



**UNIVERSIDAD DE CHILE  
FACULTAD DE CIENCIAS FÍSICAS Y MATEMÁTICAS  
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**IMPLEMENTACIÓN DE UN MECANISMO DE ESTRUCTURACIÓN DE  
DECISIONES**

**MEMORIA PARA OPTAR AL TÍTULO DE INGENIERO CIVIL EN COMPUTACIÓN**

**CHRISTIAN ALEJANDRO NAVARRO GATICA**

**PROFESOR GUÍA  
JOSÉ PINO URTUBIA**

**MIEMBROS DE LA COMISIÓN  
ALEJANDRO HEVIA ANGULO  
HUGO MORA RIQUELME**

**SANTIAGO DE CHILE  
AGOSTO 2007**

# Resumen

---

La necesidad de tomar decisiones es algo fundamental para cualquier empresa, estas decisiones deben permitir que la empresa logre un desarrollo que la lleve a crecer. Esto lleva a una inevitable serie de reuniones, donde los miembros de la empresa puedan participar y exponer sus distintos puntos de vista para encontrar respuestas a las necesidades que se presenten.

El problema con esta necesidad es el uso del tiempo, que puede ser un recurso escaso en el funcionamiento de la empresa, lo que lleva a requerir reuniones que maximicen la calidad de los resultados obtenidos.

Uno de los puntos clave para lograr esto es una preparación cuidadosa de estas reuniones, tanto al definir los temas a discutir en forma precisa como al generar el material necesario para la reunión. Debido a esto se necesita un modo de representar esta preparación de un modo que todos los participantes puedan comprender y utilizar para comunicar sus aportes a la preparación de la reunión.

Existen distintos modelos para representar esta información, que definen en forma clara esta estructura, lo que obliga a los usuarios a comprender esta estructura y a la inevitable necesidad de aprender a expresar sus ideas con los elementos que presenta el modelo para este proceso. Esto provoca un problema, ya que obliga a los usuarios a salir de la forma usual de comunicación que se utiliza entre personas, que es rico al momento de comprender y expresar ideas, pero complejo al momento de ordenarlo.

Un sistema de reuniones distribuido presenta el beneficio de guardar información y tener acceso a ésta sin las barreras de la disponibilidad de tiempo, pero para ello se debe ahondar en los aspectos de la construcción de decisiones. Esto implica la necesidad de contar con un modelo de discusión, en el cual se pueda estructurar el proceso de construcción de decisiones.

Dentro de los modelos disponibles, IBIS es simple pero con una gran capacidad para expresar de manera sencilla distintas ideas utilizando solo tres tipos de elementos (Preguntas-Respuestas-Argumentos), pero el problema sigue siendo la rigidez que se presenta a los usuarios.

El mecanismo que se presenta en este trabajo evita este problema y guía a las personas en el proceso de convertir sus ideas a una estructura que siga el modelo IBIS y así lograr un material útil para la realización de la reunión.

A través de etapas, y la guía de personas entendidas en este modelo de discusión, se logra transformar la información del lenguaje que utilizan las personas para expresar sus posiciones, a una presentación de ideas en una forma organizada que puede ser útil para otras personas, utilizando los elementos del modelo IBIS.

# Agradecimientos

---

Muchas gracias a los que han estado conmigo, en las buenas y en las malas. Tanto ha pasado que sin ustedes no lo hubiera logrado.

A mis padres, Lidia y Alfonso, que me apoyaron siempre. A mis tíos, Ema y Jaime, que son mis padres también, a ellos les agradezco todo lo que me han dado y lo que me han enseñado. Sin ellos no sería lo que soy ahora.

A mis amigos, que todos estos años estuvieron allí, animándome para que no me rindiera. Gracias Marco, que sin tu constante apoyo jamás lo hubiese logrado. A mis compañeros de primer año, José y Wilfredo, que me ayudaron a sentir esta como mi casa. Y un agradecimiento a Emilio, que me ayudó a despertar en un momento complicado.

A mis hermanos, que sus recuerdos y sus alegrías ayudaron a calmar los malos momentos. A Carlos, quien, a pesar de la distancia, siempre estuvo conmigo. Y a Teresa, una mujer fuerte que me enseñó lo que es decisión.

Y en especial a mi abuela, Luna, que más que mi abuela era mi madre. A todos los años que estuvimos juntos, y todos los que estaremos después. Te quiero mucho.

Soy una persona afortunada, jamás he estado solo, sin importar lo que pasara.

A todos ellos dedico el logro de esta meta, que es el comienzo de lo que me espera, y el final de todo lo que me ha pasado.

# Índice

---

<b>CAPÍTULO 1: INTRODUCCIÓN</b> .....	<b>1</b>
<b>CAPÍTULO 2: MOTIVACIÓN</b> .....	<b>3</b>
<b>CAPÍTULO 3: OBJETIVOS</b> .....	<b>5</b>
3.1 OBJETIVO GENERAL .....	5
3.2 OBJETIVOS ESPECÍFICOS .....	5
<b>CAPÍTULO 4: CSAV</b> .....	<b>7</b>
4.1 HISTORIA DEL DESARROLLO DE CSAV .....	7
4.2 RESOLUCIÓN COOPERATIVA DE PROBLEMAS .....	9
4.3 DEMANDAS COGNITIVAS Y COMUNICATIVAS DE LA RESOLUCIÓN COLABORATIVA DE PROBLEMAS	12
4.3.1 Precondiciones .....	12
4.3.2 Estados en la Resolución de Problemas .....	12
4.4 MAPEO DE DIÁLOGOS .....	14
<b>CAPÍTULO 5: VISUALIZACIÓN DE DISCUSIONES</b> .....	<b>18</b>
5.1 CONSTRUCCIÓN VISUAL .....	19
5.2 PUNTOS IMPORTANTES EN LA CONSTRUCCIÓN DE DIAGRAMAS Y SU REPRESENTACIÓN .....	19
<b>CAPÍTULO 6: JAVASERVER FACES</b> .....	<b>22</b>
6.1 LAS APLICACIONES WEB Y JAVASERVER FACES .....	22
6.2 LOS BENEFICIOS DE LA TECNOLOGÍA JAVASERVER FACES .....	24
6.3 ¿QUÉ ES UNA APLICACIÓN EN JAVASERVER FACES? .....	25
6.4 CICLO DE VIDA DE UNA PÁGINA JAVASERVER FACES .....	26
6.4.1 Escenarios de ciclos de vida para procesar peticiones .....	26
6.4.2 Ciclo de procesamiento estándar para peticiones .....	27
<b>CAPÍTULO 7: CONSIDERACIONES DE LIBIS</b> .....	<b>29</b>
7.1 ARQUITECTURA .....	30
7.2 MECANISMOS DE DISCUSIÓN .....	32
7.3 MEMORIA GRUPAL .....	33
7.4 MECANISMOS DE COMUNICACIÓN .....	34
7.5 MECANISMOS DE VISUALIZACIÓN .....	34
<b>CAPÍTULO 8: USO DEL SISTEMA</b> .....	<b>35</b>
8.1 ROLES DENTRO DEL SISTEMA .....	35
8.1.1 Coordinador .....	36
8.1.2 Facilitador .....	37
8.1.3 Participante u Observador de la Pre-Reunión .....	38
8.2 PREPARACIÓN DE LA PRE-REUNIÓN .....	38
8.3 CONFIGURANDO LA PRE-REUNIÓN .....	39
8.3.1 Temas de la agenda .....	40
8.4 DESARROLLO DE LA DISCUSIÓN .....	40
8.4.1 Contribuir en el Proceso de la Discusión .....	40
8.4.1.1 Aportando a la Discusión .....	41
8.4.1.2 Identificación de las Contribuciones .....	43
8.4.1.3 Notificación y Awareness .....	44
8.4.2 Monitoreando la Discusión .....	44
8.4.3 Manejar el Desarrollo de la Discusión .....	44
8.4.3.1 Agregando Nuevos Participantes .....	45

8.4.3.2 Madurez de la Discusión .....	45
<b>CAPÍTULO 9: DESARROLLO DEL SISTEMA .....</b>	<b>46</b>
9.1 MODELO DE DATOS .....	46
9.1.1 Manejo de Usuarios y Permisos.....	48
9.1.2 Manejo de Pre-Reuniones y la Jerarquía de sus Elementos .....	49
9.1.3 Administración de las etapas de LIBIS.....	51
9.1.3.1 Primera Etapa: Discusión inicial de los participantes.....	53
9.1.3.2 Segunda Etapa: Preparación y discusión del modelo preliminar IBIS .....	53
9.1.3.3 Tercera Etapa: Obtención del modelo IBIS de la pre-reunión.....	54
9.2 DISEÑO DE LAS PÁGINAS.....	55
9.2.1 Navegación en el sitio .....	55
9.2.2 Despliegue de las opciones .....	59
9.3 HERRAMIENTAS DEL SISTEMA .....	61
9.3.1 Servicio de comunicación entre usuarios.....	61
9.3.2 Awareness.....	62
9.3.3 Revisión de Discusiones .....	64
9.3.4 Revisión del Estado de los Temas y el Paso a la Siguiete Fase .....	68
9.3.4.1 Estadísticas de Participación .....	69
9.3.4.2 Evolución de los Temas .....	69
9.3.5 Administración de Usuarios y Pre-Reuniones .....	69
<b>CAPÍTULO 10: CONCLUSIONES .....</b>	<b>70</b>
<b>CAPÍTULO 11: BIBLIOGRAFÍA .....</b>	<b>72</b>

# Capítulo 1: Introducción

---

En toda organización, la necesidad de reunirse es un factor muy importante que afecta su operación. Entre las reuniones que ocurren en la organización existen distintos tipos, y de entre estos tipos, las reuniones de decisión forman una parte fundamental en el proceso de crecimiento de la organización y el éxito de los proyectos. Esto muestra claramente lo importante que son estas reuniones y lo necesario que es en este proceso el apoyo computacional.

Para garantizar el éxito en estas reuniones, es decir, procurar que sean productivas, es importante evitar que la planificación de las mismas se lleve a cabo con una pobre preparación sobre los temas que se van a tratar en la reunión, ya que esto puede llevar a tomar decisiones apresuradas o muchas veces sesgadas a lo que piensan los integrantes con mayor rango dentro de la organización, impidiendo el surgimiento de ideas innovadoras que puedan existir que lleven a un mejor resultado.

Otro problema que surge de esta situación es la pérdida de tiempo, debido a la falta de antecedentes o a la poca maduración de las ideas presentadas, lo que provoca la necesidad de programar una nueva reunión.

De haber contado con una preparación mejor, este tiempo podría ser utilizado en otras necesidades de la organización. Además se evitarían los problemas provocados a los participantes al tener que organizar su tiempo nuevamente, lo que es un factor crítico en el caso de

organizaciones donde su personal está altamente distribuido, y por lo mismo, el costo para lograr una reunión “cara-a-cara” es muy alto.

Todo esto señala la necesidad de contar con una etapa anterior a la reunión, una pre-reunión donde los participantes puedan aportar ideas, participar en discusiones y hallar soluciones previas a la agenda de la reunión, lo que permitiría definirla en detalle y así evitar los costos de una mala preparación. Esta pre-reunión sería realizada en forma asíncrona y distribuida para evitar así los costos de una reunión común. Pero es crítico en esta etapa lograr una correcta comunicación y utilización del sistema para evitar la posterior toma de decisiones erróneas que provocarías más problemas.

Una forma de lograr una comunicación transparente, donde se logre una discusión estructurada y que permita la utilización posterior de esta, es la utilización de IBIS [1], que es un buen método para presentar los elementos de la discusión en una forma ordenada.

IBIS es un modelo que permite estructurar ideas en base a tres elementos y nueve relaciones entre ellos. El modelo va dirigido a representar los elementos cruciales de una discusión, permitiendo a las personas entender y facilitar contribuciones adicionales.

Estructuralmente, el modelo cuenta con pocos elementos, por lo que a primera vista parece simple, pero esta aparente simplicidad permite la interacción entre los usuarios sin la necesidad de un adiestramiento complejo, todo esto utilizando una herramienta de groupware. El problema que presenta este modelo es la posibilidad que las personas no encuentren una forma directa de expresar sus opiniones, que surgen en una típica discusión, lo que provoca dificultades al momento de participar en la pre-reunión a través del sistema.

Esto marca la necesidad de lograr una forma de acercar el modo habitual de una discusión entre personas con un modelo estructurado como IBIS, y así permitir a las personas expresar sus ideas a través del sistema.

El modelo que se busca implementar para solucionar esto es Layered IBIS (LIBIS), que es una propuesta donde se busca mantener la información de una discusión en tres capas. Desde la primera capa, se pretende dar a los usuarios la libertad de ingresar sus observaciones, preguntas, argumentos, etc., de una manera libre de las restricciones que impone IBIS, lo importante aquí es capturar las ideas de las personas que contribuyen en la preparación de la reunión. La segunda capa está destinada al proceso de las ideas ingresadas, este proceso está guiado por un “facilitador” que ayuda a las personas a ordenar todo lo previamente ingresado. En la tercera capa, el facilitador trabaja con las ideas ordenadas obtenidas de la capa dos, las personas no pueden participar de este proceso, pero pueden observar el desarrollo del mismo. Todo este proceso ayuda a capturar las ideas y preguntas de las personas sin el costo que implica el uso de IBIS para lograrlo, además de un aprendizaje por parte de éstas, durante el proceso, del uso de IBIS y sus ventajas.

## Capítulo 2: Motivación

---

Es difícil negar la necesidad de una buena preparación antes de realizar una reunión, sin esta preparación es probable no obtener los resultados esperados o peor aún, errar el camino en el desarrollo de un proyecto, más aún cuando esta etapa del proceso de la reunión es usualmente simplificada, ignorando las mejoras al resto del proceso de toma de decisiones que se pueden obtener con una buena preparación y lograr así una reunión de gran calidad. Para lograr esto ya se han definido modelos como SISCO [2], que es una extensión de modelo conversacional IBIS, y que apoya esta etapa de preparación.

Pero estos modelos presentan algunos problemas. Las experiencias con el modelo IBIS<sup>1</sup> han mostrado que los usuarios tienen algunas dificultades al intentar asociar sus ideas con los elementos que forman el modelo IBIS, es decir, a pesar que las personas pueden entender fácilmente el resultado de una discusión presentada en un modelo de argumentación del estilo IBIS, trasladar las ideas que surgen en una discusión entre personas al modelo no es un proceso directo.

---

<sup>1</sup> Explicado en detalle en el capítulo 4.



Para lograr una mejor utilización de estas herramientas es necesario contar con un método para obtener a partir de una discusión entre personas una discusión estructurada como preparación para una reunión.

Experimentos realizados con SISCO [3,4,5] muestran que IBIS no es intuitivo para las personas al momento de soportar la dinámica de una discusión. Esto es comprensible debido al hecho que las personas no piensan en forma lineal; las personas tienen “chispazos” de inspiración, pueden volver a revisar ideas anteriores, pueden cuestionar las ideas ya aceptadas, etc., por lo que tienen sentido todos los problemas que se presentan al enfrentar a las personas con la rígida estructura de IBIS, ya que sus elementos no corresponden con los pasos que siguen las personas en el proceso de generación de ideas en una discusión. Sin embargo, la utilización de IBIS presenta muchos beneficios al presentar una discusión con una estructura que permite su comprensión en forma más sencilla. El orden y la clasificación de los elementos ayudan a comprender mejor la discusión, más aún, la estructura de la discusión deja fuera los comentarios subjetivos que no aportan a la lógica de la discusión. Estos elementos pueden constituir una evaluación, pero las evaluaciones deben ser propuestas para la reunión y no discutidas en la pre-reunión, es en la reunión cuando todos los argumentos lógicos se han expresado.

## Capítulo 3: Objetivos

---

### 3.1 Objetivo General

El objetivo general de este trabajo es implementar un sistema que mejore la utilización del modelo IBIS para soportar un sistema de preparación de reuniones. Adicionalmente este sistema permite a las personas trabajar de una manera más simple en la pre-reunión para generar como resultado la información necesaria para la reunión siguiendo el modelo IBIS.

Para lograr esto es necesario ofrecer las funcionalidades a los participantes de la pre-reunión y al “facilitador” a través del modelo LIBIS y así obtener todas las preguntas y respuestas necesarias para planear la reunión y maximizar los resultados que se puedan sacar de esta.

### 3.2 Objetivos Específicos

- **Análisis de SISCO:** Estudio de SISCO como modelo de la utilización de IBIS para el sistema a implementar. este estudio busca definir las características utilizadas en SISCO y ver como pueden ser usadas dentro del proyecto para soportar una discusión distribuida y asíncrona, además de la información necesaria para soportar un sistema de estas características.
- **Modelo Conversacional:** Desarrollar un modelo conversacional conveniente a las características deseables, definidas a partir del estudio de SISCO.

- **Estudio de LIBIS:** A partir de las características de IBIS, estudiar el modelo por capas (LIBIS) que se utilizará en el desarrollo del proyecto. Ver sus diferencias y analizar las características que debe tener el sistema para que el facilitador pueda cumplir con su labor.
- **JavaServer Faces:** Estudiar la tecnología JavaServer Faces para así utilizar sus propiedades en el desarrollo de una aplicación web.
- **Desarrollo de la Aplicación LIBIS:** Utilizando lo aprendido, desarrollar una aplicación web que, utilizando las características de JavaServer Faces, permita a los usuarios trabajar en el desarrollo de la etapa de la pre-reunión. Esta aplicación web se llama SISDEL (SIStema de Decisiones LIBIS)

## Capítulo 4: CSAV

---

### 4.1 Historia del Desarrollo de CSAV<sup>2</sup>

Las raíces de la CSAV (Computer Supported Argument Visualization) están en la búsqueda de métodos para organizar argumentos y crear “mapas de argumentos”. Dentro de estos tenemos una mezcla de disciplinas de diversas áreas, entre ellas se incluyen la filosofía y la retórica como fondo de la argumentación en general.

Los campos de investigación en tecnologías centradas en los humanos como el trabajo cooperativo soportado por computadores (CSCW, Computer Supported Cooperative Work), comunicación mediada por computadores (CMC, Computer Mediated Communication), y aprendizaje colaborativo soportado por computadores (CSCL, Computer Supported Collaborative Learning) han desarrollado sus propias versiones de CSAV para soportar la coordinación de las actividades organizacionales distribuidas, la estructuración de contribuciones para sistemas de soportes de grupos, y la creación de conversaciones en las que el aprendizaje ocurre.

En la década de los 70, Horst Rittel caracterizó una clase de problemas que él nombró “difíciles” (en inglés, “wicked”, analizados posteriormente por Jeff Conklin [7]), en contraste a

---

<sup>2</sup> Esta sección usa como base lo descrito en [6].

los problemas “sencillos” (“tame”). Estos últimos no son necesariamente problemas triviales, pero en virtud de la madurez de ciertos temas, pueden ser enfrentados con más confianza. Estos problemas están suficientemente entendidos de modo que pueden ser analizados usando métodos establecidos, y es claro cuando se ha logrado encontrar una solución al problema. Debido a esto incluso es posible el uso de un análisis automatizado, como diagnóstico médico hecho por un sistema experto.

Los problemas “difíciles”:

- No pueden ser definidos fácilmente de modo que todos los participantes estén de acuerdo con el problema a solucionar.
- Requieren juicios complejos acerca del nivel de abstracción necesario para definir el problema.
- No tienen reglas claras para detener el proceso de resolución.
- Tienen soluciones “mejor que” o “peor que”, no correctas o incorrectas.
- No se cuenta con un método objetivo para medir el éxito de la solución.
- Requieren un proceso de iteración, ya que cada intento de encontrar una solución cambia el problema.
- A menudo tienen fuertes implicaciones morales, políticas o profesionales, particularmente en caso de falla.

Rittel y Webber [8], hicieron dos observaciones comprobables con directa relevancia a esta definición: primero, que muchos problemas de diseño son “difíciles”, en contraste a los problemas “sencillos” o benignos que pueden ser modelados computacionalmente, y en segundo lugar, que un “proceso argumentativo” es la forma más efectiva de enfrentar tales problemas.

Estos problemas que carecen de planes de acción únicos, formulados en consenso o bien desarrollados, son únicos, y no tienen definida una regla para saber si se encontró la solución, porque solo hay soluciones “mejor que” o “peor que”. Es por esto que el término de la búsqueda de una solución está forzado por condiciones prácticas más que por principios racionales o científicos. Estos problemas no pueden ser resueltos por modelos formales o metodologías, estos problemas están clasificados como de “primera generación” según Rittel. En lugar de esto, un acercamiento argumentativo a estos problemas fue propuesto (un método de diseño de segunda generación). La esencia de esta perspectiva es un proceso sin un final definido y dialéctico para definir y debatir problemas colaborativamente, esto es un poderoso camino para descubrir la estructura de los problemas “difíciles”.

Esta perspectiva motivó el desarrollo del modelo IBIS (Issue Based Information System) como un medio de ayudar e incentivar la deliberación abierta de los problemas. Las tres entidades principales de IBIS son Preguntas (Issues), Respuestas (Positions) y Argumentos (Arguments), las que pueden ser unidas por relaciones como *soporta*, *objeta*, *reemplaza*, *éxito temporal de*, *más general que*, y sus inversos. Visualizado como un grafo, un IBIS crece en una red a medida que más Preguntas son ingresadas y debatidas (Figura 4.1).

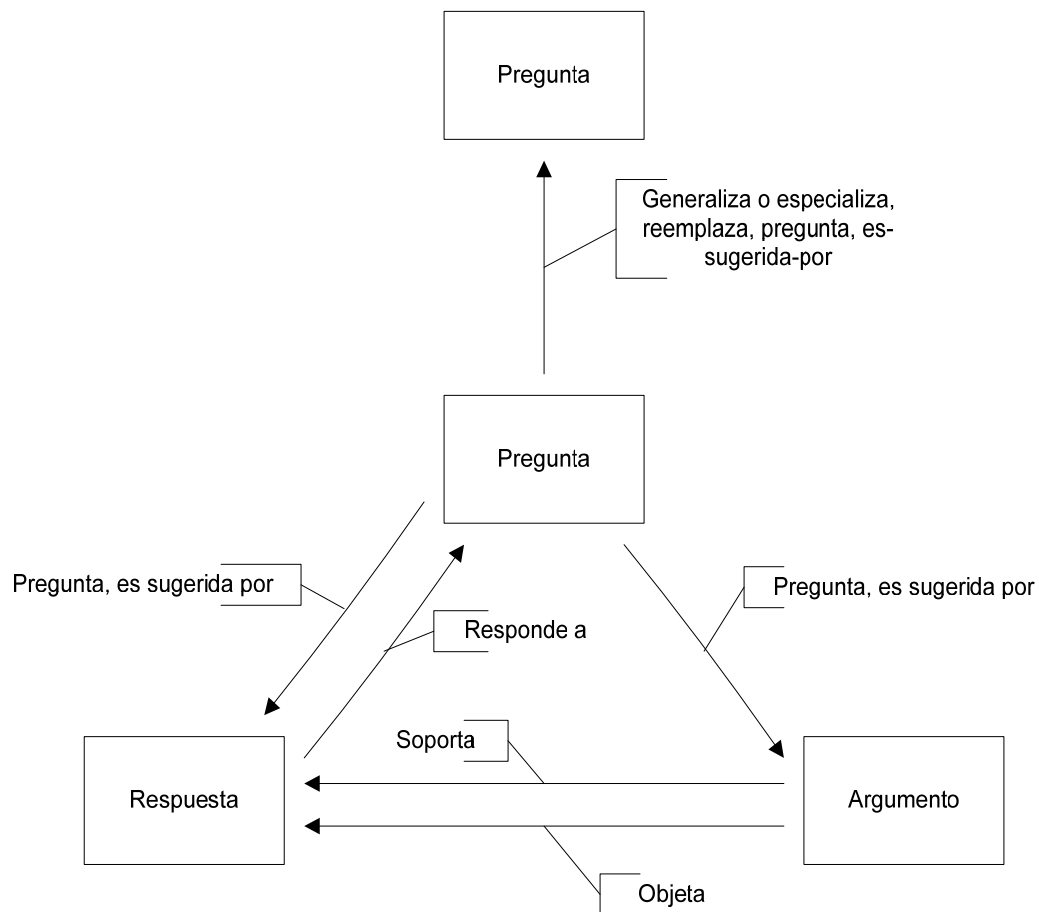


Figura 4.1 : Estructura de IBIS

En resumen, Rittel estableció una conexión entre el diseño y la argumentación. El método argumentativo para diseñar eleva la importancia del proceso de comprender un problema desde su estado inicial como un paso preliminar para usar métodos de diseño de primera generación, a la actividad central de abordar el diseño de problemas “difíciles”. Rittel y Webber, formularon la hipótesis que una forma particularmente poderosa de enfrentar tales problemas es a través de un “acercamiento argumentativo”, proponiendo a IBIS como una notación para representar los argumentos.

## 4.2 Resolución Cooperativa de Problemas<sup>3</sup>

CSAV es usado principalmente para solucionar problemas no estructurados, que son los opuestos a los problemas bien estructurados. Los problemas bien estructurados tienen:

- Una especificación completa y sin ambigüedades.
- Criterios y procedimientos claros para evaluar si se ha alcanzado la solución.

<sup>3</sup> Esta sección usa como base lo descrito en [7].

- La totalidad del conocimiento necesario para resolver el problema, que está representado en uno o más espacios, donde al menos uno de estos espacios representa el estado inicial del problema, existen los estados intermedios, y existe un estado final que representa la meta.
- Un conjunto asociado de operadores que pueden cambiar un estado del problema en otro estado.

Ejemplos de estos problemas son las *Torres de Hanoi*, el juego *Tic-Tac-Toe*, y los problemas tradicionales de los libros escolares.

El proceso para resolver este tipo de problemas a menudo sigue una progresión a través de tres etapas distintas:

- **Orientación:** Donde una representación del problema es construida.
- **Solución:** Donde los operadores son aplicados para transformar el estado del problema en el estado final o meta.
- **Evaluación:** Donde la solución y el uso de los operadores son evaluados.

Los problemas no estructurados (o poco estructurados) en cambio:

- Tienen una especificación ambigua e incompleta
- Una falta de un criterio claro para evaluar si se ha alcanzado una solución, lo que implica la inexistencia de reglas para detener el proceso.
- Hace uso de muchas fuentes de información potenciales que pueden ser usadas para representar espacios del problema, aunque no está claro cuales deben ser usadas y cómo deben ser integradas.
- No tienen un recuento de operadores que puedan ser aplicados, o un camino predeterminado desde el estado inicial a la meta.

Ejemplos de estos problemas son la composición de música, tareas de diseño y problemas de administración.

Cuando se resuelve un problema no estructurado, no se sigue un movimiento lineal a través de las etapas descritas, más bien se trabaja en soluciones parciales, regresos en el proceso para refinar la representación del problema, saltar a la evaluación de la solución parcial, revisar criterios, etcétera.

Para resolver un problema no estructurado, es necesario tener que representar el problema, definir las restricciones (constraints), aplicar operadores y evaluar las soluciones propuestas, pero dado que no se cuenta con nada de esto, primero es necesario que se dibuje el diseño sobre otras fuentes de conocimiento para determinar representaciones relevantes o beneficiosas, restricciones, operadores y criterios (estructuración del problema) y entonces justificar las decisiones tomadas. Es por este motivo que resolver problemas no estructurados es un proceso

argumentativo que requiere de razonamiento informal, el cual no es un razonamiento lógico o matemático. Este razonamiento informal requiere el uso de una estructura de los argumentos que consiste, al menos, de una petición con soporte. La evaluación de los argumentos no puede ser en términos de si los argumentos son correctos o incorrectos, pero requiere que la evaluación use otros criterios como la percepción de lo admisible de la demanda, la percepción del soporte que la razón ofrece a la demanda, y la calidad de la argumentación juzgada tomando en cuenta los contra-argumentos. Aquí, al igual que en las restricciones del problema, los individuos pueden diferir en la evaluación de la relevancia o importancia.

En una serie de investigaciones, dirigidas por Voss y sus colegas [9, 10], se aplicó una combinación del modelo de procesos de Newell y Simon (1972) [11] y el concepto de razonamiento informal a problemas en diversos campos como ciencias sociales, medicina, matemáticas y política exterior. En 1983, Voss analizó los protocolos de expertos tratando de encontrar una forma de incrementar la producción de maíz en la Unión Soviética [10]. El proceso de resolución del problema fue descompuesto en:

- **Etapas de Representación del Problema:** Donde las preguntas caracterizaron la naturaleza del problema.
- **Etapas de Solución del Problema:** Donde resolvieron el problema.

También en los análisis se distinguió entre una:

- **Estructura de Solución del Problema:** Estructura con operadores asociados para solucionar el problema.
- **Estructura del Razonamiento:** Un conjunto de operadores informales de razonamiento.

Esta línea de la investigación estudió a los expertos y principiantes que solucionaron individualmente problemas no estructurados. El análisis de sus protocolos de pensamiento utilizó un sistema de operadores de razonamiento característicos del dominio de conocimientos de las personas involucradas. El modelo subyacente en esta investigación puede ser descrito como “un agente, un modelo de representación”. Pero este modelo necesita ser extendido para incluir situaciones en las cuales hay más personas participando en la resolución del problema, las que no comparten la misma representación ni los mismos operadores para enfrentar el problema, lo que define un nuevo modelo que implica tener múltiples agentes con múltiples representaciones.

Esta idea de múltiples agentes con múltiples representaciones estaba implícita en la perspectiva de resolución de problemas originada por Rittel y el término “problema difícil”, ya visto antes.

Los problemas “difíciles” están compuestos de un conjunto de “preguntas” y “restricciones”, más que una declaración definitiva del problema. Estos problemas a menudo no son completamente entendidos hasta que se desarrolla una solución. Hay muchos participantes del proceso que tienen un gran conocimiento en diferentes aspectos del problema a ser resuelto, haciendo la resolución efectiva del problema un proceso social más que un proceso cognitivo. Es más importante obtener una solución que sea aceptada por todos los participantes del proceso de decisión que una solución correcta. Por otro lado, la descripción de estos problemas tiene



implícita ciertas suposiciones, a menudo políticas. De acuerdo a lo expresado por Rittel, los problemas “difíciles” son inherentemente multidisciplinarios, involucrando a participantes de diferentes ámbitos, y no existe un monopolio en la autoridad. Así, la resolución de estos problemas implica diferentes actores (participantes) que entregan diversas representaciones del problema, y ninguno de los participantes domina la definición del problema o su estructura. Para enfrentar estas dificultades, Rittel desarrollo IBIS. Algo esencial en el método IBIS es la ausencia de una estructuración prematura del problema, la estructura de la argumentación consiste de un número de tópicos (preguntas) y respuestas que los participantes mantienen con respecto a estas preguntas y no hay medios en el método de, por ejemplo, crear la descomposición de un problema.

### 4.3 Demandas Cognitivas y Comunicativas de la Resolución Colaborativa de Problemas<sup>4</sup>

#### 4.3.1 Precondiciones

Es necesario nombrar algunas precondiciones más o menos sociales subyacentes en la resolución de problemas colaborativos, es decir, ciertos niveles mínimos de comprensión compartida, la responsabilidad y la confianza que debe existir antes de que cualquier resolución colaborativa de problemas pueda ocurrir.

La primera precondición es un mínimo de comprensión compartida. Esta es el estado donde dos o más personas tienen expectativas equivalentes sobre una situación, esto significa que sus explicaciones de la situación y sus predicciones de cómo puede ser el desarrollo son las mismas.

La segunda precondición es la responsabilidad. La responsabilidad es el mecanismo social subyacente responsable del comportamiento de la gente, es decir, que un miembro del equipo no plagie a un compañero del equipo, tome el crédito por el trabajo realizado por otro compañero del equipo o trabaje para perjudicar a un miembro del equipo. Esta precondición no implica que los agentes necesariamente tengan las mismas metas.

La tercera y última precondición es la confianza. La confianza es la capacidad percibida de confiar en el carácter, capacidad, fuerza o veracidad de alguien o algo y es el factor decisivo en un proceso social que da lugar a una decisión de un individuo para aceptar o rechazar un riesgo basado en las expectativas de otro grupo que resolverá los requisitos de funcionamiento.

#### 4.3.2 Estados en la Resolución de Problemas

Las investigaciones en resolución de problemas distinguen tres estados generales en la resolución de un problema:

1. **Orientación:** Para determinar cual es el problema.

---

<sup>4</sup> Esta sección usa como base lo descrito en [16].

2. **Resolución del Problema:** El cual determina el proceso para resolver el problema.
3. **Evaluación:** El cual determina si el problema ha sido resuelto y si el problema que ha sido resuelto es el “correcto”.

Hay que notar que cada estado necesita información que a menudo falta en los problemas no estructurados o “difíciles”. En estos problemas, la descripción y necesidades del problema son ambiguas en el mejor de los casos, los operadores que pueden ser aplicados no son entregados, y los criterios para evaluar las soluciones están ausentes. Este proceso lineal a través de las tres etapas es sólo posible cuando se enfrentan problemas bien estructurados usando algoritmos bien conocidos. Cuando uno se enfrenta a problemas no estructurados, el proceso se mueve atrás y adelante entre las etapas de orientación, resolución del problema y evaluación. Probar una solución parcial, por ejemplo, puede ayudar a comprender mejor el problema y puede conducir a agregar o refinar las necesidades y criterios.

En el estado de orientación, los participantes estructuran el problema. En este estado ellos aclaran la descripción del problema, las necesidades y el criterio para la solución y la evaluación. Los participantes se preguntan si el problema es el que comenzaron a resolver (nivel de superficie) o el problema real yace más profundamente, escondiéndose tras el problema que ellos ven. Los participantes también necesitan clarificar, y a menudo definir, las condiciones y necesidades límites, el(los) contexto(s) en los que el problema está sumergido, etc. Usando los resultados de las soluciones intermedias y parciales, los encargados de solucionar el problema repasarán y revisarán su comprensión del problema, sus necesidades y los criterios para evaluar una solución.

En el estado de resolución, los participantes planifican como solucionar el problema y entonces intentan ejecutar la solución de una forma más o menos sistemática. Esto es a menudo un proceso conducido por la información con que se cuenta, con una estrategia de control relajada que permite a las personas del equipo moverse atrás y adelante entre soluciones parciales, estructuración del problema y evaluación.

En el estado de evaluación, tanto el proceso como el resultado serán comprobados, es decir, los participantes del proceso no sólo revisarán la correcta aplicación de los operadores, sino también determinarán si el “problema como fue planteado” ha sido resuelto. Dado que no es un criterio claro evaluar las soluciones o la correcta aplicación de operadores en problemas no estructurados, es necesario definir, repasar y revisar sus criterios.

Las fases de orientación y resolución del proceso parecen las más cruciales para la resolución colaborativa del problema porque la representación del problema que fue realizada en la fase de orientación esta fuertemente ligada al tipo de razonamiento argumentativo aceptable que puede ser usado en la fase de resolución. En equipos multidisciplinarios, las diferencias de interpretación acerca de “cuál es el problema” y de “cuál debería ser la estrategia de resolución del problema” pueden llevar a grandes dificultades si los diferentes puntos de vista no convergen en una solución. Es aquí donde la argumentación es más útil y donde el diseño y uso de CSAV juega un papel importante.

Las demandas cognitivas y comunicativas varían a través de los estados de resolución del problema y estas variaciones tienen importantes consecuencias para el diseño de ambientes CSAV. En su primer esfuerzo para estructurar un problema, las personas a menudo tienen diferentes o solo representaciones parciales del problema que se superponen para trabajar en al menos un mínimo espacio de comprensión compartido del problema. La comunicación entre las personas es relativamente sin un foco definido y basada en las preguntas que nacen en el desarrollo del problema, abandonar la sala para explorar las interpretaciones del problema, las necesidades, etc.

Otra pregunta es si el uso de herramientas de visualización de argumentos en esta etapa facilita el proceso de resolución del problema. Cualquiera que sea la preferencia, si representaciones externas son usadas en las primeras etapas, entonces es necesario que la representación de la argumentación sea sin consignación (non-committal), es decir, la notación de representación debe permitir la expresión de diversas perspectivas en el problema.

#### 4.4 Mapeo de Diálogos<sup>5</sup>

En el corazón de la técnica de mapeo de diálogos está el sistema de argumentación IBIS. El poder de IBIS proviene de tres características:

- Convierte pensamientos complejos en mapas analíticos estructurados.
- Está basado en preguntar las preguntas “correctas”.
- Es lo suficientemente simple e intuitivo para ser aprendido y usado con dificultades cognitivas indirectas relativamente bajas.

IBIS captura pensamientos complejos en diagramas. No hay límites para el tamaño o la información que guardan estos mapas IBIS. Una variedad de enlaces y mecanismos de navegación provee a los usuarios con diversas formas de organizar grandes cantidades de información. Una de las propiedades más básicas es que un mapa puede contener otros mapas, de manera que la información puede ser lógicamente agrupada. A menudo, la señal para un nuevo grupo, un nuevo mapa, es una nueva pregunta de importancia. Este nuevo sub-mapa puede ser transformado en un hyperlink y colocado en el mapa superior.

A diferencia de la mayoría de las técnicas de diagramación, IBIS tiene una gramática, que impone una estructura formal (aunque simple) en los mapas IBIS. Esta gramática impone una disciplina en los mapas de IBIS, con dos consecuencias complementarias:

- Es más difícil aprender a usar IBIS que otras técnicas libres de gramática.
- Los mapas IBIS son más rigurosos, más fieles a patrones, y más reproducibles.

Para el análisis de situaciones complejas, la estructura adicional y el rigor de IBIS crea una importante base de orden y reproductibilidad.

---

<sup>5</sup> Esta sección usa como base lo descrito en [17].

IBIS invoca una disciplina de encontrar la “pregunta” correcta. Las preguntas son el corazón de este método. Uno de los beneficios de hacer “preguntas” en una situación de trabajo colaborativo es que puede ayudar a romper la “respuesta refleja”, en la cual los participantes intercambian respuestas sin nunca estar de acuerdo en qué pregunta están respondiendo. También, la gramática de IBIS permite que una pregunta se refiera a cualquier otro elemento de IBIS, de modo que es a través de realizar nuevas preguntas que el mapa de IBIS crece. Finalmente, hay un pequeño número de tipos de preguntas en IBIS, cada tipo espera diferente tipo de respuesta, y estos tipos se conectan juntos en patrones de un nivel más alto, o “templates”. Estos patrones proveen un orden adicional y una reproductibilidad en los mapas IBIS.

Otra importante propiedad de la gramática de IBIS es que es simple e intuitivo, numerosos estudios han mostrado que las potenciales ventajas de las técnicas de estructuración retórica como IBIS son compensadas a menudo por los crecientes “gastos cognoscitivos indirectos” de aplicarlos cuando están relacionadas a una tarea substantiva. En parte, estos gastos indirectos imponen una necesidad ante el poder expresivo de cualquier formalismo que se pudo adoptar en el proceso de análisis: notaciones candidatas deben ser lo suficientemente simples e intuitivas como para que el “costo cognitivo” de usarlas sea bajo. Las experiencias prácticas a través de los años han mostrado que notaciones más sofisticadas y poderosamente expresivas que IBIS han resultado ser mucho más difíciles de aplicar de una forma transparente.

Aún así, distinguir los tres elementos básicos de IBIS – Preguntas, Respuestas y Argumentos – puede resultar difícil, este proceso se puede hacer más sencillo con práctica, y así lograr que las personas trabajen con ellos de forma muy natural.

Hay que agregar lo simple y natural que es esta notación, también es importante convenir el poder de IBIS como una notación de mapas para análisis complejos. Diferentes participantes tienen diversas ideas acerca de lo que es una “pregunta” (“issue”), especialmente cuando se enfrentan a una situación “difícil” (“wicked”). Cada participante realiza sus comentarios desde su versión del problema, pero muy a menudo no está claro cuantas versiones del problema hay, o cuáles son. Haciendo al problema algo explícito en un mapa IBIS, como preguntas, agrega una gran claridad a la discusión, así como clarifica a cuál de los problemas está relacionado un comentario dado. Los problemas “difíciles” usualmente tienen involucradas docenas de preguntas relacionadas entre sí, pero la memoria de corto plazo en los humanos es limitada, de modo que a menos que uno esté extremadamente familiarizado con toda la información relacionada a todos los problemas, la exploración sin ayuda de todas estas preguntas resulta muy confusa. El poder de IBIS como notación es que organiza todas las preguntas, respuestas, información y suposiciones de manera que todos los participantes tienen el mapa de preguntas como punto de referencia, y se pueden referir a él en lugar de intentar mantenerlo todo en sus cabezas. De esta forma IBIS contribuye al mapeo de diálogos al ser una “ayuda cognitiva” porque aumenta la cognición humana en uno de sus puntos más débiles: los límites de la memoria de corto plazo.

El mapeo de diálogos es una técnica para ayudar a los participantes en la que el Facilitador captura la discusión del grupo en tiempo real en un mapa IBIS. El mapeo de diálogos puede ser realizado en un pedazo de papel, en una pizarra, o usando un programa como QuestMap [12].

El rol del Facilitador tiene dos roles que lo componen: un rol incluye acompañar al grupo facilitando el trabajo de manera tradicional (es decir, mirando el reloj y cuidando al grupo que trabaja), y el otro rol, también conocido como “tecnógrafo”, enfocado en escuchar y escribir en la

computadora. Aunque una sola persona puede realizar ambos roles, a menudo es preferible si dos personas trabajan juntas, una como “Facilitador” y la otra como “Tecnógrafo”.

En el mapeo de diálogos el grupo activa y colaborativamente participa en la creación del mapa de IBIS. Un punto central en el mapeo de diálogos es la noción de un “despliegue compartido” (shared display), un despliegue proyectado por computadora de una aplicación de hipertexto que muestra el mapa del conocimiento y razonamiento del grupo, mostrándose en tiempo real.

El mapeo de diálogos funciona debido a la habilidad del Facilitador. El trabajo del Facilitador es servir como un enlace discreto entre la conversación del grupo y el mapa compartido. Si los miembros del grupo están solo conversando sin mirar el mapa compartido, entonces el Facilitador no está realizando su trabajo. Un trabajo hábil por parte del Facilitador significa escuchar los comentarios del grupo y capturarlos en el mapa de forma tal que clarifique los significados y agregue un valor a la conversación. Es de este modo que el grupo aprende a mirar el mapa creado por el Facilitador porque, cuando no lo hace, pierde el curso de la discusión o se mueve en círculos.

El Facilitador está a cargo de escuchar, y asegura así que hay una alta calidad de la comunicación presente en la sesión. En esencia, el proceso del Mapeo de Diálogos enfoca al grupo en la construcción de un modelo colectivo de su problema. Con este modelo visual evolucionando mientras hablan, las discusiones giran alrededor de convenir con el modelo, el cual puede, por supuesto, incluir el convenir en puntos de desacuerdo.

Por supuesto, a veces el grupo no presta mucha atención al mapa, y ocasionalmente el grupo quiere que el mapa cambie más rápidamente, o que vaya en una dirección diferente. Como con cualquier tipo de intervención en el grupo, hay muchos factores y parámetros que contribuyen al éxito o fracaso de una sesión de Mapeo de Diálogos. Una importante habilidad que un Facilitador debe aprender es mantener al grupo enfocado en el mapa y mantener la posesión del grupo sobre el mapa, es decir, mantener una relación estrecha del grupo con el mapa. Otra habilidad crítica es la capacidad de escuchar y escribir al mismo tiempo.

Debido a que en el Mapeo de Diálogos se trata de lograr hacer de forma transparente un mapa del pensamiento del grupo, a veces el Facilitador sólo escucha, escribe y edita el mapa. En otros casos, es necesaria una mayor intervención, incluyendo una guía en el proceso de discusión como sugerir tópicos o llamar a descansos en la discusión. El Facilitador debe detectar las necesidades del grupo y hacer juicios sobre qué clase de intervención debe efectuar y la dirección de esta, si es necesario, sea apropiada. El Mapeo de Diálogos no funciona en todo tipo de reunión o con cualquier tipo de grupo, a veces el mapa termina siendo nada más que una lujosa minuta de reunión.

Un problema común en el trabajo colectivo es decidir a qué nivel de detalle o abstracción necesita enfocarse el grupo. El mapeo de diálogos soluciona este problema al proveer un marco de trabajo de “puro contenido” – un modelo semiestructurado – para el problema. Los participantes pueden mantenerse enfocados en el nivel de contenidos porque está representado rígida y gráficamente en el mapa compartido. Más aún, desacuerdos sobre el contenido no son simples discusiones verbales, el Facilitador trabaja con el grupo para traer las mejores ideas de los participantes para desplegar las opciones y sus respectivos argumentos, y quizás introducir nuevas opciones. Mientras este distanciamiento emocional no hace que el debate desaparezca,

elimina la belicosidad innecesaria y permite aclarar las ideas para negociar nuevos significados y formular nuevas opciones. En muchos casos, un claro “ganador” emerge simplemente a través del proceso de diálogo abierto, y aún cuando esto no ocurra, el modelo del mapeo de diálogos presenta las opciones y las compensaciones claramente a la persona encargada de decidir.

## Capítulo 5: Visualización de Discusiones<sup>6</sup>

---

En esencia, la pregunta es como organizar el conocimiento colectivo y distribuido. No se busca reemplazar la narrativa textual como un medio de expresión. Lo que se busca es aumentar estos medios al explotar la información global en línea de una forma que las publicaciones tradicionales no pueden lograr.

Un ejemplo de esto son las publicaciones escolares, en las que la forma en las que están pensadas es como una co-evolución de notaciones con la tecnología de impresión, pero no están diseñadas en ningún caso para tomar ventaja de la infraestructura que presenta la información de hoy en día.

Los investigadores se benefician de esta organización, que permite un acceso más rápido a los recursos de documentos de investigación tales como nuevas bibliotecas digitales, pero los investigadores (como casi todas las profesiones) también están encontrando problemas, con menos tiempo para buscar un creciente número de conferencias, periódicos y reportes. Pero más allá de rastrear nuevos resultados, hay toda una nueva dimensión de análisis de la literatura.

---

<sup>6</sup> Esta sección usa como base lo descrito en [18].

Para lograr una solución a esto, hay que definir las preguntas que los investigadores, en este caso, están interesados en responder. Por supuesto que hay preguntas acerca del significado de la contribución de una investigación. Estas preguntas operan a un diferente nivel de las que son tratadas por metadatos convencionales, lo que normalmente busca sacar la inconsistencia, ambigüedad y lo incompleto.

La utilización de formularios y menús en el diseño para una aplicación web es algo relativamente definido. Aunque este diseño soporta la construcción de demandas y argumentos a un nivel técnico, y la actividad web basada en menús y la utilización de formularios es familiar, es muy útil explorar una interfaz de usuario complementaria, que permita un uso más intuitivo. Hacer demandas es esencialmente un modelado de literatura, una tarea cognitiva que requiere una construcción visual de una estructura de red. Modelar esto a través de mapas de conceptos es la forma más intuitiva para manejar la carga cognitiva, y que lleva típicamente a un refinamiento iterativo del modelo, del nombre y el tipo de los conceptos, de los tipos de conexiones (links), y de la granularidad.

## 5.1 Construcción Visual

Es necesario responder qué significa representar las contribuciones importantes, y las conexiones de estas como una red explícita de conceptos y argumentos asociados. Además definir la interfaz del usuario, de manera que pueda proveer una forma de representar esta estructura, y responder a las dificultades que se presenten en el proceso.

Uno de los puntos importantes es la disposición que tendrá el diagrama. En el de la ejemplo de la Figura 5.1<sup>7</sup> se adopta una disposición vertical, con el concepto primario arriba (el concepto “primario” depende de la lectura que se haya hecho del artículo, diferentes lectores pueden producir distintos mapas).

Otro ejemplo (tomado de [14]) se muestra en la Figura 5.2, allí se ilustra no solo demandas individuales (concepto-unión-concepto), sino también estructuras demanda-unión-demanda (resaltadas), en otras palabras, argumentos apoyando otros argumentos.

## 5.2 Puntos Importantes en la Construcción de Diagramas y su Representación

En cualquier acercamiento a la construcción de mapas de argumentos (en este caso, para cualquier acercamiento a la construcción de modelos de conceptos) hay siempre decisiones que tomar para la representación de nombres, objetos, clasificaciones, y asociación de objetos, manejar la coherencia de la estructura general, y trabajar en un nivel apropiado de granularidad.

Ser capaz de tomar estas “meta-decisiones” es una habilidad crítica para logra la creación de un mapa de los argumentos en forma colaborativa y en tiempo real, dados el tiempo y las presiones del grupo para mantener el impulso del proceso y la velocidad para capturar la información que lleva hacia notaciones “simples” como IBIS. Esta presión es menos intensa en

---

<sup>7</sup> EL modelado fue hecho con una adaptación de la herramienta Mifflin para argumentación IBIS.



un contexto de uso donde se trabaja en un modo más reflexivo, probablemente (aunque no necesariamente) en un modo propio, destilando la esencia de un trabajo en un mapa.

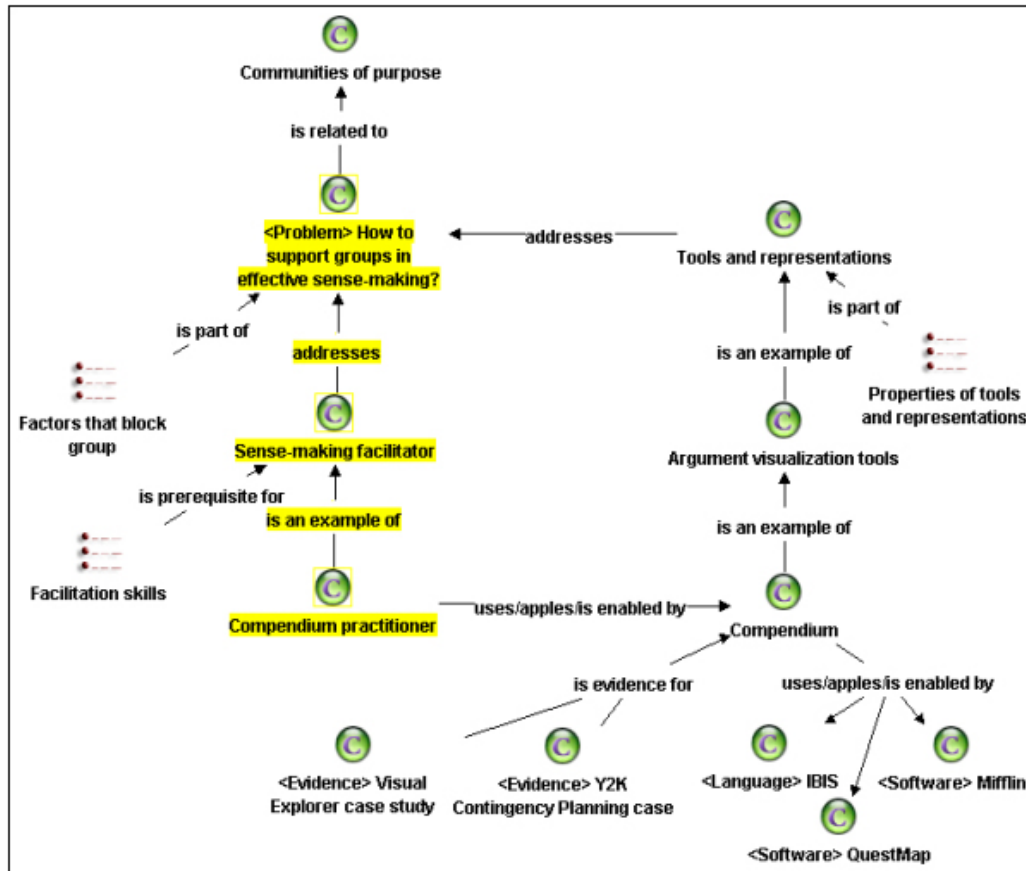


Figura 5.1 : Analisis visual de demandas para [13]

Es por esto que en un caso, donde la necesidad de soportar la interacción en tiempo real no existe, se puede utilizar notaciones más ricas, que permitan expresar las ideas en forma más clara, lo que ofrece opciones más expresivas.

Una preocupación persistente, en el diseño de esta notación, es el peligro de sobrecargar al usuario. Los elementos que tengan diferencias sutiles pueden provocar problemas al momento de usar la notación, a manos de un usuario que no sea capaz de diferenciarlos, o a la posibilidad de frustrarlo a manos de una notación con un vocabulario pequeño, con el cual no logren expresarse.

Volviendo al tema de la visualización, en el caso de las demandas, el uso de un ambiente abierto y conectado a una red con conceptos “vivos” que pueden ser usados por numerosos investigadores, coloca como un punto importante reutilizar los conceptos y demandas cada vez que sea posible.

Esto puede ser relativamente simple para conceptos concretos, como los nombres de teorías específicas, métodos, algoritmos, herramientas de software, etc. Una búsqueda rápida puede revelar el concepto si este fue creado, así el usuario puede revisarlo. En contraste, ideas complejas estarán ligadas a demandas o conjuntos de demandas de una estructura impredecible.

La búsqueda de una palabra clave puede mostrar un buen candidato para “re-usar”, o el conocimiento de un investigador en el campo de la búsqueda puede llevarlos a un documento que este investigador conoce, cuya búsqueda puede revisar, re-usar o adaptar para resolver la suya. Un grupo de investigadores puede publicar una biblioteca de conceptos y demandas representando sus publicaciones más importantes, recomendándolas para citar su trabajo, y así otros pueden usar esta biblioteca. Estos son escenarios que podemos considerar, pero que dependen de la interacción de las tecnologías y su adopción.

Otra decisión que debe ser tomada, para una representación adecuada, es como presentar la estructura. Podemos dar una pantalla en blanco para que el usuario presente sus argumentos como él desee, o dar una librería de *templates* para llenar los espacios en blanco, para géneros canónicos de ensayos en un campo dado, esto puede ayudar a los principiantes dándoles una guía para presentar sus ideas.

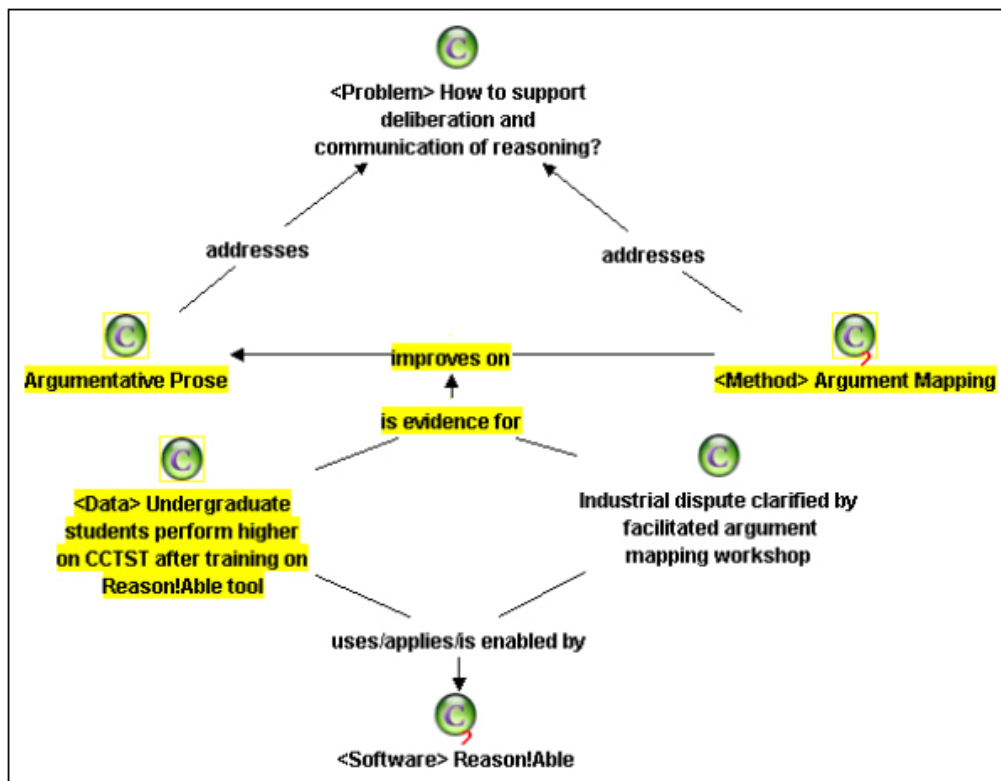


Figura 5.2 : Analisis visual de demandas para [14]

# Capítulo 6: JavaServer Faces<sup>8</sup>

---

## 6.1 Las Aplicaciones Web y JavaServer Faces

La tecnología JavaServer Faces es un framework para la interfaz de los usuarios utilizada para construir aplicaciones web que se ejecutan en un servidor Java y que despliegan la interfaz del usuario (User Interface o UI) de vuelta al cliente.

Los principales componentes de la tecnología JavaServer Face son:

- Unas APIs y una implementación de referencia para:
  - Representar los componentes UI y manejar su estado.
  - Manipular eventos, validaciones desde el servidor, y conversión de datos.
  - Definir la navegación en las páginas.
  - Soportar la internacionalización y accesibilidad.

---

<sup>8</sup> Esta sección usa como base lo descrito en [19].

- Proveer extensibilidad para todas estas características.
- Una biblioteca para JavaServer Pages (JSP) de “custom tags” para expresar los componentes UI dentro de las páginas JSP.

Este modelo de programación bien definido y la biblioteca de tag para los componentes UI que facilitan significativamente el trabajo de construir y mantener una aplicación Web con una UI localizada en el servidor. Con esto se logra que efectuando un mínimo esfuerzo se puede conseguir:

- Conectar eventos generados en el cliente a código de las aplicaciones localizado en el servidor.
- Mapear componentes UI en una página a datos localizados en el servidor.
- Construir una UI con componentes reusables y extensibles.
- Guardar y restaurar el estado de los UI más allá del tiempo de vida de los requerimientos al servidor.

Como se muestra en la Figura 6.1, la interfaz del usuario que se crea con la tecnología JavaServer Faces (representada por miUI en la figura) corre en el servidor y se despliega en el cliente.

La página JSP, `miformulario.jsp`, expresa los componentes de la interfaz del usuario con tags especiales definidos por el framework de la tecnología JavaServer Faces en vez de ser codificados por el desarrollador. La UI para las aplicaciones Web (representadas por miUI en la figura) maneja los objetos referenciados por la página JSP. Estos objetos incluyen:

- Los objetos componentes que mapean los tags en la página JSP.
- Los capturadores de eventos, validadores, y los convertidores que están registrados en los componentes.
- Los objetos del modelo que encapsulan los datos y la funcionalidad de los componentes específica para la aplicación.

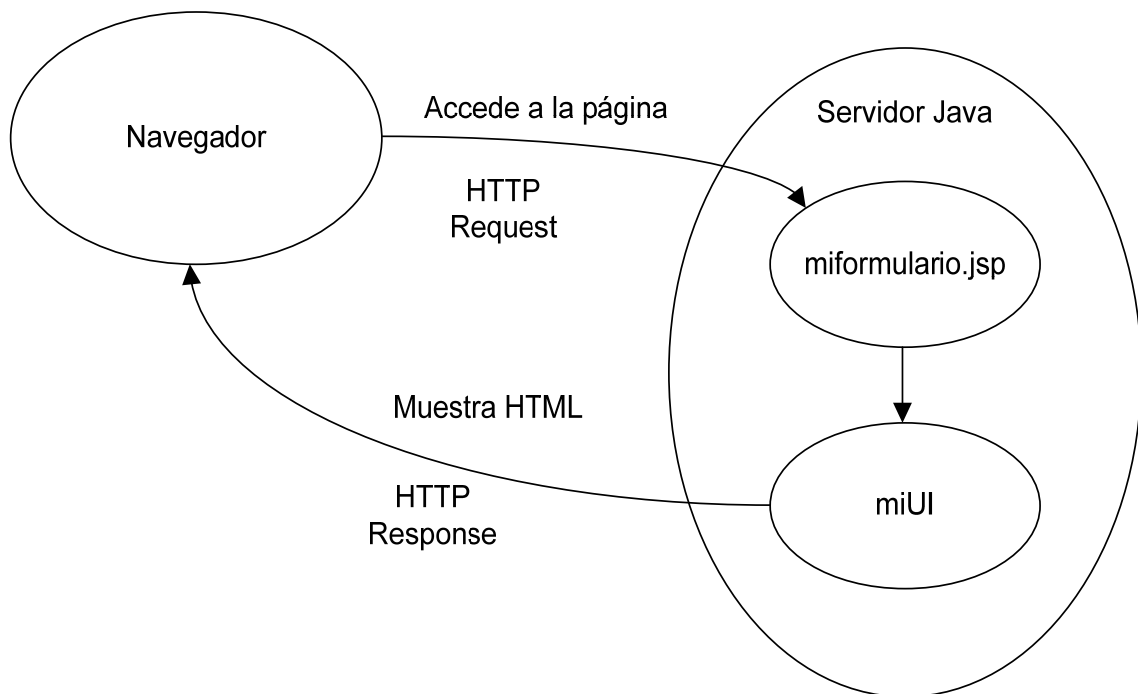


Figura 6.1 : La UI corre en el servidor

## 6.2 Los beneficios de la tecnología JavaServer Faces

Una de las grandes ventajas de la tecnología JavaServer Faces es que ofrece una clara separación entre la presentación y la lógica. Las aplicaciones Web construidas con la tecnología JSP parcialmente logran esta separación. Sin embargo, una aplicación JSP no puede mapear los HTTP Request a capturadores de eventos específicos o manejar elementos UI como objetos que manejen estado en el servidor. La tecnología JavaServer Faces permite construir aplicaciones Web que implementan una separación fina entre la lógica y la presentación tradicionalmente ofrecida por las arquitecturas UI basadas en el cliente.

La separación de la lógica de la presentación también permite a cada miembro del equipo de desarrollo de una aplicación Web enfocarse en su pieza del proceso de desarrollo, y provee un modelo simple de programación para juntar todas las piezas. Por ejemplo, los autores de páginas sin conocimientos de programación pueden usar los tags de los componentes UI de la tecnología JavaServer Faces para conectarse al código de la aplicación dentro de una página Web sin escribir ningún script.

Otro beneficio importante de esta tecnología es que da importancia a los componentes UI familiares a todos y a los conceptos de la capa Web sin limitar el desarrollo a una tecnología de scripts en particular o a un lenguaje específico. Mientras la tecnología JavaServer Faces incluye una biblioteca de “custom tags” para representar componentes en una página JSP, las APIs de esta tecnología están colocadas directamente sobre la API de JavaServlet. Esto permite:

- Usar otra tecnología de presentación además de JSP.
- Crear nuestros propios componentes directamente desde las clases de componentes.
- Generar salidas para diferentes dispositivos clientes.

Más importante aún, la tecnología de JavaServer Faces provee una arquitectura rica para manejar el estado de los componentes, procesar los datos de los componentes, validar la entrada del usuario, y manejar eventos.

### 6.3 ¿Qué es una aplicación en JavaServer Faces?

En su mayor parte, las aplicaciones en JavaServer Faces son como cualquier otra aplicación Java en Web. Corren en un contenedor Java Servlet, y típicamente contienen:

- Componentes JavaBeans (llamados objetos del modelo en la tecnología JavaServer Faces) que contienen funcionalidades y datos específicos a la aplicación.
- Capturadores de eventos.
- Páginas, como las páginas JSP.
- Clases de ayuda en el servidor, como los beans para acceder a las bases de datos.

Además de esto, una aplicación JavaServer Faces también tiene:

- Una biblioteca de tags especiales (“custom tags”) para mostrar los componentes UI en la página.
- Una biblioteca de tags especiales para representar a los capturadores de eventos, los elementos para validar las entradas, y otras acciones.
- Componentes UI representados como objetos con estado en el servidor (stateful objects).
- Validadores, capturadores de eventos, y capturadores de navegación.

Toda aplicación JavaServer Faces debe incluir una biblioteca de tags que define los tags especiales que representan los componentes UI y una biblioteca de tags para representar otras acciones del núcleo (core actions), como los validadores y los capturadores de eventos. Estas bibliotecas de tags son provistas por la implementación de JavaServer Faces.

La biblioteca de tags de los componentes elimina la necesidad de codificar a mano los componentes UI en HTML u otro lenguaje, resultando en componentes completamente reusables. Y, la biblioteca de tags para las acciones del núcleo hace sencillo registrar eventos, validaciones, y otras acciones en los componentes.

La biblioteca de tags de los componentes puede ser la biblioteca de tags *html\_basic* incluida con la implementación de referencia de la tecnología JavaServer Faces, o se puede definir una biblioteca de tags propia que muestre componentes creados por uno o que muestre otra salida diferente a HTML.

Otra importante ventaja de las aplicaciones JavaServer Faces es que los componentes UI en la página son representados como objetos con estado en el servidor. Esto permite a la aplicación manipular el estado de los componentes y conectar los eventos generados en el cliente con el código en el servidor.

Finalmente, la tecnología JavaServer Faces permite convertir y validar los datos en cada componente y reportar cualquier error antes de que la información en el servidor sea modificada.

## 6.4 Ciclo de vida de una página JavaServer Faces

El ciclo de vida de una página JavaServer Faces es similar a la de una página JSP: El cliente realiza una petición (HTTP request) por la página, y el servidor responde con la página traducida a HTML. Sin embargo, debido a las características extras que la tecnología JavaServer Faces ofrece, el ciclo de vida provee algunos servicios adicionales al ejecutar algunos pasos extras.

El donde estos pasos en el ciclo de vida son ejecutados depende de si el “request” fue originado desde una aplicación JavaServer Faces y si el “response” es generado con la fase de despliegue del ciclo de vida de la tecnología JavaServer Faces.

### 6.4.1 Escenarios de ciclos de vida para procesar peticiones

Una aplicación JavaServer Faces soporta dos diferentes tipos de respuestas (responses) y dos diferentes tipos de peticiones (requests):

- **Faces Response:** La respuesta de un servlet que fue creado como resultado de la ejecución de la fase *Render Response* del ciclo de vida del proceso de *request*.
- **Non-Faces Response:** La respuesta de un servlet que no fue creado por la ejecución de la fase *Render Response*. Un ejemplo es una página JSP que no incorpora componentes JavaServer Faces.
- **Faces Request:** La petición de un servlet que fue enviada desde un *Faces Response* previamente generada. Un ejemplo de esto es el envío (submit) de un formulario desde un componente de la interfaz del usuario de JavaServer Faces, donde la URI de la petición identifica al árbol de componentes a ser usado para procesar la petición.
- **Non-Faces Request:** La petición de un servlet que fue enviada a un componente de la aplicación, como un servlet o página JSP, en vez de ser dirigido a un árbol de componentes JavaServer Faces.

Estas diferentes respuestas y peticiones dan como resultado tres posibles escenarios para el ciclo de vida que pueden existir para una aplicación JavaServe Faces.

- **Escenario 1: Una petición No-Faces genera una respuesta Faces:** Un ejemplo de este escenario es cuando un hiperlink de una página HTML abre una página que contiene componentes JavaServer Faces. Para mostrar una Respuesta Faces desde una Petición No-Faces, una aplicación debe proveer un mapeo al *FacesServlet* en la URL a la página que contiene al componente JavaServer Faces. El servlet *FacesServlet* acepta las peticiones entrantes y la pasa a la implementación del ciclo de vida para procesarlas.
- **Escenario 2: Una petición Faces genera una respuesta No-Faces:** A veces una aplicación JavaServer Faces puede necesitar ser redirigida a un recurso de

una aplicación Web diferente o generar una respuesta que no contiene ningún componente JavaServer Faces. En estas situaciones, el desarrollador debe evitar la fase de rendering (Poner link cruzado) llamando a FacesContext, al método responseComplete. FacesContext contiene toda la información asociada con una Petición Faces particular.

- **Escenario 3: Una petición Faces genera una respuesta Faces:** Este es el escenario más común del ciclo de vida de una aplicación JavaServer Faces. Este escenario involucra componentes JavaServer Faces enviando una petición a una aplicación Faces utilizando el FacesServlet. Ya que la petición ha sido manejada por la implementación de JavaServer Faces, no son necesarios pasos adicionales por la aplicación para generar una respuesta. Todo será manejado por el ciclo de vida de JavaServer Faces.

#### 6.4.2 Ciclo de procesamiento estándar para peticiones

Este ciclo está representado por el escenario 3. La mayoría de los usuarios de esta tecnología no necesitan preocuparse con el ciclo de vida de las peticiones en JavaServer Faces. Sin embargo, sabiendo que la tecnología Faces realiza correctamente el proceso de una página, un desarrollador de aplicaciones Faces no necesita preocuparse de problemas de rendering asociados con otros UI frameworks.

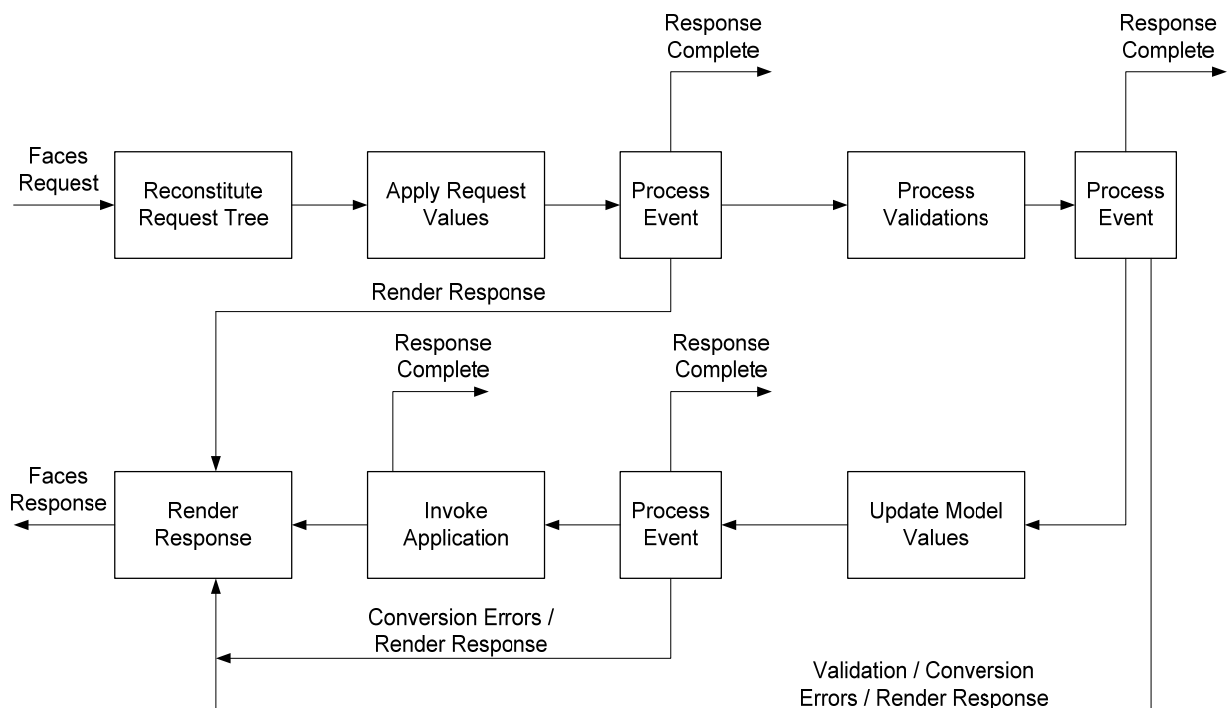


Figura 6.2 : Ciclo de Vida Request-Response de JavaServer Faces

Un ejemplo de esto involucra los cambios de estado de los componentes individuales. Si la selección de un componente como un checkbox afecta la apariencia de otro componente en la página, la tecnología JavaServer Faces manejará este evento apropiadamente y no permitirá que la página se muestre sin reflejar este cambio.



La Figura 6.2 muestra los pasos en el ciclo de vida request-response en JavaServer Faces.

## Capítulo 7: Consideraciones de LIBIS

---

Durante la pre-reunión se realiza un proceso de toma de decisiones para planificar las actividades que se realizarán en la reunión. Pero debido a la dificultad que tienen las personas para asociar sus pensamientos con los elementos que componen a IBIS es necesario el uso de un proceso para lograr que sus ideas puedan ser ingresadas sin problemas y que logren llevar a cabo una discusión para definir los elementos de este proceso.

Las personas pueden entender fácilmente las ideas de los elementos de IBIS, pero al intentar expresar sus ideas siguiendo las restricciones que impone IBIS el proceso se hace complicado, ya que el proceso de traducción de las ideas puede requerir un proceso para adaptarlas al modelo, donde se puede dar el caso que una de estas ideas necesite ser descompuesta en los tres elementos del modelo, lo que hace que este proceso pueda ser complejo y engorroso.

Un punto importante en todo este proceso es la comunicación que debe existir durante esta etapa, que permite que la preparación de la pre-reunión pueda abarcar más actividades, como la generación de material útil para la reunión, además de generar una base de conocimientos común de los temas que rodean los puntos de la agenda de la pre-reunión para todos los usuarios del sistema, y de esta forma hacer más simple el proceso de generación de ideas.

Para el sistema que se desea desarrollar, estas actividades incluirán el compartir archivos y la comunicación entre los miembros para así lograr una comunicación más constructiva.

Como en SISCO, la pre-reunión tiene como objetivo realizar las actividades para realizar la planificación de la reunión, nada más que eso. Debe evitarse el desarrollo de los temas que se tratarán en la reunión dentro de la pre-reunión.

Los participantes de la pre-reunión parten por discutir sobre los distintos temas que contiene la agenda de la pre-reunión y presentan sus ideas. LIBIS busca facilitar este proceso al acercar la discusión a los participantes, utilizando un modelo de tres niveles que presentan una estructuración creciente. La idea consiste en obtener, en el tercer nivel, las ideas ingresadas por los usuarios modeladas en IBIS.

Inicialmente los usuarios interactúan con un modelo libre de restricciones, donde ingresan sus ideas sin considerar modelo alguno, y a través de un proceso de discusión, estos usuarios pueden participar en la transformación de sus ideas para cumplir con las necesidades del modelo IBIS. El mapeo de las ideas es realizado por un facilitador, a través de las herramientas del sistema.

Uno de los puntos más difíciles en este enfoque para la pre-reunión es convencer a los participantes a que compartan sus ideas y conocimientos, y que vean el hecho que tener visiones contradictorias no debe ser visto como una pérdida de poder dentro del grupo o tomar riesgos innecesarios.

En un comienzo la participación puede ser algo temerosa o tímida. La mayoría de los participantes pueden tomar una actitud conservadora, leyendo las contribuciones de otros usuarios en vez de exponer sus propias ideas.

Uno de los puntos importantes es la necesidad de liderazgo dentro del grupo. Debido a que se trata de una nueva forma de participación dentro de un grupo, para muchos de los participantes, puede ser necesario estimular una actitud positiva frente a la nueva forma de trabajo colaborativo. Para esto es necesario contar con el facilitador, esta persona debe ejecutar su papel dentro de la pre-reunión de manera adecuada, para así evitar discusiones vacías o aburridas y lograr que estas sean dinámicas, entusiastas y que generen la sensación de compañerismo, para así evitar una participación indiferente y obtener resultados que llenen los requerimientos de la reunión.

Siguiendo los pasos del artículo [15] y considerando las características que se buscan para cumplir con las ideas de LIBIS se analizarán los puntos del diseño del sistema.

## **7.1 Arquitectura**

El objetivo del sistema es proveer de un medio de comunicación dedicado a atender las necesidades de la pre-reunión. Es por esto que un punto importante dentro de este medio de comunicación es proporcionar un lugar donde almacenar toda esta información generada.

El uso de un medio de comunicación síncrono supone una comunicación más fluida y en donde el proceso de generación de ideas es más eficiente. Pero hay que considerar el medio en el cual esta conversación se desarrolla. Un ambiente donde los usuarios deben comunicarse a través de comunicaciones en línea no permite un desarrollo ordenado de la conversación en la mayoría de los casos.

En cambio el uso de un medio de comunicación asíncrono, como son por ejemplo los foros de discusión, a pesar de no permitir una comunicación dinámica, donde los usuarios pueden llegar a conclusiones más rápidamente, permite tener un registro de esta discusión, además de permitir el almacenamiento y posterior uso de esta información por otros usuarios, que es una ventaja importante cuando se busca lograr resultados dentro de un numeroso grupo de personas.

Para lograr esto, el uso de una arquitectura ASMP (Asynchronous System to Support Meeting Preparations) parece lo indicado.

El objetivo de esta arquitectura (Figura 7.1) es dar a los usuarios una interfaz grupal, acceso a un “espacio común” (Memoria Grupal) y un medio de comunicación dentro del grupo. Para que el sistema cumpla con la idea de esta arquitectura el esquema cliente-servidor es el más adecuado para llevarlo a cabo.

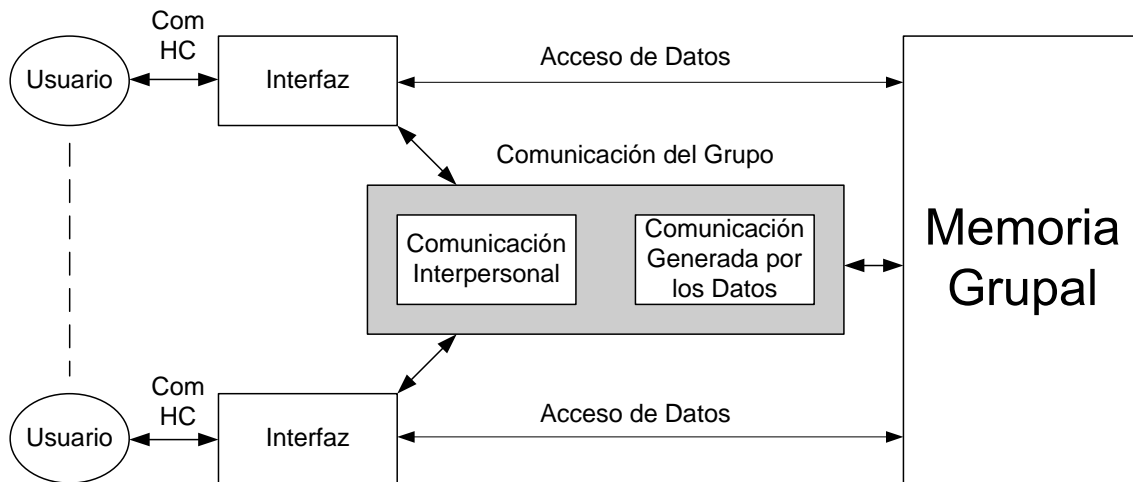


Figura 7.1 : Arquitectura ASMP

Para lograr esto las opciones son utilizar una aplicación distribuida, utilizando RMI de Java por ejemplo, o utilizando la interfaz que proporciona el uso de una aplicación web.

Una aplicación distribuida cuenta con una gran libertad al momento de diseñar y desarrollar el sistema. RMI permite el acceso al servidor sin problemas y de una manera simple, solo se necesita definir la interfaz remota y generar los “stub” y los “skeletons” necesarios. Estos elementos también se encuentran disponibles cuando uno desarrolla una aplicación web con herramientas como los applets, pero restringe las acciones que puede realizar la aplicación en la máquina cliente.

El uso de una aplicación web utilizando las herramientas que brinda JavaServer Faces, si bien cuentan con restricciones, como la necesidad de ajustarse al ciclo de vida de una aplicación JSF, cuenta con la gran ventaja de permitir al usuario acceder al sistema desde cualquier computador con acceso a Internet, sin requerir nada más que el navegador, ya que lo que genera el servidor en este caso es código HTML, por lo que está disponible para cualquier navegador.

El usuario accede al sistema a través de la web, donde JavaServer Faces responde a las peticiones del usuario ejecutando las aplicaciones en el servidor, y retornando como respuesta código HTML que puede ser visto a través de cualquier navegador.

Pero quizás la gran ventaja es la disponibilidad de un framework que resuelve muchos de los problemas para implementar la solución (acceso a la base de datos, manejo del acceso a los recursos del sistema, seguridad del sitio), además de contar con una biblioteca de beans que pueden ser usados en el desarrollo de la aplicación. Esta biblioteca facilita mucho el desarrollo, permitiendo un diseño basado en eventos y un diseño más simple.

Esta facilidad de uso es un punto importante, considerando que se busca que la aplicación sea lo más “amigable” con el usuario, y que se logre un desarrollo más sencillo y confiable del sistema.

El uso de JSF permite al usuario acceder a la Memoria Grupal para guardar y recobrar los elementos relevantes de la pre-reunión, además de posibilitar la comunicación entre ellos así como información generada por los eventos que ocurren en la pre-reunión. Todo esto brinda a los usuarios la posibilidad de compartir sus ideas en el contexto de la discusión, construyendo un espacio común y asequible para todos.

Toda la información generada en la pre-reunión será almacenada y podrá ser revisada por los usuarios, dado que la interacción entre ellos será asíncrona. Además los eventos que ocurran en el sistema darán soporte al “awareness”, necesario para que el usuario pueda estar al tanto de lo que ocurre en la pre-reunión.

## **7.2 Mecanismos de Discusión**

Debido a las características de LIBIS nos encontramos con dos distintos mecanismos de discusión. La existencia de etapas y la intención de no exigir conocimientos previos a los usuarios, exige un mecanismo simple para la primera etapa de discusión.

En Internet el mecanismo de discusión más conocidos son los foros, que tienen una estructura simple, pero sin mayor orden. Este mecanismo no exige conocimientos ni instrucciones para que los usuarios logren expresar sus ideas.

Un problema que se puede producir con este mecanismo es una pérdida del rumbo, provocando que los resultados se alejen de los objetivos de la discusión. Es aquí donde la labor del facilitador se hace necesaria para evitar estos problemas, tanto eliminando partes de la discusión que no cumplan los objetivos como instruyendo y guiando a los usuarios. También es el facilitador quien decide cuando proponer el término de la etapa y comunicar su decisión al coordinador para que este evalúe y decida que acción tomar.

En las etapas siguientes ya existe un modelo a seguir, basado en IBIS, cuyos elementos hacen referencia a los aportes realizados por los participantes en la primera etapa. Aquí es el facilitador quien guía totalmente el rumbo de la discusión, al ser él quien propone un modelo pregunta-respuesta-argumento que representa la discusión de la primera etapa. Este trabajo puede ser algo más que difícil, considerando la gran cantidad de información que se pudo generar en la etapa anterior y la complejidad del proceso de transformación, por lo que la existencia de varios facilitadores para un tema es algo útil.

Esto nos lleva a un problema, el grupo de facilitadores debe contar con herramientas para que su trabajo en grupo no provoque discusiones innecesarias que entorpezcan el trabajo en la segunda etapa.

Una opción es permitir que esta discusión se realice por medios externos al sistema, permitiendo que, debido al reducido número de facilitadores (en comparación a la cantidad de participantes), pueda llegar a acuerdos de forma más eficiente.

Otra opción es desarrollar herramientas dentro del sitio para que se realice una discusión similar a la que se produce en la primera etapa del sistema, solo que restringida a un grupo pequeño y limitada a un tema en particular.

La segunda opción permite que este proceso de toma de decisiones no requiera disponer de tiempo adicional, que desde un principio fue la motivación para desarrollar un sistema distribuido. La solución es desarrollar esta discusión en forma asíncrona, llevando un registro de los aportes y las herramientas para seguir el desarrollo de la discusión.

El papel de los participantes en esta segunda etapa es de observadores que pueden hacer acotaciones al modelo propuesto por el facilitador, quien en base a estos comentarios lo modificará a su parecer, hasta considerar que el modelo ya responde a las necesidades para realizar la reunión. El término de esta etapa también es decidido por el coordinador.

Otro punto importante de esta etapa es definir las tareas que se deben llevar a cabo antes de la pre-reunión. En esta etapa solo se proponen las tareas, aún no se decide ni fechas ni encargado de realizarla. Los participantes del tema pueden postularse para desarrollar la tarea, y en caso de no contar con postulantes o que ninguno de ellos cumpla con los requisitos del facilitador puede en la tercera etapa nombrar a los encargados de desarrollar la tarea.

En la etapa final se presenta el modelo IBIS definitivo, disponible para todos los participantes de la reunión, esto a través de la información que se puede obtener del sistema. Es en este momento en donde el resultado de las postulaciones a las tareas es anunciado y se comunica a los encargados de su asignación, además de fijar plazos para su entrega.

### **7.3 Memoria Grupal**

El propósito de la Memoria Grupal es mantener un espacio común y persistente donde guardar la información relacionada a la discusión dentro del grupo.

En un ASMP, la Memoria Grupal necesita ser diseñada considerando tres tipos de información diferentes. La primera esta relacionada con los miembros del grupo, las discusiones en las que están involucrados, los roles dentro de la discusión, por ejemplo. El segundo tipo de información tiene relación con los elementos que se definen a partir del mecanismo de discusión que se llevará a cabo, en este caso, LIBIS. El tercer tipo de información contiene lo que se necesita saber antes de la discusión, que son las decisiones tomadas antes de la discusión y los límites de la discusión. Estos son puntos importantes a considerar durante el proceso de la pre-reunión.

Uno de los objetivos de la Memoria Grupal es proveer la información requerida para permitir los mecanismos de awareness asíncronos. Pero es posible que también permita obtener persistencia en la comunicación interpersonal cuando sea necesario. Un ejemplo de esto es la necesidad de registrar una conversación entre miembros del grupo de trabajo, que se realice a través del sistema, y así contar con esto como información útil para el grupo de discusión.

## **7.4 Mecanismos de Comunicación**

La interacción dentro del grupo de discusión es uno de los principales objetivos. Es indispensable lograr una comunicación entre los participantes, pero esta comunicación debe hacerse en el contexto de la discusión que se está desarrollando, contando con la ayuda de las herramientas que presta el sistema para dar soporte a esta comunicación.

Uno de los problemas para lograr esta comunicación es la sensación de impersonalidad, en particular el anonimato y la separación física que no permite la comunicación fluida que brinda la conversación con otras personas.

En SISCO no se define qué medio de comunicación puede ser útil. Es más, se asume que los medios de comunicación convencionales están a disposición de los miembros del grupo de trabajo. Sin embargo, el énfasis del sistema está en usar como medio de comunicación la Memoria Grupal que guarda toda la información generada en la pre-reunión y en todas las que anteriormente se trabajó.

La comunicación a través de la base de datos es formal, estructurada, persistente y pública para los miembros autorizados a participar. Esto permitirá que la información generada por LIBIS sea de acceso simple y permita a su vez el acceso a todas sus etapas.

Este medio de comunicación puede parecer algo rígido si lo comparamos con los medios de comunicación utilizados usualmente por las personas. Es aquí donde se asume que otros medios de comunicación convencionales permiten llenar estos vacíos.

## **7.5 Mecanismos de Visualización**

El despliegue de la información es algo crucial, es poco lo que se puede obtener de una discusión si no se logra acceder a todos los datos útiles para realizar esa discusión.

Mientras más rica y abundante sea la información guardada en la Memoria Grupal, más importante es el cuidado que se debe tener al momento de proveer esta información a los participantes de la discusión.

En SISCO hay diferentes medios para acceder a esta información, por medio de búsquedas de texto, navegación por autor, seguir el hilo de la conversación y accediendo al material no leído. Es importante también lograr filtrar esta información dependiendo de las obligaciones del participante, así como por sus intereses.

La forma básica para visualizar en una aplicación Web es utilizar un medio basado en texto, donde la información se despliega según su jerarquía. Sin embargo, esto puede llevar a problemas debido a que este tipo de medio de comunicación suele provocar la omisión de datos. La utilización de medios más interactivos y la posibilidad de utilizar elementos conocidos por los usuarios (árboles, menús, ventanas y elementos interactivos) vistos normalmente en las aplicaciones normales. También la posibilidad de utilizar el uso de medios gráficos es una opción que puede usar el usuario para visualizar mejor la evolución de la discusión.

## Capítulo 8: Uso del Sistema

---

### 8.1 Roles dentro del Sistema

Los usuarios para los que está destinado el sistema no deben verse obligados a interiorizarse en lo que es IBIS ni sus restricciones. Es obvio pensar que después de un tiempo de uso la definición de IBIS será algo familiar, pero la ventaja de contar con un sistema donde no se requieran mayores conocimientos es la facilidad de agregar a personas que cuenten con recursos y conocimientos útiles para la pre-reunión que se desarrolla. Otra ventaja es lo simple que es participar en una discusión de este tipo, sin restricciones, y poder trabajar en ellas en grupo, para lograr un mejor resultado.

Considerando el objetivo de no requerir por parte de los usuarios un conocimiento especial en el modelamiento de la discusión a través de IBIS, el peso que recae en la persona encargada de la discusión es considerable. Es aquí donde se hace evidente la necesidad de contar con personas dentro del círculo de usuarios que puedan llevar a cabo la conversión de una discusión común al modelo IBIS.

Si bien la actuación de los usuarios dentro de la segunda etapa es algo importante, no es menos cierto que el material sobre el que se desarrolla esa discusión es más importante aún. Por lo tanto, un perfil clave dentro del sistema es el de facilitador.



Para una correcta participación es necesario definir roles, que serán asumidos por los participantes en cada tema. Cada rol tendrá responsabilidades dentro de la discusión y privilegios para acceder a distintos datos de la Memoria Grupal. Además del coordinador es necesario considerar roles adicionales para lograr resultados.

Las personas que tomarán decisiones en la reunión claramente deben participar en la discusión de la pre-reunión, para evitar así la incerteza y los errores en la pre-reunión. En SISCO existen tres roles adicionales al rol del coordinador, definidos para participar en la pre-reunión. En este caso la participación de los roles dentro del proceso se verá alterado por el uso de LIBIS y el cambio en la responsabilidad de cada rol durante la pre-reunión. En la Tabla 1 se definen los derechos de acceso para cada rol.

Acceso Rol	Restricciones		Temas		Objetivos		Elementos	
	Leer	Escribir	Leer	Escribir	Leer	Escribir	Leer	Escribir
<b>Coordinador</b>	X	X	X	X	X	X	X	
<b>Facilitador</b>	X		X		X	X	X	X
<b>Participante</b>	X		X		X		X	X
<b>Observador</b>	X		X		X		X	

Tabla 1 : Permisos de acceso por rol

### 8.1.1 Coordinador

Este coordinador es el responsable del éxito de la pre-reunión. Su principal objetivo es lograr que los participantes adquieran el conocimiento para poder cumplir con la toma de decisiones de una manera informada, y así obtener un adecuado nivel de documentación e incrementar la base de conocimientos de la Memoria Grupal.

La idea es que la persona con la autoridad administrativa apropiada pueda jugar el rol de coordinador, lo que amerita la necesidad de definir una lista de personas que tengan las características necesarias para administrar el proceso de la pre-reunión, considerando la importancia de su papel dentro de esta.

Las actividades que debe realizar el coordinador son las siguientes:

- Definir a los usuarios que participarán de la pre-reunión.
- Ejecutar el cambio de etapa de LIBIS para los temas de la pre-reunión.
- Asignar los roles que cumplirán los usuarios dentro de cada tema en la pre-reunión. Estos roles están definidos y tienen diversos deberes y atribuciones que buscan permitir un mejor desarrollo de la pre-reunión, al ingresar a las personas

que son útiles para cara tema de la agenda. Cada usuario puede jugar distintos roles en temas distintos. El coordinador también puede participar en los temas.

- Definir y manejar las restricciones de la pre-reunión. Esto implica definir o modificar las pre-decisiones que dan un piso para el desarrollo de las discusiones, y definir o modificar las restricciones que limitan lo que se puede o no se puede hacer.
- Crear y modificar los temas de la agenda de la pre-reunión. Es el coordinador quien debe definir la agenda de la discusión, también se debe definir los facilitadores para cada tema, quienes tendrán bajo su cargo la evolución de la discusión y el desarrollo de las distintas etapas dentro del tema. También definir la fecha de término del tema.
- Monitorear el desarrollo de la pre-reunión y guiar el comportamiento de los participantes.

### 8.1.2 Facilitador

El facilitador es la persona encargada de supervisar la discusión de un tema en particular. Esta persona no tiene necesariamente una posición superior en la organización, pero tiene la facultad de tomar decisiones importantes para el desarrollo del tema de conversación que tiene a su cargo, como solicitar al coordinador la presencia de una persona en particular, escoger a los participantes responsables de la ejecución de una tarea, y vigilar el desarrollo del tema para asegurar el cumplimiento de los objetivos.

Como se mencionó antes, debido a los cambios dentro del modelo que disminuye la responsabilidad por parte de los participantes de seguir un modelo, es necesario que la persona que asume este rol este conciente de la importancia que tienen sus decisiones.

Su accionar en la primera etapa se hace crucial al momento de manejar el rumbo que tome la discusión, considerando la falta de una estructura rígida. Y en las siguientes dos etapas, aunque todos participan dando opiniones sobre el rumbo que toma la transformación de los aportes de la primera etapa al modelo IBIS, es él quien decide el modelo inicial y sus cambios.

Esta transformación puede ser una carga de trabajo importante, considerando la cantidad de opiniones que pueden salir de la discusión, por lo que ante la necesidad, el coordinador puede designar más de un facilitador para un tema como ya se dijo antes. Si bien esto provoca un proceso más complejo, ya que ahora las decisiones se toman en grupo, la posibilidad de trabajo en equipo permite obtener mejores resultados en discusiones complejas, así como la reducida cantidad de facilitadores permite un trabajo más fluido. Otra ventaja es la posibilidad de elegir quienes son los encargados para el trabajo y así el coordinador se asegura que las capacidades de los encargados sean las requeridas para el trabajo.

Es la experiencia dentro de este rol lo que ayuda a que la pre-reunión se desarrolle sin problemas, y aunque se ha puesto énfasis en la importancia de sus decisiones, también es clara la necesidad de contar con habilidades para manejar el grupo de participantes y que

logren producir los resultados que se esperan. En otras palabras, que en vez de ser alguien que solo tome decisiones, sea una guía dentro del proceso. Para esto es necesario contar con las herramientas para seguir el accionar de los participantes y mantener una comunicación privada con ellos.

### 8.1.3 Participante u Observador de la Pre-Reunión

En cuanto a los participantes, el principal objetivo es que aporten en la discusión en la que han sido invitados por el coordinador, pero existen casos en donde el objetivo no es que su participación sea activa dentro de la discusión. En este caso lo que se busca es que puedan seguir el desarrollo de la discusión para obtener información que pueda ser útil en el desarrollo de otro tema dentro de la pre-reunión. En este caso su rol dentro del tema será el de observador, pudiendo seguir la evolución del tema, pero sin tener la posibilidad de aportar en él.

Otra importante herramienta es dar la posibilidad que los participantes soliciten su inclusión dentro de un tema en forma voluntaria, para así brindarles más posibilidades de obtener información que ellos consideren útil. Estas suscripciones serán administradas por el coordinador, quien podrá suspender la inclusión de un participante de la pre-reunión a un tema en especial, así también como cambiar el rol del participante dentro del tema.

El objetivo de esto es dar una mayor flexibilidad para administrar la pre-reunión y lograr obtener los mejores resultados durante su desarrollo. El coordinador, al contar con la información de los usuarios, puede ver como mejorar la discusión.

Los participantes de un tema tienen las siguientes obligaciones:

- Producir elementos de discusión (elementos de LIBIS) en los temas en los que están autorizados. Pero este punto depende de las expectativas que tiene el coordinador o el facilitador de la participación del usuario en cada tapa, que puede ser distinta, esperando gran participación en una en particular, o desarrollar tareas para la reunión.
- Ofrecerse como responsable del desarrollo de una tarea al facilitador.
- Realizar las tareas asignadas por el facilitador, quien es el que define al responsable de la tarea. La tarea debe ser realizada dentro del tiempo definido para su entrega.

## 8.2 Preparación de la Pre-Reunión

Es necesario dar soporte para la creación de la pre-reunión. Siguiendo el ejemplo de SISCO, el sistema permite que cualquier usuario pueda solicitar la creación de una pre-reunión a través del sistema. Esta decisión es tomada por uno de los coordinadores del sistema (que son usuarios con los permisos para hacerlo) y se genera la creación de la estructura en la base de datos que contendrá la Memoria Grupal de la pre-reunión (Figura 8.1).

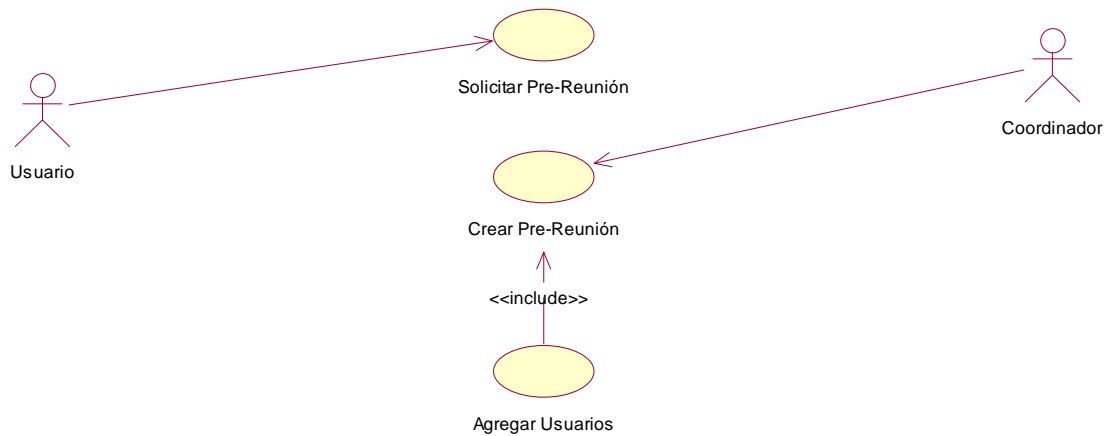


Figura 8.1 : Creación de la Pre-Reunión

### 8.3 Configurando la Pre-Reunión.

Posterior a la creación de la pre-reunión y la designación de los usuarios que participarán de ella, es necesario definir el ambiente en el que se desarrollará la pre-reunión.

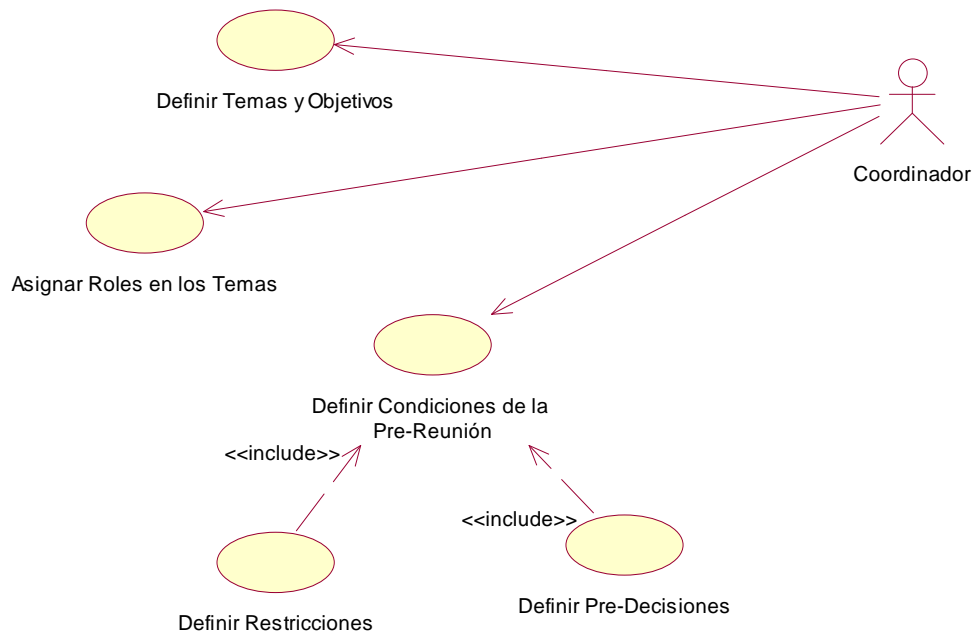


Figura 8.2 : Configuración de la Pre-Reunión

Esta etapa es importante, debido a que se define el camino que seguirá la pre-reunión, al definir los temas y los objetivos que cubrirá. Además es necesario definir las condiciones en las que se desarrolla la pre-reunión (Figura 8.2).

### 8.3.1 Temas de la agenda

La necesidad de tener definidos los temas de la discusión no se puede pasar por alto. Estos temas deben estar relacionados a las decisiones que se tomarán en la reunión que se realizará. Por lo que, quien quiera que sea la persona encargada de la reunión debe estar relacionada con el proceso de definición de los temas de la pre-reunión.

Esta especificación de temas es muy importante para el éxito de la pre-reunión, ya que define el marco de trabajo en la que se desarrolla la pre-reunión, lo que evita posibles malos entendidos entre los participantes de la discusión.

Es importante también evitar el restringir en demasía los temas, o definir metas vagas o tiempos irreales para terminar cada tema. Estos errores pueden llevar a discusiones sin sentido o apresuradas, lo que no producirá los resultados necesarios para desarrollar en forma adecuada la reunión subsiguiente.

Como en SISCO, para evitar estos problemas es necesario que el coordinador esté lo suficientemente capacitado para especificar la agenda de la de la discusión, ordenar los temas y establecer los tiempos para cada uno. Más aún, es necesario que para cada tema el coordinador establezca los objetivos de la discusión (esta tarea también puede involucrar a los facilitadores encargados del tema), que deben ser considerados durante el tiempo de la discusión. Esta estructura es la que se respetará en las tres etapas del proceso.

## 8.4 Desarrollo de la Discusión

### 8.4.1 Contribuir en el Proceso de la Discusión

Los participantes de una pre-reunión deben entrar en la discusión, y así lograr que se enfrenten las distintas posturas que puedan existir dentro del grupo. Este conflicto se necesita debido a que una pre-reunión donde los participantes estén de acuerdo sin mediar una discusión no logra un resultado satisfactorio.

Es de esperar que frente a distintos puntos dentro de la discusión las personas tengan diversas visiones. Estas deben ser expresadas dado que las personas deben ser libres para hacerlo. Son estas diferencias que brotan en la discusión las que enriquecen el trabajo del equipo.

Es posible que la primera opinión compartida en la discusión sea la que parezca la más adecuada, pero aún así es mucho mejor contar con todas las opiniones que puedan surgir del grupo de participantes y después escoger. Si esto no ocurre es aconsejable que el facilitador se preocupe de que esto ocurra. Es probable encontrar ideas que se complementen, pero para que esto suceda es necesario contar con todas las ideas que puedan crear los participantes de la pre-reunión.

Otro punto importante de compartir estas ideas es que se pueden obtener mejores resultados con el trabajo en equipo, y también se logra que el conocimiento del grupo mejore. Mucha de la información expresada en la discusión puede ser desconocida para algunos miembros del equipo, esto sin considerar el concepto de “grupo de trabajo” que se genera entre los miembros del equipo.

Considerando la primera etapa, en la que los participantes de la pre-reunión comparten sus ideas libremente, la falta de restricciones puede hacer más sencilla la tarea de llevar a término esta discusión, siempre considerando los problemas que puede traer el no manejar el curso que lleve la discusión con cuidado.

#### 8.4.1.1 Aportando a la Discusión

Las contribuciones de los participantes no deben considerarse como ideas personales, es algo importante de inculcar en ellos. Lo que se busca es que estas ideas se guarden para la etapa siguiente. Es durante la reunión donde estas opiniones son necesarias.

En la primera etapa (Figura 8.3), siguiendo el modelo que proponen los foros de discusión en Internet, los aportes que se ingresan pueden generar un nuevo hilo de discusión o ser la respuesta a una de los aportes existentes.

Esto es importante de considerar en la primera etapa de LIBIS, donde la libertad en el formato de las opiniones ingresadas puede llevar a un desorden en la discusión. Es aquí donde el facilitador debe hacer uso de sus atribuciones para manejar la discusión para evitar el descontrol.

Es en la segunda etapa (Figura 8.4) donde los elementos lógicos de IBIS se definen a partir de las discusiones. En este segundo paso es donde los participantes pueden involucrarse en la creación de los elementos de IBIS solo haciendo comentarios al trabajo del facilitador. Esto beneficia al grupo con la posibilidad de participar en el proceso de crear la estructura pregunta-respuesta-argumento de IBIS, aunque no de la forma convencional. Pero aún así, el usuario logra aprender como estructurar las ideas contando ya con un concepto de los temas y objetivos, con la posibilidad de participar en una forma menos rígida en la creación de la estructura IBIS.

Este segundo paso logra eliminar todo problema en la comprensión de los aportes de la primera fase. El facilitador creará los elementos del modelo a partir de lo generado antes, con la seguridad de que los usuarios podrán opinar libremente sobre su trabajo. Gracias a esto, el facilitador evitará que en el resultado final de la segunda etapa, las preguntas, respuestas y argumentos, sean decididos sin la libre opinión de las personas que contribuyeron. Como resultado final se obtendrá el modelo IBIS que utilizará para la reunión.

En la tercera etapa, ya contando con una versión trabajada de la discusión ajustada al formato IBIS, es posible para el facilitador decidir a partir de la discusión y de los postulantes de la etapa pasada, las tareas necesarias para realizar la reunión y los responsables de estas, así como sus tiempos de entrega (Figura 8.5).

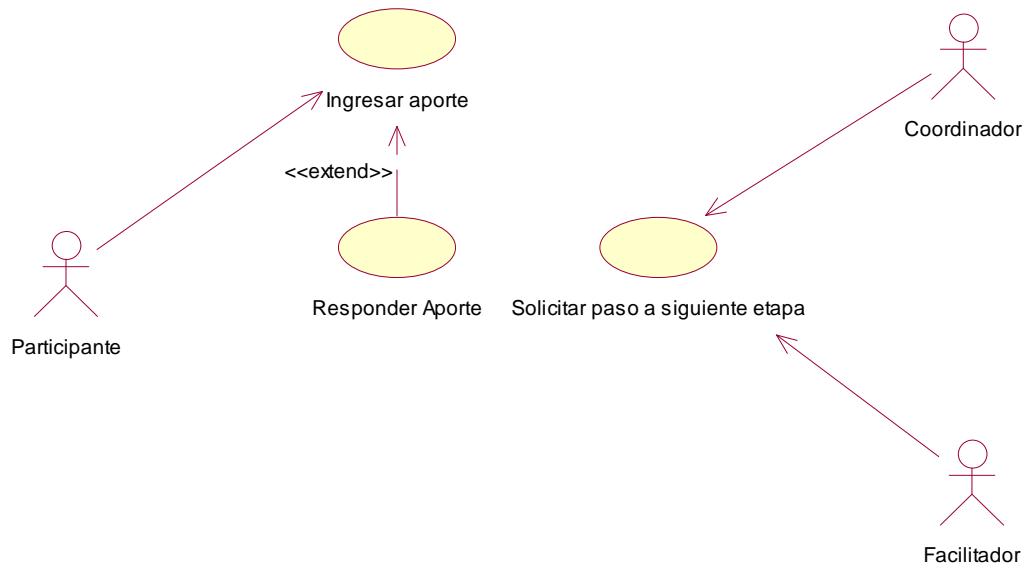


Figura 8.3 : Primera Etapa de la Discusión

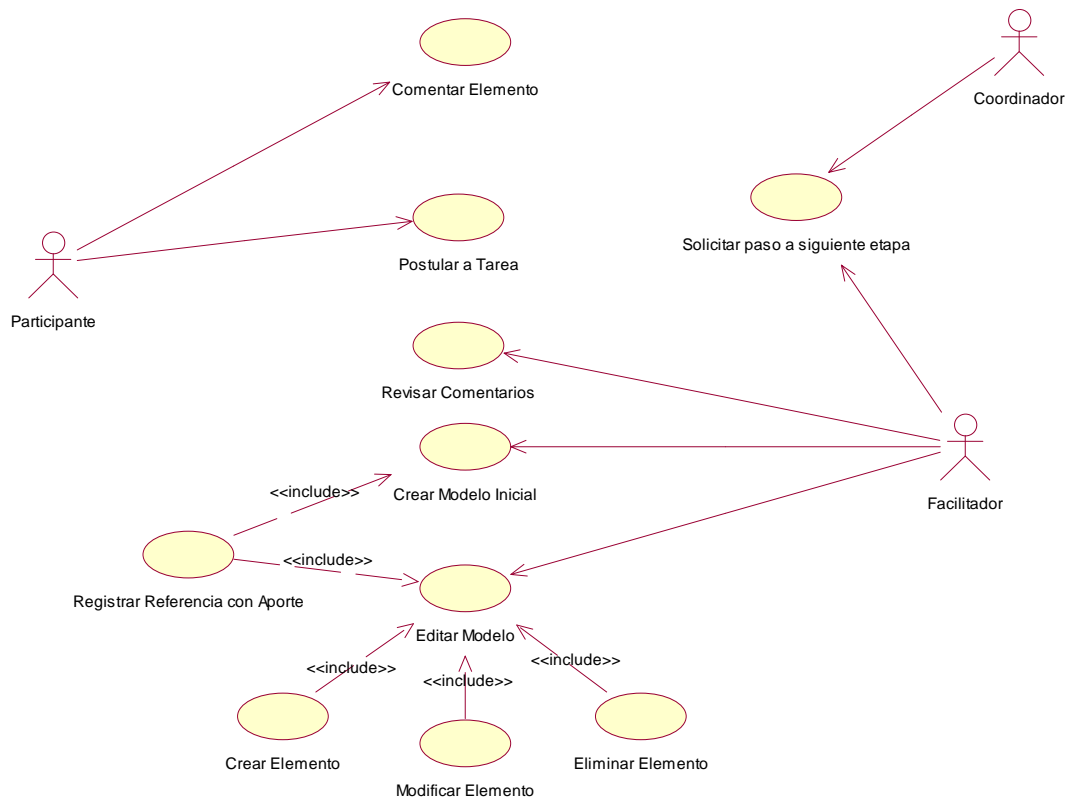


Figura 8.4 : Segunda Etapa de la Discusión

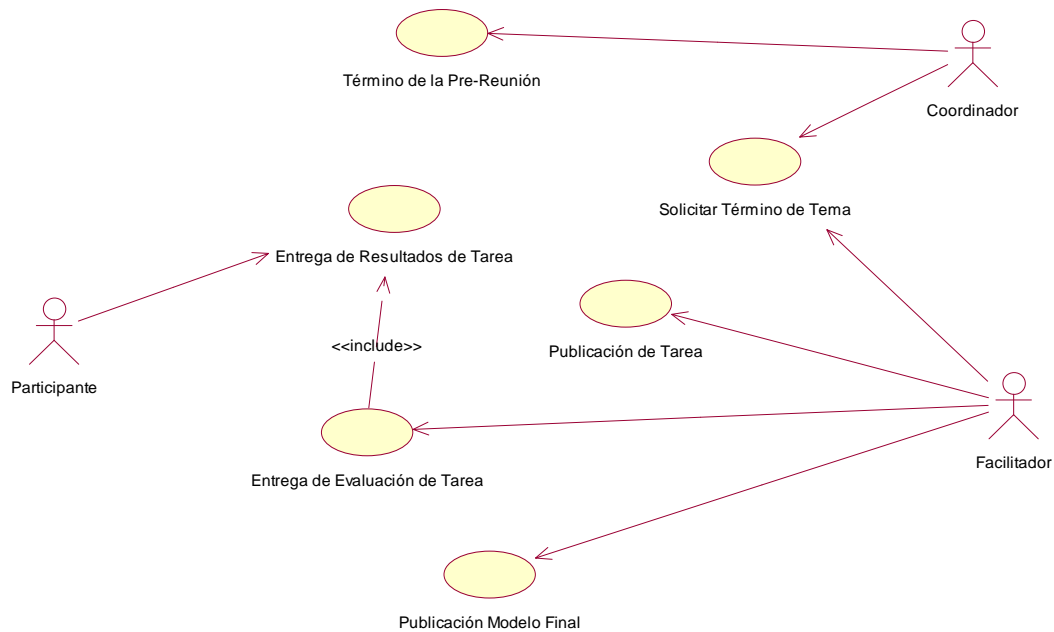


Figura 8.5 : Tercera Etapa de la Discusión

#### 8.4.1.2 Identificación de las Contribuciones

El tema del anonimato es algo de cuidado dentro de un sistema diseñado para soportar grupos de discusión. El concepto del anonimato es clave cuando existe la necesidad de lograr que los participantes expresen sus puntos de vista libremente, sin el miedo comprensible que sienten al momento de criticar a otros miembros del grupo, o que consideren que sus ideas no son populares por lo que pueden ser “perjudiciales”.

Aunque el principal objetivo del sistema es obtener ideas para la reunión, estas ideas que no necesariamente deben expresar las creencias del autor, solo deben considerarse útiles o interesantes para la reunión. Las personas no pueden evitar el sentir que expresan sus ideas y que por ellas pueden tener problemas.

Sin embargo, el anonimato presenta problemas. El anonimato ofrece la posibilidad de agredir, demasiada libertad al momento de cooperar y ansiedad por el temor de los comentarios que puedan recibir los aportes realizados.

Todo esto exige un control de la situación, y es aquí donde el facilitador encargado del tema junto al coordinador debe fiscalizar el ambiente de la discusión. El facilitador puede notificar al coordinador la situación, pero solo el coordinador es quien puede conocer la identidad del participante involucrado y decidir que hacer. El facilitador debe quedar fuera del proceso para no afectar la relación con el participante sancionado y el desarrollo del tema.

Con esto, la necesidad de evitar el ingreso de aportes sin que los participantes se identifiquen es clara. Ellos podrán decidir si el comentario está firmado o no, pero estarán



ligados en la Memoria Grupal a su aporte. Esta información estará disponible solamente para el coordinador de la pre-reunión.

#### 8.4.1.3 Notificación y Awareness

Los mecanismos de awareness son esenciales para los sistemas de soporte para grupos de trabajo. El objetivo es transformar las interacciones irregulares de los miembros del grupo en consistentes y perceptivas.

Es importante que los mecanismos para lograr esto mantengan a los miembros al tanto de lo que ocurre, avisando cuando ocurran eventos importantes y así permitir una actuación más consciente de su parte.

En sistemas “offline”, la noción del awareness está más relacionada con la pérdida de información y de contexto, debido a la interrupción de la interacción mientras un miembro del grupo se aleja de su estación de trabajo, lo cual es acrecentado por la falta de comprensión causada por la falta de interacción en tiempo real. Es por esto que la necesidad de enfrentar al usuario a los cambios ocurridos en la pre-reunión es clara.

Para el sistema a desarrollar se cuenta con las herramientas que provee JavaServer Faces, con lo que es posible lograr otras formas de despliegue y diferenciación, como el uso de viñetas y árboles, además de cambios en la página que señalen puntos que deben ser revisados o mensajes que deben ser atendidos.

#### 8.4.2 Monitoreando la Discusión

Uno de los principales objetivos es lograr que se desarrolle una discusión que entregue resultados útiles para la reunión. Esta es una tarea encargada al coordinador de la pre-reunión, y para lograr esto debe contar con funciones específicas que le permitan alentar a los participantes a que contribuyan, y así lograr enmendar los errores que puedan estar cometiendo.

Durante la discusión, el coordinador requiere herramientas que le indiquen como está la evolución de la discusión, desde un punto de vista amplio. Los niveles de participación de los contribuyentes (estadísticas de ingreso al sistema o la cantidad de contribuciones realizadas) así como la evolución de las ideas (cantidad de aportes en la primera etapa, o los comentarios al modelo sugerido en la segunda etapa) son ejemplos de la información que es necesaria.

En el desarrollo de la pre-reunión, el coordinador cuenta con herramientas para revisar la participación de los miembros del grupo, y a partir de esto comunicarse con ellos en caso de ser necesario. Esta comunicación será privada y tiene como objetivo el corregir y no castigar o exponer los problemas que pueda tener el participante en la pre-reunión.

#### 8.4.3 Manejar el Desarrollo de la Discusión

Es la primera etapa donde el manejo de la discusión es más crítico, considerando el número de usuarios que están involucrados y la falta de estructura.

Aunque la discusión en si no tiene mayor orden que una discusión entre personas, es necesario tener cuidado. Durante el proceso de discusión los límites iniciales pueden no ser los más adecuados. Un síntoma de esto es encontrar duplicados elementos en la discusión, causado por la relación muy cercana entre los temas de la agenda. Otra circunstancia es el espacio creado por la ausencia de tópicos relacionados, lo cual llevará al contribuidor a errores o contribuciones extremadamente largas.

Con las herramientas que brinda el sistema, como son el manejo de los hilos de conversación que puede realizar el facilitador, y la información individualizada con la que cuenta el coordinador, es posible manejar esta etapa en forma adecuada, siempre recordando que la falta de estructura es un punto clave.

En la segunda etapa, donde la estructura de IBIS se hace presente, la participación de los usuarios se convierte en comentarios que ayudan a perfeccionar esta estructura, pero en donde su modelamiento y sus cambios posteriores están en manos del facilitador, haciendo su evolución más manejable.

#### 8.4.3.1 Agregando Nuevos Participantes

Puede ser necesario agregar nuevos participantes, para que el desarrollo de la discusión sea mejor. Estos participantes son requeridos por su experiencia o por la presencia de un nuevo participante en la reunión posterior. También puede ser útil por la necesidad de llenar un vacío en la discusión y que el nuevo participante juegue papeles informales que tengan como objetivo el despertar la interacción entre los participantes. El coordinador es el encargado de manejar la lista de participantes, y al ser agregados su primera tarea es adquirir la información que ha sido volcada en la discusión, para esto es necesario revisar lo que encuentre en la Memoria Grupal. Este ingreso puede ser útil tanto en la primera o segunda etapa, que es donde el modelo IBIS final toma forma.

#### 8.4.3.2 Madurez de la Discusión

El proceso de discusión dentro de un tema está manejado por el facilitador. Si este considera que el tema en cuestión ya ha sido discutido en su totalidad puede solicitar su paso a la siguiente fase, el coordinador revisa el estado de la discusión, que será descrito por el facilitador (dada la falta de estructura en la primera etapa y la clara necesidad de un seguimiento de su desarrollo), y decide el paso a la siguiente fase.

Es también el coordinador quien decide si las tareas que se encargaron en la segunda fase cumplen con los requerimientos, esto junto a un reporte del facilitador, y así dar por terminado el tema. Cuando todos los temas estén terminados, la pre-reunión estará finalizada.

En el caso de que no hayan contribuciones o sean muy pocas, es necesario que el coordinador o el facilitador tomen algunas acciones para motivar la participación.

# Capítulo 9: Desarrollo del Sistema

---

## 9.1 Modelo de Datos

Para el desarrollo de la aplicación es necesario considerar la estructura de la información a manejar.

Viendo las características necesarias para los usuarios como el objetivo del sistema, y considerando la revisión que se realizó a SISCO, se llegó a un modelo de datos que permite el desarrollo de las etapas de LIBIS (Figura 9.1).

Esencialmente el modelo toma como base el usado en SISCO, en cuanto al manejo de los usuarios y las pre-reuniones.

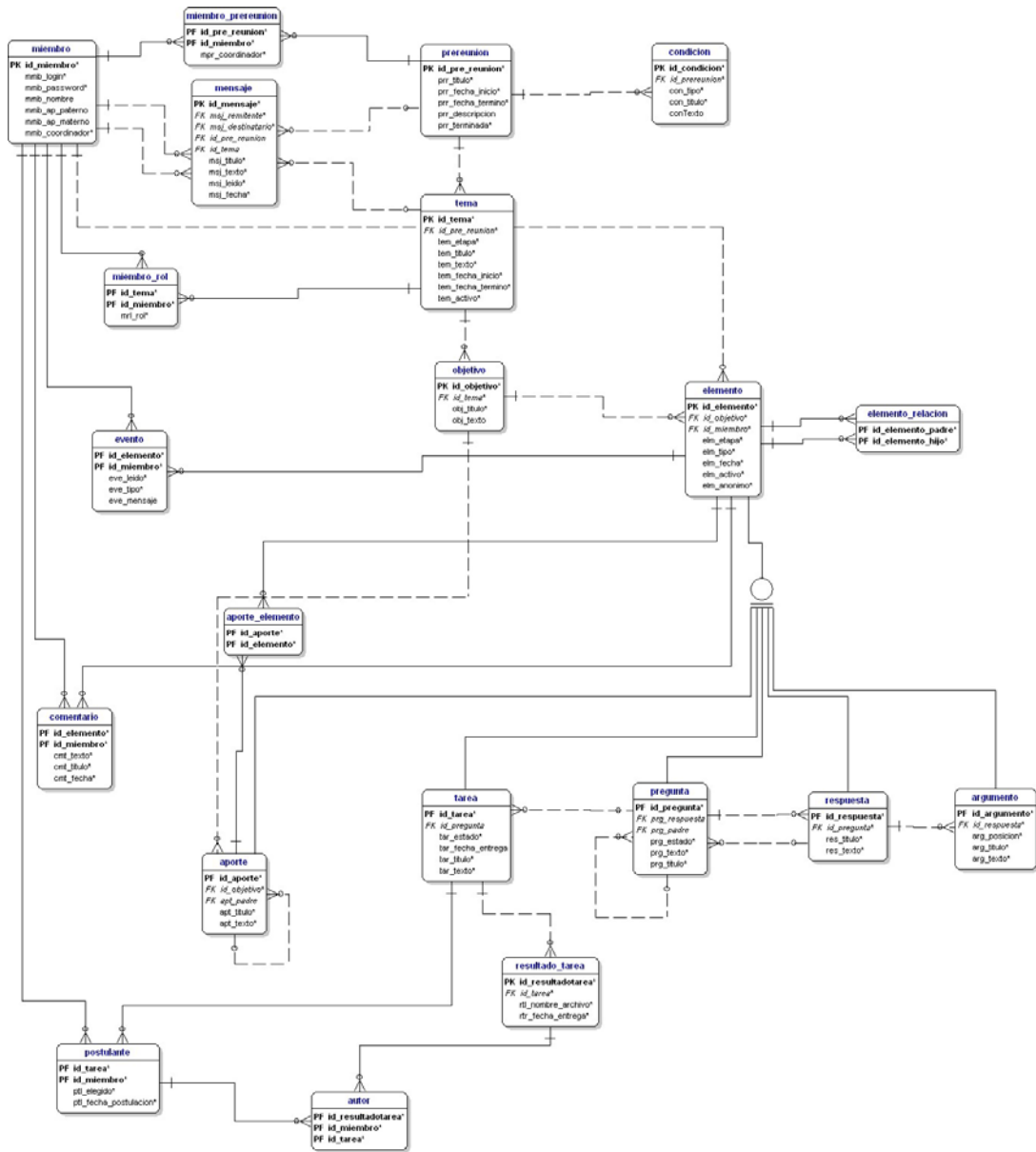


Figura 9.1 : Modelo de Datos

Una sección especial dentro del modelo es la que se refiere al manejo de los elementos de cada objetivo y su relación con el resto del modelo. Esta sección del modelo se puede dividir en tres partes, cada una de las cuales maneja una de las tres etapas del modelo LIBIS.

Un punto importante de considerar es el motor de base de datos a utilizar. Para este propósito existen dos posibilidades.

- MySQL: Una base de datos que tiene como mayor virtud su rapidez y simpleza, también tiene opciones para trabajar con transacciones.
- PostgreSQL: Una base de datos que presenta muchas prestaciones de las bases de datos comerciales.

La base de datos a utilizar es PostgreSQL 8.2, debido a sus características que aseguran un mejor manejo de los datos y la posibilidad de trabajar con transacciones, además de la seguridad que brinda.

- PostgreSQL posee una gran escalabilidad y permite un ajuste óptimo a la arquitectura del sistema (CPU, memoria), haciéndole capaz de soportar una mayor cantidad de peticiones simultáneas de manera correcta (en algunos benchmarks se dice que ha llegado a soportar el triple de carga de lo que soporta MySQL).
- Implementa el uso de rollback's, subconsultas y transacciones, haciendo su funcionamiento mucho más eficaz, y ofreciendo soluciones en campos en las que MySQL no podría.
- Tiene la capacidad de comprobar la integridad referencial, así como también la de almacenar procedimientos en la propia base de datos, equiparándolo con los gestores de bases de datos de alto nivel, como puede ser Oracle. También presenta la posibilidad de trabajar con objetos dentro de la base de datos.

#### 9.1.1 Manejo de Usuarios y Permisos

La intención detrás del manejo de usuarios es permitir asociar a los usuarios con distintas pre-reuniones (Figura 9.2), pero contando con una estructura para la obtención de esta información de una manera simple para realizar las consultas a la Base de Datos.

Pero el objetivo principal es el de poder administrar el tipo de relación que tienen los usuarios con los distintos temas que forman cada pre-reunión. Esta relación define las acciones que puede realizar dentro de cada pre-reunión en el sistema.

Cada usuario puede participar en más de una pre-reunión, así como cumplir distintos roles dentro de cada uno de los temas de cada pre-reunión.

Lo que se busca es permitir administrar esta relación de manera independiente para cada tema y así lograr que la aplicación presente los datos de la forma más simple y adecuada para el usuario.

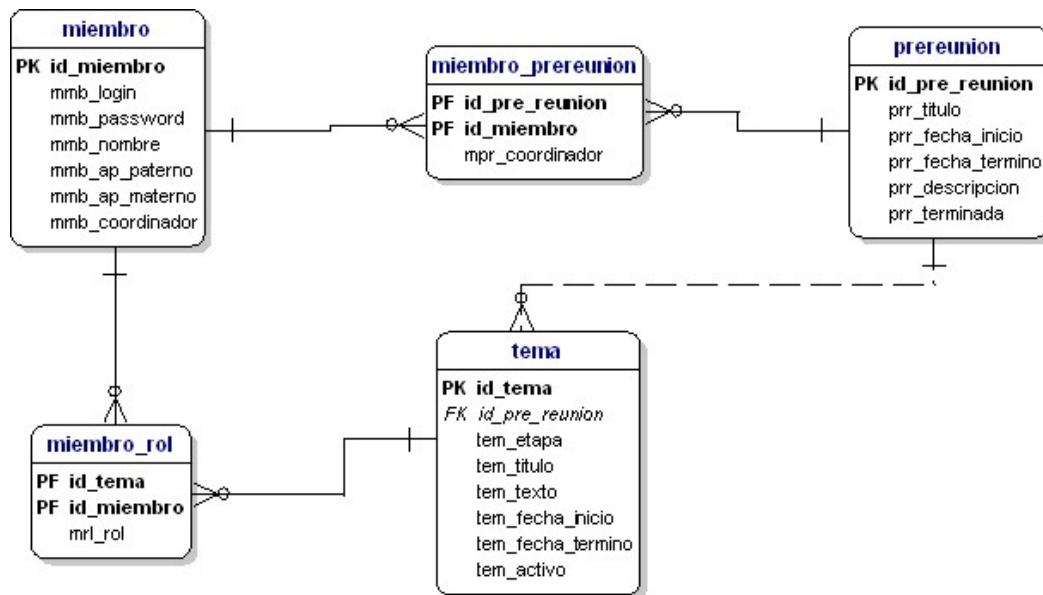


Figura 9.2 : Administración de Usuarios

Dentro de esta idea la definición de roles en la relación entre los miembros de la pre-reunión y los distintos temas es una manera de administrar estos permisos de manera centralizada y simple, y así poder asociar a los usuarios con roles que pueden realizar acciones diferentes.

### 9.1.2 Manejo de Pre-Reuniones y la Jerarquía de sus Elementos

El manejo de la información de la pre-reunión está basado en la división de esta en temas, y cada tema se divide a su vez en objetivos. De esta forma la pre-reunión se ordena y logra separar en forma clara la discusión que llevarán a cabo los participantes (Figura 9.3).

La definición de las condiciones define el marco en el cual se desarrolla la pre-reunión, esto permite poner a los usuarios en conocimiento de las reglas que deben cumplir sus aportes.

El uso de roles para cada tema permite la definición de facilitadores, que serán responsables de los objetivos asociados y el desarrollo de las discusiones dentro de las preguntas que nazcan de cada objetivo.

Otro detalle son los mensajes que se envían de cada pre-reunión. Las restricciones que se apliquen a los usuarios para decidir si pueden enviar mensajes dependen del ambiente desde el que se envíe el mensaje.

En el sistema se pueden distinguir dos ambientes, el primero fuera de la discusión, donde se manejan las pre-reuniones, y uno interno, donde se manejan los mensajes de cada pre-reunión en las que participa el usuario, estos mensajes pueden estar relacionados con la pre-reunión o con un tema particular dentro de la pre-reunión. Además están los mensajes que no tienen relación con algún elemento de la discusión.

Para esto podemos usar:

- Manejar cada tipo de mensaje en una tabla separada
- Manejar los tipos de mensajes dentro de una tabla y definir tipos de mensajes.

La primera solución permite una mejor definición de los elementos que deben estar presentes en los mensajes, cada tipo de mensaje necesita distintos elementos para poder explicar su contenido de una forma más clara. Otra ventaja es la posibilidad de agregar al modelo otros tipos de mensajes sin necesidad de modificar el modelo. Un problema que trae son los reportes que necesita el coordinador o el facilitador, las diferencias entre los tipos de mensajes puede hacer complicada su despliegue en pantalla como las consultas necesarias para obtener la información.

En la segunda solución, la ventaja es contar con un formato definido, la capacidad de agregar nuevos tipos de mensajes sin modificar el modelo, y reportes más simples de satisfacer. Los problemas están ligados a la rigida estructura que deben tener, aunque aún es posible agregar nuevas tablas, pero esto sale del diseño pensado para solucionar este problema

En este caso, debido a su simpleza y pensando que en general el espíritu del modelo es ese, la segunda opción es la más adecuada.

En el caso de ser generados en el ambiente del sistema, los mensajes serán entre usuarios, no están ligados a una pre-reunión en particular, solo el coordinador es capaz de enviar estos mensajes a los usuarios, los usuarios solo pueden responder o enviar mensajes al coordinador.

Estos mensajes también pueden estar relacionados a alguna pre-reunión en particular, en este caso los mensajes que envíen los participantes de la pre-reunión están relacionados con esa pre-reunión.

En el caso de realizarse dentro del ambiente de un tema, existe un nuevo rol que puede participar, los facilitadores pueden enviar mensajes al coordinador, a los participantes y a otros facilitadores del tema. En el caso de los mensajes a otros facilitadores, los mensajes serán enviados a todos, no podrá elegir un destinatario específico para evitar el desconocimiento de las decisiones tomadas por algún facilitador.

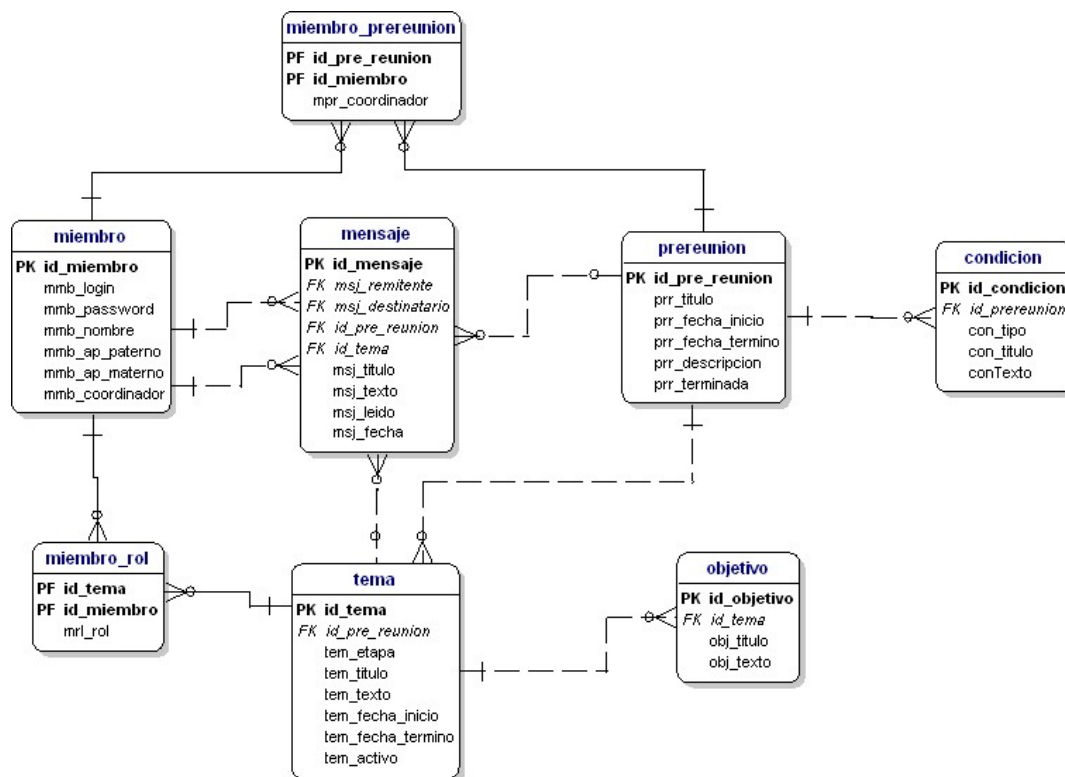


Figura 9.3 : Modelo de una Pre-Reunión

### 9.1.3 Administración de las etapas de LIBIS

Para las etapas de LIBIS, el uso de los elementos usados en SISCO es lógico, considerando que como resultado del desarrollo de la discusión se espera obtener el esquema de IBIS. Pero para lograr manejar las dos etapas previas, es necesario agregar nuevos elementos al sistema.

Para organizar la información de estas etapas, y lograr una estructura que permita relacionar estos distintos elementos entre si, el uso de sub-entidades que se relacionarán entre si, dependiendo de la estructura que se quiera dar a la discusión (Figura 9.4), es una solución que sigue el esquema original del modelo.

La alternativa a esta opción es organizar el modelo de datos de manera de separar totalmente las tres etapas del proceso, obligando a la creación de nuevas entidades que representen cada etapa, las que manejarán sus elementos.

Uno de los motivos que llevó a la elección de la primera opción fue la simplicidad del modelo y la posibilidad de realizar relaciones entre los elementos de cada etapa de manera simple, y así lograr un proceso más sencillo al momento de generar reportes e informes del desarrollo de la pre-reunión. Esta opción se complica al momento de separar las etapas.



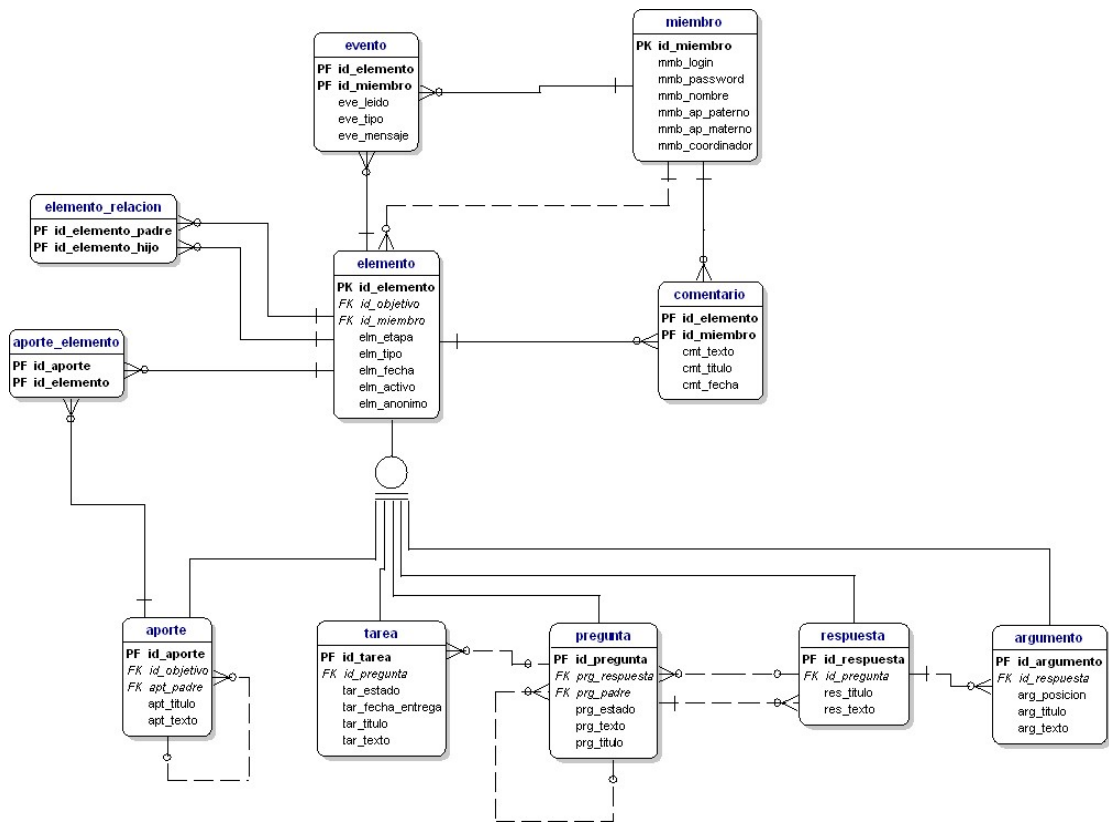


Figura 9.4 : Estructura de Elementos de LIBIS

Otro punto importante es la tabla *elemento\_relacion*. Esta tabla será administrada por el facilitador, proporcionándole una forma simple de unir distintos elementos entre las etapas, y así lograr una navegación simple entre los elementos relacionados.

### 9.1.3.1 Primera Etapa: Discusión inicial de los participantes

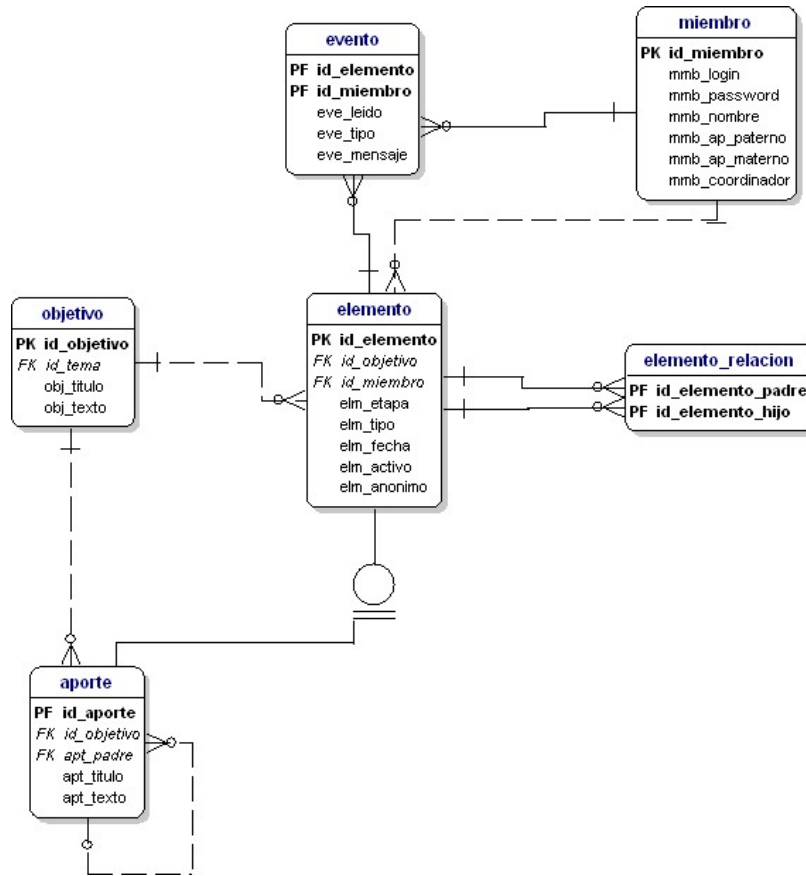


Figura 9.5 : Primera Etapa

En la primera etapa del proceso de LIBIS la estructura de la discusión es muy simple. La idea es llevar una “conversación normal”, desde el punto de vista de los usuarios. La estructura de esta es simple y fácil de usar, ya que no es necesario que sus aportes cumplan con regla alguna (Figura 9.5).

En la tabla aporte se guarda el árbol de la discusión, los participantes podrán agregar aportes a voluntad, lo mismo que responder a los aportes ya ingresados.

### 9.1.3.2 Segunda Etapa: Preparación y discusión del modelo preliminar IBIS

Esta es la etapa que involucra más al facilitador en un papel activo, y requiere que su relación con la discusión realizada en la etapa anterior sea cercana y que comprenda el ambiente en la que se desarrollo.

Los participantes en esta etapa solo son observadores con derecho a voz, lo que si no se maneja bien puede producir algunos problemas de comunicación y “sobrecarga” de opiniones.

El facilitador agregará en un comienzo el modelo preliminar, cuyo nivel de detalle dependerá de su criterio y del nivel que haya tomado la discusión en la etapa anterior. Si el nivel fue bajo y no apporto todas las ideas que eran deseables, a pesar de cumplir con los

requerimientos, es posible ingresar un modelo incompleto que fuerce la participación en la segunda etapa. Si en cambio, el nivel de la discusión fue bueno, solo es necesario “traducir” esta discusión al modelo de IBIS y revisar los comentarios que puedan surgir.

Una vez publicado el modelo la tabla “comentario” comenzará a recibir las contribuciones de los usuarios. Gradualmente el facilitador irá editando los elementos considerando las opiniones de los usuarios.

Otro elemento importante es la postulación de las tareas que se vayan definiendo durante la discusión, tareas que serán definidas y a las que se les asignará un conjunto de postulantes, definidos por el facilitador u obtenidos a través del ofrecimiento voluntario de los participantes del tema.

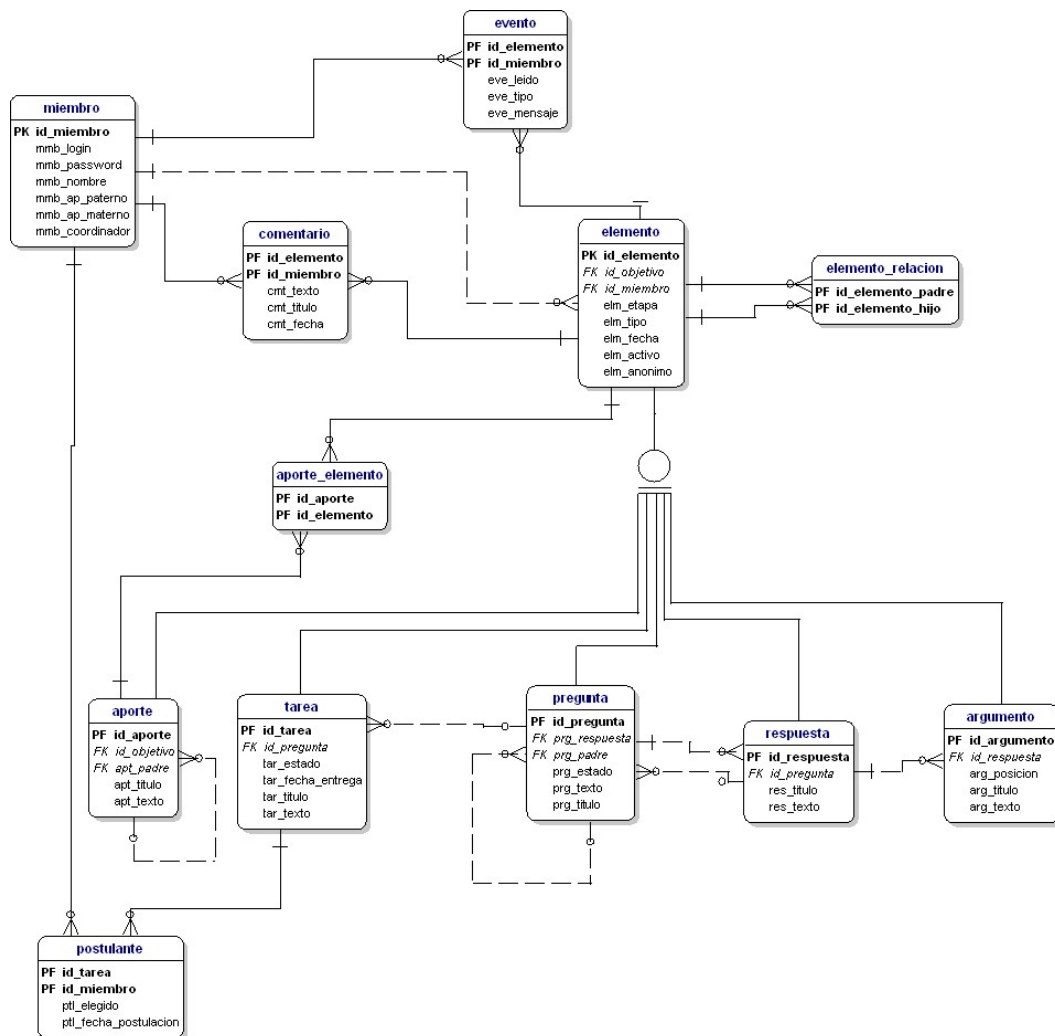


Figura 9.6 : Segunda Etapa

### 9.1.3.3 Tercera Etapa: Obtención del modelo IBIS de la pre-reunión

En esta etapa el modelo IBIS ya está definido como resultado de la discusión de la etapa anterior. Este será publicado y los usuarios podrán revisar el resultado navegando por las preguntas-respuestas-argumentos.

Es en esta etapa donde se espera que las tareas asignadas se ingresen al sistema para que estén a disposición de los participantes. La etapa se dará por terminada cuando se cuente con todo el material necesario para la reunión.

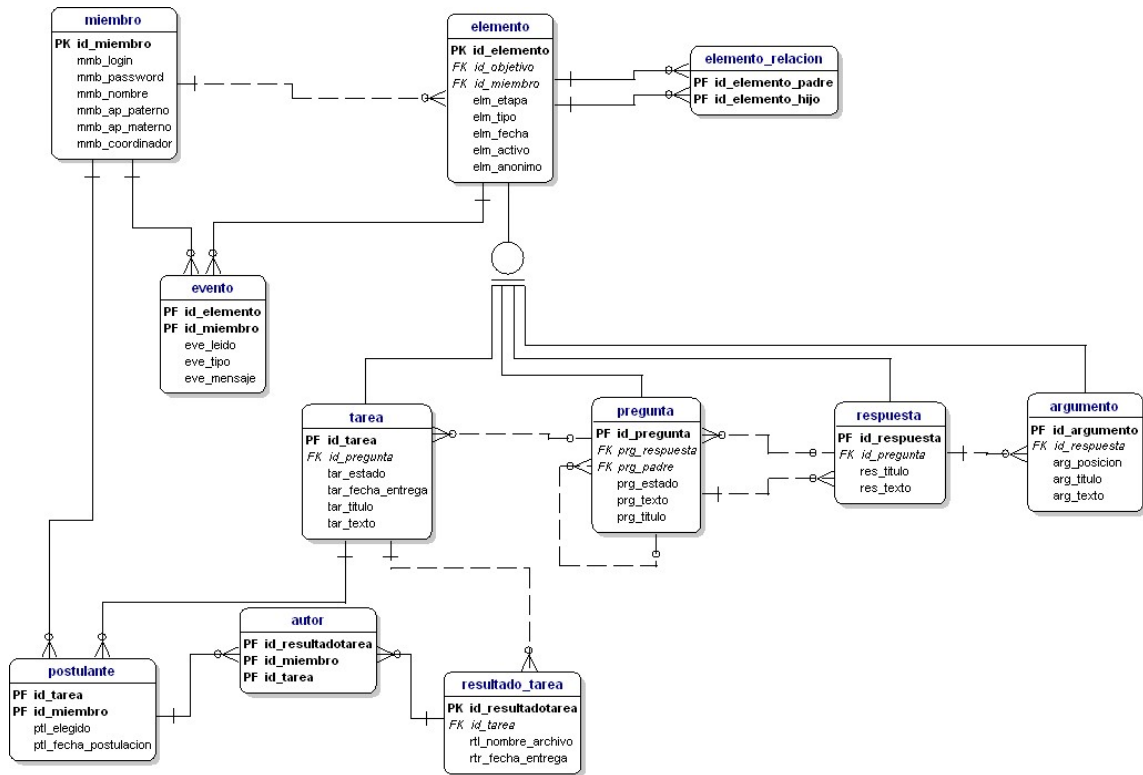


Figura 9.7 : Tercera Etapa

## 9.2 Diseño de las Páginas

Uno de los puntos más importantes es lograr un diseño que sea útil y sencillo. Los usuarios están ya habituados a modelos de interfaces que se han hecho comunes.

El problema con esto son las limitaciones que presenta el ambiente de desarrollo. Las aplicaciones web están limitadas por las capacidades que presentan los navegadores.

Gracias a las características de JavaServer Faces y los componentes que facilita la herramienta de desarrollo Sun Java Studio Creator 2 ([20]), el diseño de las páginas logra que los usuarios encuentren un ambiente conocido y fácil de utilizar.

### 9.2.1 Navegación en el sitio

La forma de navegar dentro de las opciones que brinda la aplicación debe permitir un movimiento simple dentro del sitio.

La solución debe mostrar todas las opciones en cada momento, para que el usuario pueda llegar a todas las funciones de manera simple, así como permitir saber en todo momento en que lugar del sitio se encuentra.

Hay que considerar que existen dos ambientes en donde el usuario se mueve. El primero es en el ámbito de las pre-reuniones y su estructura, y el segundo es en las discusiones en sí.

El primero cuenta con opciones de navegación limitadas, donde básicamente se encuentran las secciones de la aplicación, como son los mensajes, las pre-reuniones, y para el usuario coordinador, la administración de las pre-reuniones y los usuarios. Las opciones de administración de las pre-reuniones, debido a que son estructuras más complejas, cuentan a su vez con opciones que requieren de interfaces distintas, muy similares a las requeridas por las discusiones.

Para la segunda pantalla las opciones dentro de la página son más simples, pero es necesario recorrer los elementos que forman la discusión del objetivo, lo que implica una estructura que crece con los aportes de los usuarios y requiere una gran cantidad de espacio para poder navegar libremente en ella.

Las opciones para navegar en el sitio son:

- Utilizar árboles para modelar las opciones y su estructura.
- Utilizar menús convencionales, que permitan navegar dentro de los niveles y utilizar una ruta en la página para poder moverse entre los niveles de navegación.
- Usar un esquema maestro detalle para navegar dentro de las opciones.
- Utilizar fichas (“tabs”) para colocar las opciones.

El uso de árboles permite una navegación simple, así como la posibilidad de ver toda la estructura de navegación, permitiendo un movimiento rápido dentro del sitio. Su problema es que frente a mucha información, con muchos niveles, el mostrar toda esta información puede hacer difícil encontrar lo que estamos buscando dentro del árbol.

Los menús convencionales son algo conocido por los usuarios, están presentes en gran parte de los sitios web y son fáciles de comprender, además de que la herramienta de desarrollo permite utilizar ayuda emergente para los beans que pueden ser usados para implementarlos. El problema es que el tipo de navegación, así como la dificultad para mostrar la información de los niveles superiores los hacen una opción que hace difícil un movimiento ágil dentro del sistema.

El uso de formularios *maestro-detalle* es algo que presenta características de navegación simple, es conocido por los usuarios y, si bien no muestra toda la información de la navegación hasta el punto donde nos encontramos, permite ver gran parte del camino y las opciones en cada nivel, para movernos de manera simple. Un problema que presenta es la cantidad de espacio en la pantalla que necesita para mostrar la información.

La utilización de fichas permite un uso sencillo por parte del usuario, además de ser un mecanismo de navegación simple, que muestra todas las opciones y que permite anidar y crear un árbol de navegación al anidar conjunto de fichas. Cada conjunto de fichas muestra distintas fichas, las que pueden contener más fichas, lo que permite que el árbol de

navegación de pueda recorrer fácilmente, además de maximizar el espacio que se puede utilizar en el despliegue de los paneles que están en cada ficha. Los problemas que presenta es la gran cantidad de espacio que requiere para mostrar la opciones, lo que limita la cantidad de elementos que puede haber en cada nivel debido a la limitación que implica el ancho de la página, así como el problema que se crea cuando el árbol de opciones tiene muchos niveles, lo que produce el uso de gran parte de la pantalla para mostrar las opciones.

Considerando los dos tipos de opciones que presentan las pantallas, las opciones que satisfacen mejor las necesidades son el uso de paneles para los menús reducidos y definidos, como las opciones dentro de las pantallas, y el uso del esquema maestro detalle para la navegación en cada una de las etapas (Figura 9.8 y Figura 9.9).



Figura 9.8 : Pantalla de inicio

Otro punto importante dentro de la navegación es la forma en la que las distintas discusiones serán presentadas.

- Al seleccionar una nueva discusión, esta se muestra en la misma pantalla que muestra el estado del sistema.
- Para cada discusión, se abre una nueva ventana.

La primera opción presenta el inconveniente de perder la visión global del sitio, pero facilita el manejo de las variables de la sesión.

The screenshot displays the 'Sisdela' web application interface. At the top, the logo 'Sisdela' is prominent, followed by the user name 'Christian Navarro Gatica' and the page title 'Problemas de producción'. A navigation bar contains tabs for 'Eventos', 'Primera Etapa', 'Segunda Etapa', and 'Tercera Etapa'. The main content area is split into two panels: 'Preguntas' and 'Respuestas'. Each panel features a list of discussion items on the left and a detailed form on the right. The 'Preguntas' form includes fields for 'Título' and 'Descripción', a 'Pregunta padre' checkbox, an 'Activa' checkbox, and buttons for 'Agregar', 'Modificar', and 'Eliminar'. The 'Respuestas' form also has 'Título' and 'Descripción' fields and 'Agregar', 'Modificar', and 'Eliminar' buttons. A dropdown menu for 'Repuestas' is located below the question list, and another for 'Argumentos a favor' is below the answer list.

Figura 9.9 : Estructura de la página de una discusión

En cambio la segunda permite al usuario manejar distintas discusiones de manera paralela sin descuidar otras discusiones y más aún, permitiéndole acceder a la página central del sistema sin necesidad de abandonar la discusión.

Sin dudar, la segunda opción es la más atractiva, pero cuenta con un problema, en la aplicación, el manejo del bean de la sesión será complicado, al momento de guardar los datos de todas las ventanas abiertas y todas las acciones desarrolladas.

Para solucionar esto, los datos de cada ventana son manejados a través de request beans, que son beans especiales que tienen el único propósito de transmitir datos entre páginas. De esta forma, cuando se carga nuevamente la pantalla en una de las discusiones, la información necesaria para rescatar los datos de la discusión se entregan por este medio al recargar la página.

Este problema lleva a otro punto importante, los datos utilizados para reconocer usuarios, permisos y pre-reuniones. Estos datos definen la navegación dentro del sitio, pero su

modificación es poco habitual y una consulta a la base de datos consume recursos en el servidor, sobre todo considerando el volumen de conexiones que posiblemente deba soportar.

Nuevamente los beans ayudan a sacar esta información de la sesión. En el servidor de aplicaciones se encuentran los application beans, que permiten mantener información disponible para todas las sesiones mientras el servidor de aplicaciones se encuentre funcionando.

La solución al problema del manejo de los datos es mantener esta información en el bean de la aplicación, en este caso se utilizó una lista para almacenar el resultado de las consultas.

### 9.2.2 Despliegue de las opciones

La elección de la forma de navegar entre las distintas páginas del sistema define en cierto grado el modo de desplegar las opciones que presenta el sistema.

La decisión que hay que tomar es como mostrar las distintas opciones que presenta el sistema, considerando que existen distintas opciones para los distintos roles, y la característica que presentan las fichas de poder anidarlas para presentar nuevas opciones.

Como ya se decidió, las opciones que se presentarán a través de fichas serán las que están acotadas y definidas, es decir, los menús de la aplicación. El problema es que en general recorrer muchos niveles puede provocar problemas.

Aún con la posibilidad de recorrer las opciones de manera sencilla, el tener que salir de una ficha para poder ver la información que se necesita para ingresar o modificar la información del sistema es engorroso y molesto, además de un problema al momento de manejar las transacciones que se deben llevar a cabo con la base de datos. Es deseable no contar con más de tres niveles en el árbol de fichas. Soluciones a este problema son:

- El uso de pantallas emergentes que presenten las opciones que se pueden elegir y así poder ver la información necesaria y decidir en la otra ventana.
- Utilizar los paneles para, al usarlos anidados, presentar distintas opciones en cada panel, esto se puede hacer con el uso de un listbox con las acciones que se pueden realizar y manejando las opciones de los paneles, activando y desactivando sus propiedades (rendered y visible).

La primera opción es sencilla y con el uso de las opciones que brindan los beans de sesión que usa el Studio Creator se puede compartir la información con estas ventanas, permitiendo que sus acciones sean llevadas a la base de datos sin problemas. El problema que presenta está más relacionado con la percepción que tendrá el usuario con este método. El uso de distintas ventanas junto con las capacidades que presenta el servidor de aplicaciones permite que, a pesar de tener abierta una ventana para editar opciones presentes en la ventana principal, el usuario puede cambiar la ficha y abrir otra ventana. Si bien esto es algo extremo, no es menos cierto que es una posibilidad. Otro inconveniente es el uso del



espacio y el hecho que muchas ventanas son un problema en una pantalla que puede llegar a sentirse “pequeña”.

La utilización de paneles en cambio, permite realizar las decisiones dentro de un ambiente donde todas las variables que pueden servir para tomar las decisiones están disponibles en todo tiempo, además de obligar al usuario a decidir que hacer antes de cambiar de ficha, lo que evita inconvenientes al momento de navegar en la aplicación. Si bien la interfaz de la aplicación puede verse sobrecargada en algunos casos, el orden y la utilización de paneles anidados junto con los listbox pueden dar al usuario la posibilidad de manejar mejor la información y las opciones desplegadas.

The screenshot shows the 'Sisdel' application interface. At the top, the user is identified as 'Christian Navarro Gatica' and the current page is 'Problemas de producción'. There are four tabs: 'Eventos', 'Primera Etapa', 'Segunda Etapa' (which is active), and 'Tercera Etapa'. The main content area is divided into two sections. The top section is for adding a new question or comment. It has a 'Preguntas' sidebar with a listbox containing 'Problemas con el sistema actual de pedidos' and 'Efectos generados por los problemas'. The main form has fields for 'Título' (Title) and 'Descripción' (Description). The title is 'Problemas con el sistema actual de pedidos' and the description is 'El sistema actual de pedidos no logra satisfacer la demanda de los clientes'. There are buttons for 'Agregar', 'Modificar', and 'Eliminar', and a checkbox for 'Activa'. The bottom section is for editing an existing problem. It has a 'Cambiar problema' sidebar with a listbox containing 'Cambiar problema'. The main form has fields for 'Título', 'Autor', and 'Descripción'. The title is 'Cambiar problema', the author is 'Christian Navarro Gatica', and the description is 'El problema es el sistema actual de pedidos, sino una mala gestión en las ventas'.

Figura 9.10 : Cambio de panel an ingresar un nuevo comentario

El uso de paneles simplifica tanto la percepción que tiene el usuario de que hacer y donde hacerlo, como la navegación dentro de opciones que dependen de valores externos a los elementos del formulario, que permiten tomar las decisiones de una forma más sencilla. En la Figura 9.10 se ingresa un comentario al problema.

## 9.3 Herramientas del sistema

### 9.3.1 Servicio de comunicación entre usuarios

Uno de los puntos importantes dentro de un sitio como este son las posibilidades de comunicación y las condiciones a las que esta sometida. En un sistema así, la comunicación debe seguir con el modelo de interacción, la comunicación asíncrona, que si bien no permite una buena interacción entre los usuarios, permite un registro y un seguimiento de esta comunicación.

Como ya se discutió al analizar el modelo de datos, los usuarios tienen distintas opciones de comunicación dependiendo del lugar en el sitio en el que se encuentren, como el rol que tengan en ese lugar

En este caso, la comunicación está destinada a satisfacer la comunicación con fines de controlar y corregir el camino que está tomando la discusión. Como se dijo antes, el sistema está destinado solo a la información requerida para el desarrollo de la pre-reunión, la interacción entre los usuarios se desarrolla fuera del sistema y por medios habituales (encuentros, conversación telefónica, correo, etc.).

En la jerarquía de roles, el coordinador está encargado de obtener un buen resultado para la reunión que se realizará posteriormente. Este rol es el que posee la capacidad de comunicarse con los participantes directamente a través del sistema a través de mensajes, para que esto quede registrado pero no sea de dominio público.

El facilitador también necesita contar con herramientas de comunicación dentro del sistema, tanto con los participantes del tema como con los otros facilitadores del tema, pero dada la necesidad de no teñir su relación con los participantes con un tinte de fiscalizador (el objetivo de este rol es facilitar la comunicación), la comunicación destinada a regular e informar el comportamiento y los resultados que se observan en el desarrollo de la discusión solo está disponible para el coordinador de la pre-reunión.

La información ingresada en las discusiones que ocurren dentro de cada objetivo son catalogadas como eventos y se gatillan al momento de ingresar. Este evento es informado a los miembros de esta discusión de forma automática. En este nivel no existen los mensajes entre usuarios, toda comunicación se realiza a través de la discusión, por lo que está limitada al objetivo relacionado y todo elemento ingresado que sea considerado fuera del objetivo será eliminado por el facilitador encargado.

Otro punto importante entre la comunicación de los usuarios son las contribuciones anónimas. Es necesario contar con este tipo de aporte, debido a las presiones que existen en un grupo de trabajo, pero esto lleva a exponerse a posibles ataques o contribuciones fuera del ámbito de la discusión. Las opciones para solucionar esto son dos:

- Permitir que los usuarios aporten en forma anónima, siendo revisada posteriormente por el facilitador o el coordinador.
- Cada contribución anónima debe pasar por un control antes de ser publicada.

La primera opción, si bien permite que estas contribuciones aporten a la discusión en una forma más fluida y permitan un avance más rápido, implican un riesgo y una posible pérdida de confianza en los resultados que se pueden obtener por este medio.

La segunda opción tiene el problema de hacer que las contribuciones que realmente pueden aportar en forma significativa a la discusión se mantengan al margen y provoquen una pérdida de tiempo, e incluso que al ser ingresadas a la discusión, su aporte ya no sea el mismo, debido a que la discusión tomo otro rumbo y el volver al punto en la que el aporte fue hecho sea poco viable.

En este caso es más sensata la segunda opción, a pesar del riesgo que conlleva. Para reducir este riesgo se hace necesario brindar al facilitador la opción de examinar y decidir el destino de este tipo de contribución sin la necesidad de pasar por las manos del coordinador.

Si bien esto puede parecer una carga mayor que la que el facilitador puede soportar, se hace más clara la utilidad de permitir la participación de más de un facilitador por tema. Esta opción también tiene la ventaja de poder someter la contribución a un juicio entre distintas personas, lo que evita que la discusión se vea afectada por las decisiones, que a pesar de mucho evitarlo, son al final de cuentas subjetivas.

### 9.3.2 Awareness

Ya se explicó el medio de comunicación entre usuarios y la diferencia entre estos y los eventos en las discusiones.

Si bien los mensajes son elementos importantes dentro del sistema para lograr guiar el desarrollo de la pre-reunión, los mensajes más importantes son los eventos. Son estos los que permiten que el desarrollo de las discusiones sean útiles al propósito del sistema, al permitir que la participación de los usuarios se produzca de manera más rápida y que tengan una visión clara de los movimientos que ocurren en la pre-reunión.

Si los usuarios no son informados de los cambios no pueden saber donde es necesaria su atención, donde ocurrió un cambio que puede cambiar su percepción de la discusión.

El problema es que estos mensajes están asociados a la información de la discusión que se realiza para un objetivo en particular, y el usuario ingresa al sistema donde la información mostrada es más general (mensajes del coordinador/facilitador) asociada al sistema, a una pre-reunión o a un tema. Para solucionar este problema las opciones son:

- Tener un tipo especial de mensaje que avise al usuario para que revise un objetivo en particular.
- Despliegue de un mensaje en pantalla indicando cambios en un objetivo, una lista con avisos.

La primera opción se ajusta dentro del modelo de la aplicación, sin separarse de los medios de comunicación ya definidos, el principal problema es como lograr que el usuario logre al momento de su entrada al sistema estos mensajes.

La segunda opción presenta la ventaja que el usuario no puede evitar el notar el suceso de estos eventos. El problema es conseguir un método que no salga de la interfaz ya elegida. Al momento de escoger el uso de fichas, el hecho de permitir un mejor uso del espacio disponible en la pantalla fue un punto importante.

En este caso, ambas soluciones presentan ventajas, la primera es su forma de presentar los mensajes al usuarios, y la segunda, la imposibilidad por parte del usuario de no notar el evento.

En este caso la elección es una combinación de las dos opciones. La herramienta de desarrollo usada permite el uso de fragmentos de página, lo que permite compartir en distintas páginas un fragmento común con funcionalidades que están disponibles para toda pantalla que las incluya, aunque presentan algunas limitaciones como la imposibilidad de usar formularios virtuales (ver [20]), pero en este caso la solución no los necesita.

En el fragmento de página se pondrá un aviso llamativo (usando un color fuerte y una fuente llamativa) indicando la necesidad de revisar los mensajes recibidos de las discusiones.

La gran ventaja de esta opción es que, al utilizar este fragmento en todas las páginas de la aplicación, este mensaje será notificado al usuario en todo momento.

Esta notificación es un link que lleva al usuario a la página principal, el la ficha de los mensajes y con las lista de mensajes de eventos seleccionados. El aviso no desaparecerá hasta que todos los eventos hayan sido revisados.

El aviso indica donde se ha producido el cambio y el momento en el que ocurrió, además al tratarse en un tipo de mensaje especial, el usuario dispone de un modo de llegar a la discusión en forma directa para revisar el cambio que gatillo el mensaje, una vez hecho esto, el mensaje para ese evento ya no estará activo.



Figura 9.11 : Después de ingresar el comentario el sistema avisa de un nuevo evento

Este mensaje estará disponible para todos excepto para el usuario que lo produjo (en el caso de la figura, el usuario supo de la existencia de un nuevo evento después de ingresar su comentario, lo que provocó que la pantalla se refrescara).

Esta forma de aviso también es usado para los mensajes tradicionales, pero el movimiento se realiza en la misma página.

### 9.3.3 Revisión de Discusiones

Para lograr un mejor manejo de las pre-reuniones, las discusiones son abiertas en ventanas distintas, para que el usuario pueda revisar más de una discusión al mismo tiempo. Debido al sistema de awareness usado, el usuario está atento a los cambios que ocurren en todas las pre-reuniones en las que participa.

Como se discutió antes, el árbol de elementos que forman la discusión son desplegados usando un esquema de maestro-detalle de tres capas, cada capa presenta los elementos relacionados al elemento maestro.

En cada nivel se presenta la información del elemento seleccionado, que dependiendo de cada rol presentará distintas acciones que se podrán realizar para cada elemento. Estas

acciones están disponibles a través de paneles que contienen las opciones, de esta forma se puede restringir las acciones de cada rol en el elemento (Figura 9.12). El panel mostrado se elige a través de la lista desplegable presente en cada nivel.

**Sisdel**

Christian Navarro Gatica Problemas de producción  
Nuevos eventos

Eventos Primera Etapa Segunda Etapa Tercera Etapa

**Preguntas**

Problemas con el sistema actual  
Efectos generados por los pro

Comentarios Repuestas Comentarios Tareas Administración

Título: Problemas con el sistema actual de pedidos

Descripción: El sistema actual de pedidos no logra satisfacer la demanda de los clientes

Pregunta padre  Activa Agregar Modificar Eliminar

Título: Cambiar problema

Autor: Christian Navarro Gatica

Descripción: El problema es el sistema actual de pedidos, sino una mala gestión en las ventas

Figura 9.12 : Opciones para un usuario administrador

En el caso de la segunda etapa:

- Preguntas
  - Respuestas: Muestra las respuestas asociadas a la pregunta seleccionada, dentro de este panel se encuentra otra lista desplegable para mostrar las opciones de las respuestas.
  - Comentarios: Formulario donde los usuarios pueden ingresar sus comentarios y donde el facilitador o el coordinador pueden leerlos.
  - Tareas: Con este formulario los participantes pueden postular a realizar una de las tareas propuestas por el facilitador.
  - Administración: Herramientas para que el facilitador o el coordinador manejen distintos elementos de la pregunta.
    - Tareas.
    - Administración de comentarios anónimos.
    - Agregar, modificar o eliminar respuestas.
    - Administrar comentarios.
- Respuestas
  - Argumentos a favor: Listado con los argumentos que apoyan la respuesta.
  - Argumentos en contra: Listado con los argumentos de están en contra de la respuesta.
  - Comentarios: Formulario donde los usuarios pueden ingresar sus comentarios y donde el facilitador o el coordinador pueden leerlos.
  - Administración:
    - Administrar comentarios anónimos.
    - Administrar argumentos.
    - Administrar comentarios.
- Argumentos
  - Comentarios: Formulario donde los usuarios pueden ingresar sus comentarios y donde el facilitador o el coordinador pueden leerlos.
  - Administración

- Administrar comentarios anónimos.
- Administrar comentarios.

En el caso de la tercera etapa, solo queda recibir las tareas que han sido asignadas. El modelo de la discusión ya está definido.

Un punto importante que se debe notar es la diferencia que tiene la primera fase. Al no contar con una estructura definida, la forma de desplegar la información se estructura como un árbol, por lo que la opción tomada es el uso de un esquema similar a un foro de discusión, para esto se utiliza una tabla con los datos de cada aporte y las herramientas para ver el detalle de cada aporte, así como la posibilidad de responder o de agregar un nuevo aporte en el nivel.

The screenshot shows a forum interface with a navigation bar at the top containing 'Eventos', 'Primera Etapa', 'Segunda Etapa', and 'Tercera Etapa'. Below this is a header for 'Nivel 2 (1 - 2 de 2)'. A table lists two posts:

Título	Autor	Fecha	Nº de Respuestas
Problemas en el proceso	Teresa Neira	09-05-2007	5
El sistema y sus efectos	Monica Puebla	11-05-2007	4

Below the table, there is a 'Propiedades del aporte' dropdown menu and a 'Volver nivel anterior' button. The detailed view of the selected post shows:

**Título**  
Problemas en el proceso

**Autor**  
Teresa Neira

**Descripción**  
Es necesario abordar los problemas en el proceso de recepción en el proceso de producción si planeamos expandir la capacidad de la fábricas. El uso de medios convencionales de producción puede ser un tema a discutir. En la fábrica 5 los problemas muestran que el proceso se debe revisar

Figura 9.13 : Propiedades del aporte y las respuestas a este aporte

Esta forma de discutir dificulta la búsqueda de elementos nuevos, ya que marcar el camino que se debe seguir para llegar al elemento buscado implica la posibilidad de navegar por muchos niveles y tener que marcar el árbol para que el usuario llegue a su destino. Es aquí donde la solución de comunicar los cambios en las discusiones a través de mensajes que lleven directamente al nuevo elemento evita estos problemas.



En la primera etapa los usuarios pueden aportar en un foro abierto, donde los usuarios pueden discutir sin seguir un esquema (Figura 9.13). Para responder al aporte se despliega un formulario que cuenta con los mismos elementos, además de la opción de hacer el aporte en forma anónima (Figura 9.14). El formulario de ingreso es igual, solo que se inserta como un elemento más de la tabla de respuestas al aporte

Los participantes de la discusiones, una vez terminada la primera etapa, pueden participar en el proceso a través de comentarios (ver Figura 9.10).

Título  
Problemas en el proceso

Autor  
Teresa Neira

Descripción  
Es necesario abordar los problemas en el proceso de recepción en el proceso de producción si planeamos expandir la capacidad de la fábricas. El uso de medios convencionales de producción puede ser un tema a discutir. En la fábrica 5 los problemas muestran que el proceso se debe revisar

Responder

\* Título

\* Descripción

Anónimo Guardar

Si selecciona esta opción su publicación estará sujeta a su contenido y al aporte que signifique al desarrollo de la discusión

Figura 9.14: Responder a un aporte en la primera etapa

#### 9.3.4 Revisión del Estado de los Temas y el Paso a la Siguiente Fase

El cambio de etapa es un punto crucial dentro del desarrollo de la pre-reunión, en especial el paso a la segunda etapa, donde se realiza el complejo trabajo de traducir todos los aportes en elementos IBIS.

El hecho que esta decisión de aprobar o no un tema repose solo en el coordinador implica una gran responsabilidad, debiendo estar al tanto del desarrollo de cada tema de la pre-reunión, por lo que la comunicación entre él y los facilitadores debe ser continua, y la obtención de reportes muy necesaria.

Estos reportes consideran, fuera de las observaciones del facilitador, estadísticas de participación y la revisión de la evolución de los temas.

#### 9.3.4.1 Estadísticas de Participación

Las estadísticas de participación muestran la cantidad de aportes y comentarios del usuario, para revisar que tan importante es su participación en la pre-reunión.

En este caso, el facilitador puede reportar este problema si es necesario al coordinador, quien puede decidir sacar al usuario de la pre-reunión.

Esta información se obtiene en forma de tablas que contienen la información de las consultas a la base de datos.

#### 9.3.4.2 Evolución de los Temas

Los temas tienen una fecha límite para terminar, cada una de las fases tiene una fecha tope, que debe respetarse, si esta fecha aún no llega pero los aportes o los comentarios son bajos o nulos, se puede decidir pasar a la siguiente fase si es que el facilitador considera que la discusión ha producido los resultados que esperaba, o solicitará la presencia de nuevos participantes para renovar la discusión dentro del tema.

#### 9.3.5 Administración de Usuarios y Pre-Reuniones

En el sistema es necesario manejar a los usuarios, agregar y eliminar usuarios del sistema. Para esto:

- Se puede contar con un rol de administrador
- El coordinador puede hacerse cargo de esta tarea.

La existencia de un rol administrador es poco necesario para el desarrollo de una pre-reunión, ya que el coordinador es quien maneja de principio a fin el desarrollo de la pre-reunión. Pero es necesario tener un rol que pueda manejar estas opciones.

Permitir que todo coordinador pueda administrar el sitio también es algo incómodo, pensando que en una organización grande pueden existir muchos coordinadores, y no es necesario contar con muchos administradores.

La solución es contar con coordinadores que puedan manejar usuarios y sus privilegios. Así, si es necesaria la participación de otro usuario coordinador-administrador, el sistema pueda contar con él.

## Capítulo 10: Conclusiones

---

En un comienzo, el uso de un sistema como IBIS me resultó interesante, considerando los problemas que he enfrentado al momento de tratar de organizar y lograr una comunicación entre los miembros de un equipo de desarrollo. Si bien esto se aparta del ámbito de la aplicación desarrollada, el proceso de toma de decisiones puede ser muy complejo si no se cuenta con el tiempo necesario para llegar a un acuerdo. La capacidad de comunicación entre las personas, que se nos hace tan natural, resulta ser demasiado complejo para poder llevarlo directamente a un medio distribuido.

Los mecanismos para lograr un trabajo en grupo ordenado y eficiente dependen de muchos factores, pero principalmente de las capacidades de las personas para lograr esa comunicación y de las herramientas para poder llevar a cabo estas tareas.

Al analizar las ideas detrás de LIBIS comprendí que esto último, si bien es necesario, puede lograrse a través de un proceso de aprendizaje, donde personas más entrenadas en el modo de interactuar en ambientes de este tipo, logran modificar la forma de relacionarnos al momento de trabajar en grupo.

La importancia de los facilitadores en los resultados de la pre-reunión es clara, su capacidad para ordenar y crear un ambiente de trabajo son aptitudes que son necesarias en un grupo de trabajo. Pero más esencial aún es su capacidad de entregar este conocimiento y así lograr un mejor desempeño en las tareas necesarias para poder realizar una pre-reunión.

Las personas que participan en procesos como estos logran comprender mejor el trabajo en equipo y la necesidad de participar activamente en el proceso de toma de decisiones, aprenden lo importante que es encontrar esa pertenencia a un grupo de trabajo y a no guardar los aportes que consideren importantes por motivos ajenos a las metas del trabajo del grupo, así como a la necesidad de escuchar los aportes de los demás.

Pensando en esto, la oportunidad de integrar en este proceso a personas ajenas al ambiente de trabajo grupal da la oportunidad de educar y lograr mejores resultados. Si bien el proceso de llegar a un resultado usando LIBIS puede parecer algo lento, esta posibilidad de educar a través del trabajo permite que cada vez la necesidad de la primera etapa sea menor, y el trabajo del grupo de facilitadores menos decisivo en el resultado final.

En cuanto al desarrollo del sistema, el uso de JavaServer Faces fue muy importante, considerando todas las ventajas que tiene trabajar dentro de un framework, además de contar con un conjunto de componentes (beans) que simplificaron mucho el trabajo. El desarrollo de tecnologías como esta permite el uso de la capacidad de entrega la web para el trabajo en grupo.

Otro punto interesante es la aparición de nuevas tecnologías que pueden lograr una mayor eficiencia en este tipo de sistemas. Herramientas como AJAX, que pueden ejecutar beans que modifican solo partes de la página, entre otras cosas, permiten visualizar un futuro prometedor para las aplicaciones web.

## Capítulo 11: Bibliografía

---

1. Kunz Werner, Rittel Horst, “Issues as Elements of Information Systems”. Working Paper N° 131, 1970.
2. Bellasai G., Borges Marcos, Fuller David, Pino José A., Salgado Ana, “SISCO: A Tool to improve meetings productivity”. En: I CYTED-RITOS International Workshop on Groupware, Lisbon, Portugal. Sept. 1995. pp.149-161,.
3. Parra R, Pino J, “N-Sisco: A Notes Implementation of SISCO”. En: 1<sup>st</sup> International Workshop on Groupware (CRIWG’95), Lisbon, Portugal. Sept. 1995. pp.125-138.
4. Espinosa J, Pino J, Pollard P, “On the Development of a SISCO Implementation Using Java”. En: 3<sup>rd</sup> International Workshop on Groupware (CRIWG’97), Escorial, Spain. Oct. 1997. pp.17 - 24.
5. Cavalcanti M, Borges M, Endo M, “SISCO-RIO: An Asynchronous System to Support Meeting Preparation”. En: 3<sup>rd</sup> International Workshop on Groupware (CRIWG’97), Escorial, Spain. Oct. 1997. pp.25-32.

6. Buckingham Shum Simon, "The Roots of Computer Supported Argument Visualization". En: Visualizing argumentation: software tools for collaborative and educational sense-making. London, UK. Springer-Verlag . 2003. pp.3 - 24.
7. Conklin Jeff, "Wicked Problems and Social Complexity". En su: Dialogue Mapping: Building Shared Understanding of Wicked Problems. United Kingdom. CogNexus Insitute. Chapter 1. 2005.
8. Rittel, H., and M. Webber. "Dilemmas in a General Theory of Planning" pp, Policy Sciences, Vol. 4 Elsevier Scientific Publishing Company, Inc., Amsterdam:155-169. 1973.
9. Voss J.F., Greene T.R., Post T.A., Penner B.C., "Problem-Solving Skill in the Social Sciences" The Psychology of Learning and Motivation, Vol. 17:162-213. New York Academic Press, 1983.
10. Voss J.F., Blais J., Means M.L., Greene T.R., Ahwesh E., "Informal Reasoning and Subject Matter Knowledge in the Solving of Economics Problems by Naive and Novice Individuals". Cognition and Instruction, 3(3):296 - 302, 1986.
11. Newell A., Simon H.A, "Human Problem Solving". Englewood Clifs, N.J., Prentice-Hall, 1972.
12. Conklin Jeff, "Seven Years of Industrial Strength CSCA in an Electric Utility". Group Decision Support System Inc. [en línea] <<http://d3e.open.ac.uk/cscl99/Conklin/Conklin-paper.html>> . [consulta: 15 mayo 2007].
13. Selvin Albert, "Fostering Collective Intelligence: Helping Groups Use Visualized Argumentation". En: Visualizing argumentation: software tools for collaborative and educational sense-making. London, UK. Springer-Verlag . 2003. pp.137 - 163.
14. Van Gelder Tim, "Enhancing Deliberation Through Computer Supported Argument Visualization". En: Visualizing argumentation: software tools for collaborative and educational sense-making. London, UK. Springer-Verlag . 2003. pp.97 - 115.
15. Borges Marcos, Pino José A., Fuller David, Salgado Ana, "Key Issues in the Design of an Asynchronous System to Support Meeting Preparation". Decision Support Systems (27):269-287, 1999.
16. van Bruggen Jan M, Boshuizen Henry P.A., Kirschner Paul A., "A Cognitive Framework for Cooperative Problem Solving with Argument Visualization". En: Visualizing argumentation: software tools for collaborative and educational sense-making. London, UK. Springer-Verlag . 2003. pp.25 - 47 .
17. Conklin Jeff, "Dialog Mapping: Reflections on an Industrial Strength Case Study". En: Visualizing argumentation: software tools for collaborative and educational sense-making. London, UK. Springer-Verlag . 2003. pp.117 - 136.

18. Buckingham Shum Simon, Gangmin Li Victoria Uren, Domingue John, Motta Enrico, “Visualizing Internetworked Argumentation”. En: Visualizing argumentation: software tools for collaborative and educational sense-making. London, UK. Springer-Verlag . 2003. pp.185 - 204.
19. “The JavaServer Faces Technology Tutorial”. Sun Microsystems, 2003.
20. Stearns Beth, “Developing Web Applications Using Sun Java Studio Creator IDE”. Java Studio Creator Development Team.