lo anteriormente mencionado.

### **2.1 Toma de decisiones**

Tomar una decisión consiste en elegir la mejor opción entre un conjunto de alternativas posibles, buscando decidir de manera racional decisiones difíciles y complejas [Keeney y Raiffa, 1993], apoyándose de disciplinas clásicas como la estadística, la economía, la matemática y la inteligencia artificial para desarrollar modelos o técnicas que permitan o faciliten la toma de decisión [Sánchez, 2007].

#### **2.1.1 El problema de toma de decisiones**

Generalmente las empresas no tienen los suficientes recursos para apoyar todas las acciones disponibles en un mismo momento, esto lleva a un problema de toma de decisión, donde es necesario revisar cada una de las acciones individualmente y buscar el conjunto de éstas que satisfaga las necesidades actuales de la mejor manera posible y así se maximice el impacto en la empresa, utilizando los recursos disponibles.

Según Chernoff [Chernoff, 1987], los problemas de decisión clásicos presentan como elementos básicos:

- Uno o varios objetivos por resolver.
- Un conjunto de alternativas o decisiones posibles para alcanzar dichos objetivos.
- Un conjunto de factores o estados de la naturaleza que definen el contexto en el que se plantea el problema de decisión.
- Un conjunto de valores de utilidad o consecuencias asociados a los pares formados por cada alternativa y estado de la naturaleza.

De acuerdo con Sánchez [Sánchez, 2007], la teoría de la decisión establece una serie de criterios que permiten clasificar a los problemas de acuerdo a los siguientes puntos de vista:

- Según el número de criterios o atributos que se han de valorar en la toma de decisión.
- Según el ambiente de decisión en el que se han de tomar las decisiones.
- Según el número de expertos que participan en el proceso de decisión.

En el proceso de toma de decisión existe el Tomador de Decisión (DM), el cual es la persona o grupo cuyas preferencias son determinantes en la solución de problemas con múltiples objetivos, los cuales en ciertos casos muestran conflictos entre si [Fernández et al., 2011]; el DM será quien tome la decisión final y elija la solución que más le parezca conveniente a partir de las preferencias previamente establecidas.

### **2.1.2 Sistemas de apoyo a la toma de decisión multicriterio**

Los sistemas de apoyo a la toma de decisión multicriterio (MCDSS, por sus siglas en inglés) son herramientas computacionales utilizadas para analizar problemas de decisión de alta complejidad en un tiempo razonable en comparación de lo que tardarían en analizarse sin utilizar dichas herramientas [López, et al., 2008]. Son un subconjunto de los sistemas de apoyo a la toma de decisión (DSS) los cuales están enfocados en los problemas multiobjetivo, debido a esto, también posee las mismas propiedades y funciones que un DSS, tales como:

- Extraer y manipular información
- Soportar un conjunto de decisiones

- Es capaz de adaptarse a los cambios en el ambiente
- Se apoya en métodos cuantitativos
- Ofrece un conjunto de tamaño reducido de soluciones de buena calidad

Debido a la capacidad de manejar problemas de alta complejidad de los MCDSS, estas herramientas son comúnmente utilizadas en los problemas de cartera de proyectos públicos, los cuales se verán a más adelante dentro de este capítulo.

### **2.1.3 Métodos de análisis de decisión multicriterio**

Un componente relevante de MCDSS son los métodos MCDA, ya que estas herramientas al menos utilizan uno de estos modelos para poder analizar problemas multicriterio. El análisis de decisión multicriterio, es un término que incluye una colección de conceptos, métodos y técnicas que persiguen ayudar a los individuos o grupos a tomar decisiones que implican diferentes puntos de vista en conflicto y múltiples agentes interesados [Belton y Stewart, 2002]. Entre los métodos MCDA destacan el método NOSGA2 [Fernández et al, 2011] y los métodos ELECTRE [Roy, 1991].

Cinco componentes esta involucrados en los métodos de MCDA

- Meta
- Tomador de Decisión
- Alternativas o acciones
- Preferencias
- Conjunto de soluciones basadas en las preferencias

### **2.1.4 Modelado de preferencias**

Parte de los criterios para tomar una decisión, son las preferencias que tenga el tomador de decisiones dado un problema, esto con el objetivo de que se puede construir un modelo, tal que al aplicarse respete las recomendaciones establecidas por el DM.

El modelado de las preferencias dentro de la toma de decisión es una área de trabajo donde los expertos, en base a su conocimientos, habilidades y experiencias, valoran las

alternativas y establecen generalmente un orden de preferencia de cada una como solución preliminar del problema [Doyle, 2004; Oztürk et al., 2005; Roubens y Vincke, 1985].

A partir del modelado de preferencias, mediante métodos MCDA es posible expresar una relación de superación para cada par de soluciones factibles. De manera formal se dice que  $x$  supera a  $y$  ( $xSy$ ).

### *Relaciones de superación*

Fernández propone un sistema relacional de preferencias que toma como punto de partida el cálculo del grado de veracidad  $\sigma(x, y)$  para cada par de alternativas [Fernández, et al., 2011]. La función  $\sigma(x, y)$  mide el grado de verdad de la afirmación “ $x$  es al menos tan buena como  $y$ ”. El método ELECTRE es el método MCDA comúnmente usado para calcular  $\sigma(x, y)$  [Marzouk, 2011]. Primero se evalúa si existen suficientes motivos a favor de  $x$ ; se trata de la concordancia en términos de cantidad de criterios o de peso de los criterios favorables. Enseguida se evalúa si no se registran opiniones fuertemente contrarias a  $x$ ; esta es la discordancia, ejercida como derecho de veto, medida como una gran diferencia de puntaje en  $x$ , para alguno de los criterios bajo análisis. Finalmente la concordancia y la discordancia se integran para la obtención de  $\sigma(x, y)$ .

Para cada par de soluciones se establece un tipo de relación de preferencia el cual permite después definir el conjunto de soluciones que se apegan más a las restricciones y preferencias establecidas por el usuario. El modelo de Fernández establece el tipo de relación de preferencia considerando: el grado de veracidad  $\sigma$ , el grado de dominancia (ver sección 2.2) y las preferencias del DM. Existen métodos MCDA que permiten mapear las preferencias en parámetros del sistema relacional de preferencias de Fernández.

Considérese un umbral de credibilidad aceptable  $\lambda$  para todas las relaciones, un parámetro de asimetría  $\beta$  para la preferencia estricta y un parámetro de simetría  $\varepsilon$  para la indiferencia, donde  $0 \leq \varepsilon \leq \beta \leq \lambda$  y  $\lambda > 0.5$ , el modelo identifica una de las siguientes relaciones de preferencia para cada par de soluciones  $(x, y)$ .

*Preferencia estricta:* denotado  $xPy$ , es donde el DM tiene claro y justificada la elección de  $x$  sobre  $y$ , se define como una disyunción de las siguientes condiciones:

- $x$  domina  $y$
- $\sigma(x, y) \geq \lambda \wedge \sigma(y, x) < 0.5$
- $\sigma(x, y) \geq \lambda \wedge [0.5 \leq \sigma(y, x) < \lambda] \wedge [\sigma(x, y) - \sigma(y, x)] \geq \beta$ .

*Indiferencia:* denotado  $xIy$ , es donde el DM tiene clara y justificada una equivalencia entre ambas opciones y se define como una conjunción de las siguientes condiciones:

- $\sigma(x, y) \geq \lambda \wedge \sigma(y, x) \geq \lambda$
- $|\sigma(x, y) - \sigma(y, x)| \leq \varepsilon$ .

*Preferencia débil:* existe una duda entre  $xPy$  y  $xIy$ , representado como  $xQy$  y se define como una conjunción de las siguientes condiciones:

- $\sigma(x, y) \geq \lambda \wedge \sigma(x, y) > \sigma(y, x)$
- $\neg xPy \wedge \neg xIy$ .

*Incomparabilidad:* denotado  $xRy$ , para el DM existe un gran nivel de heterogeneidad entre las soluciones y no se puede escoger directamente una sobre otra:

- $\sigma(x, y)$  como  $xRy \Rightarrow \sigma(x, y) < 0.5 \wedge \sigma(y, x) < 0.5$ .

*k-preferencia:* existe una duda entre  $xPy$  y  $xRy$ , representado como  $xKy$  y se define como una conjunción de las siguientes condiciones:

- $0.5 \leq \sigma(x, y) < \lambda$
- $\sigma(y, x) < 0.5$
- $\sigma(x, y) - \sigma(y, x) > \beta/2$ .

## 2.2 Conceptos de optimización multiobjetivo

Muchos de los problemas reales de toma de decisión en el mundo involucran varios objetivos manejados al mismo tiempo [Pérez et al., 2007], los cuales usualmente se encuentran en conflicto, tanto por sus restricciones como por las preferencias determinadas por el cliente, por esa razón determinar la mejor alternativa no es fácil, ya que pueden existir casos en los que no exista en si una solución óptima como tal, sino que existan un



conjunto de soluciones óptimas de las cuales no se puede definir cuál es mejor que otra para un tomador de decisiones.

Según Kalyanmoy Deb [Deb, 2001], la optimización es el encontrar una o más soluciones factibles que corresponden a valores extremos de uno o más objetivos, ya sea para problemas de un sólo objetivo (optimización mono-objetivo) o de múltiples objetivos (optimización multiobjetivo).

En un problema multiobjetivo se busca el vector de variables de decisión  $\vec{x} = [x_1, x_2, \dots, x_N]^T$  que optimice [Serrano, 2007]:

$$F(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_k(\vec{x})], \quad f_i: \mathbb{R}^n \rightarrow \mathbb{R} \quad (1)$$

Sujeto a:

$$\text{Desigualdades } g_i(\vec{x}) \leq 0; \quad i = 1, 2, \dots, m \quad (2)$$

$$\text{Igualdades } h_j(\vec{x}) = 0; \quad j = 1, 2, \dots, p \quad (3)$$

Donde:

- $k$  es el número de funciones objetivo
- $n$  es el número de variables de decisión
- $m$  es el número de restricciones de desigualdad
- $p$  es el número de restricciones de igualdad

Existen tres tipos de problema multiobjetivo:

- Minimizar todas las funciones objetivo
- Maximizar todas las funciones objetivo
- Minimizar algunas y maximizar las funciones objetivo restantes

Para abordar la solución de los problemas de optimización multiobjetivo (MOP por sus siglas en inglés), existen algoritmos exactos y aproximados. Dentro de los algoritmos

aproximados, se encuentran estrategias deterministas y estocásticas. Una de las estrategias estocásticas de mayor popularidad las constituyen los algoritmos evolutivos.

Para los MOP, se busca un conjunto de soluciones que ofrezcan un equilibrio óptimo entre los diferentes objetivos, una representación común de dichas soluciones es la gráfica de sus valores solución conocida como *Frente de Pareto*, por esta razón dichas soluciones son conocidas como soluciones *óptimas de Pareto* o soluciones no-dominadas [Van Veldhuizen y Lamont, 1998].

Algunas definiciones relevantes con respecto a la optimización multiobjetivo se describen a continuación:

*Dominancia*: sean  $A$  y  $B$  soluciones factibles, se dice que  $A$  domina  $B$  en un problema con múltiples objetivos si se cumplen las siguientes condiciones:

- La solución  $A$  es al menos tan bueno como  $B$  en cada uno de los objetivos
- La solución  $A$  es estrictamente mejor que  $B$  con respecto al menos un objetivo

Si alguna de estas condiciones no se cumple, entonces no existe dominancia; al comparar dos soluciones sólo existen tres posibles resultados

- $A$  domina a  $B$
- $A$  es dominada por  $B$
- $A$  y  $B$  no se dominan (son no dominadas entre sí)

*Óptimo y Frente de Pareto*: se le llama *óptimo de Pareto* a un vector de variables de decisión cuando no existe otro vector factible de variables que lo domine, sin embargo es usual que este concepto ofrezca más de una solución, a dicho conjunto se le denomina *Conjunto de óptimos de Pareto* ( $P^*$ ). La representación de las funciones objetivo cuyos vectores de variables de decisión son no-dominados y además están en el conjunto de óptimos de Pareto es llamado el *Frente de Pareto*.

Todas las soluciones pertenecientes al frente de Pareto son igualmente buenas, y no se puede especificar si alguna de las soluciones es preferible a las otras, excepto en aquellos casos en que el DM haya definido una preferencia a priori [López, 2013].

## **2.3 Problema de cartera de proyectos públicos**

Normalmente en diversas organizaciones, existen un conjunto de proyectos que buscan llevarse a cabo, sin embargo, la cantidad de recursos que se poseen en ese momento no permiten que todos los proyectos puedan iniciarse al mismo tiempo, lo cual causa un problema, ya que es necesario saber qué conjunto de proyectos resulta más conveniente apoyar para satisfacer las demandas de dicha organización y maximizar el efecto en la sociedad, a esto se le conoce como un *problema de cartera de proyectos públicos* (SPP por sus siglas en inglés).

Los factores que se deben considerar en este problema son las preferencias que tenga el tomador de decisión con respecto a cada uno de los proyectos disponibles y las diferentes restricciones existentes, ya sea un límite presupuestal general, por área o región.

Otra dificultad a considerar para en este problema consiste en que muchas veces se genera un gran cantidad de soluciones posibles factibles, lo cual no permite analizarlas en un tiempo aceptable y una elección incorrecta puede llevar a pérdida de ganancias y recursos.

### **2.3.1 Proyecto**

Un *proyecto* es un proceso temporal, único e irreplicable que persigue un conjunto específico de objetivos [Carazo et al., 2010], por otra parte una *cartera de proyectos* es un conjunto de proyectos realizados. Debido a esto, dichos proyectos comparten los recursos disponibles y existe la posibilidad de que varios de los proyectos se complementen entre ellos, debido a esto es necesario considerar los proyectos de la cartera no sólo de manera individual sino en subconjuntos.

### **2.3.2 Características de un problema de cartera de proyectos**

La selección de proyectos de una cartera de proyectos públicos necesita de un tratamiento especial por las siguientes razones [Fernández y Navarro, 2002]:

- La calidad de los proyectos es generalmente descrita por múltiples criterios que frecuentemente están en conflicto.

- Al ser varios de sus requerimientos subjetivos, estos no se conocen de manera exacta, por ejemplo, el impacto en el bienestar social.
- Varios proyectos al ser heterogéneos resultan difíciles de comparar.

El problema objeto de estudio en esta investigación es un caso particular del problema de cartera de proyectos, cuya definición formal se presenta a continuación. Consideremos una cartera de proyectos a implementar de los cuales el DM debe elegir ciertos proyectos de dicho conjunto con el objetivo de obtener el mayor grado de satisfacción sin excederse de los recursos actuales. Considérese un vector  $p$ -dimensional  $f(i) = (f_1(i), f_2(i), f_3(i), \dots, f_p(i))$ , donde cada  $f_j(i)$  representa la contribución del proyecto  $i$  para el objetivo  $j$ , por otra parte el portafolio es usualmente modelado como un vector binario  $x = (x_1, x_2, \dots, x_N)$  donde  $N$  es el total de proyectos disponibles,  $x_i$  tiene un valor de 1 si el proyecto  $i$  es apoyado y 0 en caso contrario.

Teniendo un límite de recursos  $B$  y un costo  $c_i$  para cada proyecto, la restricción de presupuesto se puede establecer de la siguiente manera:

$$\left( \sum_{i=1}^N x_i c_i \right) \leq B \quad (4)$$

Además de la restricción de presupuesto, también deben tenerse en cuenta las restricciones por área  $k$ , en otras palabras, que rango (máximo  $U_k$  y mínimo  $L_k$ ) de la cantidad total del presupuesto se puede destinar a dicha área. Considérese también un vector  $a = (a_1, a_2, \dots, a_N)$  el cual establece a que área corresponde cada proyecto. Para lo anterior se formula la siguiente restricción:

$$L_k \leq \sum_{i=1}^N x_i g_i(k) c_i \leq U_k \quad (5)$$

Donde  $g$  está definido por:

$$g_i(k) \begin{cases} 1 & \text{si } a_i = k \\ 0 & \text{si no} \end{cases} \quad (6)$$

De igual manera, a cada proyecto le corresponde una región a la cual beneficiará. A cada una de las regiones existentes se les aplica un rango máximo y mínimo específico el cual no debe excederse. Tratando a las regiones de la misma forma que a las áreas, dichos rangos también formarán parte de las restricciones del problema.

## 2.4 Teoría de argumentación en la toma de decisiones

Los principales elementos involucrados en un esquema de soporte para ayuda a la toma de decisiones son [Ouerdane, 2009]: a) Sistemas de Recomendación, b) Teoría de la Argumentación, c) Esquemas de Argumentación y d) Juego de Diálogos, dichos puntos serán descritos dentro de este capítulo. La jerarquía del esquema de soporte entre los componentes para la ayuda a la toma de decisiones se puede verse en la Figura 2.1.

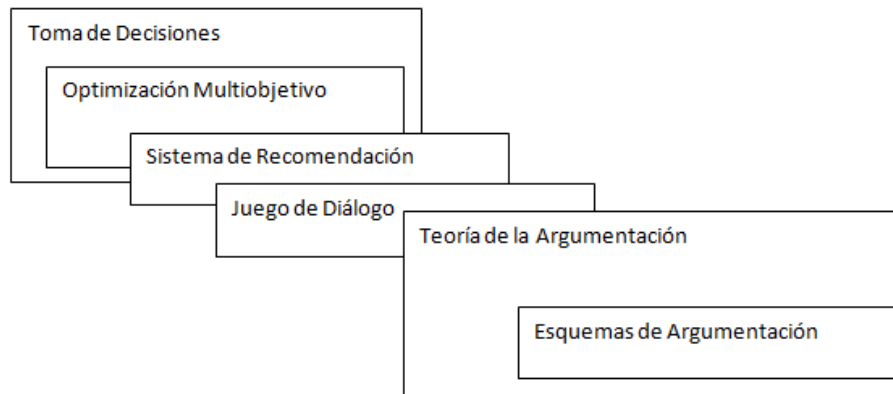


Figura 2.1: Esquema de Soporte para ayuda a la Toma de Decisiones

### 2.4.1 Sistema de recomendación

Un sistema de recomendación se enfoca en el apoyo a los usuarios para dirigirlos y que sean capaces de acceder a objetos relevantes, los cuales pertenecen a un conjunto demasiado grande para ser tomado sin ningún apoyo y que provocan un problema de sobrecarga de información al usuario. Estos sistemas intentan generar un modelo del usuario y aplicar diversas heurísticas para anticipar que información puede ser de interés para éste [Resnick y Varian, 1997].

Basándose en la búsqueda de soluciones óptimas a partir de las restricciones establecidas y el conjunto de preferencias dado por el tomador de decisiones, se puede generar un sistema que sea capaz no solamente de ofrecerle al DM la mejor opción posible,

sino también de explicar las diversas razones de dicha elección, ya que en muchos casos es común que como resultado final se muestre una solución sin explicar la razón de porqué ésta ha sido elegida sobre sus contrapartes.

Un sistema de recomendación es complejo, ya que debe ser capaz de tomar como entradas las acciones y las preferencias del DM y como salida una respuesta clara y concisa, sin embargo, existen varios factores que dificultan este proceso tales como:

- El DM no sabe expresar sus preferencias
- El DM no cuenta con toda la información del problema, por lo que sus preferencias pueden cambiar a mitad del proceso
- El DM pudiera no aceptar o comprender los resultados

Aplicando las diversas estrategias de optimización multiobjetivo en el SPP, un sistema es capaz de generar un conjunto de soluciones óptimas o muy cercanas al óptimo, las cuales se presentarán al tomador de decisión, sin embargo, muchas veces por diversas razones, la cantidad de soluciones encontradas son demasiadas y el DM es incapaz de analizar cada una de ellas de manera adecuada [Miller, 1956]. Además de que a la luz de nueva información el DM puede cambiar sus preferencias y por lo tanto cambiar las posibles soluciones. Una alternativa para resolver esta situación es el uso de un *Sistema de Recomendación* para ofrecer al tomador de decisión una de las soluciones obtenidas y justificar dicha elección aplicando diversas técnicas; esta investigación está soportada principalmente en la *Teoría de la Argumentación*.

#### **2.4.2 Teoría de la argumentación**

La teoría de la argumentación es un campo en crecimiento de la inteligencia artificial, es el proceso de construir y evaluar argumentos para justificar las conclusiones. También permite tomar decisiones prácticas de forma justificada, ofreciendo un mecanismo de razonamiento no monotónico, lo cual quiere decir que las conclusiones pueden cambiar a partir de nueva información obtenida [Ouerdane, 2009].

### *Artefacto cognitivo*

Los artefactos cognitivos son objetos creados por los humanos que buscan ayudar, mejorar o aumentar la cognición. Algunos ejemplos de artefactos cognitivos incluyen un calendario, algún recordatorio en una agenda o una computadora. Éstos no buscan solamente apoyar la memoria, sino también el razonamiento sobre clasificaciones y comparaciones [Wilson, 1999].

El proceso de ayuda a la decisión puede verse como un conjunto de artefactos cognitivos en secuencia, que se producen a través de la interacción entre analista y el cliente. Tsoukiàs [Tsoukiàs, 2011] sugiere los siguientes cuatro artefactos cognitivos:

- Una representación de la situación del problema
- Una formulación del problema
- Un modelo de evaluación
- Una recomendación final

Emitir una recomendación es el paso final en el proceso de ayuda a la decisión.

### *Lógica no monótona*

Se conoce como monotonidad al proceso de razonamiento lógico clásico en el cual una conclusión se considera cierta sin importar las adiciones al conjunto de proposiciones que nos permitieron llegar a dicha conclusión, en otras palabras, las conclusiones obtenidas no se modificarán o eliminarán sin importar que tanta nueva información relevante a dicha decisión se encuentre.

Por otra parte, existen casos en los cuales la nueva información obtenida puede provocar un cambio en la elección de una solución, muy a menudo en la vida diaria, nos enfrentamos a situaciones que están lejos de conocer de forma certera toda la información necesaria para poder llevar a cabo una decisión, es por ello que construimos conclusiones a partir de la información que se tiene disponible en ese momento, pero con la posibilidad de cambiarlas en cualquier momento considerando los nuevos datos que obtengan.

A este tipo de lógica, que se encuentra fuera del razonamiento lógico clásico, se le conoce como lógica no monótona [McCarthy, 1977; McDermott y Doyle, 1980; Reiter, 1980]. La exploración de dicha lógica se basa principalmente en la intención de poder proporcionar una manera formal para tener en cuenta la posibilidad de que las conclusiones deducidas podrían cambiar a la luz de información adicional.

### *Argumentación*

La argumentación surge como una alternativa para mecanizar el razonamiento no monotónico, ya que en los entornos de trabajo basados en argumentos se ve al problema como un proceso en cual se construyen y comparan los argumentos a favor y en contra de las conclusiones. La generación de nuevas premisas que conllevan a nueva información y argumentos que apoyan una nueva conclusión o que contra-argumentan fuertemente las ya existentes, lleva a la necesidad de adoptar la no monotonicidad en la toma de decisión. La argumentación también puede ser vista como:

*"Un tipo de proceso para tomar decisiones prácticas y justificadas. El objetivo del proceso es aclarar y resolver los problemas, y producir una justificación de la decisión que pueda resistir una evaluación crítica de un público en particular"* [Gordon y Walton, 2009].

Prakken y Sartor [Prakken y Sartor, 2002], sugieren dividir el proceso de argumentación en cuatro capas:

- *Capa lógica*: Construcción de argumentos, definiendo la forma en que la información puede ser expresada y reglas para la construcción de argumentos.
- *Capa dialéctica*: Identificación de argumentos en conflicto.
- *Capa procedimental*: Regula como se puede llevar a cabo una disputa, obtención de nuevos argumentos con nueva información, implica el uso de argumentos en el diálogo.
- *Capa estratégica*: Procesamiento racional de la disputa dentro de los límites procedimentales de la tercera capa.



## **2.5 Juegos de diálogo**

Una forma de definir la lógica argumentativa está en la forma dialéctica de los juegos de diálogo (o sistemas de diálogo). Dichos juegos modelan la interacción entre dos o más jugadores, en donde existen argumentos a favor y en contra que se intercambian de acuerdo a un conjunto de reglas y condiciones [Carlson, 2012].

Existen diferentes diálogos formales, tomando en consideración información tal como los participantes, el lenguaje de comunicación, los roles de los participantes, el objetivo del diálogo, entre otros. Debido a éste es necesario definir un conjunto de reglas para que el juego de diálogo que se lleve a cabo se mantenga coherente y en movimiento.

### **2.5.1 Reglas del juego de diálogo**

Los sistemas de diálogo definen esencialmente el principio del diálogo coherente y la condición bajo la cual una declaración hecha por un individuo se considera apropiada. Esto se logra mediante la formulación y aplicación de un conjunto de reglas durante el juego de diálogo, las cuales sirven para regular la forma en la que se lleva a cabo, definiendo aspectos tales como los puntos de inicio y terminación de mismo así como los tipos de declaración que tiene permitido realizar cada jugador en un cierto momento del juego, buscando mantener el diálogo en flujo y coherente.

Las reglas del juego de diálogo se dividen en cuatro elementos, los cuales son:

- *Reglas de locución*: reglas que definen los movimientos permitidos
- *Reglas de compromiso*: reglas que definen el compromiso a una declaración
- *Reglas de diálogo*: reglas que regulan los movimientos
- *Reglas de terminación*: reglas que definen el final del juego de diálogo

#### **2.5.1.1 Reglas de locución**

Reglas que definen que declaraciones son permitidas durante el juego de diálogo, típicamente, las locuciones legales permiten a los participantes hacer valer sus proposiciones, también les ofrece la posibilidad de cuestionar o retar proposiciones anteriores, además permite que las proposiciones afirmadas que son subsecuentemente cuestionadas o probadas justifiquen su validez.

En otras palabras, las reglas de locución son los movimientos que tienen permitidos realizar los jugadores dentro de un juego de diálogo. Para esto se toman en consideración los movimientos previos realizados por los demás jugadores, así como las declaraciones de diálogo existentes, las cuales son esenciales para inicializar un esquema de argumentación.

### **2.5.1.2 Reglas de compromiso**

Durante el juego de diálogo, a cada jugador se le asocia un almacén de compromisos, el cual es el conjunto de declaraciones y argumentos realizados por él donde el jugador está comprometido a defender dichas afirmaciones. Las reglas de compromiso definen tanto las circunstancias en las cuales un jugador se compromete con una declaración así como los efectos ocurridos dentro del almacén de compromiso que fueron consecuencia de los movimientos realizados durante el juego de diálogo.

### **2.5.1.3 Reglas de diálogo**

Como se vio anteriormente, existen reglas que definen los movimientos que se pueden realizar durante el juego de diálogo, sin embargo es necesario también tener un conjunto de reglas que regule el uso de dichos movimientos para mantener el diálogo comprensible y coherente, estas reglas son conocidas como reglas de diálogo.

Según [Prakken, 2005], las reglas de diálogo pueden ser divididas en tres conjuntos diferentes, dependiendo del objetivo de la declaración, ya sea para atacar argumentos generados por otros jugadores, responder a los ataques recibidos en las afirmaciones generadas o para actualizar el estado de una declaración previamente realizada.

## **2.5.2 Diagrama de transición de estados**

A partir de la definición de las reglas del juego diálogo se puede representar la forma en la que se llevarán a cabo los movimientos de los jugadores dentro de un diagrama de transición de estados. Dentro de este diagrama se muestran los diferentes posibles estados en los que se puede encontrar el diálogo, los movimientos que se pueden realizar en cada uno de dichos estados, así como los puntos de finalización del juego de diálogo, definidos como estados finales.

Las reglas de locución definidas para el juego de diálogo realizado se muestran dentro del diagrama como las acciones de cada uno de los estados, la disponibilidad de dichas acciones en cada estado se define en base de las reglas de diálogo y las reglas de compromiso. Finalmente, las reglas de terminación están representadas en el diagrama como los estados finales, los cuales marcan las condiciones para llegar al final del diálogo. La Figura 2.2 muestra un ejemplo de lo anterior, aplicando las reglas definidas en el trabajo de Ouerdane [Ouerdane, 2009].

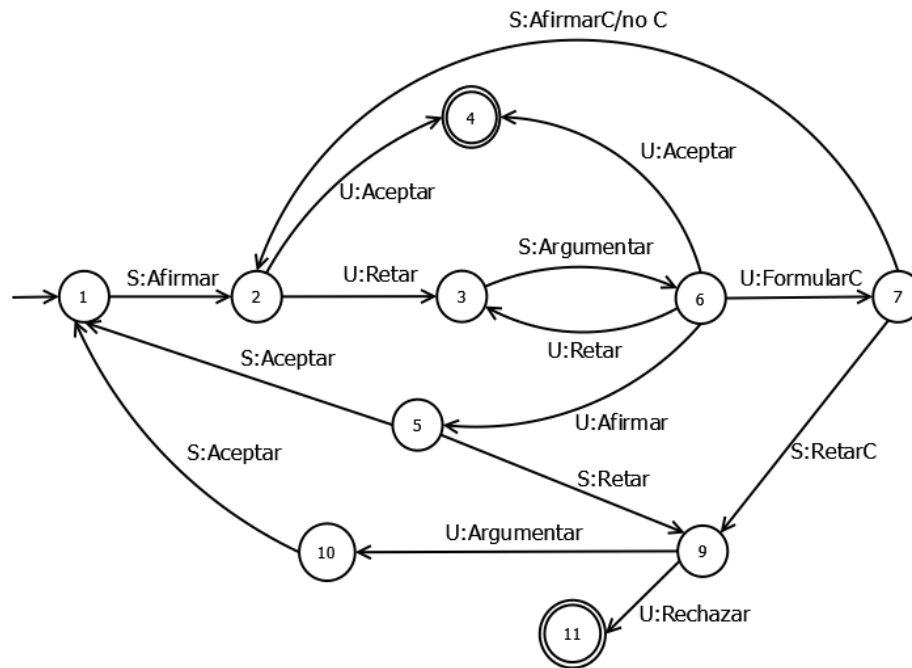


Figura 2.2: Diagrama de Transición de Estados, traducido de [Ouerdane, 2009]

## 2.6 Estándares de prueba

Dentro del juego de diálogo, las declaraciones realizadas necesitan ser evaluadas para determinar su veracidad, para esto se hace uso de los estándares de prueba, los cuales permiten agregar tanto el conjunto de argumentos a favor, como el conjunto de argumentos en contra de la declaración, comparándolos y llegando a una conclusión, resolviendo así el conflicto.

Aunque existen un gran número de estándares de prueba, para este trabajo se hizo enfoque en un conjunto específico de ellos, los cuales están orientados a la definición de una solución dentro del problema de carteras de proyectos públicos.

### 2.6.1 Mayoría simple

Para una pareja de alternativas  $a$  y  $b$ , se dice que  $a$  tiene una preferencia sobre  $b$  si dentro del conjunto de criterios  $N$  la cantidad de criterios en los cuales el valor de  $a$  supera al de  $b$  es mayor que en los cuales el valor de  $b$  supere  $a$ . La definición formal se muestra de la siguiente manera [Ouerdane, 2009].

$$a \geq b \leftrightarrow |\{i \in N: aS_i b\}| \geq |\{i \in N: bS_i a\}| \quad (7)$$

Cabe mencionar que se puede utilizar un umbral de veto en este estándar de prueba para definir si la preferencia es lo suficientemente fuerte como para ser considerada verdadera y aplicable.

### 2.6.2 Mayoría ponderada

Este estándar de prueba posee una enorme similitud con la mayoría simple, sin embargo, se apoya en los pesos de cada criterio para evaluar. En este caso cada criterio  $i$  tiene un peso  $w_i$ , se dice que  $a$  tiene una preferencia sobre  $b$  si la suma de los pesos de los criterios donde  $a$  supera a  $b$  es mayor que la suma de los pesos de los criterios donde  $b$  supera  $a$ , La representación formal es la siguiente [Ouerdane, 2009].

$$a \geq b \leftrightarrow W_{ab} = \sum_{aS_i b} w_i \geq W_{ba} = \sum_{bS_i a} w_i \quad (8)$$

De igual manera que la mayoría simple, este estándar de prueba puede utilizar un umbral de veto para definir si la diferencia entre ambas sumatorias es lo suficientemente fuerte como para que se considere una preferencia.

### 2.6.3 Orden lexicográfico

Para este estándar de prueba es necesario tener de forma previa un orden jerárquico establecido en los criterios, después se hace una comparación entre los valores de las alternativas en cada criterio siguiendo el orden establecido. Se dice que la alternativa  $a$  tiene una preferencia sobre  $b$  si el primero tiene un mejor valor que el segundo para el

criterio en evaluación. Si ambas alternativas tienen el mismo valor en dicho criterio, se usa el siguiente que se encuentre en el orden para realizar la comparación, esto se realiza hasta que se genere un desempate [Ouerdane, 2009].

El orden de jerarquía de los criterios establece que un criterio de mayor orden es infinitamente más importante que aquellos criterios que se encuentran en una posición inferior, por lo tanto si una alternativa es mejor que otra en un criterio de mayor importancia, esta tiene preferencia aun cuando la segunda supere a la primera en todos los demás criterios de menor jerarquía.

#### **2.6.4 Suma de pesos**

Para este estándar de prueba, se toma en consideración el peso de cada criterio y el valor por criterio de las alternativas, para una alternativa  $a$  se busca obtener una puntuación  $S_a$ , resultante de la sumatoria del valor de cada uno de sus criterios  $h_i$  multiplicado por el peso correspondiente a dicho criterio  $w_i$  tal como se muestra en la ecuación (9) [Ouerdane, 2009].

$$S_a(h) = \sum_{i \in N} w_i h_i(a) \quad (9)$$

En este caso se dice que la alternativa  $a$  tiene preferencia sobre  $b$  si la puntuación resultante del primero supera al segundo.

#### **2.6.5 TOPSIS**

Desarrollado por Hwang y Yoon [Hwang y Yoon, 1981], este estándar de prueba considera la distancia hacia la solución ideal y la solución ideal negativa, donde la solución que se encuentra más cerca de la primera y más lejana a la segunda es la que tiene preferencia.

Para llevar a cabo este estándar de prueba se siguen una serie de pasos:

- 1) Generar una matriz de desempeño  $D$ , donde cada fila representa una las  $m$  alternativas y cada columna representa uno de los  $n$  criterios.

$$D = \begin{bmatrix} x_{11} & x_{12} & \cdot & x_{1j} & \cdot & x_{1n} \\ x_{21} & x_{22} & \cdot & x_{2j} & \cdot & x_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ x_{i1} & x_{i2} & \cdot & x_{ij} & \cdot & x_{in} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ x_{m1} & x_{m2} & \cdot & x_{mj} & \cdot & x_{mn} \end{bmatrix}$$

- 2) Normalizar la cada celda de la matriz de desempeño utilizando la ecuación (10), donde  $x_{ij}$  representa el valor de la alternativa  $i$  con respecto al criterio  $j$  mientras que  $r_{ij}$  representa el valor normalizado.

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}} \quad (10)$$

- 3) Multiplicar la matriz normalizada por el peso del criterio correspondiente

$$V = \begin{bmatrix} w_1 r_{11} & w_2 r_{12} & \cdot & w_j r_{1j} & \cdot & w_n r_{1n} \\ w_1 r_{21} & w_2 r_{22} & \cdot & w_j r_{2j} & \cdot & w_n r_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ w_1 r_{i1} & w_2 r_{i2} & \cdot & w_j r_{ij} & \cdot & w_n r_{in} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ w_1 r_{m1} & w_2 r_{m2} & \cdot & w_j r_{mj} & \cdot & w_n r_{mn} \end{bmatrix} = \begin{bmatrix} v_{11} & v_{12} & \cdot & v_{1j} & \cdot & v_{1n} \\ v_{21} & v_{22} & \cdot & v_{2j} & \cdot & v_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ v_{i1} & v_{i2} & \cdot & v_{ij} & \cdot & v_{in} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ v_{m1} & v_{m2} & \cdot & v_{mj} & \cdot & v_{mn} \end{bmatrix}$$

- 4) Determinar la solución ideal  $V^+$  mediante la ecuación (11) y la solución ideal negativa de cada criterio  $V^-$  mediante la ecuación (12).

$$\begin{aligned} V^+ &= \{(\max v_{ij} | j \in J) \text{ o } (\min v_{ij} | j \in J'), i = 1, 2, \dots, m\} \\ &= \{v_1^+, v_2^+, \dots, v_n^+\} \end{aligned} \quad (11)$$

$$\begin{aligned} V^- &= \{(\min v_{ij} | j \in J) \text{ o } (\max v_{ij} | j \in J'), i = 1, 2, \dots, m\} \\ &= \{v_1^-, v_2^-, \dots, v_n^-\} \end{aligned} \quad (12)$$

Sea  $J$  un vector que contiene los valores normalizados de cada criterio en la alternativa  $i$ , tal que:

$$J = \{j = 1, 2, \dots, n | v_{ij}, \text{ buscando una respuesta de mayor valor}\}$$

$$J' = \{j = 1, 2, \dots, n | v_{ij}, \text{ buscando una respuesta de menor valor}\}$$

- 5) Calcular la separación de cada alternativa de la solución ideal y la solución ideal negativa para después calcular la cercanía de la solución ideal para todas las alternativas.

La separación de una alternativa a la solución ideal  $S_i^+$  se calcula utilizando la ecuación (13)

$$S_i^+ = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^+)^2} \quad (13)$$

Por otro lado la separación de una alternativa a solución ideal negativa  $S_i^-$  se calcula utilizando la ecuación (14)

$$S_i^- = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^-)^2} \quad (14)$$

Finalmente se calcula la cercanía de cada alternativa a la solución ideal denominada como  $C_i$  aplicando la ecuación (15), los valores resultantes se encuentran en un rango de 0 a 1 y deben ser ordenados, mientras más cerca se encuentre  $C_i$  de 1 mayor será su prioridad. La alternativa con mayor prioridad tiene preferencia sobre las demás.

$$C_i = \frac{S_i^-}{S_i^+ + S_i^-} \quad (15)$$

## 2.7 Esquemas de argumentación

Dentro de la teoría de la argumentación se encuentran los esquemas de argumentación, los cuales son formas de argumentos que capturan patrones estereotípicos de razonamiento humano, especialmente los anulables [Walton, 2005]. Los argumentos son presentados como reglas de inferencia general, donde dado un conjunto de premisas se puede llegar a una conclusión [Walton, 1996]. Sin embargo, debido a la naturaleza anulable de los argumentos tales esquemas no son deductivamente estrictos. Los esquemas permiten a los argumentos ser representados dentro de un contexto particular y tomar en cuenta que el razonamiento presentado puede ser modificado a la luz de nuevas pruebas o excepción a las reglas.

Los esquemas de argumentación permiten identificar las premisas y la conclusión del argumento. Las premisas se dividen en tres tipos basándose en el modelo de *Carneades*, el cual hace uso de pruebas estándar para determinar la aceptabilidad de los argumentos [Gordon y Walton, 2006]:

- *Premisas ordinarias*: aquellas que requieren de más evidencia para considerarse.
- *Suposiciones*: aquellas que pueden ser asumidas hasta ser cuestionadas.
- *Excepciones*: aquellas que no se sostienen hasta que se demuestre lo contrario.

Dichos esquemas juegan dos roles dentro del juego de diálogo [Walton, 1996]:

- Cuando se construyen argumentos, proveen un repertorio de formas de argumentos para ser considerados y una plantilla para las piezas que se necesitan.
- Cuando se ataca, los argumentos proveen un conjunto de preguntas críticas que pueden identificar las potenciales debilidades en el caso del oponente.

*Preguntas críticas*: Cada esquema de argumentación genera un conjunto de preguntas críticas específicas, las cuales representan ataques, retos o críticas, que en caso de no responderse de forma adecuada falsifican el argumento acorde en el esquema, estas preguntas son vitales para determinar la solidez y capacidad de un argumento de poder defenderse en caso de ser atacado, ya que si es incapaz de contestar estas preguntas entonces no será capaz de explicar al cliente o tomador de decisión por qué razón se eligió dicha solución sobre las demás.



Walton y Reed [Walton y Reed, 2002] afirman que los esquemas de argumentación y las preguntas críticas trabajan juntas. La primera es usada para identificar las premisas y la conclusión, mientras la segunda es usada para evaluar el argumento explorando dentro de sus potenciales puntos débiles. Por esta razón se puede decir que los esquemas de argumentación están compuestos de tres elementos:

- *Premisas*: argumentos que funcionan como soporte u oposición a la conclusión.
- *Conclusión*: declaración que se busca afirmar o rechazar.
- *Preguntas críticas*: ataques, retos o críticas, que en caso de no responderse de forma adecuada falsifican el argumento acorde en el esquema.

Para este trabajo se decidió utilizar un cierto grupo de esquemas de argumentación cuyos conceptos serán explicados dentro de esta misma sección a continuación.

### 2.7.1 Razonamiento abductivo

Este esquema de argumentación busca encontrar una explicación satisfactoria a un conjunto de hechos existentes en un cierto momento durante el diálogo. Estos hechos son considerados como verdaderos al momento de utilizar el esquema, aunque su veracidad puede ser cuestionada en momentos posteriores. La explicación puede ser definida de forma dialéctica considerando que la satisfacción que produce a las personas que buscaron dicha explicación depende del tipo de diálogo y del avance que se haya logrado en el mismo [Walton et al., 2008]. En la tabla 2.1 se muestran las premisas, conclusión y preguntas críticas correspondientes a este esquema de argumentación.

**Tabla 2.1:** Esquema de Argumentación del razonamiento abductivo

Premisas	$F$ es un conjunto de hechos
	$E$ es una explicación satisfactoria para $F$
	No existe una explicación alternativa $E'$ que sea tan satisfactoria como $E$
Conclusiones	Por lo tanto, $E$ es plausible como hipótesis.
Preguntas críticas	¿Qué tan satisfactorio es $E$ como una explicación de $F$ , aparte de las explicaciones alternativas disponibles en el diálogo?
	¿Cuánto mejor es la explicación $E$ en comparación de las alternativas disponibles?
	¿Qué tanto ha progresado el diálogo?, si el diálogo es una investigación, ¿a qué tanta profundidad ha llegado la investigación del caso?

### 2.7.2 Desde una posición para saber

Este esquema de argumentación se basa en la idea de que la persona que responde posee algo de información sobre el tema deseado, la cual se puede obtener si se le pregunta o pide a dicha persona la información que se necesita. El preguntador supone que el respondiente tiene un cierto grado de experiencia en el tema y que por lo tanto su respuesta será válida, lo cual no es necesariamente cierto [Walton et al., 2008]. La Tabla 2.2 muestra las premisas, conclusión y preguntas críticas correspondientes a este esquema.

**Tabla 2.2:** Esquema de Argumentación de un argumento desde una posición para saber

Premisas	Premisa Mayor: La fuente $a$ esta en posición para saber cosas en un cierto dominio $S$ que contiene la proposición $A$ .
	Premisa Menor: $a$ afirma que $A$ es verdadero/falso.
Conclusiones	$A$ es verdadero/falso
Preguntas críticas	¿Está $a$ en posición de saber si $A$ es verdadero/falso?
	¿Es $a$ una fuente confiable?
	¿ $a$ afirmó que es $A$ verdadero/falso?

### 2.7.3 Desde una opinión experta

Como se mencionó en el esquema anterior, la persona que responde a las preguntas no es necesariamente un experto en el tema, sino que solamente posee información sobre este que no tiene el preguntador, lo cual no marca una gran diferencia de conocimiento entre ambos lados.

El esquema de argumentación desde una opinión experta se basa en la idea de que la diferencia de conocimiento sobre el tema entre el preguntador y el respondiente es grande, al punto de que el segundo es considerado como un experto del tema, mientras que el primero no.

La idea de este esquema es que la persona que pregunta se apoya en el conocimiento de un experto para resolver un problema o llevar a cabo una acción relacionada al tema del cual está preguntando [Walton et al., 2008]. En la Tabla 2.3 se muestran las premisas, conclusiones y preguntas críticas de este esquema.

**Tabla 2.3:** Esquema de Argumentación de un argumento desde una opinión experta

Premisas	Premisa Mayor: La fuente $E$ es un experto en el campo $F$ que contiene la proposición $A$ .
	Premisa Menor: $E$ afirma que la proposición $A$ en el campo $F$ es verdadera/falsa.
Conclusiones	$A$ puede plausiblemente ser tomada como verdadera/falsa.
Preguntas críticas	¿Qué tan creíble es $E$ cómo una fuente experta?
	¿Es $E$ un experto en el campo $F$ donde se encuentra $A$ ?
	¿Qué afirmó $E$ que infiere $A$ ?
	¿Es $E$ personalmente confiable como una fuente?
	¿Es $A$ consistente con las afirmaciones de otros expertos?
	¿Es la afirmación de $E$ basada en evidencia?

#### 2.7.4 Evaluación muticriterio de parejas

Un par de soluciones existentes en un problema de múltiples criterios pueden verse con una relación tal que una de esas soluciones es al menos tan buena como la otra. Para justificar esta declaración se puede utilizar una *evaluación multicriterio de parejas*, la cual permite comparar todos los criterios considerados en el diálogo buscando tanto que las razones de soporte encontradas sean suficientes, así como no existan suficientes razones de oposición para negar la declaración establecida [Ouerdane, 2009]. La composición de este esquema de argumentación se muestra en la Tabla 2.4.

**Tabla 2.4:** Esquema de Argumentación evaluación multicriterio de parejas

Premisas	Acción $a$
	Acción $b$
	Conjunto de criterios $h$
	Existen suficientes razones de soporte $SR$
	No existen suficientes razones de oposición $OR$
Conclusiones	$a$ al menos tan buena como $b$
Preguntas críticas	¿Las razones de soporte tienen la suficiente importancia como para superar a las razones de oposición?
	¿Es la diferencia entre $SR$ y $OR$ lo suficientemente grande para poder aceptar la conclusión?

#### 2.7.5 Ad ignorantiam

Existen esquemas de argumentación que se basan en la falta de conocimiento para sostener sus argumentos. El esquema de argumentación *ad ignorantiam* establece que la falta de pruebas que apoyen un argumento es suficiente para determinar su falsedad [Walton et al., 2008], existen ciertos diálogos en los cuales este esquema de argumentación entra en efectos, un ejemplo es la presunción de inocencia durante un juicio legal. En la Tabla 2.5 se

muestran las premisas, conclusión y preguntas críticas que corresponde a este esquema de argumentación.

**Tabla 2.5:** Esquema de Argumentación de un argumento Ad Ignorantiam

Premisas	Premisa Mayor: Si $A$ fuera verdadera, entonces se sabría que $A$ es verdadera.
	Premisa Menor: $A$ no es conocida como verdadera.
Conclusiones	$A$ es falsa.
Preguntas críticas	¿ $A$ es conocido?
	¿Existen pruebas de que $A$ es falso?

### 2.7.6 Causa efecto

Este esquema de argumentación se basa en el hecho de que un evento puede ocurrir u ocurrirá si se llevó previamente a cabo una acción o situación la cual se sabe que precede a dicho evento [Walton et al., 2008]. La Tabla 2.6 muestra las diferentes premisas, conclusión y preguntas críticas pertenecientes a este esquema.

**Tabla 2.6:** Esquema de Argumentación de un argumento causa-efecto

Premisas	Premisa Mayor: Generalmente si $A$ ocurre, $B$ por consecuencia también.
	Premisa Menor: $A$ ocurre en este caso.
Conclusiones	$B$ ocurre.
Preguntas críticas	¿Qué tan fuerte es la generalización causal ( $B$ resulta debido a $A$ )?
	¿Existe evidencia lo suficientemente fuerte para garantizar la generalización causal?
	¿Existen otros factores causales que puedan interferir con el efecto del caso mencionado?

### 2.7.7 Desde la parcialidad

Durante el diálogo, es probable que uno o varios de los participantes presenten un cierto grado de preferencia respecto a un cierto argumento o elemento específico, debido a esta razón, dichos participantes pueden no tomar todos los argumentos y declaraciones existentes en el diálogo de la misma manera, dándole prioridad a aquellos que apoyen a los elementos a los que se tenga una preferencia. Es trabajo del experto del tema identificar dichas parcialidades y separarlas de los argumentos imparciales [Walton et al., 2008]. En la Tabla 2.7 se muestran las premisas, preguntas críticas y conclusión correspondiente a este esquema de argumentación.

**Tabla 2.7:** Esquemas de Argumentación de un argumento desde la parcialidad

Premisas	Premisa Mayor: Si $x$ es parcial, entonces es menos probable que $x$ haya tomado en cuenta la evidencia de ambos lados al llegar a la conclusión $A$ .
	Premisa Menor: Argumentador $a$ es parcial.
Conclusiones	Es menos probable que $a$ haya tomado en cuenta la evidencia de ambos lados al llegar a la conclusión $A$ .
Preguntas críticas	¿Cuál es el tipo de diálogo en que tanto el emisor como el receptor están participando?
	¿Qué evidencia existe de parcialidad por parte de algún participante?

# Capítulo 3

## Estado del Arte

---

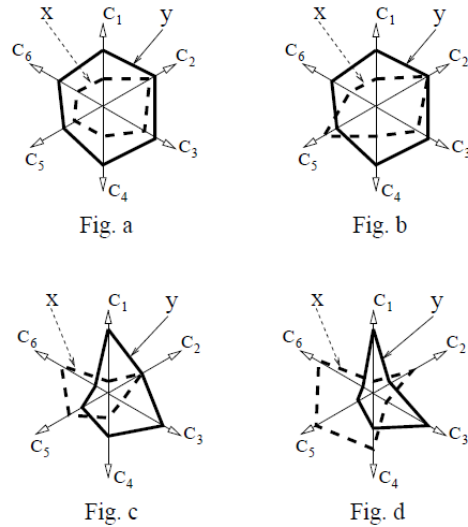
---

Dentro de este capítulo se llevará a cabo un análisis de varios trabajos previos realizados, los cuales están enfocados a una o varias áreas relacionadas con los objetivos establecidos en este documento. Estos trabajos involucran áreas tales como la utilización de los pesos para evaluar diversas soluciones y llegar a una conclusión, la aplicación de estándares de prueba y esquemas de argumentación para generar, apoyar y defender argumentos, la definición de un marco de trabajo para un SPP apoyándose de la teoría de la argumentación así como el diseño e implementación de prototipos de sistemas de recomendación para problemas de elección utilizando como apoyo los esquemas de argumentación y la teoría de la argumentación para generar recomendaciones y establecer un juego de diálogo.

### **3.1 “Argumentation of the decision made by several aggregation operators based on weights” [Labreuche, 2006]**

Labreuche describe en su trabajo una forma de aproximación dentro de la argumentación cuando se lleva a cabo la comparación entre dos alternativas, utilizando un método MCDA el cual se basa en los pesos definidos para cada uno de los criterios para su análisis.

También establece cuatro diferentes situaciones, las cuales involucran a dos alternativas del conjunto  $y$  y  $x$ , además de un conjunto de pesos  $w$  para seis diferentes criterios, los cuales representan las preferencias del tomador de decisiones. Estas situaciones pueden utilizar evaluaciones basadas en el peso de cada uno de los criterios existentes para llegar a la conclusión de que la solución  $y$  se prefiere sobre la solución  $x$ . La Figura 3.1 muestra una representación gráfica de los cuatro casos estudiados.



**Figura 3.1:** Casos considerados por Labreuche [Labreuche, 2006]

Sean  $c_i$  los criterios del problema cuyo valor aumenta conforme este más alejado su punto del centro y las variables  $x$  y  $y$  soluciones factibles, véase en el primer caso (Fig. a) que la solución  $y$  es mejor que  $x$  en cada uno de los criterios. Este caso no requiere de una evaluación más profunda ya que no existen argumentos en contra de que  $y$  es preferida sobre  $x$ .

En el segundo caso (Fig. b) la solución  $y$  es mejor que  $x$  en la mayoría de los criterios. En este caso aunque el resultado parece ser obvio, hay que tener en cuenta que un posible cambio de valores de los pesos  $w$  podría llevar a otro resultado.

El tercer caso (Fig. c) tiene un mayor grado dificultad con respecto a los anteriores, ya que la cantidad de criterios en los cuales  $y$  es mejor que  $x$  en comparación con lo contrario es muy pequeña. En este caso un cambio en los valores de  $w$  muy probablemente podría llevar a que  $x$  sea preferida sobre  $y$ .

El último caso (Fig. d) especifica que la mayoría de los criterios son mejores en  $x$  con respecto a  $y$ . Para definir que  $y$  se prefiere sobre  $x$  es necesario definir que los criterios en los cuales  $y$  es mejor que  $x$  son de mayor importancia con respecto a los demás.

Para todos estos casos, salvo el primero el cual no requiere de un proceso de evaluación adicional debido a que no existen argumentos en contra de la declaración inicial,

se utilizaron cuatro métodos MCDA para generar un argumento buscando justificar la preferencia de  $y$  sobre  $x$ .

*Suma de pesos:* este modelo se realiza comparando la sumatoria de cada uno de los pesos de los criterios de ambas alternativas. En este caso  $y$  tiene preferencia sobre  $x$  en base a la suma de pesos ( $y \geq \frac{WSum}{w} x$ ) si el puntaje resultante en  $y$  supera o es igual a la suma de pesos en  $x$ .

$$y \geq \frac{WSum}{w} x \Leftrightarrow H \frac{WSum}{w}(y) \geq H \frac{WSum}{w}(x)$$

Donde

$$H \frac{WSum}{w}(x) = \sum_{i \in N} w_i x_i$$

Sea

- $N$  = número de criterios del problema.
- $w_i$  = peso del criterio  $i$ .
- $x_i$  = valor del criterio  $i$  para la solución factible  $x$ .
- $H \frac{WSum}{w}(x)$  = sumatoria de pesos de la variable  $x$ .

*Votación:* al igual que la suma de pesos, este método utiliza una sumatoria de pesos para llevar a cabo la comparación, sin embargo, los conjuntos de comparación son diferentes. Para definir que la solución  $y$  tiene preferencia sobre  $x$  bajo votación ( $y \geq \frac{WVot}{w} x$ ) la sumatoria de pesos de aquellos criterios donde  $y$  sea mejor que  $x$  debe ser mayor o igual que la sumatoria de pesos de aquellos criterios donde  $x$  sea mejor que  $y$ .

$$y \geq \frac{WVot}{w} x \Leftrightarrow H \frac{WVot}{w}(y, x) \geq 0$$

Donde

$$H \frac{WVot}{w}(y, x) = \sum_{i \in A^+(y,x)} w_i - \sum_{i \in A^-(y,x)} w_i$$



Sea

- $A^+(y, x)$  = argumentos positivos, tales que  $y_i > x_i$ .
- $A^-(y, x)$  = argumentos negativos, tales que  $y_i < x_i$ .
- $H_w^{WVot}(y, x)$  = puntaje resultante de evaluar los argumentos positivos y negativos de la pareja de soluciones  $(y, x)$ .

*Peso mínimo:* para este modelo se consideran solo los valores de los criterios de menor importancia en el conjunto  $(1-w_i)$  descartando aquellos criterios de mayor peso por su alta importancia, y tiene preferencia sobre  $x$  bajo el peso mínimo ( $y \geq \frac{WMin}{w} x$ ) si el peor valor de los criterios considerados en  $y$  ( $H_w^{WMin}(y)$ ) es mejor o igual que el peor de los valores de los criterios considerados en  $x$  ( $H_w^{WMin}(x)$ ).

$$y \geq \frac{WMin}{w} x \Leftrightarrow H_w^{WMin}(y) \geq H_w^{WMin}(x)$$

Donde

$$H_w^{WMin}(x) = \min_{i \in N}(\max(x_i, 1 - w_i))$$

*Peso máximo:* este modelo considera solamente los valores de los criterios de mayor importancia en el conjunto  $(w_i)$  descartando aquellos criterios de menor peso por su poca relevancia, y tiene preferencia sobre  $x$  bajo el peso máximo ( $y \geq \frac{WMax}{w} x$ ) si el mejor valor de los criterios considerados en  $y$  ( $H_w^{WMax}(y)$ ) es mejor o igual que el mejor de los valores de los criterios considerados en  $x$  ( $H_w^{WMax}(x)$ ).

$$y \geq \frac{WMax}{w} x \Leftrightarrow H_w^{WMax}(y) \geq H_w^{WMax}(x)$$

Donde

$$H_w^{WMax}(x) = \max_{i \in N}(\min(x_i, w_i))$$

### **3.2 “Multiple Criteria Decision Aiding: a Dialectical Perspective” [Ouerdane, 2009]**

En su trabajo doctoral, Ouerdane presenta un enfoque para proporcionar razones subyacentes que soporten la acción seleccionada. Para la explicación, combina teoría de la argumentación, ayuda a la decisión y un lenguaje para posibilitar de manera acotada el manejo incompleto o contradictorio de piezas de información.

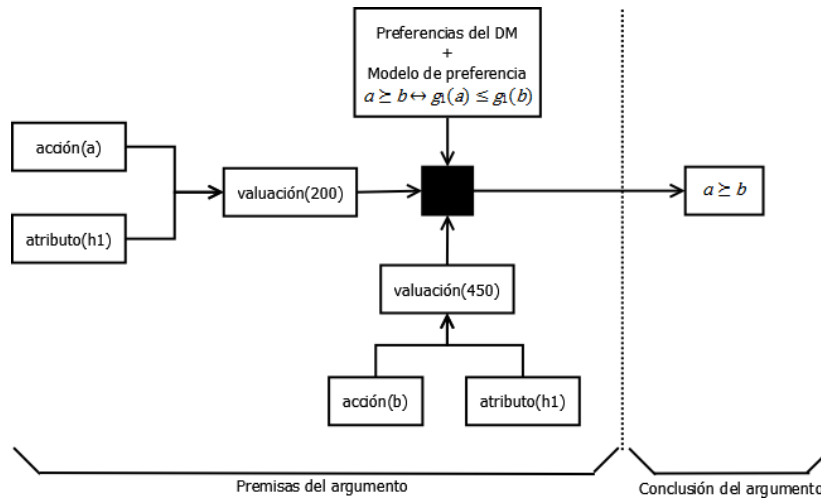
También se propone una estructura jerárquica de esquemas de argumentos para descomponer el proceso de decisión en algunos pasos cuyas premisas subyacentes se hacen explícitas, lo cual permite identificar en qué momento la información de esos pasos debe ser incorporada en el diálogo con el DM.

Dentro de su trabajo se analizaron los diferentes elementos necesarios para poder realizar de manera adecuada un juego de diálogo con el usuario, buscando no solo defender una recomendación establecida, sino también obtener nuevas preferencias definidas por el DM, las cuales pueden llevar a cambios en varios de los elementos relacionados con el diálogo y por lo tanto en la recomendación.

Para todo esto, se analizaron diversos puntos dentro de este documento:

*Identificación de premisas:* Una parte vital para los esquemas de argumentación es identificar las premisas que lo componen, ya que a partir de la validez o no de ellas se generan las suficientes razones de soporte u oposición para definir la veracidad de la conclusión de cada esquema.

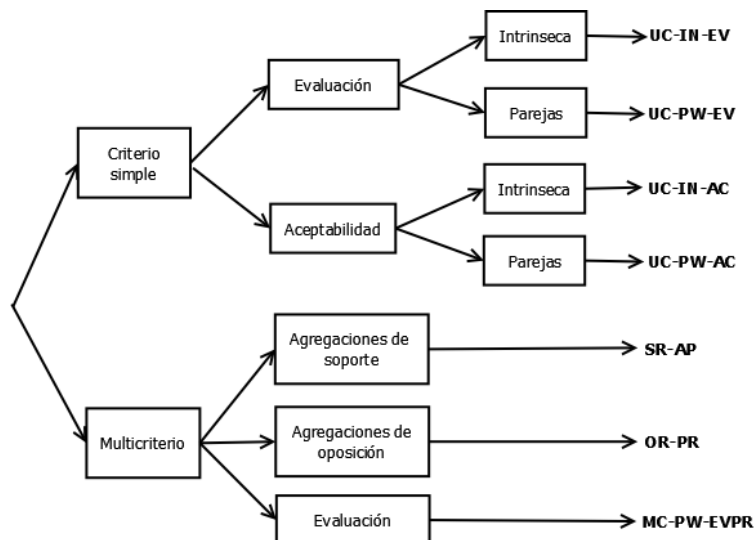
En el contexto del trabajo de Ouerdane, las premisas de un argumento que se enfocan a un solo punto de vista solo pueden estar basadas tanto en la tabla de desempeño del problema, así como la información provista por el DM. En la Figura 3.2 se muestra una de las formas en la que Ouerdane maneja las premisas de un argumento, estas varían a partir del esquema de argumentación que se maneje.



**Figura 3.2:** Ejemplo de un argumento para la evaluación de parejas, traducido de [Ouerdane, 2009]

*Esquemas de argumentación:* Se identificaron diferentes patrones bajo los cuales se puede generar un esquema de argumentación, el análisis de las premisas que lo componen y la generación de razones de soporte y oposición para el apoyo o rechazo de la conclusión establecida de manera previa. Se definieron y organizaron las clases de esquemas de argumentación utilizadas en las diferentes etapas del proceso de evaluación utilizando diferentes criterios. La Figura 3.3 muestra la clasificación establecida:

- Si están enfocados en uno a varios criterios.
- Si están enfocados en la evaluación de la acción o su aceptabilidad solamente.
- Si aplican una evaluación de parejas o intrínseca.
- Si se enfocan a razones positivas o negativas.



**Figura 3.3:** Diferentes clases de esquemas de argumentación, traducido de [Ouerdane, 2009]

- $UC-IN-EV$  = Evaluación intrínseca unicriterio.
- $UC-PW-EV$  = Evaluación unicriterio de parejas.
- $UC-IN-AC$  = Aceptabilidad intrínseca unicriterio.
- $UC-PW-AC$  = Aceptabilidad de parejas unicriterio.
- $SR-AP$  = Procesos de agregación de razones de soporte.
- $OR-PR$  = Procesos de agregación de razones de oposición.
- $MC-PW-EVPR$  = Evaluación multicriterio de parejas.

*Estándares de prueba:* Dentro del trabajo doctoral de Ouerdane, se consideraron un conjunto de estándares de prueba a utilizar para generar y defender una recomendación.

- Mayoría simple con y sin veto
- Orden lexicográfico
- Mayoría ponderada con y sin veto
- Suma de pesos
- Modelo no aditivo

De igual Ouerdane estableció un conjunto de propiedades con el objetivo de ayudar a decidir cuál de los estándares de prueba utilizar, asignándole a cada estándar un subconjunto de dichas propiedades:

- Ordinalidad
- Anonimato
- Adición con peso
- Adición con valores
- Veto
- Dependencia entre Criterios

Además de las propiedades anteriores, también se definió un esquema de argumentación para cada estándar de prueba con el objetivo de facilitar el proceso de selección de estos (ver Anexo A).

*Reglas del juego de diálogo:* En su trabajo, Ouerdane estableció que el juego de diálogo aplicado tendrá dos jugadores, el usuario y el sistema de recomendación, de igual

manera se establecieron un conjunto de reglas de locución, compromiso, diálogo y terminación, las cuales regulan el diálogo entre ambos jugadores. El conjunto de reglas de locución utilizadas se muestran en la Tabla 3.1.

**Tabla 3.1:** Reglas de locución definidas en [Ouerdane, 2009]

Movimientos	Respuestas	Rendición	Estado del reclamo
afirmar( $\phi$ )	retar( $\phi$ ) ¿Cuál prueba? formular( $C$ , $tipo$ , $\phi$ , $id\_esquema$ )	aceptar( $\phi$ )	Declarada
retar( $\phi$ )	argumentar( $\phi$ , $premisas$ , $id\_esquema$ )	retractar( $\phi$ )	Cuestionada
argumentar( $\phi$ , $premisas$ , $id\_esquema$ )	retar ( $premisas$ ) formular( $C$ , $tipo$ , $\phi$ , $id\_esquema$ ) afirmar( $\phi$ ) ¿Cuál prueba?	aceptar( $\phi$ ) aceptar( $premisas$ )	Aceptada
formular( $C$ , $tipo$ , $\phi$ , $id\_esquema$ )	retar(no- $C$ ) afirmar( $C$ ) afirmar(no- $C$ )	retractar( $\phi$ )	Rechazada
aceptar( $\phi$ )			Aceptada
retractar( $\phi$ )			Rechazada
¿Cuál prueba?	de acuerdo a ( $pf$ )		

Las reglas de compromiso definidas en el trabajo de Ouerdane son las siguientes:

- 0) Después de que un jugador hace una declaración, esta es guardada en el almacén de compromisos.
- 1) Después de retractar una declaración, ésta es borrada excepto si es información de la evidencia de la propiedad.
- 2) Retar una declaración coloca a ésta en el almacén de compromiso del oyente a menos de que ya esté ahí o si el mismo oyente retracta de manera inmediata su compromiso relacionado a este tema.
- 3) Retar una declaración coloca a ésta en la pila de preguntas bajo discusión del oyente.
- 4) Después de argumentar, el esquema de argumentación, sus premisas y la suposición son incluidas en el almacén de compromisos del emisor.

Por otro lado las reglas de diálogo están divididas en tres diferentes categorías:

### *Regla para actualizar*

- 5) En cualquier momento de su turno, el usuario puede actualizar (retractar + afirmar) una declaración de su almacén de compromisos.

### *Reglas para atacar movimientos*

- 6) No está permitido retar un tema si todavía está bajo otro reto o en compromiso.
- 7) Está permitido retar cualquier declaración a la que el otro participante se haya comprometido, excepto si va contra 1).
- 8) No se le permite al emisor declarar algo en lo que ya está comprometido o que contradiga algo a lo que está comprometido.
- 9) Está permitido hacer una declaración que contradiga el compromiso del otro jugador, a menos de que vaya en contra de 3).
- 10) Está permitido preguntar cualquier pregunta crítica relacionada con un esquema de argumentación al cual este comprometido algún otro participante mientras esta pregunta no esté ya bajo discusión.

### *Reglas para responder a movimientos*

- 11) Después de una actualización, el sistema debe informar las consecuencias.
- 12) Después de una afirmación de recomendación, se permite solamente hacer una pregunta crítica sobre la aceptabilidad de dicha recomendación.
- 13) Cada jugador toma su turno de habla avanzando una locución a la vez.
- 14) Es obligación responder a un reto con una retracción o un argumento ya sea:
  - Retractando el caso.
  - Presentando un argumento para el caso.
- 15) Después de una aceptación, se deben mencionar las consecuencias del estado actual de la recomendación.
- 16) Después de una afirmación del usuario, si existe una contradicción, informar de las consecuencias haciendo explícita dicha contradicción.
- 17) Después de que una declaración fue ofrecida en respuesta a un reto, si dicha respuesta es una instancia de substitución de algún esquema de argumentación

del juego, la locución formular  $C$  es legal, donde  $C$  es una pregunta crítica del esquema de argumentación afectado.

18) Es obligatorio responder a una pregunta crítica  $C$  aplicando uno de los siguientes movimientos.

- Si  $C$  es una suposición de su esquema de argumentación:
  - Afirmar  $C$ ;
  - No afirmar  $C$ ;
  - Retractor  $C$ .
- Si  $C$  es una excepción de su esquema de argumentación:
  - Afirmar  $C$ ;
  - No afirmar  $C$ ;
  - Retractor  $C$ ;
  - No retar  $C$ .

Finalmente las reglas de terminación definen que solamente el usuario es capaz de finalizar el juego de diálogo aceptando o rechazando de manera definitiva la recomendación final establecida por el sistema de recomendación.

*Diagrama de transición de estados:* Ouerdane utilizó un diagrama de transición de estados para representar la forma en la que se lleva a cabo un juego de diálogo entre el usuario y el sistema de recomendación, teniendo previamente definidas las reglas necesarias para poder llevar a cabo dicho diálogo.

De igual forma Ouerdane implementó un prototipo llamado *ArgDec* enfocado en problemas de elección de baja dimensión, el cual permite establecer un juego de diálogo con el usuario así como le permite visualizar de forma gráfica los argumentos realizados.

En la Figura 3.4 se muestra una captura de la interfaz gráfica de dicho sistema, en el cuadro superior izquierdo se encuentra la ventana en la cual se lleva a cabo el juego de diálogo entre el usuario y el sistema, por otro lado, en la ventana superior derecha se puede visualizar una representación gráfica de los argumentos utilizados en dicho diálogo. En esta misma figura se muestra en la parte inferior la tabla de desempeño del problema.

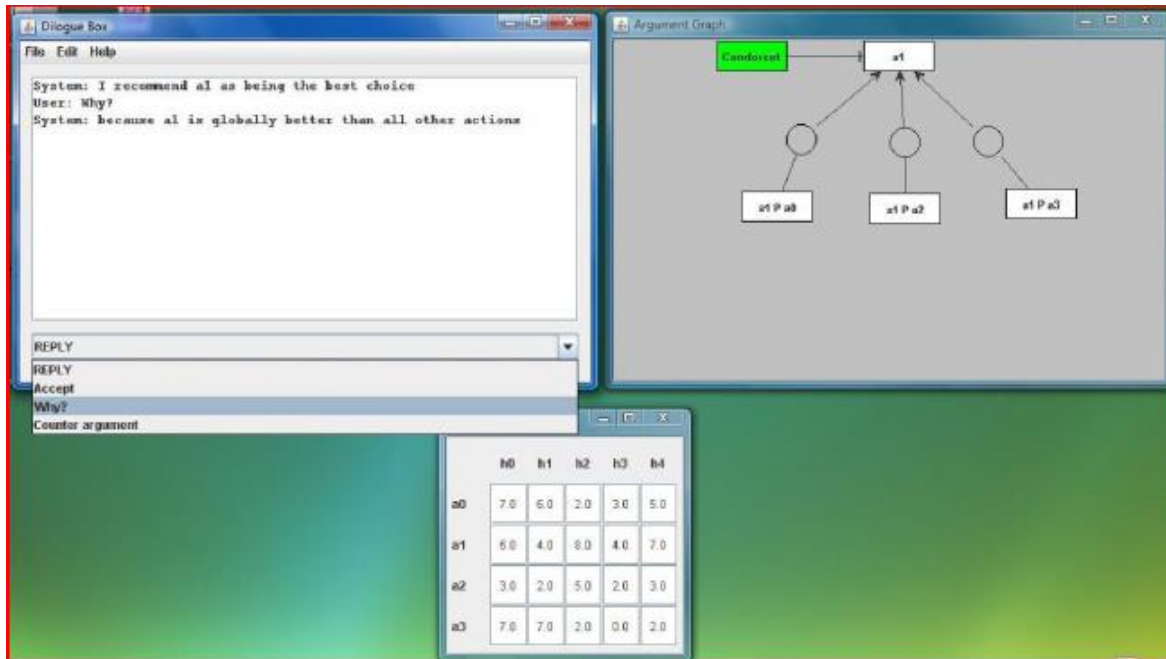


Figura 3.4: Interfaz gráfica del sistema ArgDec [Ouerdane, 2009]

### 3.3 “A Decision Support System Framework for Public Project Portfolio Selection with Argumentation Theory” [Cruz-Reyes et al., 2014]

Cruz-Reyes et al propusieron la generación de un marco de referencia para un sistema de soporte a la decisión (DSS) el cual está enfocado en el problema de cartera de proyectos públicos (SPP) mediante una caracterización de argumentos y diálogo a través de la teoría de la argumentación y conjuntos rugosos.

En otras palabras, se buscó generar un sistema de recomendación para la toma de decisiones el cual se apoya de una interfaz amigable y la generación de juegos de diálogo.

El DSS debe ser capaz de justificar sus acciones, en especial la recomendación mostrada al usuario, en este caso utilizando una aproximación basada en los argumentos, con apoyo de los esquemas de argumentación. En su trabajo el marco propuesto tiene un funcionamiento tal como se muestra en la Figura 3.5.



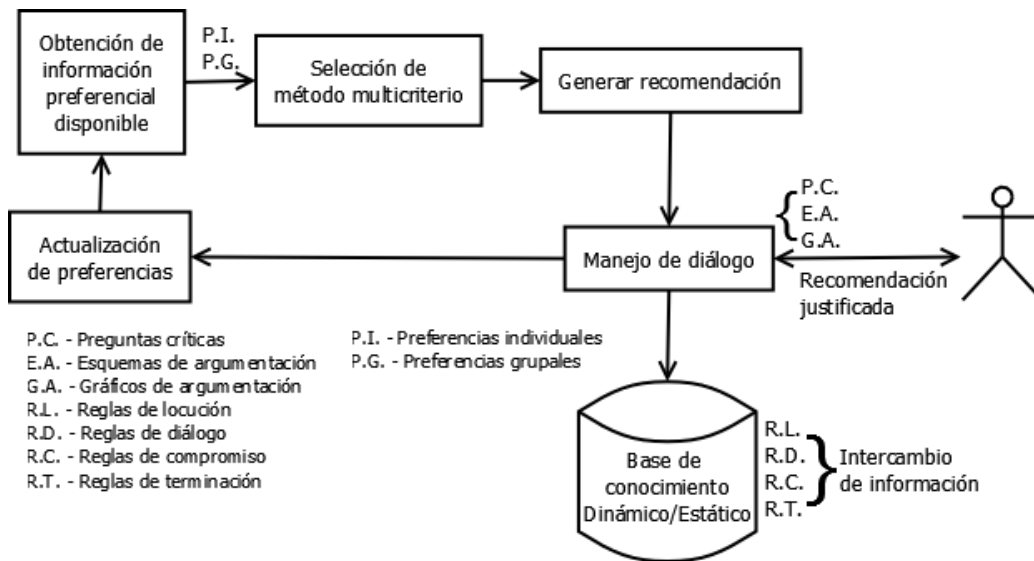


Figura 3.5: Marco de referencia propuesto, traducido de [Cruz-Reyes et al., 2014]

El proceso mostrado en la figura anterior inicia mediante la obtención de información preferencial disponible, la cual es provista por el DM, a partir de eso se selecciona el método multicriterio a utilizar para evaluar las carteras disponibles, generando de esta forma una recomendación. En este punto es donde entra la teoría de la argumentación en forma de juego de diálogo entre el DM y sistema, donde el segundo busca justificar dicha recomendación apoyándose de la base de conocimiento que se tenga en ese momento, la cual consta de dos partes.

- *Estática*: esta parte está conformada por las reglas de locución, compromiso, diálogo y terminación definidas para el juego de diálogo en cuestión, además de los esquemas de argumentación y las preferencias iniciales.
- *Dinámica*: esta parte incluye los argumentos generados durante el juego de diálogo y las preferencias actualizadas.

Conforme el diálogo va avanzado existe la posibilidad de que el DM ofrezca nueva información al marco que no conocía de manera previa, actualizando las preferencias. Debido a esto es necesario reiniciar el proceso desde la obtención de información preferencia disponible, lo cual puede llevar a un cambio en el método multicriterio seleccionado y la recomendación generada. Este ciclo se continúa hasta que el DM llegue a una conclusión y acepte o rechace de forma definitiva la recomendación del sistema.

### **3.4 Análisis comparativo**

Después de haber realizado la revisión del estado del arte, en la Tabla 3.2 se muestra un resumen de los trabajos analizados en este capítulo, así como una comparativa de ellos con el trabajo que se desarrolló en este proyecto.

Como se puede ver, aunque todos los trabajos están enfocados en la solución de un problema de selección de alternativas apoyándose de la teoría de la argumentación, solamente Cruz-Reyes et al y este proyecto están enfocados en el problema de cartera de proyectos públicos.

Por otra parte aunque existen prototipos de un sistema de recomendación implementados en los últimos dos trabajos vistos, estos solo llegan a un nivel conceptual, mientras que en este trabajo se desarrolló un prototipo funcional. También, el prototipo de Ouerdane está enfocado solamente en problemas de elección de alternativas de baja dimensión. Además, el prototipo implementado en este proyecto contiene al estándar de prueba TOPSIS dentro del conjunto de los estándares de prueba disponibles, el cual no se es utilizado por ninguno de los trabajos revisados.

Trabajo	Tipo de problema	Dimensión	Opciones de argumentos del DM	Aplicación de un sistema de recomendación	Tipos de estándares de prueba	Manejo de diagrama de transición de estados	Uso de esquemas de argumentación
Labreuche, 2006	Elección de alternativas.	No específica.	No específica.	No.	Basados en peso de criterios.	No.	No específica
Ouerdane, 2009	Elección de alternativas.	Baja.	Limitado.	Prototipo conceptual.	Basados en peso, orden lexicográfico y veto.	Si.	Si.
Cruz-Reyes et al., 2014	SPP.	Media.	No específica.	Prototipo conceptual.	No específica.	No específica.	Si.
Este trabajo	SPP.	Media.	Limitado.	Prototipo funcional.	Basados en peso, orden lexicográfico, veto y cercanía a la mejor solución.	Si, múltiples.	Si.

Tabla 3.2: Análisis del Estado del Arte

# Capítulo 4

## Propuesta de solución

---

---

Como se vio en el capítulo anterior, los sistemas de recomendación apoyados en la teoría de la argumentación han sido tratados anteriormente, por lo que se puede identificar un cierto patrón en la forma que dichos sistemas trabajan. A partir de esto se definió la forma en la cual se desarrolló este proyecto y cómo funciona su proceso de recomendación. En este capítulo se mostrará cómo se definieron cada uno de los componentes existentes en el sistema de recomendación justificada generado para este proyecto.

### 4.1 Propuesta de caracterización de tareas cognitivas en la ayuda a la decisión

Este trabajo está enfocado en dos tareas cognitivas, el *modelo de evaluación de las alternativas* y la *construcción de argumentos para la recomendación final*, ya que con ellas se establece el funcionamiento del prototipo de sistema de recomendación desarrollado. En la Figura 4.1 se muestra la manera en la que se aplican estas tareas cognitivas dentro de un marco de trabajo de apoyo a la decisión.

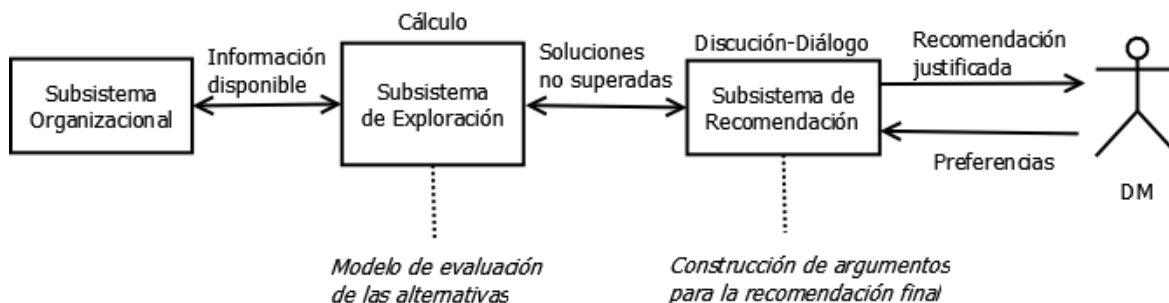


Figura 4.1: Esquema general de un framework de apoyo a la decisión.

#### 4.1.1 Modelo de evaluación de alternativas

Tsoukiàs [Tsoukiàs, 2007] define esta tarea cognitiva como el proceso en el cual se evalúan un conjunto de alternativas apoyándose de conjuntos de atributos, indicadores o

dimensiones para medir cada una de las posibles soluciones. En el caso de este proyecto, las alternativas serán las diferentes carteras existentes para el problema en cuestión; siendo cada una de estas soluciones evaluadas por diversos elementos como:

- Peso de cada uno de los criterios;
- Matriz de desempeño;
- Orden de jerarquía de criterios;
- Proyectos seleccionados por cada cartera.

Con esta información, el sistema puede definir un *estándar de prueba* a utilizar, con el cual evaluará cada una de las carteras en la instancia, definiendo para cada posible solución un valor, dicho valor posteriormente se analizará, permitiendo al sistema generar una recomendación para el usuario. Esto llevará a la construcción de argumentos de la recomendación final, el cual es el otro artefacto cognitivo que se aplica en el prototipo del sistema de recomendación de este proyecto.

Sin embargo, al existir un conjunto de varios estándares de prueba dentro del sistema, los cuales fueron explicados en el Capítulo 2, es necesario definir una forma o método que permita definir en qué momento es correcto utilizar un cierto estándar de prueba de manera específica. Para esto, los diferentes estándares de prueba utilizados son analizados para identificar posibles diferencias entre ellos.

**Mayoría simple:** En este caso, los criterios *son de igual importancia*, así como se considera solo la información ordinal existente desempeño.

**Método lexicográfico:** Para ser capaz de utilizar el *método lexicográfico* es necesario definir un *orden jerárquico lineal* en los criterios. Dicho orden establece que un criterio que se encuentre en una posición superior es infinitamente superior más importante que aquellos criterios que se encuentren en posiciones inferiores.

**Mayoría ponderada:** Este método, a diferencia de la mayoría simple, requiere que los criterios *no sean de igual importancia*.

**Suma de pesos:** En este método se hace una comparación entre *puntajes globales*, resultantes de la suma de los valores de criterios de una cartera, comparando dos alternativas para definir si existen condiciones de superación. Esto requiere de la consideración de los pesos de cada criterio.

**TOPSIS:** Para poder efectuar este estándar de prueba es necesario no solamente enfocarse en una solución óptima, sino también considerar la *lejanía* de los valores de las alternativas hacia la *peor solución posible calculada*. Lo anterior se realiza mediante el cálculo de la cercanía de cada cartera a su solución ideal, el proceso para llegar a este resultado se explicó en la sección 2.6.5 [Hwang y Yoon, 1981].

A partir de lo anterior, y con base en el trabajo doctoral de Ouerdane [Ouerdane, 2009], se definió un conjunto de propiedades que permitan identificar cuál estándar de prueba es el adecuado para utilizar en ese momento durante el juego de diálogo:

**Ordinalidad:** Al momento de comparar acciones, solamente la *información ordinal* dentro del desempeño es relevante. Esto quiere decir que la superación de una alternativa sobre otra se basa en la comparación de los valores en su desempeño.

**Anonimato:** Los criterios son intercambiables ya que tienen una *similitud* en su importancia. Un ejemplo de esto se presenta cuando no existe una preferencia específica en dichos criterios.

**Aditividad con respecto a coaliciones:** Es posible formular un valor aditivo para la importancia de un subconjunto de criterios.

**Aditividad con respecto a valores:** El valor de una alternativa es resultado de la suma de los valores en cada criterio.

**Veto:** El valor por el cual una solución supera a otra debe superar un *umbral de veto* para ser considerada lo suficientemente fuerte para ser aceptable.

**Lejanía de la peor solución:** Se debe considerar no solamente la cercanía de la solución a la mejor solución posible, sino que también debe tomarse en cuenta que tan alejada esta dicha solución de la peor solución posible que se pueda calcular.

Debido a la naturaleza de estas propiedades, algunas de estas no pueden coexistir al mismo tiempo, tal es el caso de la ordinalidad y el anonimato con la aditividad c.r.a. valores.

Para definir cuál estándar de prueba es correcto aplicar para el proceso de evaluación se utiliza el método establecido por Ouerdane [Ouerdane, 2009], en el cual cada estándar se identifica con un *conjunto de propiedades* y dependiendo de cuales propiedades estén activas o no, se elegirá aquel *estándar de prueba* que se adecue más a la información que se tiene.

Con base a la información anterior y apoyándose del trabajo de Ouerdane, se definió un conjunto de propiedades para cada uno de los estándares de prueba utilizados como se muestra en la Tabla 4.1.

El proceso de selección del estándar de prueba a utilizar consiste en identificar aquellas propiedades que estén en consideración durante un juego de diálogo y elegir el estándar que más se adecue a dicho conjunto de propiedades.

En el caso del orden lexicográfico y la mayoría ponderada, cuyos conjuntos de propiedades son idénticos, el factor que define el estándar a utilizar es si existe un orden de preferencia en los criterios la instancia evaluada, seleccionando el primero en caso de ser cierto y el segundo en caso contrario.

**Tabla 4.1:** Estándares de prueba y sus propiedades

Prueba	Ordinalidad	Anonimato	Aditividad con respecto a		Veto	Lejanía de la peor solución
			Coaliciones	Valores		
Mayoría Simple	✓	✓	✓			
Orden Lexicográfico	✓		✓			
Mayoría Ponderada	✓		✓			
Suma de Pesos				✓		
TOPSIS				✓		✓
Mayoría Simple + Veto	✓	✓	✓		✓	
Mayoría Ponderada + Veto	✓		✓		✓	

De esta manera el proceso de elección de estándar de prueba en el sistema está definido por un conjunto de reglas que le ofrecen la suficiente flexibilidad al sistema para poder desempeñarse de diferentes formas y poder llegar a una solución que ofrezca una mayor satisfacción al usuario.

#### **4.1.2 Construcción de argumentos para la recomendación final**

La segunda tarea cognitiva a la cual se enfoca este trabajo es en la *generación de la recomendación final* obtenida en el proceso de apoyo a la decisión, la cual este caso es realizado por el sistema implementado.

La recomendación final debe ser mostrada en una forma abstracta y formal utilizando el lenguaje en el cual el modelo de evaluación se encuentre trabajando, considerando un conjunto de elementos de suma importancia en el desarrollo de esta tarea [Tsoukiàs, 2008]:

- El analista debe asegurarse que el modelo es formalmente correcto;
- El cliente debe asegurarse que el modelo lo representa, que lo entiende y es capaz de usar sus conclusiones;
- La recomendación debe ser considerada legítima con respecto al proceso de decisión para el cual se pidió apoyo.

Este último punto cobra mucha importancia en el sistema implementado, ya que para que una recomendación dada por el experto sea considerada legítima, primero debe ser aceptada por el usuario. En el prototipo, el usuario no solo es capaz de aceptar o rechazar la recomendación, sino también puede *cuestionar las razones detrás de la recomendación* y analizar si las *justificaciones presentadas por el sistema son suficientes* para satisfacerlo o no. De igual manera el usuario es capaz de *manipular* ciertos elementos del modelo de evaluación, lo cual puede llevar a un cambio en la recomendación generada por el sistema.

Considerando lo anterior, el sistema está apoyado en la *teoría de la argumentación* para poder llevar a cabo el proceso de “legitimización” de la recomendación generada por el sistema, llevando a cabo las acciones mencionadas anteriormente basándose en un juego de diálogo entre el usuario y el sistema.



### 4.1.2.1 Reglas de juego de diálogo

Primeramente, es necesario definir las reglas que se seguirán dentro del juego de diálogo que se desarrollará entre el usuario y el sistema. Dichas reglas se dividen en cuatro conjuntos, los cuales se explicaron en profundidad en el Capítulo 2.

- Reglas de locución
- Reglas de compromiso
- Reglas de diálogo
- Reglas de terminación

Dentro de este prototipo se siguieron las reglas de *compromiso*, *diálogo* y *terminación* establecidas en [Ouerdane, 2009]. Sin embargo, las reglas de *locución*, aunque igualmente están fuertemente basadas en el trabajo de Ouerdane, recibieron ciertas modificaciones, tales como ofrecerle al sistema la capacidad de rechazar un argumento por parte del usuario si no satisface un cierto criterio de evaluación, así como permitir al usuario rechazar la recomendación generada por el sistema en diversos puntos del diálogo.

En la Tabla 4.2 se muestran las reglas de locución que se definieron para el prototipo.

**Tabla 4.2:** Reglas de locución para el juego de diálogo

Movimientos	Respuestas	Rendición	Estado del reclamo
afirmar( $\phi$ )	retar( $\phi$ ) ¿Cuál prueba? formular( $C$ , $tipo$ , $\phi$ , $id\_esquema$ )	aceptar( $\phi$ ) rechazar( $\phi$ )	Declarada
retar( $\phi$ )	argumentar( $\phi$ , $premisas$ , $id\_esquema$ )	retractar( $\phi$ )	Cuestionada
argumentar( $\phi$ , $premisas$ , $id\_esquema$ )	retar ( $premisas$ ) formular( $C$ , $tipo$ , $\phi$ , $id\_esquema$ ) afirmar( $\phi$ ) ¿Cuál prueba?	aceptar( $\phi$ ) aceptar( $premisas$ ) rechazar( $\phi$ )	Aceptada/ Rechazada
formular( $C$ , $tipo$ , $\phi$ , $id\_esquema$ )	retar( $no-C$ ) afirmar( $C$ ) afirmar( $no-C$ )	retractar( $\phi$ )	Rechazada
aceptar( $\phi$ )			Aceptada
retractar( $\phi$ )			Rechazada
¿Cuál prueba?	de acuerdo a ( $pf$ )		

#### 4.1.2.2 Diagrama de transición de estados

A partir del punto anterior es posible diseñar diagramas de transición de estados que permiten representar como se llevará a cabo el flujo del juego de diálogo entre el usuario y el sistema de recomendación. Estos diagramas regulan la forma en la que se lleva a cabo el diálogo, además de mostrar los movimientos disponibles para ambos jugadores en un momento específico de la interacción.

Ouerdane [Ouerdane, 2009], hace mención de la imposición de un razonamiento secuencial en los movimientos del diálogo al trabajar con este tipo de diagramas. Sin embargo, para este trabajo se utilizan dichos diagramas para limitar las posibles respuestas del usuario, ya que en este sistema no se toma en consideración la aplicación del lenguaje natural.

Dentro del prototipo se diseñaron dos diagramas de transición de estados, los cuales están basados en el establecido por Ouerdane. De estos diagramas se seleccionará uno para establecer un juego de diálogo con el usuario en una instancia, el criterio aplicado para definir que diagrama utilizar es si existe o no un orden de jerarquía en los objetivos desde la instancia, lo cual fuerza al sistema a trabajar directamente con el método lexicográfico como el estándar de prueba inicial.

**Diagrama de Transición de Estados 1 (DTE1):** Este diagrama es seleccionado si no existe una preferencia establecida hacia algún criterio desde la instancia. Se agregó un movimiento el cual le permite al sistema rechazar la sugerencia del usuario en caso de no existir las suficientes razones de soporte para respaldar dicha afirmación, explicándole el motivo a este y continuando con el proceso de recomendación, esto se hace para permitirle al usuario analizar sus decisiones, en base a datos objetivos.

En la Figura 4.2 se muestra la representación de DTE1, mostrando las acciones posibles para cada estado. Mientras que en la Figura 4.3 se detallan las condiciones y procesos aplicados durante la interacción, destacando aquellos procesos que invocan esquemas de argumentación.

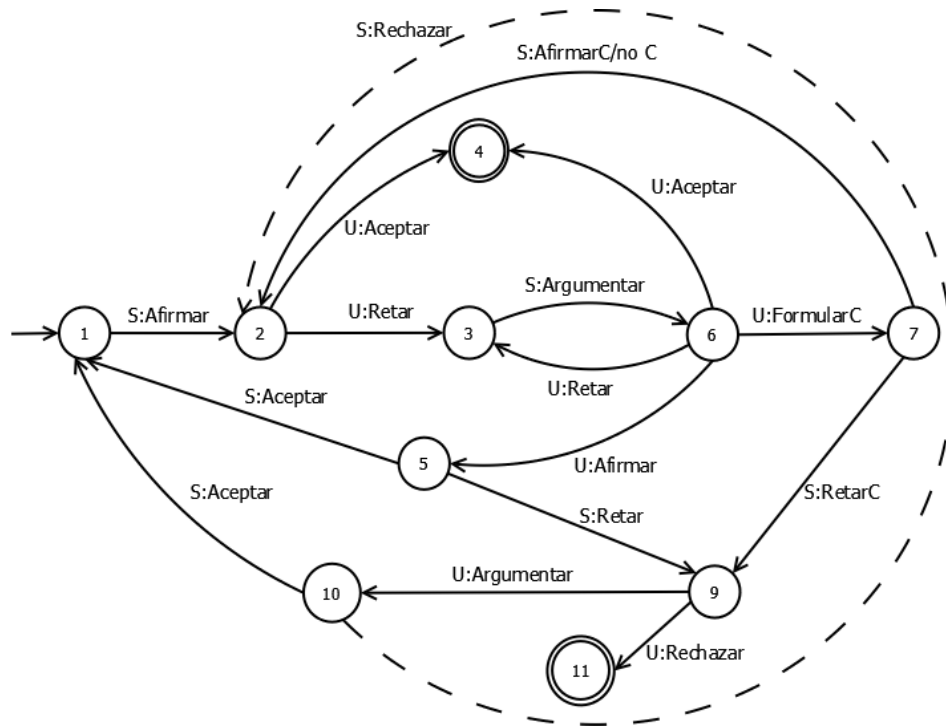


Figura 4.2: Diagrama de Transición de Estados 1 (DTE1).

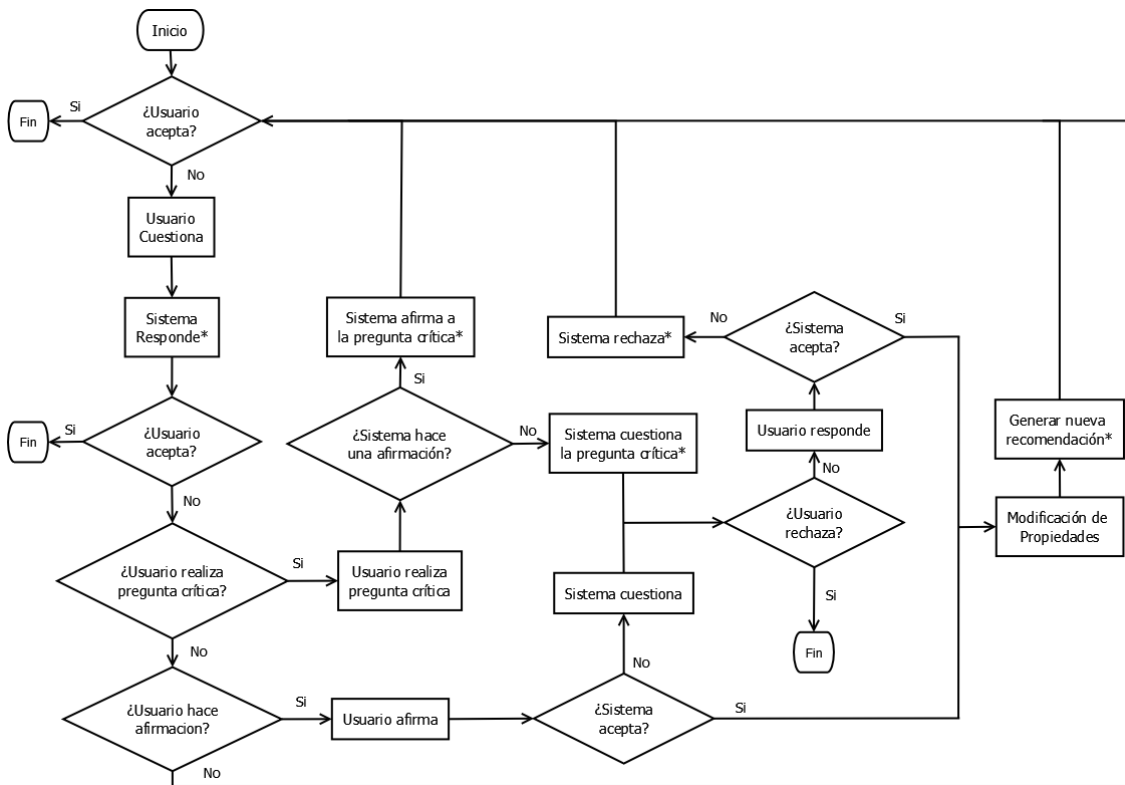
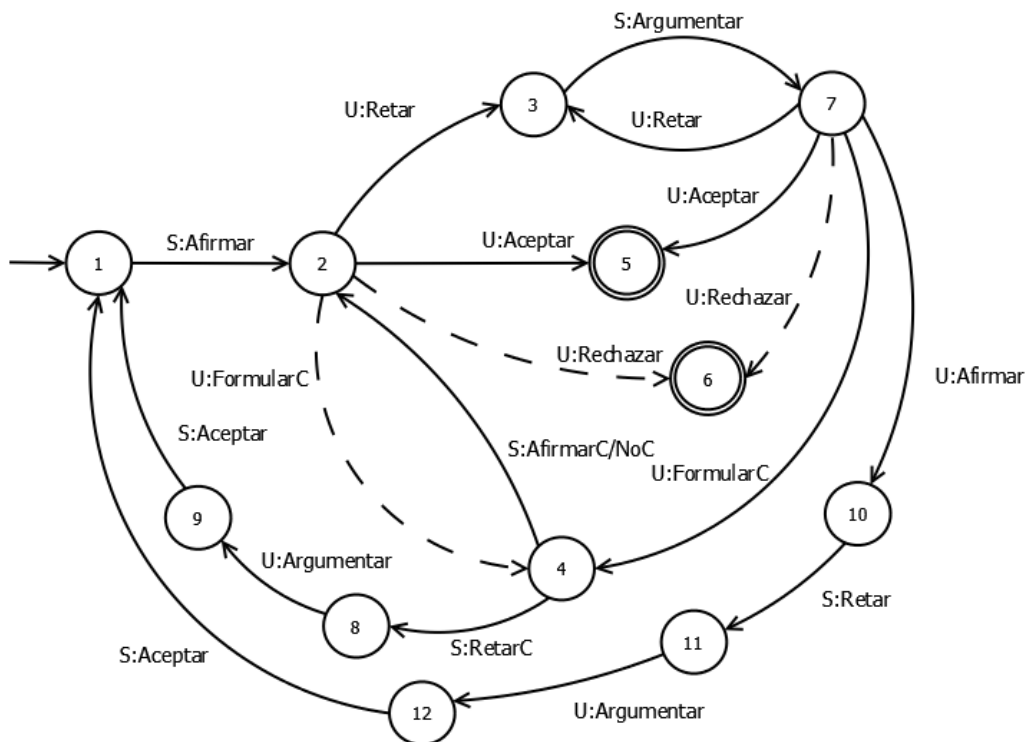


Figura 4.3: Diagrama de Flujo de DTE1.

**Diagrama de Transición de Estados 2 (DTE2):** Este diagrama es seleccionado si existe desde la instancia un orden jerárquico establecido en los criterios de esta. Este diagrama le permite al usuario la generación de pregunta críticas desde que el sistema hace una recomendación con el objetivo de mejorar su experiencia, ofreciéndole un mejor entendimiento de la instancia y el sistema en menor tiempo. De igual manera se le da la oportunidad de rechazar de manera definitiva la recomendación del sistema en dos puntos diferentes, al momento que el sistema genera la recomendación y después de que el sistema justifica su elección. La oportunidad de rechazo desde esos puntos le ofrece una mayor flexibilidad en su toma de decisión, lo cual a la vez lleva al usuario a sentirse satisfecho con su decisión, siendo este el objetivo final de un sistema de recomendación [Jameson et al, 2015].

De igual manera que DTE1, en la Figura 4.4 se muestra la representación del diagrama de transición de estados, mientras que en la Figura 4.5 se representa un diagrama de flujo que detalla las condiciones y procesos durante la interacción.



**Figura 4.4:** Diagrama de Transición de Estados 2 (DTE2).

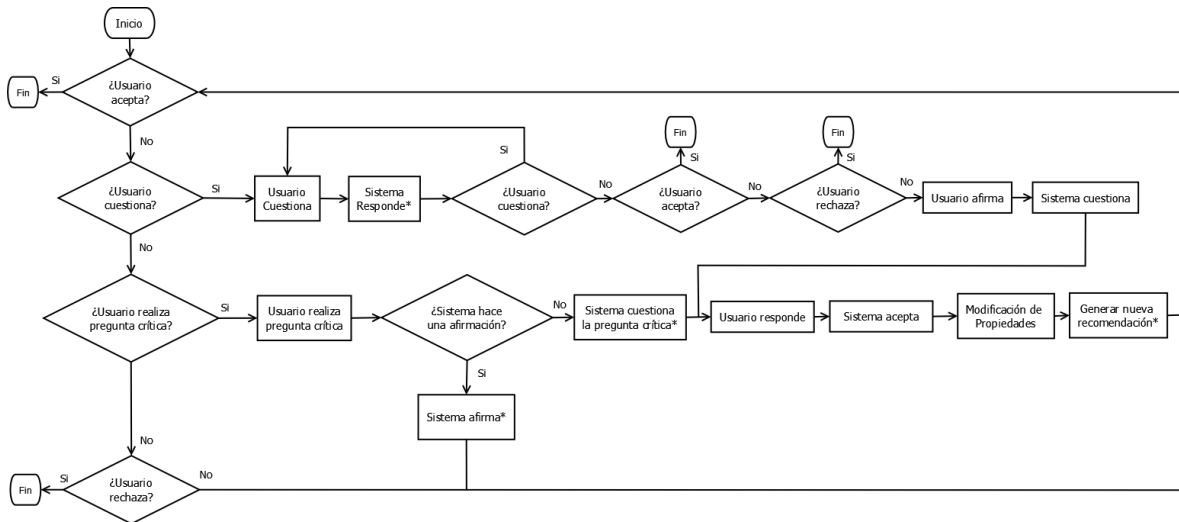


Figura 4.5: Diagrama de Flujo de DTE2.

#### 4.1.2.3 Esquemas de argumentación

El sistema se apoya en los esquemas de argumentación para poder llevar a cabo la aceptación o rechazo de una declaración o argumento, así como para obtener nueva información, la cual puede llevar a cambios en los valores de criterios de la instancia o en las propiedades de los estándares de prueba.

Esto a su vez puede dirigir a un cambio en el estándar de prueba a utilizar para la generación de la recomendación, así como en la solución que el sistema recomienda (véase el Anexo B para ver un ejemplo de lo anterior).

Los esquemas de argumentación utilizados en este sistema buscan evitar problemas tales como un posible ciclado en el proceso de recomendación, esto debido a que el usuario es capaz de modificar las propiedades y criterios en cualquier momento. Para ello se limitó la capacidad de alteración de datos hasta un cierto número de ciclos. Considerándose como ciclo cada vez que el sistema genera una nueva recomendación o acepta cambios en propiedades de estándar de prueba y preferencias.

El conjunto de esquemas de argumentación que se utilizan dentro del sistema, así como la aplicación de cada una de ellas se muestran en la Tabla 4.3.

**Tabla 4.3:** Esquemas de argumentación

<b>Esquema de Argumentación</b>	<b>Aplicación</b>
Razonamiento abductivo	Selección de un <b>estándar de prueba</b> a utilizar en el juego de diálogo.
Desde una posición para saber	Generación de una recomendación en <b>los ciclos iniciales</b> , capaz de modificar propiedades y aceptar nueva información.
Desde una opinión experta	Generación de recomendación en <b>ciclos avanzados</b> , no acepta cambios en propiedades para evitar ciclos infinitos.
Evaluación multicriterio de parejas	<b>Comparación</b> entre la recomendación y otra alternativa.
Ad ignorantiam	El sistema no genera inferencias, solo toma como <b>verdadero</b> la información obtenida.
Causa-efecto	Manejo de los <b>cambios</b> de propiedades o estándar de prueba.
Desde la parcialidad	Manejo de las <b>preferencias</b> del usuario por orden jerárquico.

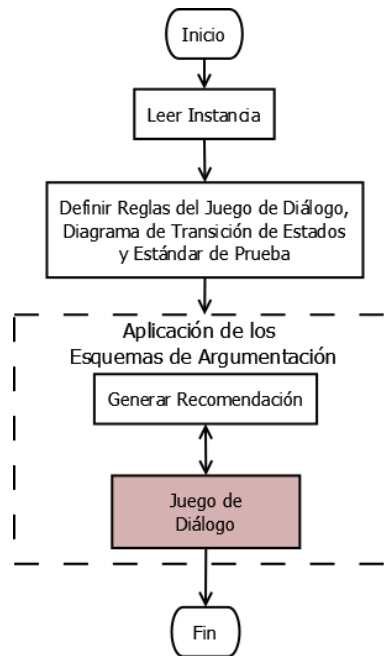
## 4.2 Propuesta de arquitectura

El sistema de recomendación con justificación desarrollado para este proyecto requiere de un conjunto de elementos clave para su funcionamiento.

- Reglas del Juego de Diálogo
- Diagrama de Transición de Estados
- Estándar de Prueba
- Juego de Diálogo
- Esquemas de Argumentación

Para conjuntar estos elementos en un proceso capaz de llevar a cabo los objetivos deseados, el sistema de recomendación está desarrollado de tal forma que siga una serie de pasos tal como se muestra en la Figura 4.6, donde inicialmente se hace una lectura de la instancia. Después se definen las reglas del juego de diálogo, el diagrama de transición de

estados a utilizar y estándar de prueba inicial, posteriormente, el sistema genera una recomendación y entra en un juego de diálogo con el usuario en el cual existe la posibilidad de que se genere un cambio del estándar de prueba o los esquemas de argumentación en uso, lo cual puede llevar a la definición de una nueva recomendación.



**Figura 4.6:** Propuesta de solución

#### 4.2.1 Diagrama modular – primera capa

Buscando llevar a cabo los pasos anteriores dentro de la aplicación, se identificaron cuatro procesos principales a realizar por parte del sistema para poder llevar a cabo un proceso de recomendación de manera adecuada.

1) *Cargar Instancia*: El sistema lee un archivo de texto, el cual provee al sistema con la información inicial necesaria para poder llevar a cabo una recomendación y después establecer un juego de diálogo entre éste y el usuario.

Los datos manipulados por este proceso son:

- Si el problema para el conjunto de objetivos es de maximización (buscando el mayor valor en cada uno de los criterios) o minimización (buscando el menor valor en cada uno de los criterios).

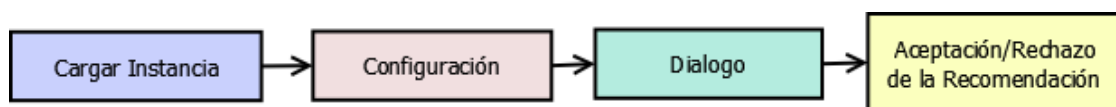
- Número de alternativas.
- Número de criterios.
- Peso de los criterios.
- Orden de preferencia de los criterios.
- Matriz de desempeño (alternativa/objetivo).
- Número de carteras obtenidas de forma previa.
- Tope presupuestal.
- Umbrales de veto.
- Composición de las carteras (cartera/alternativa) y costo presupuestal por cartera.

2) *Configuración*: A partir de la información obtenida en el proceso anterior, el sistema la analiza para determinar la configuración inicial de diversos elementos tales como las reglas de locución, el diagrama de transición de estados y el estándar de prueba, para de esta manera poder generar una recomendación inicial.

3) *Diálogo*: En este proceso el usuario y el sistema entran en un intercambio de argumentos, en el cual el sistema buscará convencer al usuario para que acepte la recomendación establecida por el primero, sin embargo, el usuario también es capaz de rechazar la recomendación actual o manipular la configuración inicial para forzar al sistema a generar una nueva recomendación a partir de sus preferencias.

4) *Aceptación/Rechazo de la Recomendación*: El usuario puede concluir si la recomendación obtenida es satisfactoria a su parecer o si ésta no lo convence, aun cuando sea la mejor solución disponible de manera definitiva. Se establece una opción de rechazo debido a que se hace consideración del factor humano con respecto a las decisiones del usuario, buscando que este se sienta satisfecho con su decisión, siendo este el objetivo final de un sistema de recomendación [Jameson et al, 2015].

En la Figura 4.7 se muestra el orden de ejecución con el que se ejecutan los módulos.



**Figura 4.7:** Diagrama Modular – Primera Capa



## 4.2.2 Diagrama modular – segunda capa

Profundizándose dentro de cada uno de los módulos, se puede ver que varios de ellos requieren de un subconjunto de procesos internos para que se lleven a cabo su respectivo objetivo de manera adecuada. En la Figura 4.8 se puede ver con más detalle los módulos que componen a cada uno de los elementos definidos en la sección anterior.

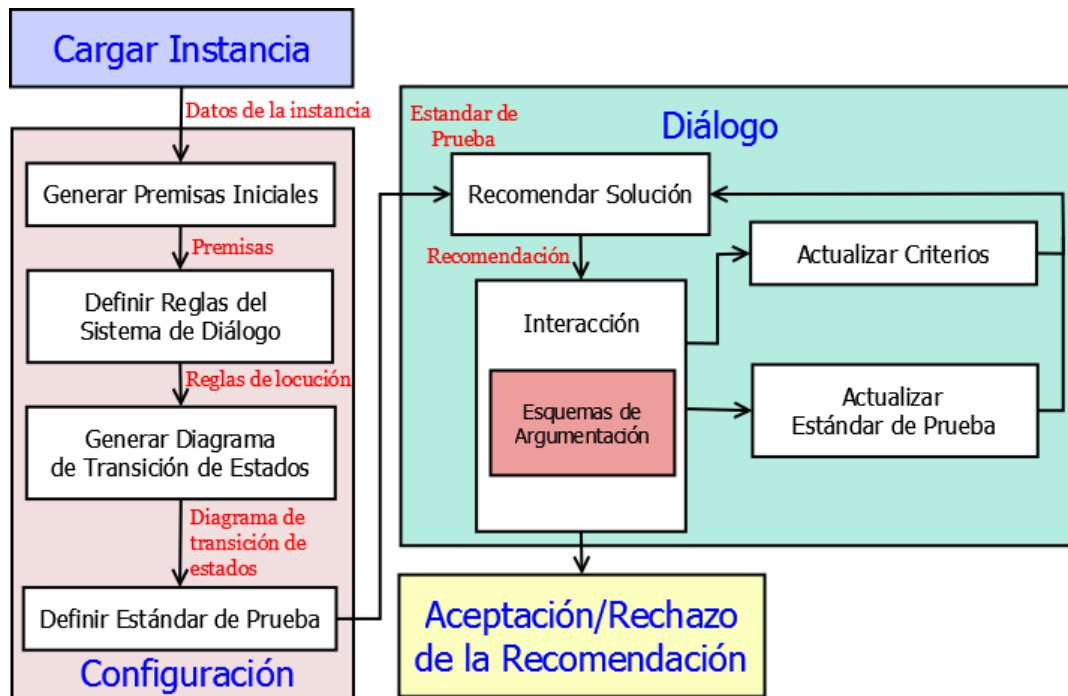


Figura 4.8: Diagrama Modular – Segunda Capa

En base esta figura, es necesario conocer a mayor detalle cada uno de los procesos internos en los módulos de configuración y diálogo para comprender el funcionamiento dentro de ellos.

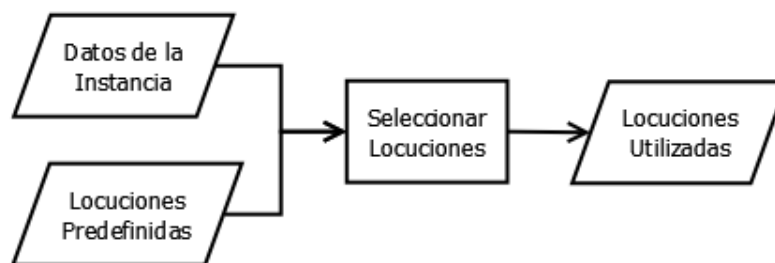
### 4.2.2.1 Módulo de configuración

El proceso de configuración está compuesto por los siguientes módulos:

*Generar Premisas Iniciales:* A partir de los datos de la instancia, el sistema genera un conjunto de premisas, que serán utilizadas en el juego de diálogo. Dichas premisas servirán para definir la conclusión de cada uno de los argumentos utilizados en el diálogo

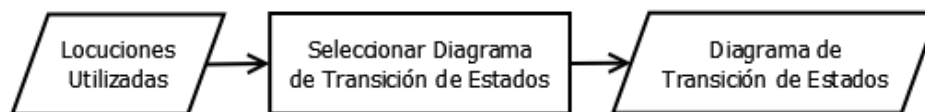
con el objetivo de llegar a un acuerdo entre sistema y el usuario, ya sea de aceptación o rechazo.

*Definir Reglas del Sistema de Diálogo:* El sistema inicializará las reglas de compromiso, diálogo y terminación que estén predefinidas para su uso en el juego de diálogo, Además, utilizando los datos de la instancia, el sistema seleccionará un conjunto de reglas de locución buscando facilitar el diálogo entre el usuario y el sistema. Cabe aclarar que una vez definidas estas reglas no se podrán modificar hasta cargar nuevamente una instancia. En la Figura 4.9 se puede ver cómo funciona dicho proceso de selección.



**Figura 4.9:** Definición de las Reglas de Locución

*Generar Diagrama de Transición de Estados:* En base a las reglas de locución formuladas, se selecciona el diagrama de transición de estados a utilizar en el juego de diálogo que se llevará a cabo después. Debido a que las reglas del juego de diálogo solo se modifican cuando el sistema lee una instancia, el diagrama de transición de estados no sufre ningún cambio en algún otro punto. La Figura 4.10 muestra de forma gráfica su funcionamiento.



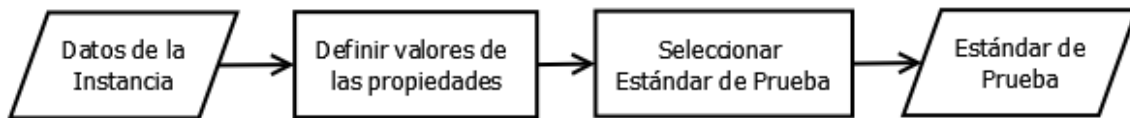
**Figura 4.10:** Generar Diagramas de Transición de Estados

*Definir Estándar de Prueba:* Apoyándose de los datos recibidos en la instancia, específicamente en los pesos de los criterios y si existe un orden de preferencia en estos, el sistema buscará seleccionar el estándar de prueba inicial que más convenga utilizar, para posteriormente ser capaz de recomendar una solución que satisfaga al usuario o se acerque lo suficiente a sus preferencias. Este proceso se puede ver representado en la Figura 4.11.

Para el estándar de prueba inicial se analizan cuatro de las propiedades:

- Ordinalidad: verdadera si ninguno de los pesos normalizados de los criterios es igual o mayor que 0.6.
- Anonimato: verdadero a menos de que exista un orden de preferencia en los criterios.
- Aditividad c.r.a. coaliciones: verdadera si la ordinalidad es verdadera.
- Aditividad c.r.a. pesos: falsa si existe ordinalidad o anonimato.

El veto y la lejanía a la peor solución se consideran falsos por defecto para la selección del estándar de prueba inicial.

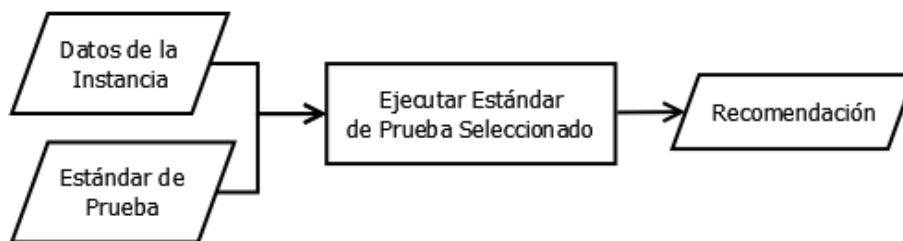


**Figura 4.11:** Definir Estándar de Prueba

#### 4.2.2.2 Módulo de diálogo

Por otro lado, los módulos que componen el proceso de diálogo son los siguientes:

*Recomendar Solución:* Utilizando los datos de la instancia y aplicando el estándar de prueba seleccionado en el proceso de configuración, el sistema analiza el conjunto de carteras disponibles dentro de la instancia y elige aquella que ofrezca el mejor resultado para el estándar en uso, dicha cartera se define como la recomendación. La Figura 4.12 muestra su proceso de manera gráfica.



**Figura 4.12:** Recomendar Solución

*Interacción:* En este módulo es donde se lleva a cabo el juego de diálogo entre el usuario y el sistema, en el cual puede aceptarse o rechazarse la recomendación, proporcionar nueva información al sistema o hacer cambios en los valores de ciertos criterios o propiedades de los estándares de prueba con el objetivo de generar una nueva recomendación.

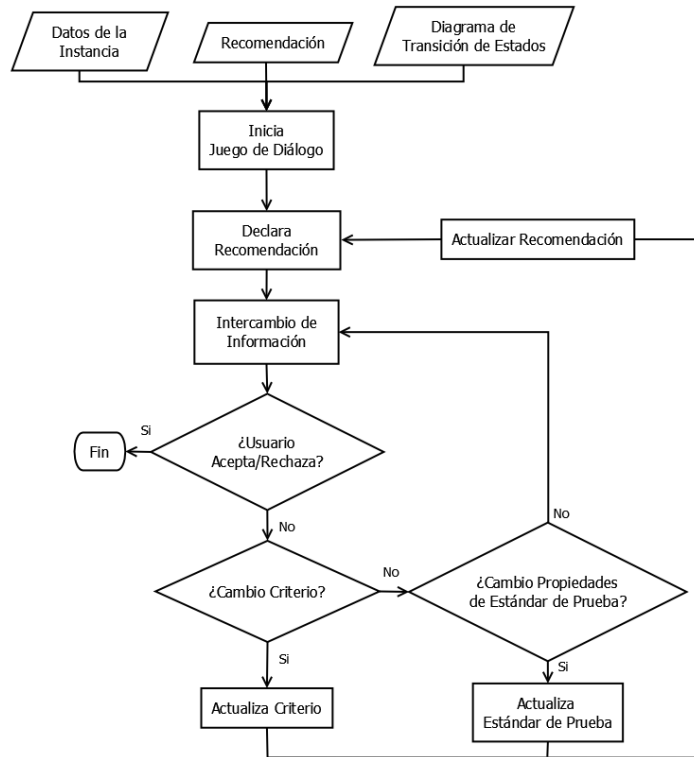
*Actualizar Criterios:* Puede ser utilizado en cualquier momento de la recomendación o interacción. Genera una nueva recomendación a partir de la nueva información.

*Actualizar Estándar de Prueba:* Si el usuario hace una declaración enfocada hacia alguna de las propiedades de los estándares de prueba, el sistema puede modificar el estado actual de las propiedades afectadas por dicha declaración, por lo que debe volverse a generar una nueva recomendación utilizando el estándar de prueba más recomendable en base a los nuevos conocimientos obtenidos.

El proceso de interacción, así como las actualizaciones de criterios y del estándar de prueba en uso se pueden utilizar para representar la estructura del juego de diálogo, en donde tras haber leído la instancia y obtenido sus datos, haber definido el diagrama de transición de estados a utilizar además de que el sistema haya seleccionado una de las carteras disponibles como la recomendación inicial, se puede llevar a cabo el juego de diálogo entre el usuario y el sistema.

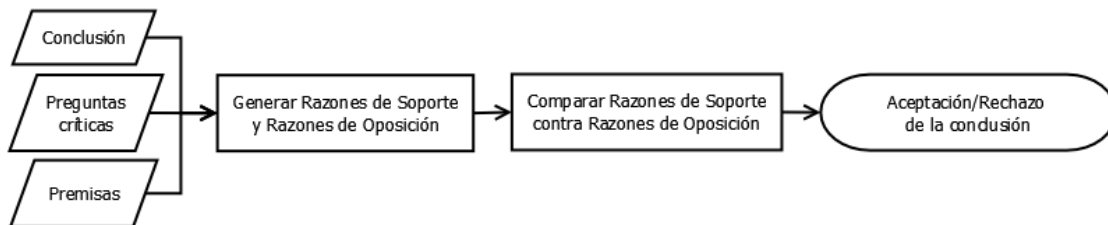
A partir de este punto ambos jugadores entran en un intercambio de argumentos, donde el usuario puede modificar los valores de los criterios en una o varias de las alternativas o hacer una declaración dentro del juego de diálogo que cambie las propiedades de los estándares de prueba, lo cual provoca que se ejecute el proceso correspondiente al cambio realizado.

El juego de diálogo finaliza cuando el usuario acepta o rechaza de manera definitiva la recomendación establecida por el sistema en ese momento en vez de continuar con el diálogo, en la Figura 4.13 se muestra una representación gráfica de lo anterior.



**Figura 4.13:** Estructura del juego de diálogo

*Esquemas de Argumentación:* Dentro del módulo de interacción se hace uso de los esquemas de argumentación para definir la aceptación o rechazo de cada declaración establecida durante el juego de diálogo, incluyendo la modificación de propiedades de los estándares de prueba además de la aceptación o rechazo de la recomendación actual. Este proceso inicia con la recopilación de la declaración realizada, considerada como conclusión, las premisas relacionadas y el conjunto de preguntas críticas del esquema de argumentación en uso, posteriormente se generan las razones de soporte y oposición a partir de dichas premisas y se comparan para definir si la declaración tiene las suficientes razones de soporte para que sea aceptada. En la Figura 4.14 está representado este proceso de manera gráfica.



**Figura 4.14:** Estructura básica de un esquema de argumentación

# Capítulo 5

## Desarrollo del prototipo

---

---

Habiéndose desarrollado una propuesta de solución para el problema que se busca resolver en este trabajo, la cual fue explicada en el capítulo anterior, el siguiente paso a realizar es la implementación de dicha propuesta, buscando que el prototipo resultante sea capaz de llevar a cabo las acciones deseadas. En este capítulo se analizará cómo se llevó a cabo el desarrollo de cada uno de los diferentes elementos que conforman el prototipo funcional implementado en este trabajo.

### 5.1 Configuración de una instancia de selección de cartera

El primer paso que realiza el prototipo es la obtención de datos para poder llevar a cabo un proceso de recomendación, dichos datos están definidos en una instancia previamente creada por el usuario. En la sección 4.2.1 se mencionó la información que el sistema requiere para poder realizar las tareas próximas, sin embargo, cada elemento requiere de un cierto formato para que el prototipo sea capaz de captar los datos que hay en la instancia.

- 1) Definir si el problema es de maximización (escribir "max") o minimización (escribir "min")
- 2) Número de alternativas
- 3) Número de objetivos
- 4) Peso de cada objetivo separando cada uno por un espacio
- 5) Orden de preferencia (Si no existe preferencia, escribir 0 1 2 3, caso contrario, definir el orden deseado)
- 6) Tabla de desempeño (alternativa/objetivo). En cada renglón escribir los valores de los criterios separando cada valor por un espacio

- 7) Número de carteras
- 8) Tope presupuestal
- 9) Umbrales de veto separando cada valor por un espacio
- 10) Composición de las carteras (cartera/alternativa) y costo presupuestal. En cada renglón escribir los valores de las alternativas separando cada valor por un espacio y escribiendo al final del renglón el costo presupuesta.

## 5.2 Interfaz gráfica de usuario y mapa de navegación

Para este prototipo se implementó una interfaz gráfica que le permite al usuario definir la instancia con la desea trabajar, establecer un juego de diálogo con el sistema e incluso manipular propiedades dentro del mismo para que se adapten a sus preferencias. Dicha interfaz está compuesta de un conjunto de ventanas las cuales le permiten al usuario realizar las actividades mencionadas previamente. Esta interfaz fue desarrollada en inglés con el objetivo de permitir su difusión de manera internacional. En esta sección se hará enfoque únicamente de la ventana principal, que se muestra en la Figura 5.1, la cual aparece desde el inicio de la ejecución del sistema.

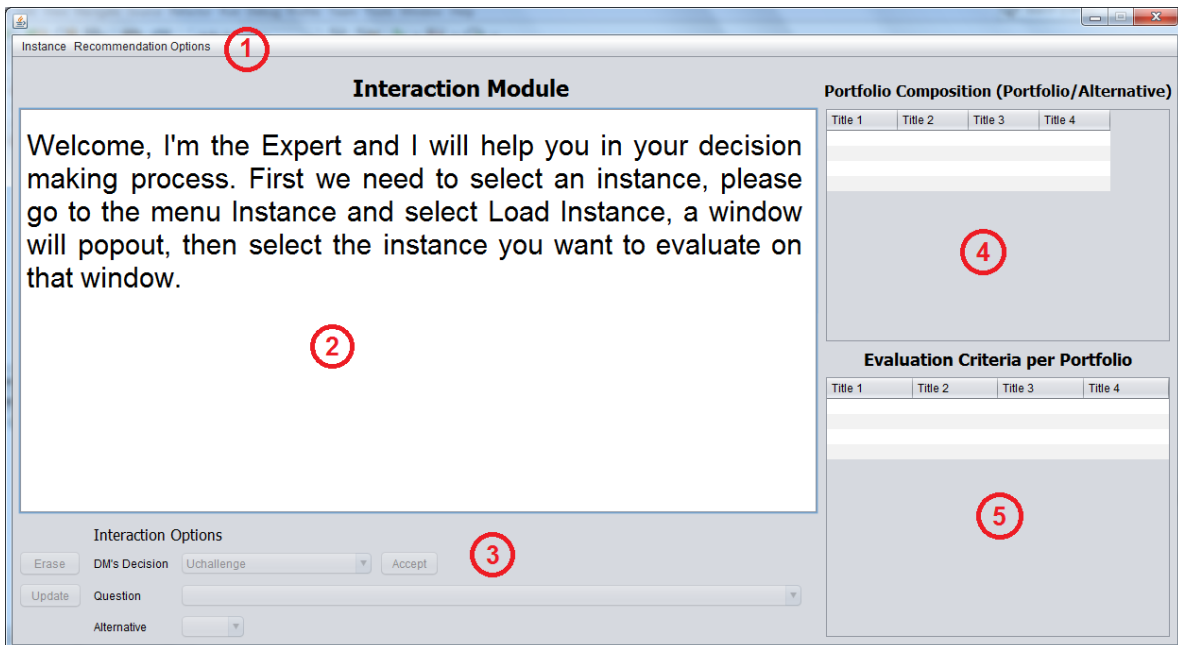


Figura 5.1: Ventana principal.

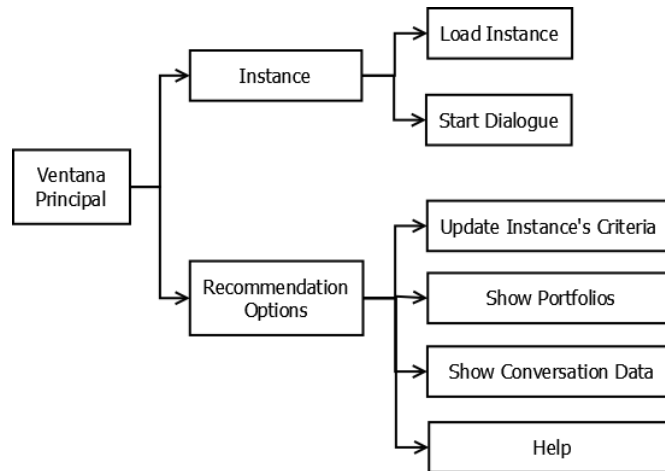
Los elementos marcados en la figura previa se explican a continuación:

- 1) Barra de menús: Menús que se permite hacer diversas acciones relacionadas a la instancia y sus propiedades.
  - a. Menú *Instance*: Este menú permite al usuario leer instancias, así como iniciar y reiniciar un juego de diálogo en base a la instancia leída.
  - b. Menú *Recommendation Options*: Este menú permite al usuario actualizar los valores de las alternativas; visualizar la composición, valor por criterio y presupuesto de las carteras; ver el estado actual del juego de diálogo, así como ver una ventana de ayuda para apoyarse en el uso de este sistema de recomendación.
- 2) Pantalla de diálogo: En este cuadro se muestran las acciones seleccionadas por el usuario, así como las respuestas del sistema.
- 3) Opciones de interacción: En este espacio es donde el usuario lleva a cabo su diálogo con el sistema, dividiendo las opciones de diálogo en tres segmentos:
  - a. *DM's Decision*: Acción realizada por el DM.
  - b. *Question*: Declaración relacionada con la acción seleccionada por el DM.
  - c. *Alternative*: En caso de existir múltiples alternativas en la declaración esta sección se activará para seleccionar la opción deseada.
- 4) Composición de las carteras (Cartera - Alternativa): En esta tabla se muestran las carteras con su correspondiente vector binario de alternativas.
- 5) Criterios de evaluación por cartera: En esta sección se muestran los valores por criterio de cada una de las carteras.

A continuación, se presenta en la Figura 5.2 el mapa de navegación correspondiente a las diferentes ventanas que se generan al seleccionar los submenús de este prototipo.

En las siguientes secciones de este capítulo se explicará el funcionamiento de cada uno de los diferentes elementos dentro del este sistema de recomendación.





**Figura 5.2:** Mapa de navegación.

### 5.3 Proceso de generación de la recomendación inicial

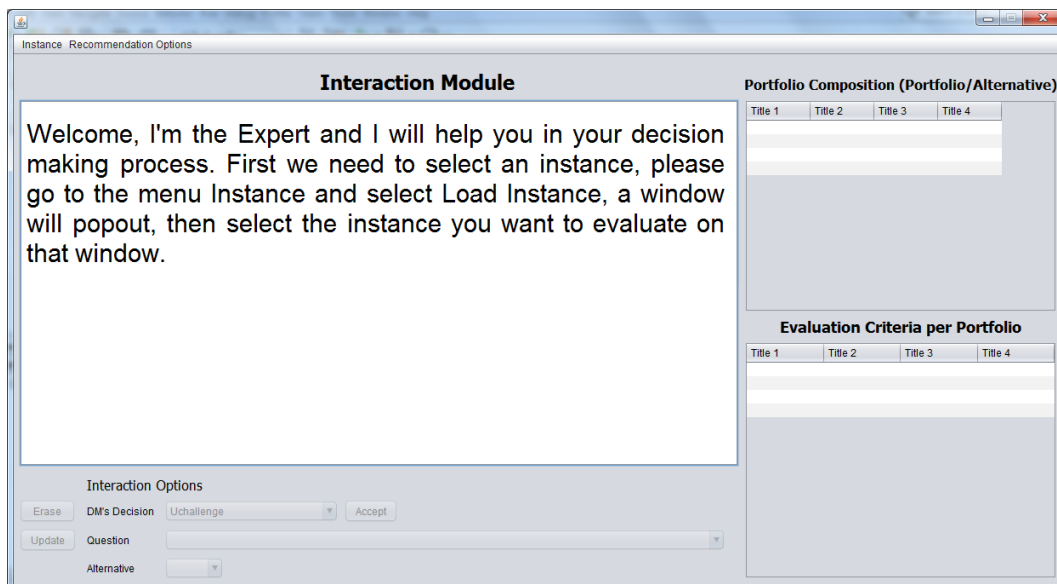
Antes de que el usuario sea capaz de entrar en un juego de diálogo con el sistema, debe interactuar con este de tal forma que le permita generar una recomendación inicial, la cual marcará el punto de partida para el intercambio de argumentos entre el usuario y el prototipo. Esa interacción previa al diálogo consta de la lectura de la instancia.

#### 5.3.1 Lectura de la instancia

Al inicio de la ejecución del sistema se le muestra al usuario un mensaje de saludo dentro del cuadro de texto utilizado para el diálogo, tal como se ve en la Figura 5.3. En dicho mensaje se le pide al usuario, como paso inicial, hacer la selección de la instancia que se utilizará durante el proceso de apoyo a la toma de decisión. Para esto el usuario deberá ir al menú *Instance* y seleccionar la opción *Load Instance* tal como se lo pide el sistema en el mensaje de saludo. El contenido de la instancia se mostró en la sección 4.2.1 y se puede ver un ejemplo de su formato en el Anexo E.

#### Función *read*

Después de eso se le mostrará una ventana emergente al usuario, en esta ventana el usuario deberá seleccionar la instancia a utilizar. Cuando una instancia es seleccionada, se ejecuta la función *read* la cual se encarga de leer el archivo seleccionado y extraer la información necesaria que hay en este. Los datos obtenidos se almacenan tal como se muestra en la Tabla 5.1.



**Figura 5.3:** Inicio de ejecución del sistema.

**Tabla 5.1:** Variables de la función *read*.

<b>Variable</b>	<b>Valor almacenado</b>
<i>max</i>	max = 1 si el problema es de maximización, max = 0 si el problema es de minimización
<i>performance_table</i> [][]	Tabla de desempeño (alternativa/criterio)
<i>original_table</i> [][]	Respaldo de <i>performance_table</i>
<i>new_table</i> [][]	Tabla de desempeño (cartera/criterio)
<i>order</i> []	Orden de preferencia
<i>weight</i> []	Peso normalizado de cada criterio
<i>veto</i> []	Veto de cada criterio
<i>portfolio</i> []	Composición de las carteras
<i>budget</i> []	Presupuesto por cartera
<i>max_budget</i>	Tope presupuestal

### **Función *calculateNewTable***

Dentro de la función *read* también se ejecuta la función *calculateNewTable*, la cual calcula el valor total por criterio en cada cartera basándose en los valores obtenidos en la tabla de desempeño, almacenados en la variable *performance\_table*, así como en la composición de las carteras, la cual se guardó en la variable *portfolio*. A continuación, se

muestra el código correspondiente a esta función, como se puede ver, los valores resultantes de la operación que utiliza las dos variables previamente mencionadas se almacenan en la variable *new\_table*.

```

public static void calculateNewTable(){
    for (int i = 0; i < port; i++)
        Arrays.fill(new_table[i],0);

    for (int i = 0; i < port; i++) {
        for (int j = 0; j < m; j++) {
            for (int k = 0; k < n; k++) {
                new_table[i][j] += (performance_table[k][j]*portfolio[i][k]);
            }
            System.out.print(new_table[i][j]+" ");
        }
        System.out.println("");
    }
}

```

### 5.3.2 Definición del estándar de prueba a utilizar

Después de seleccionar una instancia y que el sistema haya finalizado el proceso de lectura de instancia, se mostrará otro mensaje en el cuadro de texto pidiendo ir nuevamente al menú *Instance*, seleccionando la opción *Start Dialogue*, esto con el objetivo de permitirle al sistema iniciar el juego de diálogo con el usuario, esto se muestra en la Figura 5.4.

**Portfolio Composition (Portfolio/Alternative)**

	a0	a1	a2	a3	a4	a5	a6	a7	a8	a
P0	0	0	1	1	0	0	1	0	0	0
P1	0	0	1	1	0	0	1	0	1	0
P2	0	1	1	1	0	0	1	0	0	0
P3	0	1	0	1	0	0	0	0	1	1
P4	0	1	0	1	0	0	0	1	1	0
P5	0	0	1	1	0	0	1	0	1	1
P6	0	0	1	1	0	0	1	0	1	0
P7	0	0	1	1	0	1	1	0	0	0
P8	0	1	1	1	0	0	1	0	1	0
P9	0	0	1	1	0	1	1	0	0	0

**Evaluation Criteria per Portfolio**

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Portfolio 0	86530	62840	3430	4440
Portfolio 1	86860	61360	3810	4290
Portfolio 2	83845	62385	3770	4320
Portfolio 3	78960	52190	4190	4205
Portfolio 4	79780	50995	4200	4175
Portfolio 5	82090	57905	3970	4285
Portfolio 6	89030	60280	3725	4420
Portfolio 7	88595	64625	3155	4300
Portfolio 8	85670	57580	4010	4330
Portfolio 9	79935	69110	3505	4410

Figura 5.4: Mensaje del sistema después de la lectura de la instancia.

Al momento de seleccionar esa opción, se ejecutan un conjunto de procesos los cuales servirán para generar la recomendación inicial. El primero de estos procesos es la definición de propiedades y selección del estándar de prueba a utilizar para la evaluación de las soluciones.

### **Función *defineProperties***

Para llevar a cabo lo anterior se ejecuta la función *defineProperties*, la cual se divide en dos secciones, la primera sección corresponde a la definición de las propiedades previo a iniciar el juego de diálogo, siendo ésta a la cual nos enfocaremos en esta sección. Para establecer si cada una de las propiedades de estándar de prueba disponibles se deben utilizar o no se aplican las condiciones que se muestran en la Tabla 5.2.

**Tabla 5.2:** Condicionales de activación de las propiedades de los estándares de prueba.

<b>Propiedad</b>	<b>Condicional</b>
Ordinalidad	Activo a menos que el valor de uno de los elementos de <i>weight</i> sea igual o supere 0.6.
Anonimato	Activo a menos que los elementos de la variable <i>order</i> no sigan un orden ascendente de 0 a 4.
Aditividad c.r.a. coaliciones	Activo solo si ordinalidad está activo.
Aditividad c.r.a. valores	Activo solo si ordinalidad y anonimato están inactivos.
Veto	Inactivo por defecto
Lejanía de la peor solución	Inactivo por defecto

La propiedad de *veto* se maneja inactiva por defecto debido a que existen varios estándares de prueba que tienen versiones con y sin veto, por lo que la opción de manejar esta propiedad se le ofrece al usuario durante el juego de diálogo. Por otra parte, la *lejanía de la peor solución* se maneja de igual manera falsa por defecto por el hecho de que el sistema utiliza comparaciones sencillas entre las carteras durante el proceso de generación de recomendación inicial, permitiendo al usuario aplicar estándares más complejos durante el juego de diálogo.

### **Función *analysis***

Después de definir las propiedades de estándar de prueba se aplica la función *analysis*, donde el sistema las analiza y las compara con las pertenecientes a cada uno de los estándares de prueba disponibles. Se selecciona aquel estándar cuya mayor cantidad de propiedades correspondientes se encuentren activas en ese momento en el sistema.

### **5.3.3 Selección de la recomendación**

Con el estándar de prueba seleccionado, ahora se puede seleccionar una de las carteras como la recomendación inicial, para ello es necesario aplicar la función *proofStandardSelection*, la cual a su vez llamará a la función correspondiente del estándar de prueba elegido por el sistema.

### **Función *proofStandardSelection***

La función *proofStandardSelection* está compuesta de un conjunto de condicionales las cuales mandan llamar a la función correspondiente al estándar de prueba seleccionado, estas funciones encontrarán la opción de mejor calidad entre las carteras disponibles siguiendo los pasos de su estándar de prueba correspondiente. En el proceso de la generación de la recomendación inicial, la selección del estándar de prueba corresponde al resultado obtenido por la función *analysis*, la cual se menciona en la sección anterior. A continuación se presenta en la Tabla 5.3 un listado que muestra la función que se ejecuta a partir del estándar de prueba seleccionado.

**Tabla 5.3:** Funciones de los estándares de prueba.

<b>Función</b>	<b>Estándar de prueba seleccionado</b>
<i>majority</i>	Mayoría simple con y sin veto
<i>lexicographic</i>	Orden lexicográfico
<i>weightedMajority</i>	Mayoría ponderada con y sin veto
<i>weightedSum</i>	Suma de pesos
<i>TOPSIS</i>	TOPSIS

La razón por la que estos *proofStandardSelection* y *analysis* trabajan de manera separada es debido a que además de poder modificar propiedades, los usuarios son capaces de modificar directamente el estándar de prueba en uso, la forma de hacer esto se explicará más adelante dentro de este capítulo.

#### **5.3.4 Definición de las reglas del sistema de diálogo**

Ya teniendo una solución definida como la recomendación inicial, es necesario definir como se llevará a cabo el juego de diálogo entre el usuario y el sistema, para esto primero se deben definir las reglas del sistema de diálogo.

Las reglas de compromiso, diálogo y terminación al ser fijas para cualquier juego de diálogo, se generan desde el arranque del programa y se define la forma en la que serán utilizadas en un diálogo cuando se identifica el diagrama de transición de estados que se vaya a utilizar para este. Sin embargo, es necesario definir el conjunto de reglas de locución que se van a aplicar para cada juego de diálogo, dichas reglas se elegirán mediante un proceso de selección.

#### **Función *generateSTD* (primera parte)**

Para definir las reglas de locución que se utilizarán durante el juego de diálogo se hace uso de la función *generateSTD*, la cual también se utiliza para generar el diagrama de transición de estados a utilizar, esta parte de la función se explicará en la siguiente sección. Como se mencionó en la sección 4.1.2.2 el factor que define que diagrama de transición de estados será elegido depende de si existe o no una preferencia en los criterios dentro de la instancia. A partir del DTE elegido se seleccionan un conjunto de reglas de locución las cuales se van a aplicar en el próximo juego de diálogo entre el sistema y el usuario.

#### **5.3.5 Generación del diagrama de transición de estados**

Después de haber definido las reglas de locución que se utilizarán en el juego de diálogo, es posible generar un diagrama de transición de estados para definir el flujo del intercambio de argumentos entre el usuario y el sistema. Este diagrama también definirá el punto inicial del diálogo, así como los puntos donde se llega a la finalización de este.

### **Función *generateSTD* (segunda parte)**

El sistema ejecuta la segunda parte de la función *generateSTD* para generar el diagrama de transición de estados a utilizar, como se mencionó en la sección anterior, esta función decide cual DTE utilizar a partir de la existencia de una preferencia en los criterios dentro de la instancia. Se utilizará DTE1 en el juego de diálogo si no existe una preferencia de criterios, caso contrario se usará DTE2.

A partir de lo anterior se generan los estados y acciones correspondientes a diagrama de transición de estados que se utilizará en el juego de diálogo, definiendo de igual manera su estado inicial y estados finales. Para la creación del diagrama correspondiente se hace uso una librería externa, llamada *Java FiniteStateMachine*, (<https://sourceforge.net/projects/javafsm/>). Esta librería también permite que se lleven a cabo todos los movimientos que son necesarios dentro del diagrama de transición de estados.

### **5.4 Juego de diálogo**

Ya teniendo definida la recomendación inicial, las reglas del sistema de diálogo y el diagrama de transición de estados a utilizar es posible iniciar el juego de diálogo entre el usuario y el sistema.

Esto se refleja dentro del cuadro de texto de diálogo en el prototipo, donde se generan un conjunto de párrafos, el primero menciona el estándar de prueba seleccionado para la recomendación generada; el segundo párrafo explica el funcionamiento de dicho estándar de prueba y cómo se seleccionó una cartera del conjunto.

Después de eso se menciona la cartera recomendada en otra línea seguida por un párrafo donde se le explica al usuario los posibles movimientos que puede realizar dentro del sistema, ya sea para continuar el juego de diálogo o checar el estado actual del mismo, todo esto se muestra en la Figura 5.5.

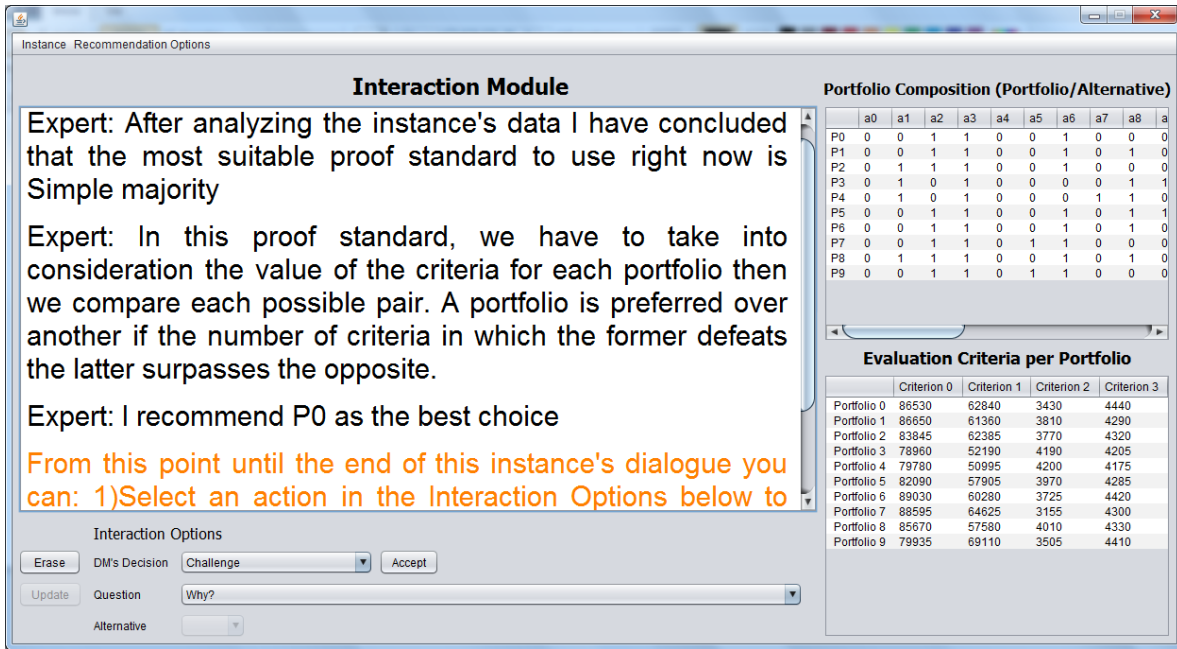


Figura 5.5: Generación de la recomendación inicial.

A partir de este punto el usuario puede interactuar en la sección llamada *Interaction Options*, ubicada en la parte inferior izquierda de la ventana, esta sección le permitirá al usuario dialogar con el sistema, así como llegar a un estado final de aceptación o rechazo de una recomendación, ya sea inicial o alguna generada durante el juego de diálogo. A continuación, se presenta en la Figura 5.6 la forma en la que está representada esta sección dentro del prototipo, así como en la Tabla 5.4 se muestra una explicación de cada uno de sus elementos.

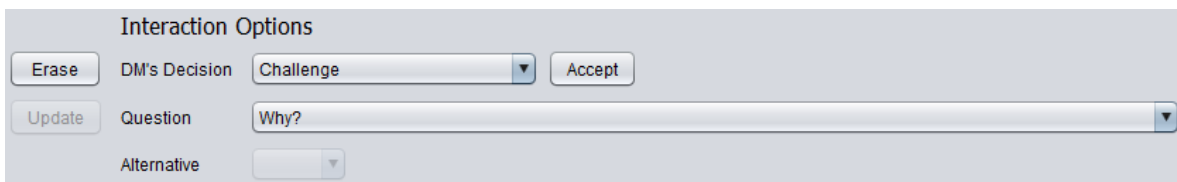


Figura 5.6: Opciones de interacción.

#### 5.4.1 Acciones del tomador de decisiones

Como se mencionó en la sección anterior, el elemento *DM's Decision* dentro de la sección *Interaction Options* le permite al usuario seleccionar el comando de acción a llevar a cabo en el estado actual del juego de diálogo, las acciones existentes en el prototipo se presentan en la Tabla 5.5.



**Tabla 5.4:** Elementos de *Interaction Options*.

<b>Elemento</b>	<b>Uso</b>
<i>DM's Decision</i>	Comando de la acción realizada por el DM.
<i>Question</i>	Declaración relacionada con la acción seleccionada por el DM.
<i>Alternative</i>	En caso de existir múltiples alternativas en la declaración esta sección se activará para seleccionar la opción deseada.
Botón <i>Accept</i>	Ejecuta la decisión del DM.
Botón <i>Erase</i>	Borra el texto del cuadro de texto del diálogo.
Botón <i>Update</i>	Actualiza los valores de la tabla de desempeño.

**Tabla 5.5:** Acciones disponibles en *DM's Decision*.

<b>Comando</b>	<b>Acción realizada</b>
<i>Accept</i>	Acepta la recomendación actual y finaliza el diálogo.
<i>Retract</i>	Rechaza la recomendación actual y finaliza el diálogo.
<i>Challenge</i>	Reta las decisiones del sistema.
<i>Argue</i>	Genera un argumento a favor o en contra de la recomendación / Toma una decisión
<i>Assert</i>	Sugiere una recomendación para que el sistema la analice.
<i>Pose Critical Question</i>	Genera una pregunta crítica / Permite modificar el estándar de prueba actual o sus propiedades / Permite definir una preferencia en un criterio.

Sin embargo, no todas las acciones existentes están disponibles en todo momento para el usuario. A partir del estado dentro del diagrama de transición de estados donde se encuentre el juego de diálogo en ese momento, solo se le permitirá elegir al usuario las acciones disponibles que tiene dicho estado. A continuación, se explicará a mayor detalle cada uno de los comandos de acción disponibles en el prototipo.

### **Comando de acción *Challenge***

Cuando el usuario selecciona la acción *Challenge* y la ejecuta, reta las decisiones tomadas por el sistema con respecto a la recomendación actual, esto aplica para ambos diagramas de transición utilizados por el sistema de recomendación.

El usuario tiene dos preguntas disponibles al momento de seleccionar este comando como la acción a realizar:

- “*Why?*”: Pregunta la razón por la cual fue elegida la recomendación actual sobre las demás
- “*Why is this alternative better than other solutions?*”: En esta pregunta, se activa el elemento *Alternative*, permitiéndole al usuario seleccionar cualquier otra cartera que no sea la recomendación. El objetivo de esta pregunta es cuestionar al sistema él porqué se seleccionó dicha recomendación sobre otra posible solución en específico.

Para ambas preguntas el sistema utiliza la función *userChallenge* con el objetivo de analizarlas y responderlas de manera adecuada. Para ello esta función realiza un conjunto de pasos para cumplir dicho objetivo.

- 1) Identifica la pregunta seleccionada por el usuario.
- 2) Revisa si existe una recomendación forzada, si se seleccionó una recomendación haciendo una consideración al presupuesto o de los valores en los criterios. Todas estas condiciones se explicarán más adelante dentro de este capítulo.
- 3) Revisa el estándar de prueba utilizado para generar dicha recomendación. Este paso no se realiza si la recomendación actual fue definida por forzamiento o consideración de presupuesto.
- 4) Selecciona la respuesta correspondiente a partir de la información obtenida en los pasos anteriores.

### **Comando de acción *Pose Critical Question***

Este comando le permite al usuario realizar una pregunta crítica con respecto a los esquemas de argumentación utilizados en ese momento durante el juego de diálogo, las cuales se le muestran en el elemento *Question*. Dichas preguntas son respondidas por el sistema, apoyándose de la función *systemAssert\_C*, la cual tiene un conjunto predefinido de respuestas para preguntas críticas. Esta función identifica la pregunta realizada y envía la respuesta correspondiente al usuario.

Sin embargo, esto no es lo único que le permite realizar, ya que si la cantidad de veces que haya sido actualizada la recomendación (ciclos) es menor o igual de cuatro, valor definido en el sistema como *cycle\_threshold*, se le muestran al usuario tres opciones más:

“*Is the current proof standard the best option?*”: Esta opción cuestiona el estándar de prueba actual utilizado por el sistema, para este caso el sistema responde que es el estándar de prueba que mejor se acomoda a las propiedades activas en ese momento.

“*Does the system have enough information to define a proof standard?*”: Esta opción cuestiona las propiedades de los estándares de prueba que se encuentran actualmente activas, en este caso el sistema menciona que la información es suficiente, pero que el usuario es libre de modificarlas.

“*Does the system have a preference for a certain criterion?*”: En esta opción el usuario pregunta si existe alguna preferencia para un cierto criterio. En caso de existir el sistema le responderá mencionando el orden de preferencia de criterios actual, en caso contrario se le menciona que no existe un orden jerárquico.

### **Comando de acción *Assert***

Dentro del sistema, el usuario es capaz de proponer otra cartera diferente a la recomendada por el sistema como una posible nueva recomendación, esto es posible mediante el comando de acción *Assert*. Este comando permite al usuario realizar dos acciones utilizando la función *userAssert*, a partir de la opción seleccionada en el elemento *Question*.

La opción “*But I prefer another solution over the recommended one*” le permite al usuario seleccionar una cartera diferente a la recomendada por el sistema, para que posteriormente cuando se llame el comando *Argue*, el cual se analizará después en esta misma sección, se haga una comparación entre la recomendación y dicha cartera.

Sin embargo, existe la posibilidad de que el usuario obligue al sistema a reconocer su selección como la recomendación establecida aún cuando las condiciones apoyen a la cartera recomendada previamente por el sistema. La forma de llevar a cabo esto es diferente para cada DTE.

En DTE1 es necesario afirmar repetidas veces una misma cartera, aún cuando el sistema haya rechazado previamente tanto la afirmación como los argumentos seleccionados por el usuario. El número de veces seguidas que se debe repetir la afirmación para esto es de tres, para superar un umbral de convencimiento definido en el sistema.

Sin embargo, en DTE2, al ofrecer un juego de diálogo más condescendiente, solo es necesario afirmar una vez la preferencia hacia una cartera en específico para que el sistema acepte el cambio de recomendación.

Con lo anterior, es fácil suponer que existe la posibilidad de que la nueva recomendación, la cual fue establecida por el usuario y no el sistema, no sea exactamente la mejor disponible. En consideración de esto, el sistema se encarga de recordarle al usuario varias veces durante el juego de diálogo que la recomendación actual fue forzada por él.

En caso de que el usuario desee regresar a la recomendación generada por el sistema, solo basta con seleccionar la opción “*I'd like to use the system's recommendation instead of mine*”, esto eliminará la recomendación actual, eligiendo aquella que el sistema considere la mejor en ese momento.

### **Comando de acción *Argue***

Los comandos anteriores le permiten al usuario la oportunidad de convencer al sistema de realizar un cambio, ya sea en el estándar de prueba utilizado o la cartera determinada como la recomendación, sin embargo, estos cambios son cuestionados por el sistema por lo que se requiere que sean apoyados por un argumento

El comando *Argue* le permite al usuario seleccionar un argumento para defender su afirmación. Cuando el usuario elige una de las tres preguntas explicadas en el comando de acción *Pose Critical Question*, el sistema manda a llamar a la función *userArgue* para analizar si se realizó algún cambio, aplicando estos para que al final del ciclo se llamen a las funciones correspondientes a la selección previamente realizada. Los pasos que realiza esta función son los siguientes:

- 1) Identifica la opción seleccionada por el usuario en el elemento *Question*.
- 2) Selecciona la acción correspondiente a dicha opción.

- 3) Genera un reporte del cambio realizado, el cual se le mostrará al usuario como parte del diálogo.

La acción realizada por sistema en esta función depende de la pregunta seleccionada en el comando de acción *Pose Critical Question*.

“*Is the current proof standard the best option?*”: El sistema le ofrece al usuario la posibilidad de cambiar el estándar de prueba utilizado para la generación de una recomendación. Forzando al sistema a evaluar nuevamente las carteras disponibles, llamando a la función correspondiente al estándar de prueba seleccionado, generando una recomendación nueva que puede o no ser la misma. Esta opción se ofrece con objetivo de acelerar el proceso de apoyo a la decisión para usuarios con un conocimiento previo de los estándares de prueba utilizados en este sistema.

“*Does the system have enough information to define a proof standard?*”: El sistema es capaz cambiar el estándar de prueba a medida que cambien las propiedades de los mismos debido a la interacción. Sin embargo, también se le da la posibilidad al usuario de realizar un argumento positivo o negativo enfocado a la información actual que tiene el sistema sobre las propiedades de los estándares de prueba si lo desea. En caso de hacerlo, dicho argumento puede llevar a la alteración del estado de alguna de estas propiedades, por lo que el sistema deberá reevaluar el estado actual de cada una ellas y cambiar el estándar de prueba utilizado en caso de ser necesario, lo cual a su vez lleva a un posible cambio en la nueva recomendación.

“*Does the system have a preference for a certain criterion?*”: El usuario tiene la posibilidad de establecer una preferencia en un criterio. Si el usuario define un nuevo orden jerárquico automáticamente el sistema utiliza el orden lexicográfico como el estándar de prueba a utilizar y evalúa nuevamente las carteras disponibles usando la función *lexicographic*, para definir una nueva recomendación.

Este comando también utiliza la función *userArgueAssert* cuando el usuario propone como solución una cartera diferente a la recomendación del prototipo, después de que el usuario haya llamado al comando de acción *Assert*. El sistema evalúa si el argumento es lo suficientemente fuerte para sostener dicha afirmación.

La respuesta del sistema ante la afirmación varía dependiendo del DTE en uso, siendo capaz de rechazarla en DTE1 si el argumento no puede sostenerla, mientras que en DTE2 siempre se acepta la afirmación, sin embargo, si el argumento es demasiado débil, en otras palabras, el sistema considera que existe una mejor alternativa que la seleccionada por el usuario, se le hace saber a éste para que lo tenga en consideración, los argumentos que se pueden seleccionar se explican a continuación.

La opción *“I like my choice more than the recommendation”* permite al sistema considerar la posibilidad de cambiar la recomendación actual, sin embargo, la única manera de convencerlo con esta opción es mediante el forzado de la recomendación.

Por otro lado la opción *“I have a preference for a certain criterion”* le permite al usuario seleccionar un criterio para establecer un orden jerárquico donde dicho criterio es el de mayor importancia, de ahí el sistema hace una comparación entre la recomendación actual y la seleccionada por el usuario, mencionándole al usuario si dicho argumento es verdadero o falso y generando una nueva recomendación, apoyándose del nuevo orden jerárquico establecido, utilizando como estándar de prueba el método lexicográfico.

En caso de existir de manera previa un orden jerárquico, también se le presenta al usuario la opción *“The discrepancy between the preferred criteria values on the recommended portfolio and another solution isn't significant”* el cual compara los valores del criterio de mayor orden entre la recomendación y la selección, cambiando la recomendación a la seleccionada por el usuario si la diferencia entre ambas es menor del 10% en base del valor del criterio de la recomendación actual.

Así como el usuario es capaz de comparar valores de los criterios, también es capaz de comparar presupuestos, al seleccionar la opción *“The budget for this portfolio is too close to the maximum limit, I would like to choose another one”* el sistema realiza dos comparaciones, en la primera evalúa si el presupuesto de la recomendación tiene al menos una separación de su 10% con respecto al tope presupuestal, en caso de no ser cierto, el argumento se considera falso, en caso contrario se evalúa si el presupuesto de la selección es al menos 10% de su valor menor que el de la recomendación, si esto es cierto entonces el sistema considera el argumento verdadero y realiza el cambio.

### 5.4.2 Aplicación de los esquemas de argumentación

Dependiendo del comando de acción seleccionado por el usuario, así como del estado del DTE utilizado en donde se encuentre en ese momento el juego de diálogo, se utiliza un esquema de argumentación en caso de ser necesario, tal como se vio en la sección 4.1.2.2 del capítulo anterior.

A continuación se explica la aplicación de cada uno de los esquemas de argumentación dentro del prototipo, así como el momento en el que se utiliza para cada uno de los diagramas de transición de estados utilizados:

#### Razonamiento abductivo

La aplicación de este esquema de argumentación tiene como objetivo definir el estándar de prueba a utilizar en el sistema para generar una recomendación, debido a esto es necesario aplicar el razonamiento abductivo en cualquier momento que se vea afectada la condición del estándar de prueba en uso o el estado de alguna de las propiedades de los estándares de prueba. Dichos momentos dentro del juego de diálogo son los siguientes:

- Después de que el sistema llevó a cabo la lectura de una instancia, previo a la generación de la recomendación inicial.
- Al reiniciar el juego de diálogo mediante la selección del submenú *Start Dialogue*.
- Al momento que el usuario selecciona una alternativa del comando de acción *Argue* después de haber realizado una de las tres preguntas mencionadas en la sección anterior en el comando de acción *Pose Critical Question*.
- Al seleccionar la opción “*I have a preference for a certain criterion*” en el comando *Argue*, después de que el usuario haya afirmado preferencia sobre alguna otra cartera diferente a la recomendada.

Este esquema utiliza dos funciones para apoyarse, en caso de que se haya modificado de manera directa el estándar de prueba se utiliza la función *changePS*, por otro lado si se modificó alguna de las propiedades de los estándares de prueba entonces se aplica

la función *defineProperties* para redefinir las propiedades y *analysis* junto con *proofStandardSelection* para seleccionar el nuevo estándar de prueba.

### **Argumento desde una posición para saber**

Este esquema de argumentación se utiliza para la generación de recomendaciones del sistema mientras la cantidad de ciclos ocurridos en el juego de diálogo sea menor que el umbral de ciclo (definido en el sistema como *cycle\_threshold*), de igual manera se utiliza para defender la recomendación generada cuando el usuario pregunte la razón por la cual se seleccionó dicha cartera como solución.

Los momentos dentro del juego de diálogo donde se aplica este esquema son los siguientes:

- Después de que el sistema llevo a cabo la lectura de una instancia y haberse definido un estándar de prueba a utilizar.
- Al reiniciar el juego de diálogo mediante la selección del submenú *Start Dialogue*.
- Al definir una nueva recomendación, siempre y cuando el número de ciclos sea menor que el umbral de ciclo.
- Cuando el usuario selecciona la pregunta “*Why?*” en el comando *Challenge*.
- Al momento que el usuario selecciona las tres preguntas mencionadas en la sección anterior en el comando de acción *Pose Critical Question*.

Esta función se apoya de varias funciones, primero las funciones correspondientes a los estándares de prueba utilizados, ya que estas proveen una recomendación. De igual manera cuando se selecciona una de las tres preguntas mencionadas en el último punto se apoya de *systemAssert\_C* para responder y ofrecer una posible modificación. Finalmente, se apoya en la función *userChallenge* cuando el usuario selecciona la pregunta “*Why?*” para responder a esta.

### **Argumento desde una opinión experta**

Este esquema de argumentación tiene el mismo funcionamiento que el argumento desde una posición para saber, con la diferencia de que este esquema solo se utiliza cuando



el total de ciclos que lleva el juego de diálogo actual supera al umbral de ciclo. Al tener el mismo objetivo que el esquema anterior, se utiliza en los mismos momentos y se apoya de las mismas funciones. Con la excepción de que cuando este esquema de argumentación esta en uso, las preguntas mencionadas cuando se explicó el comando de acción *Pose Critical Question* no aparecen por lo que la función *systemAssert\_C* no apoya a este esquema.

### **Evaluación multicriterio de pareja**

La aplicación de este esquema de argumentación se centra en la comparación de los diferentes valores de dos carteras, la que el sistema tiene definida como la recomendación y una seleccionada por el usuario. Esta situación se presenta varias veces durante el juego de diálogo, dichos momentos son:

- Cuando el usuario selecciona la pregunta “*Why is this alternative better than other solutions?*” en el comando *Challenge*, ya que esta opción le permite al usuario seleccionar una alternativa.
- Cuando el usuario selecciona una pregunta del comando *Argue*, después de que el usuario haya afirmado preferencia sobre alguna otra cartera diferente a la recomendada.

Las funciones que apoyan a este esquema son *userChallenge* para el primer punto al permitir comparar dos carteras apoyándose del estándar de prueba seleccionado y *userArgueAssert* para el segundo, el cual permite comparar valores de un cierto criterio, así como el valor de los presupuestos de la cartera recomendada por el sistema y la seleccionada por el usuario.

### **Causa-Efecto**

Debido a que durante el juego de diálogo es bastante probable que dentro del sistema se vayan cambiando los estándares de prueba en uso, sus propiedades activas e inactivas, además de que se genere, elimine o modifique un orden jerárquico para los criterios también es probable que la recomendación establecida por el sistema haya cambiando.

Por esta razón es necesario utilizar el esquema de argumentación causa-efecto, el cual se enfoca en los cambios ocurridos en diversos elementos dentro del sistema mientras se lleva a cabo el juego de diálogo. Los momentos en los que se utiliza este esquema son los siguientes:

- Al momento que el usuario selecciona una alternativa del comando *Argue* después de haber realizado una de las tres preguntas mencionadas en la sección anterior en el comando de acción *Pose Critical Question*.
- Cuando el usuario fuerza al sistema a aceptar una cartera como la nueva recomendación utilizando el comando *Assert*.
- Al seleccionar la opción “*I have a preference for a certain criterion*” y eligiendo un criterio como el de mayor preferencia en el comando *Argue*, después de que el usuario haya sugerido otra cartera como la recomendación utilizando el comando de acción *Assert*.

Las funciones que apoyan a este esquema de argumentación son todas aquellas que se encuentran involucradas en algún cambio en los estándares de prueba o en sus propiedades, las cuales son *userArgue*, *userArgueAssert*, *changePS*, *defineProperties* y *proofStandardSelection*, cuya aplicación se mencionó anteriormente en este capítulo.

### **Desde la parcialidad**

Este esquema de argumentación se utiliza cuando está definido un orden de jerarquía en los criterios, esto ocurre en DTE2 desde la lectura de la instancia y en DTE1 solo si se establece un orden de manera explícita durante el juego de diálogo. Existen diversos momentos en los que se aplica este esquema los cuales son:

- Cuando el sistema evalúa las carteras de la instancia utilizando la función *lexicographic*.
- Cuando el usuario selecciona una pregunta utilizando el comando *Challenge* si es que existe un orden de preferencia en los criterios.
- Al seleccionar la opción “*I have a preference for a certain criterion*” en el comando *Argue*, después de que el usuario haya afirmado preferencia sobre alguna otra cartera diferente a la recomendada.

Este esquema está apoyado por varias funciones, primero la función *lexicographic* permite generar una recomendación apoyándose de un orden de preferencia de criterios. Por otro lado, las funciones *userChallenge* y *userArgueAssert* le permiten al sistema utilizar dicho orden como respuesta a la pregunta que el usuario hace sobre la recomendación del sistema, así como para comparar esta cartera con otra seleccionada por el usuario en base al valor de un criterio que el usuario considera como el de mayor importancia.

### **5.4.3 Generación y respuesta de las preguntas críticas**

Uno de los elementos de los esquemas de argumentación son las preguntas críticas, las cuales deben ser respondidas para mantener la veracidad de dichos esquemas. Dentro del sistema, las preguntas pueden ser seleccionadas por el usuario eligiendo el comando de acción *Pose Critical Question*.

Cada esquema de argumentación tiene un conjunto predeterminado de preguntas críticas, las cuales se generan dentro del sistema cuando su correspondiente esquema se encuentra en uso. Debido a esta razón las preguntas que se le permiten seleccionar al usuario pueden variar dependiendo del estado en el que se encuentre el juego de diálogo en ese momento, ya que como se mencionó en la sección anterior existen momentos en los que se activan y utilizan ciertos esquemas de argumentación.

Al momento que el usuario hace la selección de una pregunta, el sistema se apoya de la función *userPose\_C* para identificar la pregunta elegida y responderla de manera adecuada. Sin embargo, debido a que ciertas preguntas del comando *Pose Critical Question* permiten hacer modificaciones en varios elementos involucrados en el juego de diálogo, tales como el estándar de prueba y el orden de preferencia de los criterios, es necesario separar estas preguntas de aquellas que solo necesitan una respuesta que no modifica las propiedades del sistema.

Por esta razón además de apoyarse de la función previamente mencionada, las preguntas que involucran modificación utilizan *systemChallenge\_C* para dirigir el juego de diálogo hacia el cambio correspondiente. Por otro lado, las preguntas que pueden ser respondidas por el sistema sin la necesidad de hacer un cambio se apoyan tanto de

*userPose\_C* como de la función *systemAssert\_C*, la cual busca la respuesta adecuada para la pregunta crítica seleccionada por el usuario.

#### 5.4.4 Actualización del estándar de prueba utilizado

Como se mencionó anteriormente en este capítulo, durante el juego de diálogo se le permite al usuario modificar ciertos elementos dentro del sistema, que llevan a posibles cambios en la recomendación establecida por este, uno de esos cambios es la modificación del estándar de prueba utilizado para definir la cartera que el prototipo considera como la mejor en ese momento.

También se hizo mención de los diferentes puntos dentro del juego de diálogo donde el usuario puede cambiar el estándar de prueba a utilizar para la generación de la recomendación por parte del sistema, ya sea eligiendo directamente el estándar a utilizar o modificando cuales de sus propiedades se encuentran activas, dichos momentos son:

- Cuando se elige una alternativa del comando de acción *Argue* después de haber seleccionado una de las tres preguntas generadas en *Pose Critical Question* cuando el esquema de argumentación *argumento desde una posición para saber* se encuentra activo.
- Cuando se establece una preferencia hacia algún criterio en específico al seleccionar la alternativa “*I have a preference for a certain criterion*” dentro del comando *Argue* previamente habiendo sugerido el usuario como solución una cartera diferente a la recomendada por el sistema utilizando el comando *Assert*.

El proceso de actualización del estándar de prueba en uso para la definición de una recomendación por parte del sistema se muestra en los siguientes puntos, los cuales están representados en la Figura 5.7:

- Llevar a cabo uno de los puntos anteriores para iniciar el proceso de actualización.
- En caso de haber realizado el primer punto, la función *userArgue* se encarga de hacer las modificaciones establecidas por el usuario. Si se realizó el segundo punto entonces la función que cubre esto es *userArgueAssert*.

- Si se llevó a cabo un cambio directo de estándar de prueba o se definió un orden jerárquico en los criterios entonces se llama a la función *psChange* para actualizar las propiedades a fin que de solo estén activas las correspondientes al estándar de prueba utilizado. Por otro lado si se activó o desactivó una propiedad de los estándares de prueba entonces se llaman a la funciones *defineProperties* para redefinir las propiedades acoplándose al cambio y *analysis* para analizar cuáles de dichas propiedades se encuentran activas.
- Finalmente se llama a la función *proofStandardSelection* para seleccionar el nuevo estándar de prueba a utilizar.

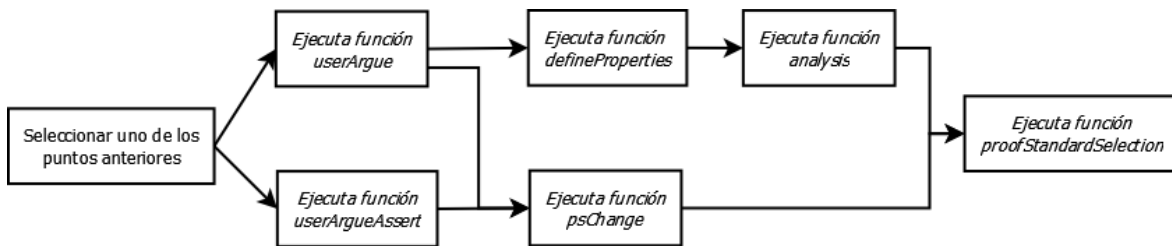


Figura 5.7: Actualización del estándar de prueba.

#### 5.4.5 Cambio de recomendación

Cuando el sistema enfrenta modificaciones en diversas propiedades involucradas en el juego de diálogo, es probable que tenga la necesidad de cambiar la recomendación establecida. Existen tres situaciones en las cuales el sistema puede cambiar dicha recomendación y seleccionar una nueva del conjunto de carteras dentro de la instancia:

*Modificación del estándar de prueba en uso:* si el estándar de prueba utilizado es cambiado aplicando el proceso visto en la sección anterior, entonces es necesario volver a evaluar las carteras existentes en la instancia bajo las reglas establecidas por el estándar de prueba seleccionado. Para llevar a cabo esto el sistema llama a la función correspondiente al nuevo estándar de prueba, los cuales se muestran en la Tabla 5.3, después de eso se le avisa al usuario mediante el cuadro de diálogo el nuevo estándar de prueba en uso así como la nueva recomendación generada.

*Forzado de la recomendación:* como se mencionó en la sección 5.4.1 al explicar el comando de acción *Assert*, es posible obligar al sistema a aceptar una cartera sugerida por

el usuario como la recomendación, de esta manera el usuario puede manipular el juego de diálogo para que se adapte a sus preferencias. Sin embargo, el sistema se encarga de recordarle en varias ocasiones durante el diálogo al usuario el hecho de que su elección puede no ser la más adecuada con la información que se tiene en ese momento.

Además, el sistema mantiene su propia recomendación almacenada dentro de otra variable durante el juego de diálogo, para que en caso de que el usuario se retracte de su elección y desee utilizar nuevamente la recomendación del sistema pueda hacerlo, la manera de llevar a cabo esto se mencionó de igual manera en la sección 5.4.1.

Cuando el sistema lleva a cabo un cambio de recomendación por cualquiera de estos dos métodos, se incrementa el número de ciclos ocurridos en el juego de diálogo, lo cual afecta el esquema de argumentación en uso (desde una posición para saber/desde una opinión experta). Debido a esto el usuario solo tiene un número limitado de modificaciones por instancia, esto está diseñado para evitar que el diálogo entre en un ciclo infinito.

*Modificación de los valores de los criterios en las alternativas:* durante cualquier etapa del juego de diálogo, el usuario es capaz de actualizar los valores de los criterios en las alternativas en caso de ser necesario, se puede acceder a esta opción de dos maneras:

- Seleccionando el submenú *Update Instance's Criteria* dentro del menú *Recommendation Options*.
- Seleccionando el comando *Update* el cual se encuentra en *DM's Decisión*.

Después de seleccionar cualquiera de esas dos opciones se abrirá una nueva ventana, la cual muestra la tabla de desempeño (alternativa/criterio), como se ve en la Figura 5.8. Los valores de dicha tabla son obtenidos mediante la función *update*. El usuario es libre de modificar dicha tabla y grabar los cambios, cuando esto ocurra se ejecutará la función *modifyUpdate*, la cual identificará los cambios realizados y grabará la nueva tabla de desempeño, calculando los nuevos valores, después de esto la ventana se cerrará.

Si el usuario está satisfecho con los cambios realizados, debe presionar el botón *Update* que aparece en la sección *Interaction Options* en la ventana principal, al hacer esto el sistema analizará los nuevos valores para después establecer una nueva recomendación la

cual se le mostrará al usuario. Además, debido a que los criterios de evaluación por cartera fueron modificados, se reinicia el número de ciclos a cero ya que debe recopilarse nueva información para el sistema se considere un experto, véase el Anexo C para ver un ejemplo de lo anteriormente mencionado.

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Alternative 0	3200	2000	165	300
Alternative 1	6255	1640	385	390
Alternative 2	5680	6940	270	445
Alternative 3	8965	4195	355	415
Alternative 4	6550	6560	315	440
Alternative 5	6740	6290	150	350
Alternative 6	9055	7165	375	485
Alternative 7	4170	3015	410	285
Alternative 8	9735	2860	480	330
Alternative 9	3350	4210	400	315
Alternative 10	8595	7270	150	265
Alternative 11	9070	2430	455	360
Alternative 12	9930	5825	420	385
Alternative 13	4675	4505	425	490
Alternative 14	8065	4030	165	425
Alternative 15	7910	7665	240	320
Alternative 16	9860	6265	415	350
Alternative 17	3175	6240	225	445
Alternative 18	9660	655	320	475
Alternative 19	1150	4500	415	400
Alternative 20	3245	5950	105	275

**Figura 5.8:** Actualización de la tabla de desempeño.

## 5.5 Visualización de apoyo a la decisión

Con lo anterior mencionado, el usuario es capaz de llevar a cabo un juego de diálogo dentro del prototipo de sistema de recomendación desarrollado en este trabajo (véase el Anexo D para un ejemplo). Sin embargo, además de lo anterior, dentro de la interfaz gráfica que se le provee al usuario para interactuar con el sistema, se implementaron diversas opciones que tienen el objetivo de apoyar al usuario durante el uso del prototipo, buscando facilitar su experiencia con este y reducir el tiempo que le pueda llevar al usuario acoplarse a su funcionamiento. Dichas opciones son las siguientes:

- Visualización de las características de las carteras.
- Visualización de las condiciones actuales del juego de diálogo.
- Visualización de una ventana de ayuda que explica diversos elementos del sistema.

### 5.5.1 Características de las carteras

Esta opción le permite al usuario visualizar diferentes propiedades de las carteras existentes en la instancia a fin de apoyarse en esos datos para poder ser capaz de tomar decisiones de mayor calidad al tener mayor información de las soluciones en comparación. Las propiedades que se toman en consideración para esta visualización son:

- La composición de las carteras (cromosoma)
- El valor de los criterios de evaluación de cada cartera
- El presupuesto necesitado para llevar a cabo cada cartera

Para llevar a cabo esta opción, el usuario debe seleccionar el submenú *Show Portfolios* que se encuentra dentro del menú *Recommendation Options*, al hacer esto el sistema lleva a cabo la función *show\_solutions*, la cual compila la información de las carteras de la instancia y las muestra en una nueva ventana, tal como se ve en la Figura 5.9. Debe hacerse mención que el usuario solo es capaz de visualizar estas propiedades y no de modificarlas, si desea modificar algún valor de los criterios dentro de la instancia debe hacerlo como hace mención la sección 5.4.5.

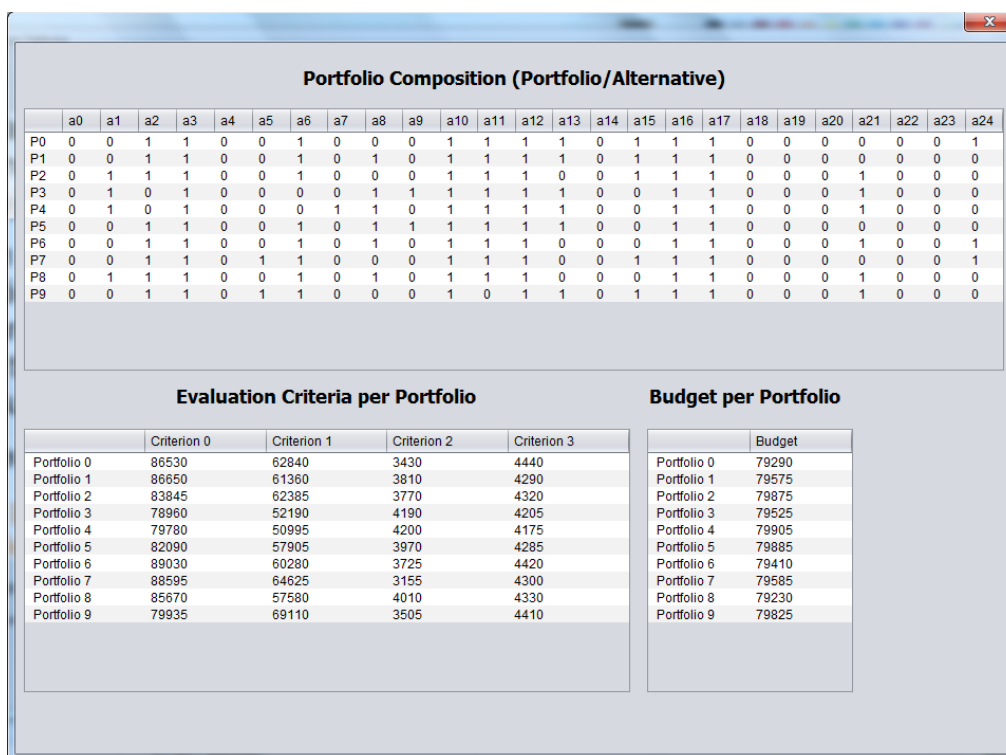


Figura 5.9: Visualización de carteras.



### 5.5.2 Condiciones actuales del juego de diálogo

Para facilitar el entendimiento del flujo del juego de diálogo se le ofrece al usuario la opción de ver el estado actual de diversas propiedades involucradas en este, esto con el objetivo de permitirle al usuario comprender las razones por las cuales el sistema toma ciertas decisiones sobre diversos elementos dentro del juego de diálogo durante el proceso, tales como propiedades relacionadas, estándares de prueba o la recomendación misma. Las propiedades que se toman en consideración en esta visualización son:

- El tipo de problema (maximización/minimización).
- La cantidad de ciclos ocurridos y si el estándar de prueba es modificable todavía.
- El orden de preferencia actual.
- El estándar de prueba en uso.
- La recomendación establecida por el sistema.
- La recomendación forzada por el usuario (en caso de existir).
- El estado actual de cada una de las propiedades de los estándares de prueba.

Esta opción se lleva a cabo cuando el usuario selecciona el submenú *Show Conversation Data* el cual se encuentra dentro del menú *Recommendation Options*, al hacer esto el sistema llama a la función *showData*, la cual recupera toda la información mencionada en los puntos anteriores y se la muestra al usuario en una nueva ventana, tal como se puede ver en la Figura 5.10.

### 5.5.3 Ayuda temática

Finalmente, la última opción de apoyo que se le ofrece al usuario es una ventana la cual se le muestran varios elementos de ayuda con el objetivo de facilitarle la comprensión de diversos conceptos o acciones que puede realizar dentro del juego de diálogo. Dicha ventana, mostrada en la Figura 5.11, muestra los siguientes conceptos al usuario:

- Un pequeño glosario con conceptos relacionados al problema de carteras de proyectos.
- Una corta explicación de cada uno de los comandos de acción en *DM's Decision*.
- Un listado de los estándares de prueba contenidos en el sistema, así como una explicación de su funcionamiento.

- Una explicación del contenido de los menús del sistema.

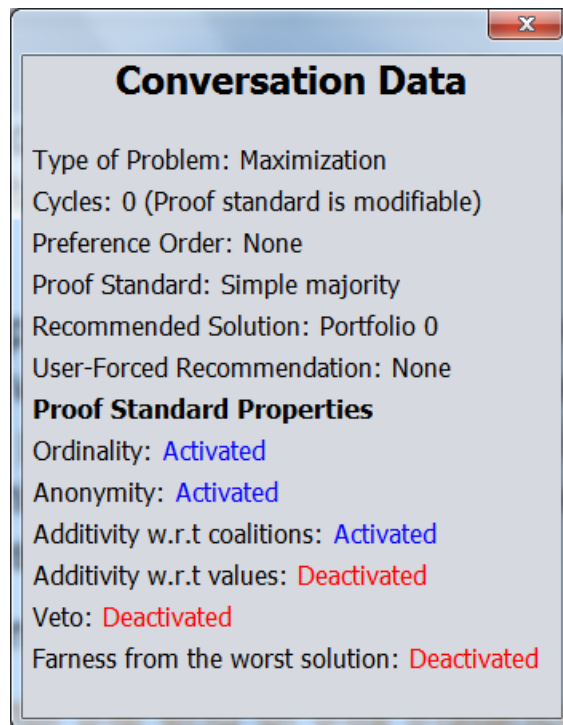


Figura 5.10: Visualización de los datos de la conversación.

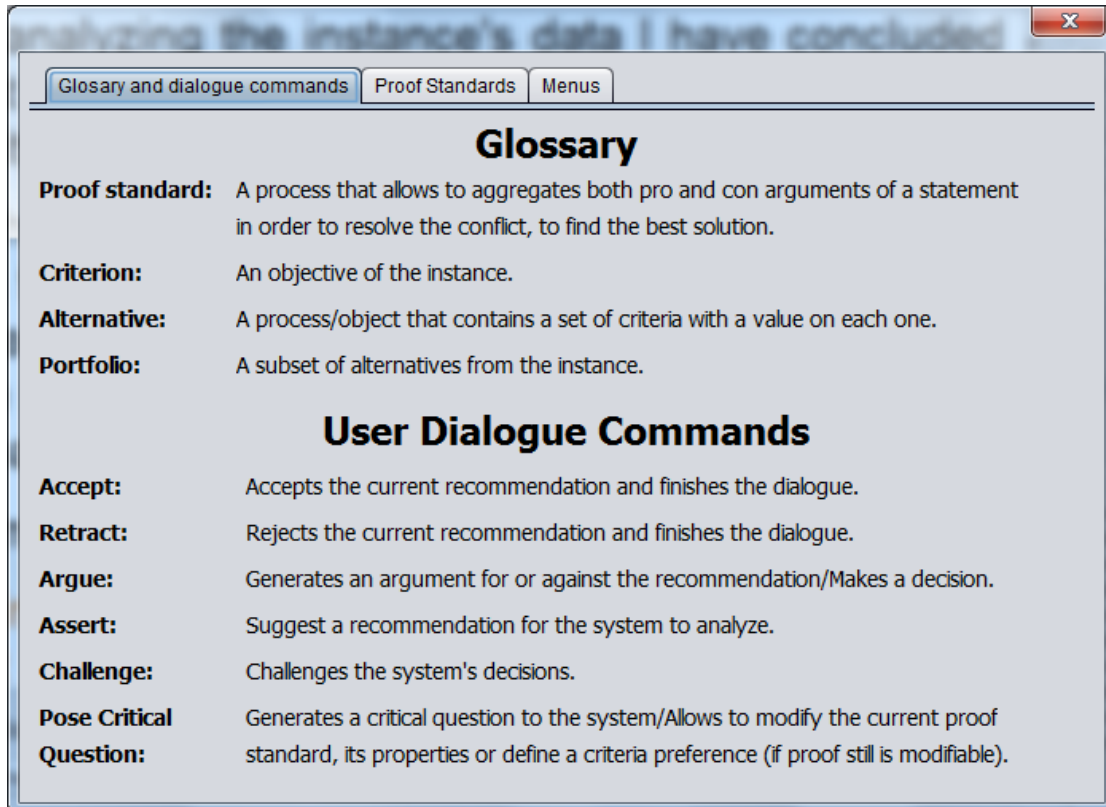


Figura 5.11: Visualización de la ventana de ayuda.

# Capítulo 6

## Experimentación y resultados

---

---

Teniendo en consideración que el trabajo realizado en este proyecto era un prototipo, fue necesario llevar a cabo un conjunto de pruebas las cuales tuvieron como objetivo medir la usabilidad del sistema desarrollado. Sin embargo, hacer dichas evaluaciones no fue tarea fácil, inicialmente debido a que los trabajos revisados dentro del estado del arte no presentan una evaluación de usabilidad para calificar su sistema propuesto, por lo que realizar una comparación directa no era posible. Además, había una amplia variedad de elementos que requerían ser calificados, mientras que por otro lado fue necesario tener en cuenta que dichas evaluaciones debían ser simples y cortas para facilitar el trabajo de los evaluadores. En este capítulo se explicará cómo se generaron las evaluaciones necesarias y como llevaron a cabo.

### **6.1 Análisis de métodos de evaluación usados en literatura relacionada**

Previo a llevar a cabo el proceso de evaluación del prototipo, se tomaron en cuenta dos aspectos de suma importancia los cuales buscan que dicha evaluación no solo cubra todos los elementos que se desean calificar, sino que también la manera en la que se revisan estos por parte de los evaluadores sea lo más fácil y simple de entender posible.

El primer aspecto a considerar es la forma en la que se debe llevar a cabo la evaluación, para esto es necesario tener claramente definidos ciertos elementos dentro de este proceso, los cuales son:

- El número de evaluaciones realizadas
- La cantidad de personas involucradas en cada muestra
- La duración de la evaluación
- El tipo de medición utilizado por los evaluadores

Debido a que la definición de esos cuatro puntos anteriores es una tarea complicada, se realizó una investigación en la cual se analizaron diversos trabajos los cuales tienen una cierta relación con este proyecto. Además de hacer consideración de los puntos anteriores, se revisaron los siguientes elementos:

- Si el trabajo está enfocado en la optimización
- El objetivo del estudio
- Si existen casos de prueba públicos

Los resultados de esta investigación se muestran representados en la Tabla 6.1, como se puede ver varios de estos trabajos no están completamente relacionados al tema, ya que algunos de ellos están más enfocados en recomendaciones de páginas web, esto debido a que encontrar evaluaciones para sistemas de recomendación no es tan común. Lo anterior se debe a que llevar a cabo experimentos controlados en una muestra aleatoria utilizando un sistema de recomendación resulta ser costoso tanto en recursos como en tiempo [Adomavicius y Tuzhilin, 2005].

Después de haber analizado y definido la forma en la que se llevará a cabo el proceso de evaluación, es necesario saber cuáles son los elementos que serán revisados por los evaluadores, en otras palabras cuáles serán las preguntas que se les aplicarán a estos al finalizar su evaluación para calificar al sistema.

Al igual que el aspecto anterior, la definición de las preguntas a realizar a los evaluadores no es un proceso fácil, ya que como se mencionó antes dentro de esta sección, dichas preguntas no solo deben cubrir todos los elementos que se desean evaluar, sino que también deben hacerse de tal manera que los evaluadores no sientan una sobresaturación, lo cual no solo alentaría el proceso, sino que reduciría la calidad de la misma evaluación.

Para este caso se analizaron tres diferentes cuestionarios, los cuales están enfocados a la evaluación de la usabilidad en los sistemas de recomendación:

- Cuestionario post-estudio de usabilidad del sistema (*PSSUQ*)
- Escala de la usabilidad del sistema (*SUS*)
- Calidad de la experiencia del usuario en sistema de recomendación (*ResQue*)

Trabajo	Optimización	Objetivo de estudio	Medición	¿Casos de estudio?	Tamaño muestra	Número de experimentos	Duración
[Karacapilidis y Papadias, 2001]	No específica	Toma de decisión colaborativa	Satisfacción del usuario (Fácil de usar, de uso agradable, estimula y organiza discusión, tiempo de adaptación al programa, comodidad de uso y deseo de reutilización)	No específica	3 profesores de medicina 46 estudiantes de medicina 12 doctores 16 estudiantes de preparatoria Estudiantes de ingeniería mecánica <b>9 personas en promedio por prueba en primera etapa</b> <b>3-6 personas en promedio por prueba en segunda etapa</b>	Aproximadamente 9 en la primera etapa 12 en segunda etapa	2 semanas a 5 meses
[Walker et al. 2004]	Si	Toma de decisión y comprobación de hipótesis	Cantidad de recomendaciones aceptadas y descartadas, satisfacción del usuario.	Solo imágenes	16 personas para adaptación y modo 21 personas para concisión	4 por persona (2 en adaptación/modo 2 en concisión)	Hora y media por persona
[Balabanovic et al. 1996]	No específica	Recomendación de páginas web	Sistema de puntaje	No específica	1 a 6 personas por día	1 por persona	24 días, 20 minutos por sujeto
[Asnicar y Tasso, 1997]	No específica	Recomendación de páginas web	Evaluación de la calidad de varios aspectos del sistema	No específica	4 personas	9 sesiones por persona	No específica
[Adomavicius y Kwon, 2007]	Si	Recomendación (sin argumentación)	Porcentaje de precisión de la recomendación	Solo mención	155 personas	14 revisiones en promedio por persona	No específica

**Tabla 6.1:** Resultados de la investigación de evaluaciones previas.

*Cuestionario post-estudio de usabilidad del sistema (PSSUQ)*: creado por Lewis [Lewis, 1995] busca evaluar la satisfacción del usuario sobre la usabilidad del sistema, dándole la oportunidad de proveer una evaluación general del sistema utilizado. Este cuestionario está compuesto de 19 preguntas bajo una escala de 1 a 7 para su evaluación. Estas preguntas se enfocan en diversos elementos dentro del mismo, los cuales son:

- Facilidad de uso
- Facilidad de aprendizaje
- Simplicidad
- Eficacia
- Información
- Interfaz de usuario

*Escala de la usabilidad del sistema (SUS)*: Brooke [Brooke, 1996] desarrolló un método de evaluación de usabilidad cuya ventaja es su tamaño, ya que al ser solamente de diez preguntas, el costo en esfuerzo y tiempo para aplicarse es muy poco. Dichas preguntas son evaluadas utilizando un rango de 1 a 5 donde 1 es total desacuerdo y 5 es total aprobación. A pesar de su tamaño, este cuestionario cubre tres aspectos esenciales respecto a la evaluación de usabilidad, los cuales son la eficacia, eficiencia y satisfacción.

*Calidad de la experiencia del usuario en sistema de recomendación (ResQue)*: A diferencia del cuestionario anterior, este utiliza una gran cantidad de preguntas tomando en cuenta una variada cantidad de aspectos relacionados con la satisfacción del usuario al momento de interactuar con el sistema [Pu et al., 2011]. Este modelo estructura la evaluación en cuatro componentes principales:

- Cualidades percibidas por el usuario
- Creencias del usuario
- Actitud subjetiva del usuario
- Intención conductual

Estos componentes se dividen en un total de 13 diferentes secciones para formar un total de 60 preguntas que conforman el cuestionario, sin embargo, debido a la posible

sobresaturación que puede causar esto, existe una versión reducida la cual está compuesta de solamente 15 preguntas del listado inicial, las cuales son las más relevantes. Igual que el modelo anterior, este cuestionario utiliza un rango de valores de 1 a 5 para calificar cada una de sus preguntas.

## **6.2 Método de evaluación propuesto para recomendación de carteras**

Teniendo en base tanto la investigación realizada en la sección anterior, así como la consideración de que este trabajo es un prototipo, se definió la forma en la que se llevó a cabo el proceso de evaluación para del sistema de recomendación de este trabajo. Estableciéndose como objetivos el obtener retroalimentación por usuarios externos al programa para llevar a cabo las modificaciones necesarias, así como medir los niveles de calidad, satisfacción y comodidad de los usuarios al utilizarlo.

Para esto, se decidió trabajar el proceso de evaluación del prototipo en dos diferentes etapas, con el objetivo de calificar el desempeño y usabilidad del sistema tanto durante la etapa de desarrollo del mismo buscando obtener una retroalimentación respecto al funcionamiento del sistema, así como después de haber finalizado la versión final del prototipo en este trabajo para evaluar la calidad de uso del mismo.

### **6.2.1 Primera etapa de evaluación y resultados**

En esta etapa se llevó a cabo la evaluación del sistema de recomendación utilizando como evaluadores a un grupo de doctores en ciencias de la computación, así como estudiantes de posgrado tanto de maestría como de doctorado en ciencias de la computación.

El número de personas involucradas en esta etapa del proceso de evaluación fue de cinco y la duración de la evaluación fue de 30 minutos por persona. Debido a que las personas en esta etapa tenían conocimiento previo del campo, se les permitió trabajar libremente con el prototipo.

Esta actividad se realizó durante el proceso de desarrollo del prototipo del sistema de recomendación, debido a esta razón, para etapa de evaluación se hizo principalmente consideración de dos aspectos:

- *Funcionalidad general del sistema*: Los usuarios probaron el sistema con el objetivo de analizar y evaluar la funcionalidad de los diversos elementos del prototipo, así como la efectividad de las respuestas del sistema ante las acciones realizadas por el usuario. En este caso, cualquier duda relacionada con la funcionalidad hecha por los expertos se buscaba responder y transformar en una corrección a realizar en el prototipo en caso de ser necesario, con el objetivo de mejorar la calidad de este.
- *Búsqueda de errores y recomendaciones*: Los usuarios buscaron errores fatales, de diseño y de lógica en el sistema. De igual forma los usuarios formularon diversas recomendaciones para la mejora de la calidad del sistema, buscando que este sea más claro y comprensible con personas con menor experiencia en el campo.

Se obtuvieron los siguientes resultados a partir de esta evaluación:

- Era necesario generar una explicación de mayor detalle al momento de realizar una recomendación, en otras palabras, explicar la razón por la que seleccionó una cartera sobre las demás.
- Se debían explicar a mayor detalle los elementos correspondientes al sistema (los elementos mostrados en el submenú *Show Conversation Data*)
- Mostrarle al usuario los conceptos de *Cartera*, *Criterio* y *Alternativa*, así como el orden de preferencia de criterios en caso de ser necesario, buscando con esto facilitar el diálogo. De igual forma no mostrarle al usuario el DTE en uso, ya que eso genera confusión.
- Hacer que el prototipo fuese capaz de manipular problemas de maximización y minimización.
- Desactivar algunos elementos de la pantalla principal si no son necesarios.

A partir de esto se llevaron a cabo las siguientes correcciones en el prototipo, las cuales se implementaron en la versión final del mismo, cuyo desarrollo se explicó a detalle en el capítulo anterior.



- Cada cambio de estándar de prueba es seguido por una explicación de su funcionamiento y la razón por la que se generó la recomendación actual.
- Se agregaron descripciones a cada uno de los elementos que se encuentran en la ventana *Conversation Data*, así como el orden de preferencia de criterios actual.
- Se agregaron los conceptos de *Alternativa*, *Criterio* y *Cartera* en la ventana *Help*.
- Se modificó el sistema para que fuera capaz de trabajar con problemas de maximización y minimización.
- Se hicieron correcciones generales en el aspecto gráfico de la interfaz.

### 6.2.2 Segunda etapa de evaluación y resultados

Durante esta etapa del proceso de evaluación se buscó valorar la satisfacción del usuario con respecto a la usabilidad de la versión final del prototipo para este trabajo. Por esta razón se les pidió a múltiples personas externas a este proyecto que utilizaran el sistema de recomendación y que generaran conclusiones a partir de la experiencia recibida.

Buscando evaluar el prototipo de manera exhaustiva, se decidió utilizar dos grupos de siete personas diferentes, cada grupo constó de estudiantes de maestría y doctorado en ciencias de la computación. A cada grupo se le dio un periodo de tiempo de 50 minutos para llevar a cabo el proceso de evaluación de manera completa.

Los pasos que se llevaron a cabo durante el proceso de evaluación de usabilidad fueron los siguientes:

- 1) *Introducción al sistema*: en este paso se les mostró a los usuarios el prototipo del sistema de recomendación, dándoles una explicación del funcionamiento de este. De igual manera se les mostró los diferentes componentes que este posee, las posibles acciones que pueden llevar a cabo y la forma en la que reacciona el sistema a cada uno de los movimientos realizados por el usuario.
- 2) *Realización de un problema introductorio con el objetivo de adentrar al usuario al uso del prototipo*: en este punto se les puso enfrente a los usuarios una situación con la cual estuvieran familiarizados pidiéndoles llevar a cabo un conjunto de pasos, para ver el problema designado véase el Anexo E.

- a. Generación de una tabla de desempeño.
  - b. Análisis manual de un conjunto de carteras, seleccionando una o más que les atraiga.
  - c. Desarrollo de una instancia utilizando valores previamente definidos y su tabla de desempeño.
  - d. Aplicación del prototipo del sistema de recomendación utilizando la instancia generada en el punto anterior.
- 3) *Simulación de un problema real de mayor complejidad con una instancia previamente creada*: con el objetivo de evaluar completamente las capacidades del prototipo, se decidió llevar a cabo la experimentación con dos grupos diferentes. El primer grupo trabajó con una instancia sin preferencias definidas en sus criterios, mientras que el segundo grupo utilizó una instancia con preferencias. Para ver las instancias utilizadas para cada grupo véase el Anexo G
- 4) Aplicación de un cuestionario para evaluar la usabilidad del prototipo y observaciones por parte del usuario. En el Anexo F se adjunta el problema del punto anterior mostrado a los evaluadores, así como el cuestionario utilizado

Para el punto final de la evaluación se tomaron en consideración varios aspectos relacionados con el sistema de recomendación, estos puntos fueron:

- Diseño
- Funcionalidad
- Facilidad de uso
- Aprendizaje
- Satisfacción
- Resultado y uso futuro

Para cada uno de estos puntos se estableció un conjunto de preguntas dentro del cuestionario, buscando obtener como resultado una evaluación clara de cada uno de dichos aspectos. Las preguntas generadas están basadas en el modelo de evaluación *PSSUQ* propuesto por Lewis [Lewis, 1995], utilizando la adaptación realizada por Zins en su

trabajo [Zins et al., 2004]. Para evaluar cada pregunta se utilizó como medición una calificación entre 0 a 10, donde 0 significaba completo desacuerdo y 10 completo acuerdo.

#### *Diseño*

- 1) Me agradó utilizar la interfaz del sistema.
- 2) La organización de la información presentada por el sistema era clara.
- 3) La interfaz fue agradable de usar.

#### *Funcionalidad*

- 4) El sistema tiene las funciones y capacidades que esperaba.
- 5) La información recuperada por el sistema me ayudo a terminar mis actividades de manera efectiva.
- 6) Los proyectos seleccionados por la recomendación del sistema son convenientes para mi inversión.
- 7) Encuentro la opción de forzado de recomendación útil.

#### *Facilidad de uso*

- 8) El sistema fue simple de utilizar.
- 9) Fue fácil encontrar la información que necesitaba.
- 10) La información de ayuda para el sistema proporcionada fue clara.
- 11) En general, este sistema es fácil de utilizar.

#### *Aprendizaje*

- 12) Fue fácil aprender a utilizar el sistema.
- 13) La información dada por el sistema fue fácil de entender.
- 14) Las explicaciones que proporciona el sistema en el diálogo facilitan la toma de decisiones.
- 15) Considero que se requiere información previa sobre uso del sistema.

#### *Satisfacción*

- 16) Me sentí cómodo utilizando este sistema.
- 17) Disfruté construir mi plan de inversión utilizando este sistema.

18) En general, estoy satisfecho con el sistema.

### *Resultado y uso futuro*

19) Pude completar mis tareas utilizando este sistema.

20) Pude completar mis tareas rápidamente utilizando este sistema.

21) Pude completar de manera eficiente mis tareas utilizando este sistema.

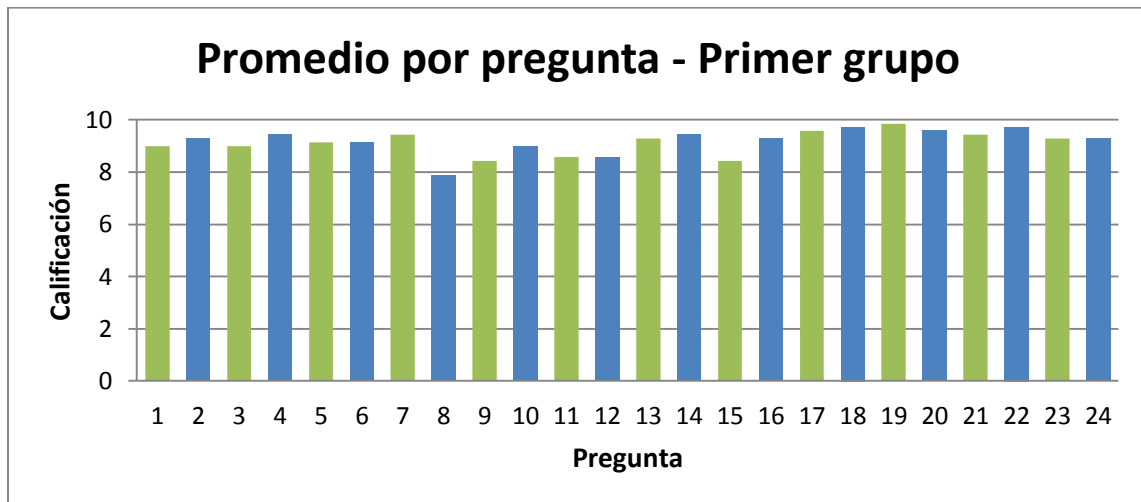
22) Siento que podría volverme productivo rápidamente si utilizó este sistema.

23) El sistema fue capaz de convencerme que las recomendaciones tenían un valor.

24) Con mi experiencia actual utilizando el sistema. Creo que lo usaría de manera regular.

### **Resultados**

El primer grupo, el cual manejó una instancia sin preferencias, generó los siguientes resultados, representados gráficamente en la Figura 6.1:



**Figura 6.1:** Promedio por pregunta del primer grupo.

Como se puede ver cada una de las preguntas se mantuvo sobre el promedio de 8, con la excepción de la pregunta número 8, la cual preguntaba si el sistema resultó simple de utilizar, esto debido a que dos de los siete evaluadores le dieron un 3 y un 6 a este aspecto, aún cuando el resto de los usuarios calificó esta pregunta con valor mayor de 8.

También se calculó el promedio por sección, obteniendo los siguientes resultados.

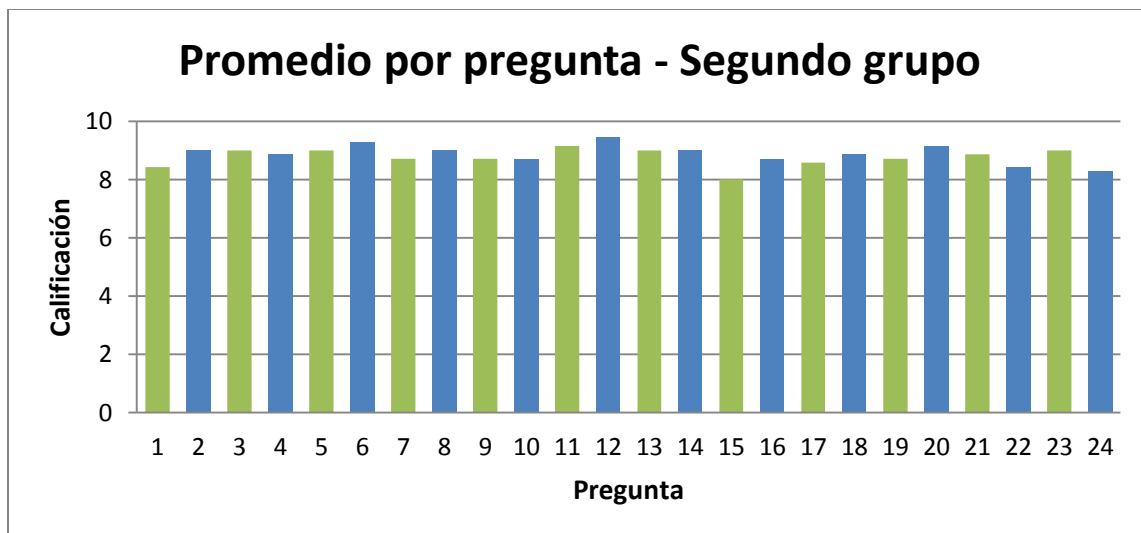
**Tabla 6.2:** Promedio por sección del primer grupo.

Diseño	9.0952381
Funcionalidad	9.28571429
Facilidad de uso	<b>8.46428571</b>
Aprendizaje	8.92857143
Satisfacción	<b>9.52380952</b>
Resultado y uso futuro	<b>9.52380952</b>

Como se puede ver en la tabla anterior, tanto las secciones de satisfacción como de resultado y uso futuro recibieron el mayor puntaje. Por otro lado, la sección de facilidad de uso recibió el menor puntaje.

Aparte se generaron observaciones por parte de los usuarios mencionando el buen diseño así como la inconveniencia de la superposición de algunas ventanas y la falta de capacidad para redimensionar estas.

Por otra parte, se obtuvieron los siguientes resultados del segundo grupo, el cual utilizó una instancia con preferencias, la Figura 6.2 representa de manera gráfica los resultados de esta evaluación:



**Figura 6.2:** Promedio por pregunta del segundo grupo.

De igual manera que en el primer grupo, los promedios de cada pregunta se mantuvieron igual o superiores a 8. En este caso, la pregunta con un menor puntaje fue la número 15, la cual corresponde a “*Considero que se requiere información previa sobre uso del sistema*”, con un promedio de 8, esto debido a que tres personas le dieron un puntaje menor que 7.

También se obtuvieron los promedios por sección para este grupo, los cuales resultaron de la siguiente manera:

**Tabla 6.3:** Promedio por sección del segundo grupo.

Diseño	8.80952381
Funcionalidad	<b>8.96428571</b>
Facilidad de uso	8.89285714
Aprendizaje	8.85714286
Satisfacción	<b>8.71428571</b>
Resultado y uso futuro	8.73809524

Como se puede ver, en este caso los promedios no alcanzaron un valor de 9 en ninguna sección. También en este caso, la sección de satisfacción del usuario obtuvo un menor puntaje mientras que la sección de funcionalidad obtuvo el mayor puntaje. Esto debido a que el grupo tuvo un mayor número de evaluadores pertenecientes al doctorado en ciencias de la computación, cuyos estándares de evaluación son más estrictos.

Este grupo realizó observaciones respecto a la necesidad de crear un tutorial de uso del sistema, la capacidad de generar una cartera nueva para comparar contra las existentes, resultados más visuales y la superposición de las ventanas.

Finalmente se hizo una comparación entre los promedios por pregunta y sección de ambos grupos, los resultados se muestran en las Tablas 6.4 y 6.5 respectivamente.

Como puede verse la mayor diferencia entre ambos grupos se presenta en la pregunta número 22, la cual corresponde a “*Siento que podría volverme productivo rápidamente si utilizó este sistema*”, mientras que la menor diferencia le corresponde a la pregunta número 3, la cual es “*La interfaz fue agradable de usar*”.

**Tabla 6.4:** Diferencia entre los promedios por pregunta de ambos grupos.

Pregunta	Grupo 1	Grupo 2	Diferencia
1	9	8.42857143	0.57142857
2	9.28571429	9	0.28571429
3	9	9	0
4	9.42857143	8.85714286	0.57142857
5	9.14285714	9	0.14285714
6	9.14285714	9.28571429	0.14285714
7	9.42857143	8.71428571	0.71428571
8	7.85714286	9	1.14285714
9	8.42857143	8.71428571	0.28571429
10	9	8.71428571	0.28571429
11	8.57142857	9.14285714	0.57142857
12	8.57142857	9.42857143	0.85714286
13	9.28571429	9	0.28571429
14	9.42857143	9	0.42857143
15	8.42857143	8	0.42857143
16	9.28571429	8.71428571	0.57142857
17	9.57142857	8.57142857	1
18	9.71428571	8.85714286	0.85714286
19	9.85714286	8.71428571	1.14285714
20	9.57142857	9.14285714	0.42857143
21	9.42857143	8.85714286	0.57142857
22	9.71428571	8.42857143	<b>1.28571429</b>
23	9.28571429	9	0.28571429
24	9.28571429	8.28571429	1

De igual manera se calculó la diferencia por sección entre ambos grupos, mostrándose los resultados en la Tabla 6.5. Para esta tabla, la mayor diferencia se encuentra en la sección de satisfacción, mientras que la sección de aprendizaje muestra la menor diferencia entre ambos grupos.

En general, existe una mejor calificación en los resultados del primer grupo, sin embargo la diferencia en 5 de las 6 secciones es menor de 0.5, así como también los usuarios del segundo grupo encontraron una mayor facilidad de uso del sistema a comparación del primero.

**Tabla 6.5:** Diferencia entre los promedios por sección de ambos grupos.

<b>Sección</b>	<b>Grupo 1</b>	<b>Grupo 2</b>	<b>Diferencia</b>
Diseño	9.0952381	8.80952381	0.28571429
Funcionalidad	9.28571429	8.96428571	0.32142857
Facilidad de uso	8.46428571	8.89285714	0.42857143
Aprendizaje	8.92857143	8.85714286	<b>0.07142857</b>
Satisfacción	9.52380952	8.71428571	<b>0.80952381</b>
Resultado y uso futuro	9.52380952	8.73809524	0.78571429

Al ser los promedios por pregunta y por sección mayor de 8 en su gran mayoría se puede decir que el prototipo tuvo un cierto éxito en el grupo de evaluación. Sin embargo existen un conjunto de aspectos a considerar, uno de ellos el hecho de que el segundo grupo otorgó en general un puntaje menor al dado por el primer grupo, por lo cual se puede decir que el uso de DTE1, el cual se utiliza si la instancia no tiene una preferencia predefinida, en un juego de diálogo provee un mejor aprendizaje y satisfacción. Sin embargo, no se puede menospreciar el hecho de que DTE2, el cual se utiliza si existe una preferencia definida desde la instancia, tiene una mayor facilidad de uso, por lo que ninguno de los dos puede descartarse y ambos tienen utilidad dentro del prototipo.

Por otro lado, considerando tanto las observaciones como las calificaciones otorgadas, se puede concluir que es necesario introducir de manera previa al usuario antes de que este utilice el prototipo, ya sea mediante capacitación o un tutorial.

También, considerando el aspecto gráfico, se puede afirmar que el sistema requiere de una mayor cantidad de resultados visuales, ya que aunque el texto pueda ser suficiente para algunas personas, existen otras que prefieren una representación en imágenes o gráficas. De igual manera se debe considerar las propiedades gráficas de la ventana principal así como las emergentes del prototipo

Finalmente se debe considerar el hecho de que en trabajos futuros se puede extender las acciones que puede llevar a cabo el usuario dentro del sistema. Un ejemplo de esto es la capacidad de poder diseñar una cartera dentro de la interfaz del sistema de recomendación y compararla con las previamente existentes en la instancia.



# Capítulo 7

## Conclusión y trabajos futuros

---

---

En este capítulo se presentan las conclusiones de este proyecto de tesis además de las aportaciones generadas por este trabajo. De igual forma se detallan algunos puntos que permitirán expandir este trabajo de investigación en el futuro.

### 7.1 Conclusiones

En este proyecto se propuso caracterizar las tareas cognitivas que están involucradas en proceso de ayuda a la toma de decisiones, buscando de esta manera guiar la interacción con los usuarios mediante un sistema de recomendación.

Para llevar a cabo lo anterior, inicialmente se definieron las diferentes tareas cognitivas involucradas en el proceso de apoyo a la decisión, identificando aquellas en las que se genera una posible interacción entre el usuario y el sistema de recomendación. Para este trabajo se hizo enfoque en dos tareas cognitivas del proceso, el *modelo de evaluación de alternativas* y la *construcción de argumentos para la recomendación final*. Para evaluar la calidad de la propuesta se desarrolló un prototipo experimental cuyo nivel de usabilidad se determinó con usuarios reales.

Para la primera tarea se establecieron los estándares de prueba, los cuales mediante la definición de sus propiedades le permiten seleccionar un modelo el cual evaluará el conjunto de alternativas existentes, apoyándose de los diferentes atributos que poseen cada uno de dichas soluciones.

Para la segunda tarea se hizo apoyo de los esquemas de argumentación y la teoría de la argumentación, ambos definidos dentro de un juego de diálogo, lo cual le permitía al usuario establecer una interacción con el sistema de recomendación. Esto con el objetivo de

permitirle al DM entender las razones por las cuales una solución tiene un cierto grado de preferencia sobre las demás del conjunto.

Dicha interacción entre el usuario y el sistema de recomendación se lleva a cabo mediante un prototipo, el cual fue implementado como parte de los objetivos de este proyecto. Este prototipo buscaba facilitar el proceso de toma de decisiones de los usuarios aplicando ambas tareas cognitivas previamente mencionadas mientras se llevaba a cabo un juego de diálogo, buscando que el DM estuviera completamente satisfecho con respecto a su decisión final, lo cual es el objetivo primordial de un sistema de recomendación.

El prototipo implementado en este proyecto fue evaluado por personas con un alto nivel de conocimiento en el campo con el objetivo de analizar la existencia de posibles fallos o carencias dentro del sistema, buscando hacer las correcciones necesarias. Posteriormente se les permitió a personas cuyo nivel de conocimiento del campo iba desde un conocimiento amplio hasta nulo utilizar el sistema para evaluar la usabilidad de este en un ambiente real. Los resultados mostraron una aprobación promedio satisfactoria del 89.91% por parte de los usuarios. Sin embargo, existieron diversas observaciones con respecto al prototipo, dichas observaciones se consideraron como posibles trabajos futuros y se mencionarán más adelante dentro de este mismo capítulo.

## **7.2 Aportaciones**

La aportación principal de este proyecto fue la caracterización de tareas cognitivas de un tomador de decisiones, particularmente para el modelo de evaluación de alternativas y la construcción de argumentos para la recomendación final.

- Para la primera de estas tareas se utilizaron diversos métodos MCDA como estándares de prueba con el fin de evaluar el conjunto de alternativas existentes dentro de un problema de carteras de proyectos públicos (ver sección 2.6).
- Se hizo uso de la teoría de la argumentación, mediante la aplicación de un juego de diálogo apoyado por los esquemas de argumentación para llevar a cabo la construcción de argumentos para la recomendación final (ver sección 2.7).

- Para el juego de diálogo, se propuso el uso de múltiples conjuntos de reglas de locución, y por consiguiente múltiples diagramas de transición de estados (ver sección 2.5).

Para evaluar la calidad de la caracterización propuesta, se llevo a cabo la generación de un prototipo de sistema de recomendación para el proceso de toma de decisiones aplicado al problema de carteras de proyectos públicos, el cual está apoyado de estándares de prueba y esquemas de argumentación. Las aportaciones realizadas por este proyecto dentro del prototipo fueron las siguientes:

- La implementación del método MCDA TOPSIS dentro de un sistema de recomendación.
- La implementación de la capacidad del sistema para rechazar una recomendación definida por el usuario si los argumentos con los que defiende dicha recomendación no son lo suficientemente fuertes.
- La implementación de una solución forzada, la cual le permite al usuario definir una solución como la recomendación del sistema, aun cuando los argumentos no sean lo suficientemente fuertes para defender esto.
- La implementación de una visualizador del estado de varios diferentes elementos dentro del juego de diálogo, tal como los ciclos, el estándar de prueba utilizado y sus propiedades activas e inactivas, entre otros.

El prototipo desarrollado fue calificado por personas con un alto y medio nivel de conocimiento dentro del campo de apoyo a la toma de decisión mediante la aplicación de una experimentación grupal con el uso del prototipo. Se diseñó un procedimiento de evaluación de prototipos de apoyo a la decisión basado en encuestas estandarizadas de usabilidad (ver sección 6.2.2), hasta nuestro conocimiento no existen procedimientos de este tipo en el área de toma de decisiones multicriterio.

### **7.3 Trabajos futuros**

A partir de los resultados obtenidos tanto en la implementación del prototipo así como en la evaluación realizada en este trabajo, se plantean los siguientes trabajos futuros:

- La capacidad de generar una nueva cartera en la instancia durante el juego de diálogo.
- La posibilidad de utilizar varios diagramas de transición de estados en el mismo juego de diálogo.
- La aplicación de mayores apoyos visuales dentro de la interfaz (gráficas, barras, figuras, poder personalizar el nombre de los criterios, carteras y alternativas, manipulación más sencilla de la interfaz, entre otros).
- Permitirle al usuario hacer afirmaciones abiertas, en vez de forzarlo a seleccionar una opción de un conjunto de soluciones predefinidas.
- Tener un tutorial del uso del sistema para los que no estén familiarizados en el proceso de toma de decisiones.
- Utilizar diferentes modelos MCDA a los aplicados en este trabajo.
- Desarrollar un módulo que permita al DM introducirse a la conceptualización de los estándares de prueba e identifique aquel que le corresponde.

# Referencias

- [Adomavicius y Tuzhilin, 2005] Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6), 734-749.
- [Belton y Stewart, 2002] Belton, V; Stewart, B. (2002). *Multiple Criteria Decision Analysis. An Integrated Approach*. Kluwer Academic Publishers, Massachusetts.
- [Brooke, 1996] Brooke, J. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 189(194), 4-7.
- [Carazo et al., 2010] Carazo, A. F., Gómez, T., Molina, J., Hernández-Díaz, A. G., Guerreo, F. M., and Caballero, R. (2010). Solving a comprehensive model for multiobjective project portfolio selection. *Computers & Operations Research*, 37(4):630–639.
- [Carlson, 2012] Carlson, L. (2012). *Dialogue games: An approach to discourse analysis* (Vol. 17). Springer Science & Business Media.
- [Chernoff, 1987] Chernoff, H. (1987). *Elementary Decision Theory*. Dover Publications.
- [Cruz-Reyes et al., 2014] Cruz-Reyes, L., Trejo, C. M., Irrarragorri, F. L., & Santillán, C. G. G. (2014). A Decision Support System Framework for Public Project Portfolio Selection with Argumentation Theory. In *Recent Advances on Hybrid Approaches for Designing Intelligent Systems* (pp. 467-479). Springer International Publishing.
- [Deb, 2001] Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. Interscience series in systems and optimization. John Wiley & Sons, LTD.
- [Doyle, 2004] Doyle, J. (2004). Prospects for preferences. *Computational Intelligence*, 20(2):111–136.
- [Fernández et al. 2011] Fernández, E., López, E., Navarro, J., Vega, I. (2011): “Aplicación de metaheurísticas multiobjetivo a la solución de problemas de cartera de proyectos públicos con una valoración multidimensional de su impacto”, *Gestión y Política Pública*, vol. XX, no. 2, 2011, pp. 381- 432.
- [Fernández y Navarro, 2002] Fernández, E. and Navarro, J. (2002). A genetic search for exploiting a fuzzy preference model of portfolio problems with public projects. *Annals OR*, 117(191-213):191–213.
- [García, 2010] Rodríguez, R. G., Reyes, L. C., Irrarragorri, F. L., & Santillán, C. G. (2010) *Híper-heurístico Aplicado al Problema de Cartera de Proyectos Sociales*.
- [Gordon y Walton, 2006] T.F. Gordon and D. Walton (2006). The Carneades argumentation framework. Using presumptions and exceptions to model critical questions. In P.E Dunne and T.J.M. Bench-Capon, editors, *Proceedings of the 1st International Conference on Computational Models of Natural Argument (COMMA'06)*, pages 195–207.
- [Gordon y Walton, 2009] T.F. Gordon and D. Walton (2009). Proof burdens and standards. In I. Rahwan and G. Simari, editors, *Argumentation in Artificial Intelligence*. Springer-verlag.
- [Hwang y Yoon, 1981] Hwang, C. L., & Yoon, K. (1981) *Multiple Attribute Decision Making-Method and Applications, A State-of-the-Art Survey*. Springer-Verlag, New York
- [Jameson et al., 2015] Jameson, A., Willemsen, M. C., Felfernig, A., de Gemmis, M., Lops, P., Semeraro, G., & Chen, L. (2015). Human decision making and recommender systems. In *Recommender Systems Handbook* (pp. 611-648). Springer US.

- [Keeney y Raiffa, 1993], Keeney, R.L., and Raiffa, H. (1993) *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press.
- [Labreuche, 2006] Labreuche C. (2006). Argumentation of the decision made by several aggregation operators based on weights. *Proceedings of the 11th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'06)*, Paris, France; 683–690.
- [Lewis, 1995] Lewis, J. R. (1995). IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7(1), 57-78.
- [López, et al., 2008] López, J.C.L., González, E.F., & Alvarado, M.T. (2008). Special Issue on Multicriteria Decision Support Systems. *Computación y Sistemas*, 12(2).
- [López, 2013] López, J. (2013). *Optimización multi-objetivo* (Doctoral dissertation, Facultad de Informática).
- [Marzouk, 2011] Marzouk, M. M. (2011). ELECTRE III model for value engineering applications. *Automation in Construction*, 20(5), 596-600.
- [McCarthy, 1977] J.L. McCarthy (1977). Epistemological problems of artificial intelligence. In R. Reddy, editor, *Proceedings of the 5th International Joint Conference on Artificial Intelligence(IJCAI'77)*, pages 1038–1044.
- [McDermott y Doyle, 1980] D. McDermott and J. Doyle (1980). Nonmonotonic logic 1. *Artificial Intelligence*, 13:41–72.
- [Miller, 1956] Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological review*, 63(2), 81.
- [Navarro, 2005] J. Navarro (2005). *Herramientas inteligentes para la evaluación y selección de proyectos de investigación-desarrollo en el sector público*. PhD thesis, Doctoral Thesis, Universidad Autónoma de Sinaloa, Sinaloa, México.
- [Ouerdane, 2009] *Multiple Criteria Decision Aiding: a Dialectical Perspective* (2009). PhD thesis, Université Paris Dauphine.
- [Oztürk et al., 2005] Oztürk, M., Tsoukiàs, A., and Ph. Vincke. (2005). Preference Modeling, pages 27–72. In: *State of the Art in Multiple Criteria Decision Analysis*, M. Ehrgott, S. Greco and J. Figueira (Ed.). Wiley Series on Intelligent Systems. Springer-Verlag.
- [Pérez et al., 2007] Pérez, F. Molina, J., Caballero, R., Coello, C., Hernández, A., (2007). Hibridación de métodos exactos y heurísticos para el problema multiobjetivo. *Journal Economic Literature: C61; C63. XV Jornadas de ASEPUMA y III Encuentro Internacional*.
- [Prakken y Sartor, 2002] Prakken, H., & Sartor, G. (2002). The role of logic in computational models of legal argument: a critical survey. In *Computational logic: Logic programming and beyond* (pp. 342-381). Springer Berlin Heidelberg.
- [Prakken, 2005] Prakken, H. (2005). Coherence and flexibility in dialogue games for argumentation. *Journal of logic and computation*, 15(6), 1009-1040.
- [Pu et al., 2011] Pu, P., Chen, L., & Hu, R. (2011, October). A user-centric evaluation framework for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems* (pp. 157-164). ACM.
- [Reiter, 1980] R. Reiter (1980). A logic for default reasoning. *Artificial Intelligence*, 67(1, 2):81–132.

- [Resnick y Varian, 1997] Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40(3), 56-58.
- [Roubens y Vincke, 1985] Roubens, M., and Ph. Vincke. (1985). *Preference modeling*. Springer-Verlag.
- [Roy, 1991] Roy, B. (1991). The outranking approach and the foundations of ELECTRE methods. *Theory and decision*, 31(1), 49-73.
- [Sánchez, 2007] Sánchez, P. (2007). *Modelos para la combinación de preferencias en toma de decisiones: herramientas y aplicaciones*. Tesis Doctoral, Universidad de Granada.
- [Serrano, 2007] Serrano, V. (2007). *Métodos para reducir evaluaciones en algoritmos evolutivos multi-objetivo, basados en aproximación de funciones*. Tesis de Maestría, Centro de Investigación y de Estudios avanzados del Instituto Politécnico Nacional, Departamento de Ingeniería Eléctrica sección de Computación.
- [Tsoukiàs, 2007] A. Tsoukiàs (2007). On the concept of decision aiding process. *Annals of Operations Research*, 154(1):3– 27.
- [Tsoukiàs, 2008] Tsoukiàs, A. (2008). From decision theory to decision aiding methodology. *European Journal of Operational Research*, 187(1), 138-161.
- [Tsoukiàs, 2011] A. Tsoukiàs (2011), *Aiding to Decide. Concepts and Issues*. In *Evaluation and Decision Models, Real Case Studies*. Springer, Berlin.
- [Van Veldhuizen y Lamont, 1998] Van Veldhuizen, D., and Lamont, G. (1998). *Evolutionary Computation and Convergence to a Pareto Front*. Late Breaking Papers at the Genetic Programming 1998 Conference, edited by John R. Koza. 221-228. Stanford, CA: Stanford University Bookstore, July.
- [Walton, 1996] Walton, D (1996). *Argumentation schemes for Presumptive Reasoning*. Mahwah, N. J., Erlbaum.
- [Walton, 2005] D.N. Walton (2005) . Justification of argument schemes. *Australasian journal of logic*, 3:1–13.
- [Walton y Reed, 2002] Walton, D., & Reed, C. (2002, July). *Argumentation schemes and defeasible inferences*. In *Workshop on computational models of natural argument, 15th European conference on artificial intelligence* (pp. 11-20).
- [Walton et al., 2008] Walton, D., Reed, C., & Macagno, F. (2008). *Argumentation schemes*. Cambridge University Press.
- [Wilson, 1999] Wilson, R. et al. (1999). *The MIT Encyclopedia of the Cognitive Sciences*. Cambridge, the MIT Press.
- [Zins et al., 2004] Zins, A. H., Bauernfeind, U., Del Missier, F., Venturini, A., & Rumetshofer, H. (2004). *An experimental usability test for different destination recommender systems*, proceedings of ENTER 2004, Cairo, Egypt.

# Anexo A

## Esquemas de argumentación de los estándares de prueba

Ouerdane en su trabajo [Ouerdane, 2009] se apoya tanto de un conjunto de propiedades específicas, así como de esquemas de argumentación para poder llevar a cabo la selección correcta del estándar de prueba a utilizar en el proceso de apoyo a la toma de decisiones. A continuación se mostrarán los argumentos diseñados en su trabajo, así como el esquema correspondiente al estándar de prueba TOPSIS.

**Tabla A.1:** Esquema para un argumento desde mayoría simple [Ouerdane, 2009].

<b>Premisas</b>	Un conjunto de criterios $a \geq b \leftrightarrow  \{i \in N: aS_i b\}  \geq  \{i \in N: bS_i a\} $ Los suficientes criterios para usar mayoría simple	$\{h_1, \dots, h_n\}$ $a \geq b$
<b>Conclusión</b>	Mayoría simple es el estándar seleccionado	$r_i \approx$ mayoría simple

**Tabla A.2:** Esquema para un argumento desde el método lexicográfico [Ouerdane, 2009].

<b>Premisas</b>	Un conjunto de criterios Un orden lineal $a \sim_i b$ $a \geq_k b$ Los suficientes criterios para usar el método lexicográfico	$\{h_1, \dots, h_n\}$ $> l$ $\forall j > k$
<b>Conclusión</b>	Orden lexicográfico es el estándar seleccionado	$r_i \approx$ lexicográfico

**Tabla A.3:** Esquema para un argumento desde mayoría ponderada [Ouerdane, 2009].

<b>Premisas</b>	Un conjunto de criterios Un conjunto de coeficientes de importancia $a \geq b \leftrightarrow W_{ab} = \sum_{i: aS_i b} w_i \geq W_{ba} = \sum_{i: bS_i a} w_i$ Los suficientes criterios para usar mayoría ponderada	$\{h_1, \dots, h_n\}$ $\{w_1, \dots, w_n\}$ $a \geq b$
<b>Conclusión</b>	Mayoría ponderada es el estándar seleccionado	$r_i \approx$ mayoría ponderada

**Tabla A.4:** Esquema para un argumento desde suma de pesos [Ouerdane, 2009].

<b>Premisas</b>	Un conjunto de criterios Un conjunto de coeficientes de importancia $\text{puntaje}(a)$ $\text{puntaje}(a) \geq \text{puntaje}(b)$ Los suficientes criterios para suma de pesos	$\{h_1, \dots, h_n\}$ $\{w_1, \dots, w_n\}$ $\sum_{i=1}^n w_i h_i(a)$ $a \succeq b$
<b>Conclusión</b>	Suma de pesos es el estándar seleccionado	$r_i \approx$ suma de pesos



**Tabla A.5:** Esquema para un argumento desde TOPSIS.

<b>Premisas</b>	Un conjunto de criterios	$\{h_1, \dots, h_n\}$
	Un conjunto de coeficientes de importancia	$\{w_1, \dots, w_n\}$
	Una matriz de desempeño normalizada multiplicada por el peso de los criterios	$V$
	Separación de la solución ideal	$S_i^+ = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^+)^2}$
	Separación de la solución ideal negativa	$S_i^- = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^-)^2}$
Cercanía a la solución ideal	$C_i = \frac{S_i^-}{S_i^+ + S_i^-}$	
	$cercanía(a) \geq cercanía(b)$	$C_a \geq C_b$
<b>Conclusión</b>	TOPSIS es el estándar seleccionado	$r_i \approx \text{TOPSIS}$

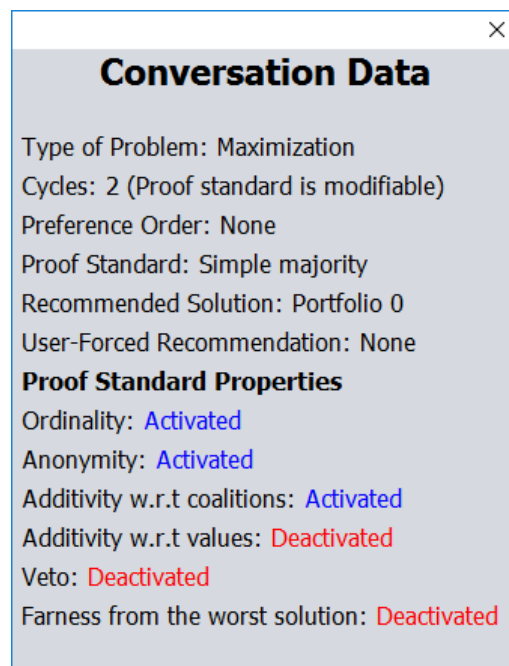
# Anexo B

## Ejemplo de aplicación de un esquema de argumentación en el prototipo

Como se menciona en este trabajo, el sistema se apoya de los esquemas de argumentación para llevar a cabo un juego de diálogo junto con el usuario. En este anexo se mostrará un ejemplo del funcionamiento de un esquema de argumentación dentro del prototipo.

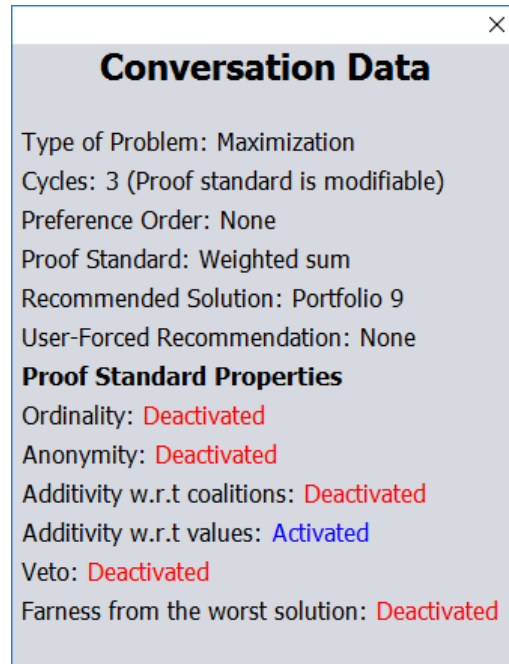
Supongamos que durante un juego de diálogo, el estándar de prueba en uso es la mayoría simple (véase Figura B.1), mientras que las propiedades activas son:

- Ordinalidad
- Anonimato
- Aditividad c.r.a. coaliciones



**Figura B.1:** Estándar de prueba actual y propiedades activas.

Sin embargo, durante el mismo diálogo el usuario decide activar la propiedad de aditividad c.r.a. valores, por efecto y debido a su naturaleza, se deben desactivar tanto la ordinalidad como el anonimato, lo cual a su vez lleva a la desactivación de la aditividad c.r.a. coaliciones, tal como se muestra en la Figura B.2.



**Figura B.2:** Estándar de prueba actual y propiedades activas después de la modificación.

Como se puede ver, la modificación de propiedad también llevó al cambio en el estándar de prueba en uso, esto debido a que se hizo uso del esquema de argumentación del razonamiento abductivo para evaluar nuevamente el conjunto de propiedades y determinar el estándar de prueba.

Considerando las premisas del esquema, definidas en la Tabla 2.1 en el Capítulo 2 se define  $F$  como conjunto de propiedades activas e inactivas de los estándares de prueba, mientras que  $E$  y  $E'$  definen dos estándares de prueba en comparación.

Debido a que hubo un cambio en las propiedades, el sistema revisa nuevamente este conjunto y concluye que la mayoría simple ya no es la alternativa  $E$  más satisfactoria para  $F$  sino que la suma de pesos es ahora el estándar que más satisface los valores actuales de  $F$ , ya que no existe otro estándar de prueba  $E'$  que tenga un mayor grado de satisfacción para  $F$ . Todo lo anterior está apoyado por el listado de propiedades por estándar de prueba definido en la Tabla 4.1 del Capítulo 4.

# Anexo C

## Ejemplo de actualización de los valores de los datos de una instancia

Parte de las acciones disponibles que tiene el usuario dentro del prototipo es la capacidad de actualizar los valores de la tabla de desempeño de la instancia en cualquier momento. En esa sección se mostrará un ejemplo de cómo se lleva a cabo este procedimiento así como la forma en la que el sistema responde a dichos cambios.

Inicialmente, como se puede ver en la Figura C.1 el sistema tiene a la cartera 9 como la solución recomendada. También como se observa en la Figura C.2 se han llevado a cabo dos ciclos y el estándar de prueba en uso es TOPSIS.

The screenshot shows a software interface titled "Instance Recommendation Options" with a central "Interaction Module". The dialogue text is as follows:

Expert: Yes, however, you can modify the status of a property if you want.  
User: (Argue)  
Expert: Additivity w.r.t. values activated  
Expert: With this property modification, the proof standard currently on use is TOPSIS  
Expert: In this proof standard, the best and worst possible solution for each portfolio are calculated, the portfolio that is closer to the best solution and farther to the worst is chosen.  
Expert: According to the new information, I recommend P9

Below the dialogue is the "Interaction Options" section with buttons for "Erase", "Update", and "Accept", and dropdown menus for "DM's Decision" (set to "Challenge"), "Question" (set to "Why?"), and "Alternative".

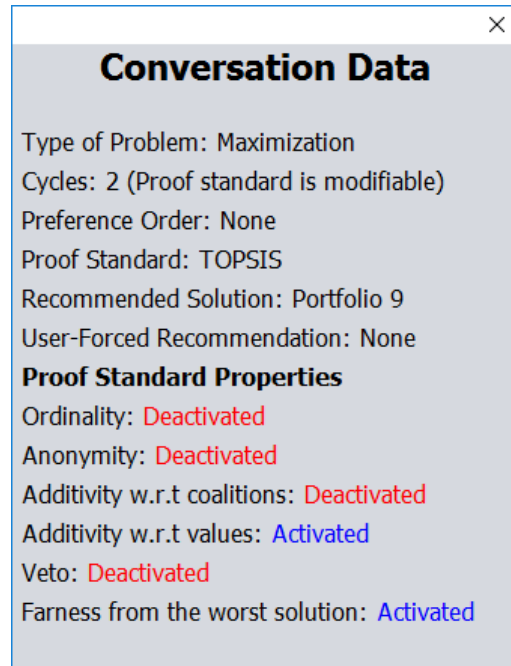
On the right side, there are two tables:

	a0	a1	a2	a3	a4	a5	a6	a7	a8	a
P0	0	0	1	1	0	0	1	0	0	0
P1	0	0	1	1	0	0	1	0	1	0
P2	0	1	1	1	0	0	1	0	0	0
P3	0	1	0	1	0	0	0	0	1	1
P4	0	1	0	1	0	0	0	1	1	0
P5	0	0	1	1	0	0	1	0	1	1
P6	0	0	1	1	0	0	1	0	1	0
P7	0	0	1	1	0	1	1	0	0	0
P8	0	1	1	1	0	0	1	0	1	0
P9	0	0	1	1	0	1	1	0	0	0

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Portfolio 0	86530	62840	3430	4440
Portfolio 1	86650	61360	3810	4290
Portfolio 2	83845	62385	3770	4320
Portfolio 3	78960	52190	4190	4205
Portfolio 4	79780	50995	4200	4175
Portfolio 5	82090	57905	3970	4285
Portfolio 6	89030	60280	3725	4420
Portfolio 7	88595	64625	3155	4300
Portfolio 8	85670	57580	4010	4330
Portfolio 9	79935	69110	3505	4410

Figura C.1: Recomendación actual durante un juego de diálogo en el prototipo.



**Figura C.2:** Datos actuales de la conversación durante un juego de diálogo en el prototipo.

Después el usuario accede a la opción de actualización de criterios desde el menú *Recommendation Options*, y define que los valores de los criterios 0 y 1 de la alternativa 1 tienen un valor de 10000 dentro de la matriz de desempeño, tal como se muestra en la Figura C.3.

**Performance Table**

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Alternative 0	3200	2000	165	300
Alternative 1	10000	10000	385	390
Alternative 2	5680	6940	270	445
Alternative 3	8965	4195	355	415
Alternative 4	6550	6560	315	440
Alternative 5	6740	6290	150	350
Alternative 6	9055	7165	375	485
Alternative 7	4170	3015	410	285
Alternative 8	9735	2860	480	330
Alternative 9	3350	4210	400	315
Alternative 10	8595	7270	150	265
Alternative 11	9070	2430	455	360
Alternative 12	9930	5825	420	385
Alternative 13	4675	4505	425	490
Alternative 14	8065	4030	165	425
Alternative 15	7910	7665	240	320
Alternative 16	9860	6265	415	350
Alternative 17	3175	6240	225	445
Alternative 18	9660	655	320	475
Alternative 19	1150	4500	415	400
Alternative 20	3245	5950	105	275

Save

**Figura C.3:** Actualización de los valores en la tabla de desempeño.

De ahí, el usuario presiona el botón *Save* el cual guarda los valores y cierra la ventana, después presiona el botón *Update* en la ventana principal ya estando satisfecho con los cambios realizados. A continuación el sistema reevalúa la tabla de desempeño y genera una nueva recomendación acorde a los nuevos valores. El sistema ahora determina como nueva recomendación la cartera 8, dando aviso de esto al usuario tal como se ve en la Figura C.4. Nótese en esa misma figura que varios valores de los criterios 0 y 1 de las diferentes carteras en la *tabla de criterios de evaluación por cartera* han cambiado, esto acorde a los nuevos valores definidos.

The screenshot shows a software interface titled "Interaction Module" with a sub-header "Instance: Recommendation Options". The main area contains a chat log with the following text:

User: (Argue)  
 Expert: Farness from the worst solution activated  
 Expert: With this property modification, the proof standard currently on use is TOPSIS  
 Expert: In this proof standard, the best and worst possible solution for each portfolio are calculated, the portfolio that is closer to the best solution and farther to the worst is chosen.  
 Expert: According to the new information, I recommend P9  
 Expert: Data have been modified, I recommend P8 as the best choice.

Below the chat log are "Interaction Options" including buttons for "Erase", "Update", and "Accept", along with dropdown menus for "DM's Decision" (set to "Challenge"), "Question" (set to "Why?"), and "Alternative".

On the right side, there are two tables:

**Portfolio Composition (Portfolio/Alternative)**

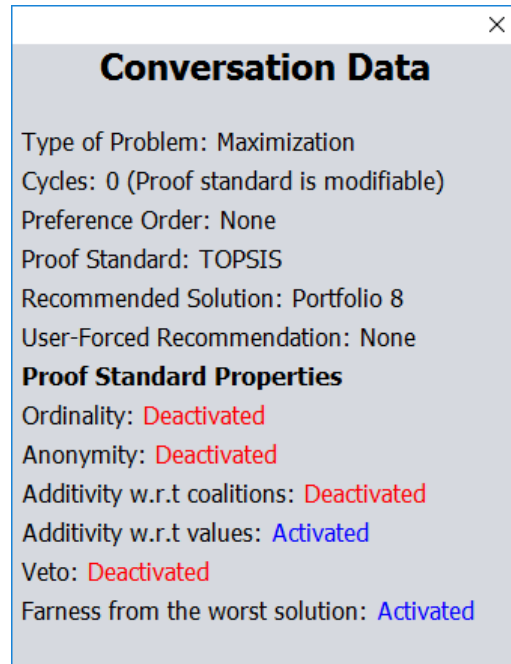
	a0	a1	a2	a3	a4	a5	a6	a7	a8	a9
P0	0	0	1	1	0	0	1	0	0	0
P1	0	0	1	1	0	0	1	0	1	0
P2	0	1	1	1	0	0	1	0	0	0
P3	0	1	0	1	0	0	0	0	1	1
P4	0	1	0	1	0	0	0	1	1	0
P5	0	0	1	1	0	0	1	0	1	1
P6	0	0	1	1	0	0	1	0	1	0
P7	0	0	1	1	0	1	1	0	0	0
P8	0	1	1	1	0	0	1	0	1	0
P9	0	0	1	1	0	1	1	0	0	0

**Evaluation Criteria per Portfolio**

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Portfolio 0	86530	62840	3430	4440
Portfolio 1	86650	61360	3810	4290
Portfolio 2	87590	70745	3770	4320
Portfolio 3	82705	60550	4190	4205
Portfolio 4	83525	59355	4200	4175
Portfolio 5	82090	57905	3970	4285
Portfolio 6	89030	60280	3725	4420
Portfolio 7	88595	64625	3155	4300
Portfolio 8	89415	65940	4010	4330
Portfolio 9	79935	69110	3505	4410

**Figura C.4:** Nueva recomendación a partir de los valores actualizados.

Si se revisan los datos actuales de la conversación se podrá ver que el número de ciclos transcurridos dentro del juego de diálogo ha sido reiniciado a cero, sin embargo, tanto las propiedades activas así como el estándar de prueba en uso siguen siendo los mismos, esto para permitirle al usuario y al sistema mantener un cierto avance dentro del diálogo buscando acelerar el proceso de toma de decisiones. Lo anterior se muestra en la Figura C.5.



**Figura C.5:** Datos actuales de la conversación después de la actualización de valores.

# Anexo D

## Ejemplo del desarrollo de un juego de diálogo en el prototipo

En esta sección se muestra un ejemplo de un juego de diálogo entre dos jugadores, en este caso el usuario y el sistema, así como su representación en el prototipo de sistema de recomendación implementado para este trabajo.

Juego de diálogo:

INICIO

1. Experto: Recomiendo  $x$  como la mejor solución. (*S:Afirmar*).
2. Usuario: ¿Por qué? (*U:Retar*).
3. Experto: Porque globalmente  $x$  es mejor que el resto de las alternativas. (*S:Argumentar*).
4. Usuario: ¿Son las razones para soportar la recomendación lo suficientemente fuertes? (*U:FormularC*).
5. Experto: Son los suficientemente fuertes para el estándar de prueba actual. (*S:AfirmarC*).
6. Usuario: ¿Por qué  $x$  es mejor que  $y$ ? (*U:Retar*).
7. Experto: Porque  $x$  supera a  $y$  en los criterios  $c$ . (*S:Argumentar*).
8. Usuario: ¿El sistema tiene la suficiente información para definir un estándar de prueba? (*U:FormularC*).
9. Experto, Si, sin embargo si desea cambiar alguna propiedad puede hacerlo. (*S:RetarC*).
10. Usuario: Quiero usar un umbral de veto (Veto activado). (*U:Argumentar*).
11. Experto: Veto activado (*S:Acceptar*).
12. Experto: Con esta modificación de propiedades, el estándar de prueba en uso actualmente es mayoría simple con veto.
13. Experto: En este estándar de prueba, se aplica un umbral de veto durante la evaluación de la mayoría simple.
14. Experto: Acorde a la nueva información obtenida, recomiendo  $z$  como la mejor opción (*S:Afirmar*).



15. Usuario: ¿Por qué? (*U:Retar*).
16. Experto: Porque globalmente  $z$  es mejor que el resto de las alternativas. (*S:Argumentar*).
17. Usuario: Yo prefiero la alternativa  $y$ . (*U:Afirmar*).
18. Experto: ¿Por qué? (*S:Retar*).
19. Usuario: Porque el presupuesto de la recomendación está muy cerca del límite superior. (*U:Argumentar*).
20. Experto: La cartera  $y$  tiene un rango menor del 10% de su presupuesto con respecto a  $z$  (*S:Rechazar*).
21. SOLUCIÓN ACEPTADA (*S:Aceptar*).

FIN

El juego de diálogo anterior se puede representar en el prototipo, en las siguientes figuras se mostrará el desarrollo del diálogo anterior dentro del sistema de recomendación.

Inicio y paso 1

The screenshot shows a software window titled "Instance Recommendation Options" with an "Interaction Module" section. The expert's advice is displayed in a text area, and the "Evaluation Criteria per Portfolio" table is visible on the right.

**Interaction Module**

Expert: After analyzing the instance's data I have concluded that the most suitable proof standard to use right now is Simple majority

Expert: In this proof standard, we have to take into consideration the value of the criteria for each portfolio then we compare each possible pair. A portfolio is preferred over another if the number of criteria in which the former defeats the latter surpasses the opposite.

Expert: I recommend P0 as the best choice

From this point until the end of this instance's dialogue you can: 1)Select an action in the Interaction Options below to

**Interaction Options**

Erase DM's Decision Challenge Accept

Update Question Why?

Alternative

**Portfolio Composition (Portfolio/Alternative)**

	a0	a1	a2	a3	a4	a5	a6	a7	a8	a
P0	0	0	1	1	0	0	1	0	0	0
P1	0	0	1	1	0	0	1	0	1	0
P2	0	1	1	1	0	0	1	0	0	0
P3	0	1	0	1	0	0	0	0	1	1
P4	0	1	0	1	0	0	0	1	1	0
P5	0	0	1	1	0	0	1	0	1	1
P6	0	0	1	1	0	0	1	0	1	0
P7	0	0	1	1	0	1	1	0	0	0
P8	0	1	1	1	0	0	1	0	1	0
P9	0	0	1	1	0	1	1	0	0	0

**Evaluation Criteria per Portfolio**

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Portfolio 0	86530	62840	3430	4440
Portfolio 1	86650	61360	3810	4290
Portfolio 2	83845	62385	3770	4320
Portfolio 3	78960	52190	4190	4205
Portfolio 4	79780	50995	4200	4175
Portfolio 5	82090	57905	3970	4285
Portfolio 6	89030	60280	3725	4420
Portfolio 7	88595	64625	3155	4300
Portfolio 8	85670	57580	4010	4330
Portfolio 9	79935	69110	3505	4410

Figura D.1: Inicio y paso 1 del juego de diálogo.

Pasos 2 a 7.

**Interaction Module**

User: (Challenge) Why is P0 the best option?

Expert: Because P0 is globally better than the other alternatives.

User: (Pose Critical Question) Are the reasons that support the recommendation strong enough?

Expert: They are strong enough for the current proof standard.

User: (Challenge) Why is P0 a better option than P1?

Expert: Because P0 is better in the criteria 1 3 than the option P1.

Interaction Options

Erase DM's Decision Accept Accept

Update Question Alternative

**Portfolio Composition (Portfolio/Alternative)**

	a0	a1	a2	a3	a4	a5	a6	a7	a8	a
P0	0	0	1	1	0	0	1	0	0	0
P1	0	0	1	1	0	0	1	0	1	0
P2	0	1	1	1	0	0	1	0	0	0
P3	0	1	0	1	0	0	0	0	1	1
P4	0	1	0	1	0	0	0	1	1	0
P5	0	0	1	1	0	0	1	0	1	1
P6	0	0	1	1	0	0	1	0	1	0
P7	0	0	1	1	0	1	1	0	0	0
P8	0	1	1	1	0	0	1	0	1	0
P9	0	0	1	1	0	1	1	0	0	0

**Evaluation Criteria per Portfolio**

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Portfolio 0	86530	62840	3430	4440
Portfolio 1	86650	61360	3810	4290
Portfolio 2	83845	62385	3770	4320
Portfolio 3	78960	52190	4190	4205
Portfolio 4	79780	50995	4200	4175
Portfolio 5	82090	57905	3970	4285
Portfolio 6	89030	60280	3725	4420
Portfolio 7	88595	64625	3155	4300
Portfolio 8	85670	57580	4010	4330
Portfolio 9	79935	69110	3505	4410

Figura D.2: Pasos 2 a 7 del juego de diálogo.

Pasos 8 a 11

**Interaction Module**

User: (Challenge) Why is P0 a better option than P1?

Expert: Because P0 is better in the criteria 1 3 than the option P1.

User: (Pose Critical Question) Does the system have enough information to define a proof standard?

Expert: Yes, however, you can modify the status of a property if you want.

User: (Argue)

Expert: Veto activated

Interaction Options

Erase DM's Decision Challenge Accept

Update Question Why? Alternative

**Portfolio Composition (Portfolio/Alternative)**

	a0	a1	a2	a3	a4	a5	a6	a7	a8	a
P0	0	0	1	1	0	0	1	0	0	0
P1	0	0	1	1	0	0	1	0	1	0
P2	0	1	1	1	0	0	1	0	0	0
P3	0	1	0	1	0	0	0	0	1	1
P4	0	1	0	1	0	0	0	1	1	0
P5	0	0	1	1	0	0	1	0	1	1
P6	0	0	1	1	0	0	1	0	1	0
P7	0	0	1	1	0	1	1	0	0	0
P8	0	1	1	1	0	0	1	0	1	0
P9	0	0	1	1	0	1	1	0	0	0

**Evaluation Criteria per Portfolio**

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Portfolio 0	86530	62840	3430	4440
Portfolio 1	86650	61360	3810	4290
Portfolio 2	83845	62385	3770	4320
Portfolio 3	78960	52190	4190	4205
Portfolio 4	79780	50995	4200	4175
Portfolio 5	82090	57905	3970	4285
Portfolio 6	89030	60280	3725	4420
Portfolio 7	88595	64625	3155	4300
Portfolio 8	85670	57580	4010	4330
Portfolio 9	79935	69110	3505	4410

Figura D.3: Pasos 8 a 11 del juego de diálogo.

Pasos 12 a 16

**Interaction Module**

Expert: Veto activated

Expert: With this property modification, the proof standard currently on use is Simple majority + veto

Expert: In this proof standard, a veto threshold is applied during the Simple majority evaluation.

Expert: According to the new information, I recommend P6

User: (Challenge) Why is P6 the best option?

Expert: Because P6 is globally better than the other alternatives.

Interaction Options

Erase DM's Decision [dropdown] Accept

Update Question [dropdown]

Alternative [dropdown]

**Portfolio Composition (Portfolio/Alternative)**

	a0	a1	a2	a3	a4	a5	a6	a7	a8	a
P0	0	0	1	1	0	0	1	0	0	0
P1	0	0	1	1	0	0	1	0	1	0
P2	0	1	1	1	0	0	1	0	0	0
P3	0	1	0	1	0	0	0	0	1	1
P4	0	1	0	1	0	0	0	1	1	0
P5	0	0	1	1	0	0	1	0	1	1
P6	0	0	1	1	0	0	1	0	1	0
P7	0	0	1	1	0	1	1	0	0	0
P8	0	1	1	1	0	0	1	0	1	0
P9	0	0	1	1	0	1	1	0	0	0

**Evaluation Criteria per Portfolio**

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Portfolio 0	86530	62840	3430	4440
Portfolio 1	86650	61360	3810	4290
Portfolio 2	83845	62385	3770	4320
Portfolio 3	78960	52190	4190	4205
Portfolio 4	79780	50995	4200	4175
Portfolio 5	82090	57905	3970	4285
Portfolio 6	89030	60280	3725	4420
Portfolio 7	88595	64625	3155	4300
Portfolio 8	85670	57580	4010	4330
Portfolio 9	79935	69110	3505	4410

Figura D.4: Pasos 12 a 16 del juego de diálogo.

Pasos 17 al final del juego de diálogo.

**Interaction Module**

User: (Assert) But I prefer the solution P1 over the recommended solution.

Expert: Why?

User: (Argue) The budget for the recommended portfolio is close to the maximum budget limit, so I prefer P1.

Expert: The portfolio P1 has less than a 10% range from its value to portfolio P6.

User: (Accept)

----Final State: Accepted----

END OF THE EXECUTION

Interaction Options

Erase DM's Decision [dropdown] Accept

Update Question [dropdown]

Alternative [dropdown]

**Portfolio Composition (Portfolio/Alternative)**

	a0	a1	a2	a3	a4	a5	a6	a7	a8	a
P0	0	0	1	1	0	0	1	0	0	0
P1	0	0	1	1	0	0	1	0	1	0
P2	0	1	1	1	0	0	1	0	0	0
P3	0	1	0	1	0	0	0	0	1	1
P4	0	1	0	1	0	0	0	1	1	0
P5	0	0	1	1	0	0	1	0	1	1
P6	0	0	1	1	0	0	1	0	1	0
P7	0	0	1	1	0	1	1	0	0	0
P8	0	1	1	1	0	0	1	0	1	0
P9	0	0	1	1	0	1	1	0	0	0

**Evaluation Criteria per Portfolio**

	Criterion 0	Criterion 1	Criterion 2	Criterion 3
Portfolio 0	86530	62840	3430	4440
Portfolio 1	86650	61360	3810	4290
Portfolio 2	83845	62385	3770	4320
Portfolio 3	78960	52190	4190	4205
Portfolio 4	79780	50995	4200	4175
Portfolio 5	82090	57905	3970	4285
Portfolio 6	89030	60280	3725	4420
Portfolio 7	88595	64625	3155	4300
Portfolio 8	85670	57580	4010	4330
Portfolio 9	79935	69110	3505	4410

Figura D.5: Pasos 17 hasta el final del juego de diálogo.

# Anexo E

## Problema de introducción de la segunda etapa de evaluación

Durante la segunda etapa de evaluación, cada uno de los evaluadores trabajó con un par de problemas enfocados al problema de cartera de proyectos públicos aplicando toma de decisiones. En esta sección se muestra el primero de dichos problemas, el cual sirve como introducción tanto a SPP como al prototipo del sistema de recomendación implementado.

Problema de introducción

*Tú has ahorrado \$20000 de lo que has recibido por tu beca CONACYT, por lo que has decidido utilizar ese dinero para cubrir ciertas necesidades. Deseas hacer 5 alternativas de inversión:*

- *Laptop nueva - \$9000*
- *Computadora de escritorio nueva - \$7000*
- *Aire acondicionado para su cuarto - \$3000*
- *Reparaciones de su automóvil - \$10000*
- *Smartphone nuevo - \$2000.*

*Debido a que tu ahorro no te permite comprar todo lo que quieres, debes seleccionar cuál de ellos elegir tomando en cuenta cuatro criterios de igual importancia, donde ninguna de ellas se prefiere sobre la otra:*

- *Apoyo a su estudio, ya sea para realizar trabajo o transportarse más rápido.*
- *Satisfacción personal.*
- *Recreación fuera del estudio.*
- *Comodidad recibida.*

*Para apoyarte en tu decisión final, utilizarás un sistema experto que te proporcionará una recomendación, la cual podrás discutir con este.*

Las actividades que debes llevar a cabo son las siguientes:

**Paso 1.** Definir un valor entre 0-100 en la **tabla de desempeño**, en la cual evalúas que tanto aporta cada alternativa de inversión en cada criterio.

	Estudio	Satisfacción	Recreación	Comodidad
Laptop				
Computadora				
AC				
Automóvil				
Smartphone				

**Paso 2.** Con el fin de apoyarte, un sistema de optimización generó la **tabla de composición de carteras**, donde se colocó un 1 si la alternativa de inversión se lleva a cabo en esa combinación y 0 en caso contrario. En la última columna se escribió el total a pagar de esa cartera. Selecciona aquella o aquellas que te sean más atractivas.

	Laptop	Computadora	AC	Auto	Smartphone	Total
Cartera 1	1	1	1	0	0	\$19000
Cartera 2	1	1	0	0	1	\$18000
Cartera 3	1	1	0	0	0	\$16000
Cartera 4	1	0	1	0	1	\$14000
Cartera 5	1	0	1	0	0	\$12000
Cartera 6	1	0	0	1	0	\$19000
Cartera 7	1	0	0	0	1	\$11000
Cartera 8	1	0	0	0	0	\$9000
Cartera 9	0	1	1	1	0	\$20000
Cartera 10	0	1	1	0	1	\$12000
Cartera 11	0	1	1	0	0	\$10000
Cartera 12	0	1	0	1	1	\$19000
Cartera 13	0	1	0	1	0	\$17000
Cartera 14	0	1	0	0	1	\$9000
Cartera 15	0	1	0	0	0	\$7000
Cartera 16	0	0	1	1	1	\$15000
Cartera 17	0	0	1	1	0	\$13000

Cartera 18	0	0	1	0	1	\$5000
Cartera 19	0	0	1	0	0	\$3000
Cartera 20	0	0	0	1	1	\$12000
Cartera 21	0	0	0	1	0	\$10000
Cartera 22	0	0	0	0	1	\$2000

**Paso 3.** Hacer un archivo .txt que contenga la información de este problema, grabarlo como *introducción.txt* donde cada punto equivale a un renglón, excepto cuando se pida escribir una tabla. A continuación las indicaciones y un ejemplo de su escritura:

- 1) Tipo de problema, en este caso al ser de maximización, escribir “max”.
- 2) No. de pagos que se quieren realizar.
- 3) No. de criterios considerados.
- 4) Pesos normalizados de los criterios, en este caso se consideran iguales, escribir “0.25 0.25 0.25 0.25”.
- 5) Orden de preferencia, en este caso no se considera una preferencia específica, escribir “0 1 2 3”.
- 6) La tabla de desempeño siguiendo el siguiente formato:
  - a. En cada renglón escribir los valores de los criterios por posible pago, separando los valores por un espacio, Ej.
  - b. 30 50 80 60
  - c. 10 50 30 70
  - d. 90 80 100 10
  - e. 20 60 80 15
  - f. 100 10 100 10
- 7) El número de carteras obtenidas, en este caso tienen 22, escribir “22”.
- 8) Tope presupuestal, como se dispone de \$20000, escribir “20000”.
- 9) Umbral de veto, para concluir que una cartera es mejor que otra en un criterio, la diferencia debe superar 10. Escribir “10 10 10 10”.
- 10) La tabla de composición de carteras, seguir el mismo formato que se utilizó para escribir la tabla anterior.

## Ejemplo

```
max
5
4
0.25 0.25 0.25 0.25
0 1 2 3
30 50 80 60
10 50 30 70
90 80 100 10
20 60 80 15
100 10 100 10
22
20000
10 10 10 10
1 1 1 0 0 19000
1 1 0 0 1 18000
1 1 0 0 0 16000
1 0 1 0 1 14000
1 0 1 0 0 12000
1 0 0 1 0 19000
1 0 0 0 1 11000
1 0 0 0 0 9000
0 1 1 1 0 20000
0 1 1 0 1 12000
0 1 1 0 0 10000
0 1 0 1 1 19000
0 1 0 1 0 17000
0 1 0 0 1 9000
0 1 0 0 0 7000
0 0 1 1 1 15000
0 0 1 1 0 13000
0 0 1 0 1 5000
0 0 1 0 0 3000
0 0 0 1 1 12000
0 0 0 1 0 10000
0 0 0 0 1 2000
```

**Paso 4.** Ejecute el sistema de recomendación, cargando la instancia generada en el punto anterior.

# Anexo F

## Segundo problema de la segunda etapa de evaluación

Para la segunda etapa de evaluación cada evaluador realizó dos problemas tal como se mencionó en el Anexo E. En esta sección se muestra el segundo de dichos problemas, el cual incluye un cuestionario donde se le permitió a cada usuario calificar las diferentes características del sistema, dicho cuestionario está basado en el modelo de evaluación PSSUQ [Lewis, 1995] y con la adaptación realizada en [Zins et al., 2004].

Problema:

*Cuatro ciudades aledañas están planeando llevar a cabo un total de 25 proyectos sociales buscando mejorar la calidad de vida de los ciudadanos que viven ahí. Sin embargo, antes de presupuestar dichos proyectos se presentó un desastre natural, lo cual mermó gravemente los fondos de estas localidades. Debido a esto, se puede llevar a cabo solo un subconjunto de los proyectos; cada ciudad ofreció una lista donde se definía un cierto nivel de satisfacción para cada proyecto. Por otro lado, se contrató a un analista, el cual generó un conjunto de posibles combinaciones de los proyectos que se pueden llevar a cabo.*

**Nota:** El objetivo que te corresponde depende del grupo al que hayas sido asignado. Si se te asignó al grupo que va implementar una instancia sin preferencia, tu objetivo correspondiente es el objetivo sin preferencia. Si se te asignó al grupo que implementará una instancia con preferencia, tu objetivo correspondiente es el objetivo con preferencia.

- **Objetivo sin preferencia:** “Ahora necesitan saber cuál combinación apoyar, buscando satisfacer a la mayor cantidad posible de esas ciudades”.
- **Objetivo con preferencia:** “Ahora necesitan saber cuál combinación apoyar. Se decidió satisfacer principalmente a una de las cuatro ciudades debido a que es la que genera un mayor ingreso al estado”.



Las actividades que debes llevar a cabo son las siguientes:

**Paso 1.** Ejecutar el sistema de recomendación, cargando la instancia correspondiente a tu objetivo e iniciando el diálogo:

- **Objetivo sin preferencia:** instancia\_evaluacion\_max.txt
- **Objetivo con preferencia:** instancia\_evaluacion\_preferencia\_max.txt

**Paso 2.** Interactuar con el sistema, llevando a cabo un diálogo con el experto buscando encontrar una solución satisfactoria.

**Paso 3.** Al finalizar tu actividad con el sistema, realizar la evaluación mostrada en la siguiente página.

## Evaluación

A continuación, se presenta un conjunto de preguntas relacionadas a ciertos aspectos del programa que acabas de utilizar para evaluar su usabilidad. El puntaje funciona en una escala de 0 a 10 donde 0 es completa insatisfacción y 10 es satisfacción total, por favor responde con sinceridad. Después, si tiene algún comentario extra que hacer al sistema, apúntelo en la sección de comentarios.

Pregunta	Puntaje
Me agradó utilizar la interfaz del sistema.	
La organización de la información presentada por el sistema era clara.	
La interfaz fue agradable de usar.	
El sistema tiene las funciones y capacidades que esperaba.	
La información recuperada por el sistema me ayudo a terminar mis actividades de manera efectiva.	
Los proyectos seleccionados por la recomendación del sistema son convenientes para mi inversión.	
Encuentro la opción de forzado de recomendación útil.	
El sistema fue simple de utilizar.	
Fue fácil encontrar la información que necesitaba.	
La información de ayuda para el sistema proporcionada fue clara.	
En general, este sistema es fácil de utilizar.	
Fue fácil aprender a utilizar el sistema.	
La información dada por el sistema fue fácil de entender.	
Las explicaciones que proporciona el sistema en el diálogo facilitan la toma de decisiones.	
Considero que se requiere información previa sobre uso del sistema.	
Me sentí cómodo utilizando este sistema.	
Disfruté construir mi plan de inversión utilizando este sistema.	
En general, estoy satisfecho con el sistema.	
Pude completar mis tareas utilizando este sistema.	
Pude completar mis tareas rápidamente utilizando este sistema.	
Pude completar de manera eficiente mis tareas utilizando este sistema.	
Siento que podría volverme productivo rápidamente si utilizó este sistema.	
El sistema fue capaz de convencerme que las recomendaciones tenían un valor.	
Con mi experiencia actual utilizando el sistema. Creo que lo usaría de manera regular.	

**Comentarios:**

# Anexo G

## Instancias utilizadas en el juego de diálogo de ejemplo y en la segunda etapa de evaluación

En esta sección se mostrarán las instancias utilizadas para el ejemplo del juego de diálogo mostrado en el Anexo D, así como aquellas utilizadas durante la segunda etapa de evaluación, con la excepción de la instancia del problema de introducción, el cual ya se mostró en su anexo correspondiente (Anexo E).

La primera instancia fue utilizada tanto en el ejemplo de juego de diálogo mostrado en el Anexo D, como en el grupo de evaluadores que utilizaron una instancia sin preferencia en la segunda etapa de evaluación.

instancia\_evaluacion\_max.txt

```
max
25
4
0.05 0.36 0.39 0.20
0 1 2 3
3200 2000 165 300
6255 1640 385 390
5680 6940 270 445
8965 4195 355 415
6550 6560 315 440
6740 6290 150 350
9055 7165 375 485
4170 3015 410 285
9735 2860 480 330
3350 4210 400 315
8595 7270 150 265
9070 2430 455 360
9930 5825 420 385
4675 4505 425 490
8065 4030 165 425
7910 7665 240 320
9860 6265 415 350
3175 6240 225 445
9660 655 320 475
1150 4500 415 400
3245 5950 105 275
5350 6750 480 460
```

```

6050 2505 285 305
9190 2395 160 290
9615 4340 100 480
10
80000
1500 1200 75 75
0 0 1 1 0 0 1 0 0 0 1 1 1 1 0 1 1 1 0 0 0 0 0 0 1 79290
0 0 1 1 0 0 1 0 1 0 1 1 1 1 0 1 1 1 0 0 0 0 0 0 0 79575
0 1 1 1 0 0 1 0 0 0 1 1 1 0 0 1 1 1 0 0 0 1 0 0 0 79875
0 1 0 1 0 0 0 0 1 1 1 1 1 1 0 0 1 1 0 0 0 1 0 0 0 79525
0 1 0 1 0 0 0 1 1 0 1 1 1 1 0 0 1 1 0 0 0 1 0 0 0 79905
0 0 1 1 0 0 1 0 1 1 1 1 1 1 0 0 1 1 0 0 0 0 0 0 0 79885
0 0 1 1 0 0 1 0 1 0 1 1 1 0 0 0 1 1 0 0 0 1 0 0 1 79410
0 0 1 1 0 1 1 0 0 0 1 1 1 0 0 1 1 1 0 0 0 0 0 0 1 79585
0 1 1 1 0 0 1 0 1 0 1 1 1 0 0 0 1 1 0 0 0 1 0 0 0 79230
0 0 1 1 0 1 1 0 0 0 1 0 1 1 0 1 1 1 0 0 0 1 0 0 0 79825

```

La segunda instancia fue utilizada por el grupo de evaluadores, a los cuales se les pidió usar una instancia con preferencia durante la segunda etapa de evaluación.

instancia\_evaluacion\_preferencia\_max.txt

```

max
25
4
0.05 0.36 0.39 0.20
1 2 0 3
3200 2000 165 300
6255 1640 385 390
5680 6940 270 445
8965 4195 355 415
6550 6560 315 440
6740 6290 150 350
9055 7165 375 485
4170 3015 410 285
9735 2860 480 330
3350 4210 400 315
8595 7270 150 265
9070 2430 455 360
9930 5825 420 385
4675 4505 425 490
8065 4030 165 425
7910 7665 240 320
9860 6265 415 350
3175 6240 225 445
9660 655 320 475

```

1150 4500 415 400  
3245 5950 105 275  
5350 6750 480 460  
6050 2505 285 305  
9190 2395 160 290  
9615 4340 100 480

10

80000

1500 1200 75 75

0 0 1 1 0 0 1 0 0 0 1 1 1 1 0 1 1 1 0 0 0 0 0 0 1 79290  
0 0 1 1 0 0 1 0 1 0 1 1 1 1 0 1 1 1 0 0 0 0 0 0 0 79575  
0 1 1 1 0 0 1 0 0 0 1 1 1 0 0 1 1 1 0 0 0 1 0 0 0 79875  
0 1 0 1 0 0 0 0 1 1 1 1 1 1 0 0 1 1 0 0 0 1 0 0 0 79525  
0 1 0 1 0 0 0 1 1 0 1 1 1 1 0 0 1 1 0 0 0 1 0 0 0 79905  
0 0 1 1 0 0 1 0 1 1 1 1 1 1 0 0 1 1 0 0 0 0 0 0 0 79885  
0 0 1 1 0 0 1 0 1 0 1 1 1 0 0 0 1 1 0 0 0 1 0 0 1 79410  
0 0 1 1 0 1 1 0 0 0 1 1 1 0 0 1 1 1 0 0 0 0 0 0 1 79585  
0 1 1 1 0 0 1 0 1 0 1 1 1 0 0 0 1 1 0 0 0 1 0 0 0 79230  
0 0 1 1 0 1 1 0 0 0 1 0 1 1 0 1 1 1 0 0 0 1 0 0 0 79825