



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

**“IMPLEMENTACION DE LOS MODULOS DE
ADMINISTRACION DE CONVOCATORIA DE
ARTICULOS Y EVALUACION DE ARTICULOS PARA EL
PORTAL WEB DEL COMPONENTE 8 DEL PROYECTO
VLIR-ESPOL UTILIZANDO MERODE COMO
METODOLOGIA DE ANALISIS Y J2EE COMO
ARQUITECTURA DE DISEÑO”**

TESIS DE GRADO

Previo a la obtención del Título de:

**INGENIERO EN COMPUTACIÓN
ESPECIALIZACIÓN EN SISTEMAS TECNOLÓGICOS**

Presentada por:

Herrera Layana Salomón Alberto

GUAYAQUIL – ECUADOR

2007

AGRADECIMIENTO

A Dios, Jesús y la Virgen María, por ser la fortaleza, la paz y la guía en mi caminar.

A mis padres Jorge y Patricia por su apoyo, comprensión y amor durante todo mi carrera.

A mi tío Javier Layana y Nancy Herrera grandes maestros en mis estudios.

Al Sub-componente de Ingeniería de Software del proyecto VLIR-ESPOL, especialmente al Ing. Carlos Monsalve, a la Ing. Mónica Villavicencio y a la Ing. Verónica Macías.

A todas las personas que me brindaron su apoyo en la realización de este proyecto.

DEDICATORIA

A Dios por estar siempre conmigo y darme la fortaleza para seguir adelante.

A mis padres que siempre están a mi lado.

A mis hermanos Patricia y Jorge y a mi sobrina Milena a quienes amo mucho.

A toda mi familia, por la unión y el respaldo.

A mis queridos y siempre recordados amigos de la Rama Estudiantil IEEE-ESPOL.

A todas aquellas personas que dejaron muchas enseñanzas en mi carrera universitaria.

TRIBUNAL DE GRADUACIÓN

Ing. Holger Cevallos
SUBDECANO DE LA FACULTAD

Ing. Verónica Macías
DIRECTORA DE TESIS

Ing. Carlos Monsalve
MIEMBRO DEL TRIBUNAL

Ing. Katherine Chiliza
MIEMBRO DEL TRIBUNAL

DECLARACIÓN EXPRESA

“La responsabilidad del contenido de esta Tesis de Grado, me corresponde exclusivamente; y el patrimonio intelectual de la misma a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL”.

(Reglamento de Graduación de la ESPOL).

Salomón Alberto Herrera Layana

RESUMEN

El presente proyecto muestra la implementación de los módulos de convocatoria de artículos y evaluación de artículos para el portal web del Componente 8 del proyecto VLIR-ESPOL, basada en el uso de la metodología Orientada a Objetos MERODE, Model-based, Existence - dependency Relationship Object - oriented Development, en la etapa de análisis y la arquitectura J2EE, Java 2 Enterprise Edition, en la implementación de los mismos. Los módulos implementados automatizan el proceso de administración de convocatoria de artículos y la evaluación de artículos, recibidos durante la organización de eventos que lleve a cabo el Componente 8.

Por otro lado, se demuestra como el uso de la metodología MERODE, que promete dar como resultado el mantenimiento fácil y controlado del software desarrollado, puede ser fácilmente adaptada a los patrones de diseño de la Arquitectura J2EE.

Para determinar la efectiva relación y adaptación que iba a tener trabajar con MERODE y J2EE, se diseñaron métricas, para relacionar el avance del

proyecto y el número de cambios producidos en cada una de las capas que demanda la metodología.

En el primer capítulo se describe los antecedentes que han originado la implementación de los módulos de administración y evaluación de artículos, también se plantean los objetivos que se espera alcanzar, la justificación del desarrollo de los módulos y quienes serán los usuarios y administradores de los mismos.

En el segundo capítulo presentamos el fundamento teórico en que se basará la solución, haciendo una descripción general de la Metodología MERODE y sus técnicas, además se describirá y justificará el uso de la Arquitectura J2EE durante el diseño.

En el tercer capítulo realizamos un análisis de los módulos, especificando los requerimientos funcionales y no funcionales más relevantes, para luego mostrar el uso que se dio a la metodología MERODE, describiendo el resultado de haber aplicado las respectivas técnicas. Al final listaremos los casos de uso y los involucrados.

En el cuarto capítulo se explicarán los modelos diseñados para los módulos

partiendo de la Arquitectura general del sistema para dar paso al modelo de objetos y al modelo lógico de la base de datos.

En el quinto capítulo se hace un análisis de la implementación realizada y la comunicación entre capas, basado en un Análisis realizado en MERODE y el diseño en J2EE, luego se plantea un plan de pruebas del sistema, un diseño para la recepción de métricas que permitan medir el avance del proceso de desarrollo de los módulos, y finalmente los resultados de las pruebas y las métricas obtenidas.

Finalmente se darán las respectivas conclusiones y recomendaciones, adjuntando los anexos y haciendo referencia a la bibliografía utilizada.

ÍNDICE GENERAL

	Pág.
RESUMEN	VI
ÍNDICE GENERAL	IX
ABREVIATURAS	XII
INDICE DE FIGURAS.....	XIII
INDICE DE TABLAS	XIV
 CAPITULO 1	
1. ANTECEDENTES Y JUSTIFICATIVOS.....	1
1.1. Antecedentes.....	1
1.2. Justificación	5
1.3. Objetivos.....	6
 CAPITULO 2	
2. FUNDAMENTOS TEORICOS.....	8
2.1. Descripción de la metodología MERODE	8
2.1.1. Técnicas de MERODE	12
2.1.1.1. El gráfico de la dependencia de existencia (EDG. Existence Dependency Graph)	12
2.1.1.2. La tabla de eventos de objetos (OET, Object-Event Table)	
	17

2.1.1.3.	Restricciones de secuencia y Maquina de Estado Finito .	20
2.2.	Conceptos relacionados a J2EE.....	21
2.2.1.	Descripción de la Arquitectura J2EE	21
2.2.2.	Justificación del uso de la Arquitectura J2EE.....	27
2.2.3.	Patrones de diseño utilizados.....	28
2.2.3.2.	Composite View.....	28
2.2.3.3.	Model View Controller.....	37
2.3.	Conceptos relacionados a AJAX	40
2.3.1.	Descripción de la tecnología AJAX.....	40
2.3.2.	Justificación del uso de la tecnología AJAX	49

CAPITULO 3

3.	ANÁLISIS DEL SISTEMA.....	50
3.1.	Requerimientos funcionales	50
3.2.	Requerimientos no funcionales	57
3.3.	Modelo del dominio del sistema usando técnicas de MERODE	58
3.3.1.	El gráfico de la dependencia de existencia	58
3.3.2.	La tabla de eventos de objetos.....	67
3.3.3.	Máquina de estados finitos relevantes: FSM del objeto Artículo	69
3.4.	Definición de roles en los módulos	72

CAPITULO 4

4. DISEÑO DEL SISTEMA.....	73
4.1. Modelos de diseño	73
4.1.1. Modelos de Componentes.....	73
4.1.2. Arquitectura basada en patrones de diseño.....	75
4.1.2.1. Composite View	75
4.1.2.2. Model View Controller	78

CAPITULO 5

5. IMPLEMENTACIÓN Y PRUEBAS	84
5.1. Capas de implementación y comunicación entre capas.....	84
5.1.1. Implementación basada en el análisis con MERODE y J2EE en el diseño	84
5.2. Plan de Pruebas	86
5.3. Diseño de métricas tomadas	92
5.4. Resultados de las Pruebas y Métricas tomadas.....	94

CONCLUSIONES Y RECOMENDACIONES.....	95
--	-----------

ANEXOS**BIBLIOGRAFÍA**

ABREVIATURAS

AJAX	Asynchronous JavaScript And XML
MVC	Model View-Controller
J2EE	Java 2 Enterprise Edition
EDG	Existence Dependency Graph
ER	Entity Relation
FSM	Finite State Machines
GB	Giga Bytes
GHz	Giga Hertz
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
JDBC	Java DataBase Connectivity
JSP	Java Server Pages
KULeuven	Katholic University Leuven
MB	Mega Bytes
Mbits	Mega bits
MERODE	Model-based, Existence – dependency Relationship Object – oriented Development
MHz	Mega Hertz
MOSES	Methodology for Object Oriented Software Engineering of Systems
OET	Object-Event Table
PIII	Pentium III
RAM	Random Access Memory
API	Application Programming Interface
JDBC	Java Database Connectivity
RMI	Remote Method Invocation
EJB	Enterprise JavaBeans
HTTP	Hypertext Transfer Protocol
DOM	Document Object Model
HTML	HyperText Markup Language
JMS	Java Messaging Service
S_AP	Alfabeto del objeto de clase P
XML	Extensible Markup Language

INDICE DE FIGURAS

Figura 2.1 Representación de las capas en MERODE.....	8
Figura 2.2 Notación Relación-Entidad simplificada para el EDG.....	12
Figura 2.3 Ejemplo empresa: Modelamiento típico.....	14
Figura 2.4 Ejemplo empresa: Modelamiento basado en el dominio.....	15
Figura 2.5 OET.....	18
Figura 2.6 Ciclo de vida por defecto representado como FSM del objeto obj1.....	20
Figura 2.7 Aplicaciones multicapas.....	21
Figura 2.8 Distribución de capas de una aplicación J2EE.....	22
Figura 2.9 Vista de dos aplicaciones J2EE en capas.....	24
Figura 2.10 Pantalla de una página modular	31
Figura 2.11 Diagrama de clase que representa el patrón Composite View..	32
Figura 2.12 Ciclo de un servicio en la capa web	37
Figura 2.13 Modelos de aplicaciones web sincrónicas y asincrónicas.....	41
Figura 3.1 Gráfico de Dependencia de Existencia – EDG.....	57
Figura 3.2 Arquitectura del Sistema.....	68
Figura 4.1 Modelo de Componentes.....	72
Figura 4.2 Arquitectura del Sistema de Información MERODE.....	72
Figura 4.3 Composición de la página artículos_asignados.jsp.....	74
Figura 4.4 Código para armar la composición de la página artículos_asignados.jsp.....	76
Figura 4.5 Ejemplo de aplicación del MVC en el ingreso de una pregunta en una evaluación.....	77
Figura 4.6 Código en ingreso_pregunta.jsp.....	79
Figura 4.7 Clase F_Pregunta.java cuando declara una instancia de la clase del modelo Pregunta.java.....	80
Figura 4.8 Clase Pregunta.java cuando declara una instancia de a clase BaseDatos.java.....	80
Figura 4.9 Clase BaseDatos.java cuando invoca a la base de datos para ejecutar el procedimiento para ingresar los datos de la pregunta.....	81

INDICE DE TABLAS

Tabla 3.1 OET – Objeto Convocatoria.....	66
Tabla 3.2 OET – Objeto Tema.....	66
Tabla 3.3 OET – Objeto Tema_Articulo.....	66
Tabla 3.4 Usuarios del Sistema.....	70
Tabla 5.1 Diseño de la plantilla utilizada para realizar la planificación del desarrollo de los módulos.....	90
Tabla 5.2 Diseño de la plantilla utilizada para comparar las horas de trabajo real vs. trabajo estimado inicial.....	91
Tabla 5.3 Diseño de la plantilla utilizada para identificar los tipos de cambios que se iban solicitando.....	92
Tabla 5.4 Tiempos utilizados en el desarrollo de los módulos.....	96
Tabla 5.5 Tiempos utilizados en el desarrollo del proyecto.....	97
Tabla 5.6 Tabla de cambios de fondo, forma y error en los componentes.....	98
Tabla 5.4 Tiempos utilizados en el desarrollo de los módulos.....	96

CAPITULO 1

1. ANTECEDENTES Y JUSTIFICATIVOS

1.1. Antecedentes

La Escuela Superior Politécnica del Litoral, ESPOL, y el Consejo de Universidades Flamencas de Bélgica, mantienen un proyecto de cooperación mutua, el proyecto VLIR-ESPOL, dividido en 8 componentes con dos objetivos básicos:

- Servir a la resolución de problemas en áreas vitales del Ecuador y
- Transferencia rápida a los sectores productivos.

Uno de sus componentes, el número ocho, tiene como objetivo principal el desarrollo de la capacidad de la educación e investigación para la Ingeniería de Software, Telecomunicaciones y Robótica. Este

componente está dividido en 3 subcomponentes: Visión y Robótica, Telecomunicaciones e Ingeniería de Software.

Entre los proyectos ejecutados en el componente, se encuentra la implementación de un portal para administrar sus principales actividades, entre ellas, un módulo para administrar los eventos que organicen cada uno de los subcomponentes durante el periodo de trabajo, denominado año VLIR. Esto sin lugar a dudas, constituye una herramienta de ayuda y soporte para las gestiones y trabajos de investigación que realizan. Sin embargo, ciertos procesos que implican la organización de eventos, aún no automatizados, son manejados en forma manual, lo que retrasa o dificulta el proceso de manejo o recepción de ponencias para eventos, específicamente se hace referencia a la convocatoria y evaluación de artículos para ciertos eventos relevantes en cada uno de los subcomponentes, así como la publicación de los trabajos de investigación.

Actualmente durante la organización de eventos en los que se desarrolla la convocatoria de artículos, el comité organizador debe preparar un mail de difusión de la convocatoria, una base de datos de contactos para difundirla, crear una cuenta personalizada para cada

evento y así empezar a realizar la convocatoria de trabajos de investigación.

Transcurrido el tiempo hasta llegar a la fecha máxima de publicación establecida, se va recibiendo mail por mail a la cuenta de correo electrónico del evento, cada artículo o trabajo de investigación propuesto por los autores.

Un vez recibidos los artículos y categorizados por los temas propuestos en la convocatoria, se procede a asignarlos a los revisores previamente contactados y seleccionados.

Cada revisor recibe uno o varios correos electrónicos que contienen las indicaciones del proceso de evaluación que debe seguir, y adjunto encontrará dos o más archivos; uno es la plantilla de evaluación que debe llenar y el resto serán los artículos que debe evaluar.

Una vez que se reciben todos los correos electrónicos de los revisores con los artículos evaluados, se procede a definir qué artículos quedarán como ponencias del evento.

Posteriormente, se envían dos tipos de correos electrónicos, uno para cada autor cuyo artículo no fue aceptado y otro para aquellos autores cuyos artículos han sido aceptados y que deberán realizar las correcciones propuestas por los revisores.

Finalmente, cada autor debe enviar de vuelta a los organizadores, en un correo electrónico, las correcciones del artículo, hasta la fecha máxima establecida, para luego ya considerarlos como ponentes del evento.

Por lo expuesto ha surgido la necesidad de automatizar este proceso para mejorar el manejo y la dinámica de administrar los artículos de investigación que se reciben en los diferentes eventos del Componente.

Por otro lado, uno de los objetivos principales del subcomponente de Ingeniería de Software, es el estudio y difusión de una metodología de análisis para el desarrollo de software, llamada MERODE, es por esto que esta tesis a más de automatizar el proceso de convocatoria y evaluación de artículos, permitirá analizar cómo esta metodología afecta el proceso de análisis para obtener software de calidad.

1.2. Justificación

Actualmente el componente ocho cuenta con un portal web que le permite administrar la organización de eventos para cada uno de sus subcomponentes, sin embargo, cuando se requiere recibir trabajos de investigación para que sean considerados como ponencias de eventos, el portal no cuenta con un módulo para administrar la convocatoria y la evaluación de estos trabajos recibidos.

Al implementar estos módulos, se logrará tener una herramienta que ayude a mejorar el manejo de estos procesos y que proveerá un mejor soporte para administradores, autores y revisores.

Para lograr estos objetivos e implementar los módulos, se usará durante el análisis la metodología MERODE, la cual permite mediante un análisis Orientado a Objetos, identificar los objetos de la lógica del negocio involucrados en los procesos y distribuirlos en tres capas diferentes: Capa interna (Objetos de Negocio), capa intermedia (Objetos de Sistema de Información) y capa externa (Interfaz con el usuario).

Este esquema permite tener modularidad, mismo que ayudará a disminuir la propagación de cambios, para luego hacer una óptima y más real estimación de costos por modificaciones.

Además la metodología al trabajar en capas permitirá dividir el sistema en subsistemas y al definir el modelo del dominio del negocio separa claramente los detalles tecnológicos del análisis de los módulos.

Conociendo la forma como trabaja la metodología, utilizando tres capas, y tomando de referencia los requerimientos recibidos, se ha procedido a utilizar la arquitectura J2EE para el diseño de los módulos, a través del uso de varios de sus patrones para al final demostrar cómo la metodología puede fácilmente adaptarse a una arquitectura distribuidas en capas.

1.3. Objetivos

Luego de haber analizado las necesidades actuales y definiendo la solución adecuada, el desarrollo de esta tesis tiene por objetivos:

- Diseñar e Implementar un módulo para manejar y administrar la convocatoria de artículos para eventos que organice el componente.

- Diseñar e Implementar un módulo para realizar de forma automática la evaluación de los artículos que se reciben para los eventos.
- Analizar cómo el uso de la metodología de Análisis MERODE afecta el desarrollo del sistema.

CAPITULO 2

2. FUNDAMENTOS TEORICOS

2.1. Descripción de la metodología MERODE

MERODE es una metodología de análisis orientada a objetos, fue diseñada en el Departamento de Ciencias Económicas Aplicadas de la Universidad Católica de Leuven, Bélgica, y ha sido probado y aplicado en varias compañías en Bélgica y Holanda [1].

Las siglas MERODE significan Model-based Existence-dependency Relationship Object-oriented Development (Desarrollo Orientado a Objetos basado en el modelo de las relaciones de Dependencia de Existencia) [1].

La característica más sobresaliente de MERODE es que usa el principio de dependencia de existencia, lo cual facilita la flexibilidad y fácil modelamiento del negocio, y al mismo tiempo el uso de métodos formales para asegurar la calidad del sistema resultante [1].

Se describe a continuación la metodología y la definición y uso de sus técnicas de análisis.

MERODE es una metodología basada en el modelamiento del dominio del negocio y asegura la construcción de software basado en el modelo del negocio [1].

MERODE divide al sistema en capas, distribuyendo los objetos sobre cada una de las capas e identifica los objetos por fases. A continuación se muestra un esquema general de estas capas:

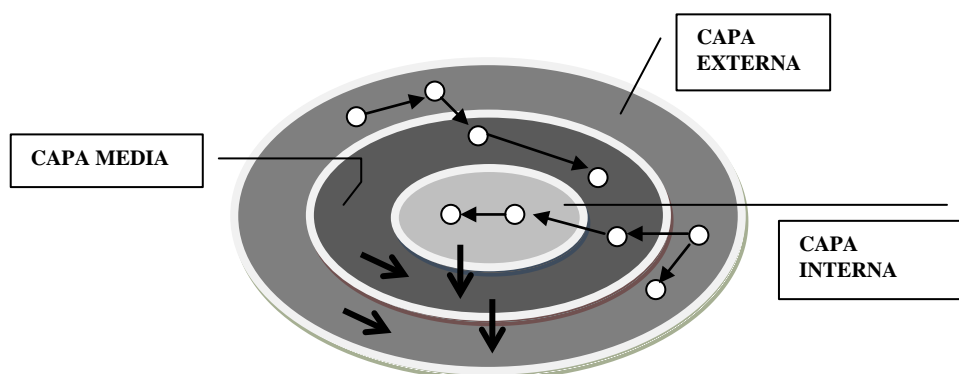


Figura 2.1 Representación de las Capas en MERODE

La Capa Interna representa la fase de modelamiento del Negocio. En esta etapa se identifican los tipos de objetos del negocio y los tipos de evento del negocio, para luego construir el gráfico de dependencia de existencia (EDG) y la tabla de objetos-eventos (OET). Finalmente se especifica las secuencias de control, de atributos y métodos y todas aquellas reglas de control adicionales que surjan del análisis realizado.

En la capa media donde se representa la fase de modelamiento funcional, se establecen todos los servicios de información que se van a necesitar, esto es, identificar tipos de objetos de información, la generación de eventos del negocio (servicios de entrada), la construcción del diagrama de especificación del servicio, y así concluir con la determinación de atributos y métodos.

La tercera capa es la externa, aquí se analiza la fase de Modelamiento de Interfaz de Usuario [1].

Los objetos de cada capa solo pueden comunicarse entre objetos de la misma capa o con aquellos de la capa inferior inmediata, lo que conlleva a que los cambios de un objeto solo afecten a objetos de capas superiores y así, los objetos más internos serán los que menos cambien, a diferencia de los externos que son lo que más cambian.

Como se mencionó anteriormente la característica de dependencia de existencia es la más importante en la metodología MERODE. Esta característica es basada en la notación de la vida de un objeto, es decir, el periodo que existe entre el momento de su creación hasta el momento de su eliminación, por tanto, siendo P y Q dos tipos de objetos, el principio afirma que P es dependientemente existente de Q si y solo si la vida de cada ocurrencia de p de tipo P está embebida en la vida de una y siempre la misma ocurrencia de q de tipo Q [1].

Para el caso anterior, p es llamado “objeto dependiente” y su existencia depende de la de q, y a su vez, q es llamado “objeto maestro”.

Refiriéndonos al periodo de vida de los objetos dependiente y maestro, la vida de un objeto dependiente no puede empezar antes que la del objeto maestro y así mismo, no puede terminar después que la del objeto maestro.

2.1.1. Técnicas de MERODE

2.1.1.1. El gráfico de la dependencia de existencia (EDG. Existence Dependency Graph)

Siendo M el conjunto de clases en el Modelo del Dominio, el EDG es un ordenamiento parcial de estas clases tal que:

- a) Una clase nunca depende de sí misma
- b) El gráfico es acíclico.

Representando gráficamente la relación de dependencia de existencia, el tipo de objeto maestro debe ser colocado arriba del tipo de objeto dependiente y ambos deben conectarse con una línea.

La cardinalidad es indicada por símbolos de círculos y flechas según sea el caso de la relación. En la figura 2.2. se muestran los casos posibles.

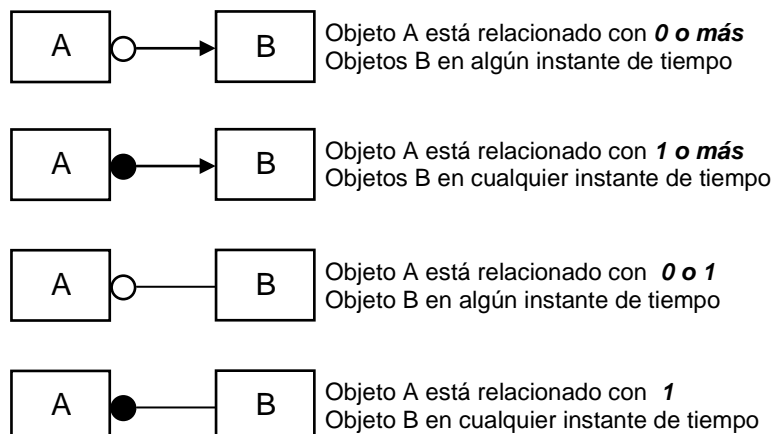


Figura 2.2 Notación Relación-Entidad simplificada para el EDG

Como se muestra en la figura 2.2, la circunferencia debe ser colocada del lado del objeto maestro y la flecha o línea debe llegar al objeto dependiente.

Para cada uno de estos 4 casos, el objeto B está relacionado con exactamente 1 y siempre el mismo objeto A.

Esta cardinalidad de la relación de dependencia de existencia del EDG, establece el número de ocurrencias de un tipo de objeto dependiente que pueden depender de un objeto maestro en un instante determinado. Para esta notación se ha

tomado el esquema simplificado de relación-entidad (ER) que contiene los mismos tipos y conexiones que el EDG. A continuación se define las diferentes representaciones que podrían ocurrir:

Una cardinalidad de muchos indica que puede haber más de un objeto dependiente del objeto maestro en un instante de tiempo.

Una cardinalidad de uno indica que puede haber máximo 1 objeto dependiente del objeto maestro en un instante de tiempo.

Dada las dos cláusulas anteriores la opcionalidad y la cardinalidad son especificadas únicamente en relación al tipo de objeto maestro.

Para graficar un EDG debemos considerar las siguientes cláusulas:

- o Solo se usarán relaciones binarias de tipo “uno a muchos”, “uno a uno”, “cero a muchos” o “cero a uno”.

- Las relaciones siempre expresarán dependencia de existencia.
- La cardinalidad del objeto dependiente siempre será de uno.
- La relación por parte del objeto dependiente siempre es mandatoria.

Para ilustrar la aplicación del EDG usaremos un ejemplo:

Una empresa tiene asignados empleados por departamentos. La forma en que normalmente se representaría esta relación es la siguiente:

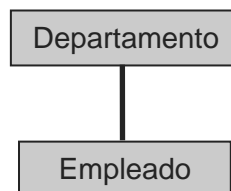
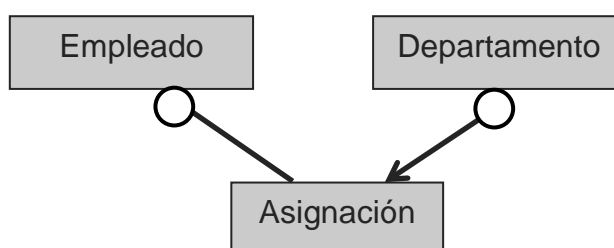


Figura 2.3 Ejemplo Empresa: Modelamiento típico

En la figura 2.3 se muestra una representación incorrecta en MERODE por cuanto no hay dependencia de existencia, es decir Departamento

sigue existiendo si Empleado deja de existir y viceversa.

Como podemos notar en la figura 2.4, para resolver este inconveniente debemos recordar que cuando trabajamos con desarrollo basado en el dominio, la comunicación se maneja a través de los eventos del negocio comunes. Los eventos del negocio son como un contrato entre los 2 objetos y son considerados también objetos del modelo conceptual. *Departamento* es un tipo de objeto maestro, y *Asignación* es su objeto dependiente. *Empleado* es un tipo de objeto maestro y *Asignación* su dependiente y por lo tanto la relación quedaría modelada de la siguiente manera:



**Figura 2.4 Ejemplo empresa: Modelamiento
basado en el dominio**

Y podemos verificar que en un mismo instante de tiempo:

- Un Departamento puede tener muchas asignaciones de empleados.
- Una Asignación de un Empleado puede ser hecha a máximo un Departamento
- Un Empleado puede ser asignado a un solo Departamento

2.1.1.2. La tabla de eventos de objetos (OET, Object-Event Table)

El OET es una vista más de MERODE. Sirve para representar los aspectos dinámicos del dominio, y muestra todos aquellos objetos del negocio que son afectados por cada evento del negocio y la forma en que son afectados.

Los eventos del negocio ocurren en algún instante de tiempo, no tienen duración, suceden en el mundo real y son atómicos (no pueden ser separados en sub-eventos).

Para identificar los eventos del negocio debemos tener presente:

- Que por cada objeto del negocio se tiene mínimo dos eventos del negocio, uno creador y otro finalizador, y
- Que por cada objeto del negocio es posible tener uno o varios eventos modificadores.

El OET es una tabla que contiene:

- Una fila por cada tipo de evento del negocio y una columna por cada tipo de objeto del negocio.
- Cada celda contiene un espacio en blanco, una “C”, una “E” o una “M”, donde:
 - “C” representa un evento creador.
 - “E” representa un evento finalizador.
 - “M” representa un evento modificador.
 - El espacio se usa cuando el objeto de la columna correspondiente no es afectado por el evento de la fila correspondiente.

- Por cada columna hay al menos una fila con una “C” y una fila con una “E”.
- $c(P)$, $m(P)$ y $e(P)$ son los conjuntos de todos los tipos de eventos para los cuales respetivamente hay una “C”, una “M” o una “E” en la columna de P.
- La unión de $c(P)$, $m(P)$ y $e(P)$ nos da como resultado lo que se conoce como el alfabeto de P, que está formado por todos los eventos que afectan a P.

Cada tipo de evento debe ser relevante para al menos un tipo de objeto. Cada fila debe tener al menos una columna con una “C”, “M” o “E” como se muestra en la figura 2.5.

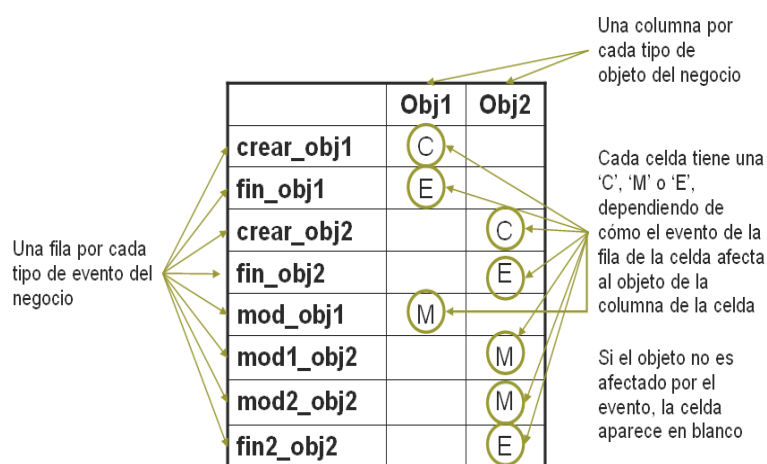


Figura 2.5 OET

2.1.1.3. Restricciones de secuencia y Máquina de Estado

Finito

Las restricciones de secuencia son aquellas restricciones de secuencia de eventos específicas para cada tipo de objeto.

Para modelar las restricciones de secuencia, MERODE utiliza la Máquina de Estado Finito (FSM). La máquina de estado finito contiene dos ejes, el progreso hacia el estado final es representado en el eje horizontal y las partes cíclicas de la secuencia son representadas en el eje vertical.

El evento del negocio puede afectar a varios tipos de objetos. Si el evento ocurre, todos los objetos afectados por el evento deben correr el método afectado al mismo. El evento en sí solo será exitoso si todos los objetos que son afectados se encuentran en un estado en el que se permite la ejecución del evento.

La figura 2.6 se muestra el ciclo de vida por defecto que tienen todos los objetos del negocio, uno para su creación, su modificación y su eliminación.

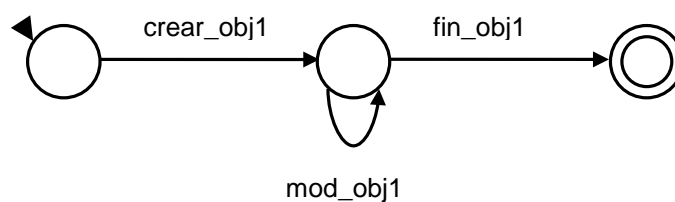


Figura 2.6 Ciclo de vida por defecto representado como FSM del objeto obj1

2.2. Conceptos relacionados a J2EE

2.2.1. Descripción de la Arquitectura J2EE

Java 2 Enterprise Edition o Java EE, es una plataforma de programación, parte de la Plataforma Java, para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java con arquitectura de múltiples capas distribuidas, basándose ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones. [5]

Java EE incluye varias especificaciones de API, tales como JDBC, RMI, e-mail, JMS, Servicios Web, XML, entre otros y

define cómo coordinarlos; también configura algunas especificaciones únicas para componentes de Java EE, estas incluyen Enterprise JavaBeans, servlets, portlets (siguiendo la especificación de Portlets Java), Java Server Pages y varias tecnologías de servicios web, soportados a través de intercambio de datos integrados basados en estándares abiertos y protocolos XML (Extensible Markup Language). Esto permite al desarrollador crear una Aplicación de Empresa portable entre plataformas y escalable, a la vez que integrable con tecnologías anteriores. Otros beneficios añadidos son, por ejemplo, que el servidor de aplicaciones puede manejar transacciones, seguridad, escalabilidad, concurrencia y gestión de los componentes desplegados, de manera que los desarrolladores pueden concentrarse más en la lógica de negocio de los componentes, que en las tareas de mantenimiento de bajo nivel.

En la figura 2.7 se muestra un esquema de aplicaciones multicapas. En la capa del usuario se encuentran las vistas con las que interactúan el cliente, en la capa lógica está toda la lógica del sistema y en la capa de datos están los repositorios de datos.

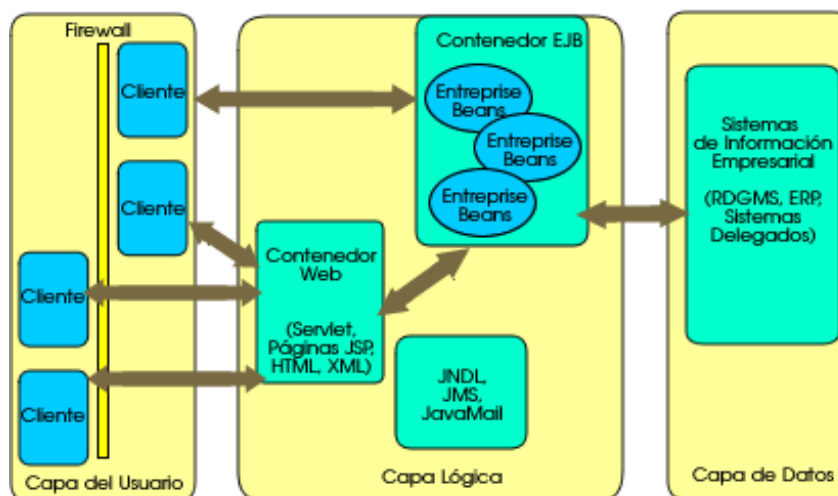


Figura 2.7 Aplicaciones Multicapas

La plataforma J2EE como se muestra en la figura 2.8, usa un modelo de aplicación multicapa distribuido para aplicaciones empresariales. La aplicación empresarial o lógica de la empresa esta dividida entre componentes dependiendo su función y al integrar una aplicación J2EE, dichos componentes son instalados en diferentes máquinas dependiendo del nivel del ambiente multicapa J2EE al cual pertenecen.

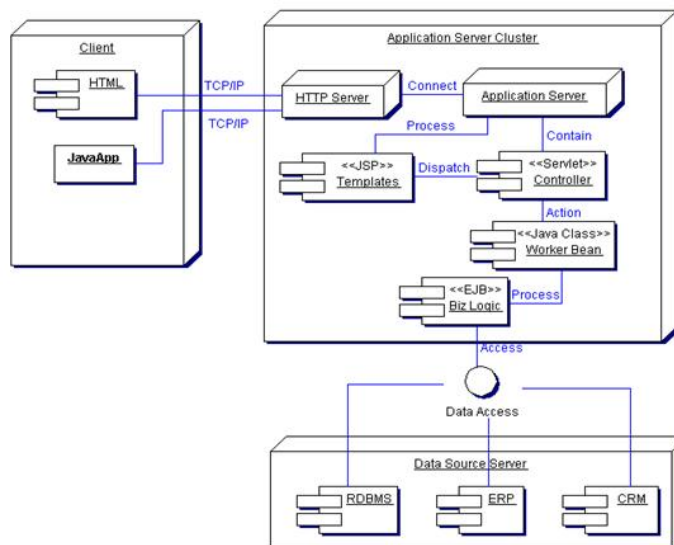


Figura 2.8 Distribución de capas de una aplicación J2EE

Las partes de la aplicación J2EE mostradas en la Figura 2.8 son presentadas en Componentes que se mencionan a continuación:

- Los Componentes de la Capa Cliente corren sobre la máquina del cliente.
- Los Componentes de la Capa Web corren en el servidor J2EE.
- Los Componentes de la Capa Lógica corren en el servidor J2EE.
- Los Componentes de la Capa de Datos (Enterprise Information System, EIS) corren en el servidor EIS.

Aunque una aplicación J2EE puede estar conformada de tres o cuatro capas como muestra la figura 2.9, las aplicaciones J2EE son generalmente consideradas aplicaciones de tres capas ya que se encuentran distribuidas principalmente en tres lugares: en las máquinas clientes, la máquina del servidor J2EE, y la base de datos. En las aplicaciones de tres capas, por lo general, el servidor de aplicaciones es considerado de dos capas, uno que enlace del servidor con el cliente y otro con el almacenamiento final.

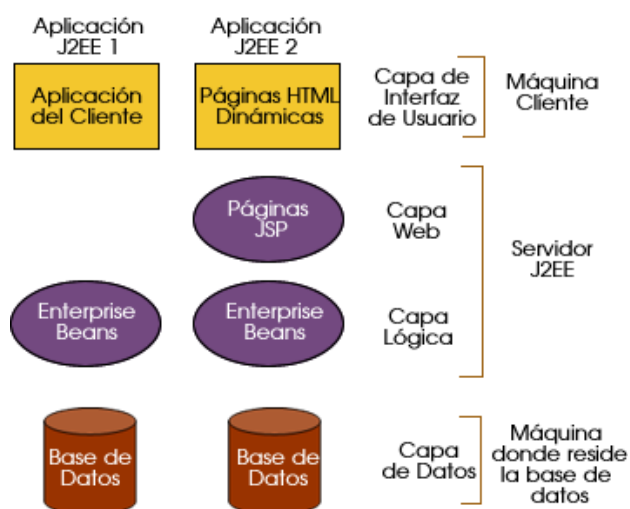


Figura 2.9 Vista de dos aplicaciones J2EE en capas

2.2.1.2. Componentes J2EE

Las aplicaciones J2EE son integradas por algunos componentes. Un componente J2EE es una unidad

funcional del software contenido por si mismo que es ensamblado dentro de una aplicación J2EE con sus clases y fuentes que se comunican con otros componentes. La especificación J2EE define los siguientes componentes:

- Aplicaciones Clientes y applets son componentes que corren en el cliente.
- Componentes Tecnológicos como Java Servlet y JavaServer Pages (JSP) son componentes web que corren sobre el servidor.
- Componentes Enterprise JavaBeans (EJB) son componentes comerciales que corren sobre el servidor.

Los componentes J2EE están escritos en Java y son compilados de la misma forma como cualquier programa en este lenguaje. La diferencia entre los componentes J2EE y las clases Java “estándar” es que los componentes J2EE son desarrollados en una aplicación J2EE, probados que se encuentren

bien desarrollados y cumpliendo con la especificación J2EE, y son estructurados según el campo de producción, donde estén siendo ejecutados y administrados por el servidor J2EE.

2.2.2. Justificación del uso de la Arquitectura J2EE

Como se describió en la sección anterior, MERODE distribuye los objetos del negocio en capas (capa interna, capa media y capa externa), lo cual permite una mejor administración y mantenimiento del software final.

Por otro lado, J2EE es una Arquitectura de diseño basada en la distribución de aplicaciones en capas, lo que permite un mejor manejo de la información que se está procesando, es por esto, que a más de tener MERODE distribuido en capas y J2EE diseñado para aplicativos en capas, esta integración de ambos, nace como un requerimiento del componente ocho del Proyecto VLIR-ESPOL, de utilizar una arquitectura Open Source, que permita sobre todo la portabilidad del sitio web y su implementación, en cualquier sistema operativo y mantener activo el funcionamiento del portal sin mayores inconvenientes.

2.2.3. Patrones de diseño utilizados

Para construir aplicaciones J2EE robustas, se debe conocer técnicas avanzadas de análisis, diseño y desarrollo.

Sun propone una serie de patrones para ayudarnos a la construcción de aplicaciones java, los cuales nos ayudan a garantizar la correcta implementación de estas aplicaciones.

Entre todos los patrones propuestos por Sun para diseños de sistemas bajo una Arquitectura J2EE y teniendo presente la distribución en capas que la metodología MERODE propone se han seleccionado los patrones: Composite View y Model View Controller.

2.2.3.2. Composite View

Las aplicaciones Web reales agregan mucha información distinta (contenidos) en sus páginas, muchas veces poco relacionada con nuestro negocio (noticias, encuestas, estadísticas, entre otras).

Normalmente, es necesario utilizar técnicas avanzadas de composición de estas páginas para simplificar su construcción y mantenimiento.

Las JSP, Java Server Pages, y los servlets proporcionan mecanismos sencillos para poder incluir porciones de una página en otra (de modo estático o dinámico). Estos mecanismos, normalmente son insuficientes por si mismos por lo que tenemos que utilizar algunas técnicas más elaboradas en incluso recurrir a productos especializados como gestores de contenidos.

Este patrón proporciona un mecanismo para combinar modularmente, las porciones atómicas de una visión en un entero compuesto, las páginas son construidas poniendo código en el formato dentro de cada visión. A continuación, algunas situaciones donde se puede aplicar este patrón:

- Las porciones atómicas del contenido de la vista cambian con frecuencia.

- Varias vistas compuestas utilizan subvistas similares, como una tabla de inventario de clientes. Estas porciones atómicas se decoran con una plantilla de texto o aparecen en diferentes localizaciones dentro de la página.
- Los cambios de distribución son más difíciles de manejar y el código es más difícil de mantener cuando las subvistas se embeben directamente y se duplican en varias vistas.
- Frecuentemente, embeber las porciones cambiantes de la plantilla de texto, dentro de las vistas también afecta potencialmente a la disponibilidad y administración del sistema. El servidor podría necesitar reiniciarse antes de que los clientes vean las modificaciones o actualizaciones de estas plantillas en cada componente.

Utilizar las vistas compuestas que se componen de múltiples subvistas atómicas. Cada componente de la

plantilla se puede incluir dinámicamente en el conjunto y la disposición de la página se puede manejar independientemente del contenido [3].

Esta solución nos facilita la creación de una vista compuesta basada en la inclusión y la sustitución de los fragmentos dinámicos y estáticos modulares de la plantilla.

Por otro lado, promueve la reutilización de las porciones atómicas por un diseño modular. Es apropiado utilizar una vista compuesta para generar las páginas que contienen, componentes que se combinan de varias maneras. Este panorama ocurre, por ejemplo, con los sitios que incluyen sub-vistas independientes, tales como alimentaciones de noticias e información del tiempo, en una sola página.

Otra ventaja de este patrón es que los diseñadores Web disponen del prototipo del sitio, tapando el contenido estático en cada uno de las regiones de la

plantilla. Mientras avanza el desarrollo de sitio, el contenido real se sustituye para estos espacios.

La figura 2.10 muestra una captura de una pantalla del sitio de Java de Sun, java.sun.com. Se identifican cuatro regiones: Navegación, búsqueda, historia, y títulos. Mientras que el contenido para cada uno de estos componentes se puede originar de diversas fuentes de datos, se presentan juntos para crear una sola página compuesta.

Proyecto VLIR-ESPOL, Componente 8 - Windows Internet Explorer

EN English (United States)

http://localhost:8080/AppVlr8/

Proyecto VLIR-ESPOL, Componente 8

Inicio | Quiénes somos | Contáctenos

Encabezado

Ingreso al Sistema

Accesos rápidos

Navegación

Suscribirse
Eventos por realizar
Eventos pasados
Publicaciones

Mejora de procesos de Software ISO

INGENIERÍA DE SOFTWARE

INGRESO AL SISTEMA:
Correo:
Clave:
Ingresar
Suscribirse ¿Olvidó su contraseña?

PUBLICACIONES

OCT|2004.- Modelamiento basado en el dominio: Estado del... [más](#)
FEB|2006.- Detection of isolated nematodes in clutter environments... [más](#)
SEP|2005.- Estudio de Sistemas Inalámbricos de Espectro... [más](#)

TELECOMUNICACIONES

BIENVENIDO A NUESTRO PORTAL
A través de nuestro portal ahora podrá enterarse de todos los trabajos e investigaciones en las Areas de Ingeniería de Software, Telecomunicaciones y Robótica y visión por Computadora.

AREAS DE INVESTIGACION

INGENIERIA DE SOFTWARE
Aplicación de MERODE en el desarrollo de software en el Ecuador. Plan de Métricas para Empresas de Software. Plan de Métricas p...
CENTRO DE VISION Y ROBÓTICA
Análisis y procesamiento de Imágenes Biomédicas. Modelado 3D. Aplicaciones de Visión y Robótica. Robótica móvil.
TELECOMUNICACIONES
Redes de Acceso por Cable. Redes de telefonía móvil de acceso por radio. Redes de Acceso fijo inalámbrico.

ROBÓTICA

Cooperación y apoyo con empresas, industrias y universidades nacionales e internacionales.

Nuestra Trayectoria...
¡Nuestro respaldo!
Asesoramiento por distinguidos profesionales

ÚLTIMAS TECNOLOGÍAS

© Copyright 2006 Componente 8 - Proyecto VLIR-ESPOL. Todos los derechos reservados

Figura 2.10 Pantalla de una página modular

La figura 2.11 muestra el diagrama de la clase que representa el patrón Composite View.

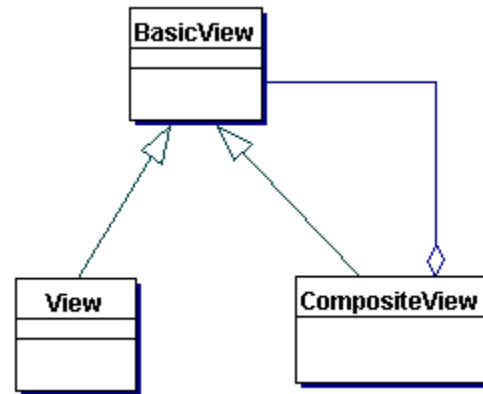


Figura 2.11 Diagrama de clase que representa el patrón Composite View

A continuación se mencionan las ventajas de usar este patrón:

Mejora la Modularidad y la Reutilización.- Este patrón promueve un diseño modular. Es posible reutilizar porciones atómicas de una plantilla, como una tabla de consulta de stocks, en varias vistas y redecorar estas porciones reutilizadas con información diferente. Este patrón permite que la tabla se mueva dentro de su propio módulo e incluirla simplemente donde sea necesario. Este tipo de

distribución y composición dinámica reduce la duplicación, facilita la reutilización y mejora la facilidad de uso.

Mejora la Flexibilidad.- Una implementación sofisticada podría incluir condicionalmente fragmentos de plantillas de vista basándose en decisiones en tiempo de ejecución, como los roles de usuario o las políticas de seguridad.

Mejora el Mantenimiento y la usabilidad.- Es mucho más eficiente manejar cambios en porciones de una plantilla cuando la plantilla no está codificada directamente en las marcas de la vista. Cuando se mantienen separadas de la vista, es posible modificar porciones modulares del contenido de la plantilla independientemente de su distribución. Además, esos cambios están disponibles inmediatamente para los clientes, dependiendo de la estrategia de implementación. Las modificaciones en la distribución de una página también se manejan más fácilmente, ya que los cambios están centralizados.

A continuación se mencionan las desventajas de usar este patrón:

Reduce la flexibilidad.- Agregar piezas atómicas para mostrarlas juntas y crear una sola vista presenta algunos potenciales problemas de presentación, ya que las sub-vistas son fragmentos de páginas. Esta es una limitación que se puede convertir en un problema de flexibilidad. Por ejemplo, si una página JSP está generando una página HTML utilizando una página principal que incluye tres sub-vistas, y cada una de las sub-vistas incluye las etiquetas de apertura y cierre de HTML (es decir, <HTML> y </HTML>), entonces la página compuesta no será válida. Por lo tanto, es importante cuando utilizemos este patrón tener cuidado de que las sub-vistas no deben ser vistas completas. Se deben contabilizar estrictamente las etiquetas utilizadas para crear vistas compuestas válidas, y así corregir este problema de flexibilidad.

Impacto en el Rendimiento.- Generar una presentación que incluye numerosas sub-vistas podría empeorar el rendimiento. La inclusión en tiempo de ejecución de sub-vistas resultará en un retardo cada vez que se presente la página a un cliente. En un entorno con Acuerdos de Nivel de Servicio que obligan a tiempos de respuesta específicos, dichas bajadas de rendimiento, aunque típicamente sean mínimas, podrían no ser aceptables. Una alternativa es mover la inclusión de las sub-vistas al tiempo de la traducción, aunque esto limita a que las sub-vistas sólo cambien cuando se re-traduce la página.

2.2.3.3. Model View Controller

Como se mencionó anteriormente, las aplicaciones Web unen mucha información que tienen poca relación entre sí.

Model View Controller (MVC) es uno de los modelos más recomendados para el diseño de aplicaciones interactivas. Organiza los objetos en una de las tres

categorías: *modelos* para mantenimiento de los datos, *vistas* para mostrar todo o una porción de los datos y controlador para manejar los eventos que afectan las vistas o el modelo [2].

MVC separa detalles del diseño (persistencia, presentación, y control de los datos), disminuye la duplicación de código, centraliza el control de la aplicación y hace que los cambios o actualizaciones sean más fácilmente manejables.

Un diseño MVC puede centralizar el control de funcionalidades como el uso de seguridad, de sesión, y el flujo de la pantalla. MVC puede adaptarse a nuevas fuentes de datos, creando código que adapta la nueva fuente con las pantallas. Finalmente, MVC define claramente las responsabilidades de las clases que participan.

Las aplicaciones de la Capa Web J2EE sirven peticiones del HTTP. En el más alto nivel, la Capa Web hace cuatro cosas básicas: interpreta peticiones

del cliente, envía esas peticiones a la lógica del negocio, selecciona la vista siguiente para mostrar, y genera y entrega la vista siguiente.

La figura 2.12 presenta el ciclo de un servicio en la capa web. Cuando el cliente envía un requerimiento, este es enviado a la capa lógica del sistema para su procesamiento y una vez procesado el sistema determina la siguiente vista que deberá mostrarse hacia el cliente.

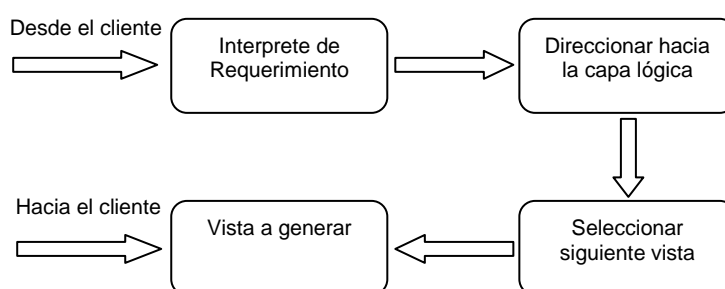


Figura 2.12 Ciclo de un servicio en la capa web

El controlador de la Capa Web recibe cada petición entrante del HTTP e invoca la operación solicitada de la capa lógica del negocio. De acuerdo con los resultados de la operación y del estado del modelo, el controlador entonces selecciona la vista siguiente

para mostrar. Finalmente, el controlador genera la vista seleccionada y la transmite al cliente para la presentación.

2.3. Conceptos relacionados a AJAX

2.3.1. Descripción de la tecnología AJAX

El término “Ajax” fue acuñado por Jesse James Garret en su artículo “Ajax: A New Approach to Web Applications”.

AJAX es el acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas. Éstas se ejecutan en el cliente, es decir, en el navegador de los usuarios, y mantiene comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre la misma página sin necesidad de recargarla. Esto significa aumentar la interactividad, velocidad y usabilidad en la misma [7].

AJAX es una combinación de tres tecnologías ya existentes:

- XHTML y CSS para la presentación de información.

- Document Object Model (DOM3) para visualizar dinámicamente e interactuar con la información presentada.
- XML, XSLT para intercambiar y manipular datos.
- XMLHttpRequest para recuperar datos asíncronamente.
- Javascript como nexos de unión de todas estas tecnologías.
- No requiere plugins o capacidades específicas de ciertos navegadores.

AJAX no constituye una tecnología en sí, sino que es un término que engloba a un grupo de varias tecnologías que trabajan conjuntamente.

En el desarrollo clásico tanto de aplicaciones como sitios web, la comunicación con el usuario es síncrona, es decir:

- El usuario realiza una petición al servidor.
- El servidor envía la página solicitada.
- El usuario comienza a “leer” la página.
- Llega un momento dado en el cual el usuario desea cambiar de página y mediante formularios o

links realiza una petición al servidor volviendo al paso número 1.

En cambio la comunicación asíncrona no implica sincronismo ante los eventos del usuario, sino que ante cualquier evento del usuario nosotros podemos proceder en consecuencia, es decir:

- El usuario realiza una petición al servidor
- El servidor envía la página solicitada
- El usuario comienza a “leer” la página
- Llega un momento dado en el cual el usuario quiere cambiar de página o alterar la información que contiene la misma. En ese momento dado la aplicación teniendo definidos los eventos posibles actúa en consecuencia a la interacción que haya podido suceder. Entonces podremos cambiar al usuario de página, o por el contrario modificar la misma para satisfacer al usuario. Estos cambios no implican cambiar de página.

A continuación, en la figura 2.13 se muestran varias imágenes que diferencian la comunicación sincrónica y asíncrona. En la primera parte, un clásico modelo sincrónico presenta cómo la

transmisión de datos desde que el cliente envía un requerimiento hasta que el sistema devuelve un resultado, es determinado por el tiempo de respuesta del procesamiento de la solicitud, en este caso en el cliente se actualizará la vista completa y mientras no ocurra esto, no podrá navegar en el resto de la página hasta estar completamente recargada. Por otro lado, en la segunda parte de la figura, se presenta la comunicación asincrónica, donde el cliente no percibe el tiempo en que el sistema se demora en procesar un requerimiento, en este caso en el cliente no ocurre una actualización de la página completa puesto que AJAX solo trabaja sobre el área que se desea actualizar y mientras tanto el cliente puede seguir navegando otra parte de la página.

En el modelo sincrónico el cliente debe esperar la actualización de la página mientras se termina de presentar el resultado de su requerimiento, mientras que en el modelo asincrónico el cliente puede seguir navegando en la página sin necesidad de esperar por la respuesta del sistema.

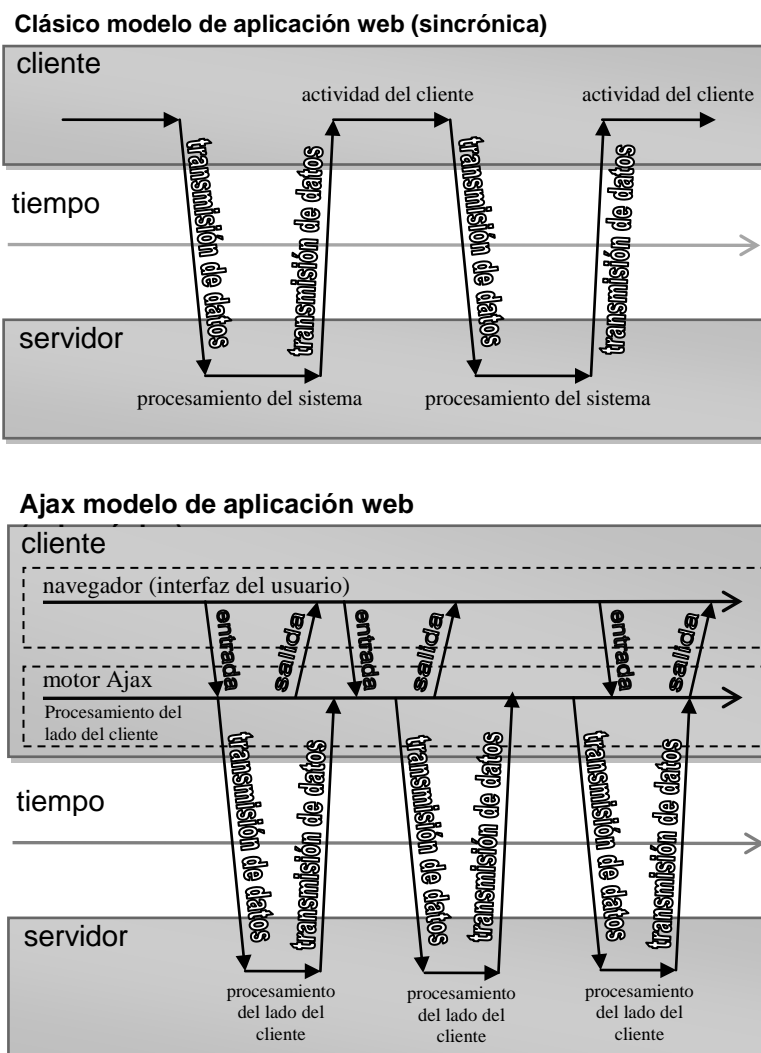


Figura 2.13 Modelos de aplicaciones web síncronas y asíncronas

Es por esto que una aplicación AJAX termina con la idea “arrancar-frenar-arrancar-frenar”. Sin Ajax cada interacción de un usuario generaría un requerimiento HTTP para poder mostrar la nueva información, pero mediante AJAX las peticiones al servidor pueden relegarse al “motor AJAX” que es quien procesaría las peticiones contra el servidor, permitiendo o

no al usuario interactuar con la página mientras el servidor está procesando la información, evitando de este modo los tiempos de carga en los cuales el usuario ve como su reloj de arena da vueltas mientras su navegador trata de componer la nueva página mientras recibe la información del servidor [6].

El funcionamiento de AJAX consiste en la posibilidad de poder ante cualquier evento o la negación del mismo ejecutar código pudiendo alterar la estructura (DOM) de la página.

Como ejemplo de sistemas web actuales que utilizan esta tecnología tenemos a **Google**.

Google, buscador por excelencia demostró otra vez hace más de un año el potencial de la tecnología y de sus servidores. Google Suggest permite comprobar instantáneamente el número de resultados que una búsqueda va a tener mientras se está escribiendo la misma.

Este ejemplo demuestra la utilidad de AJAX para realizar búsquedas rápidas (ej: Datos personales, números de teléfono) o solucionar problemas a la hora de auto completar campos específicos dentro de formularios.

En la parte práctica copiaremos el ejemplo de google para buscar páginas web haciendo un pequeño código donde buscaremos el número de teléfono de una persona simplemente escribiendo una parte de su nombre o apellido.

Ajax nos sirve principalmente para diseñar y programar interfaces de usuario mucho más allá de la web, rompiendo las limitaciones que la "sincronía" supone y abriendo una nueva puerta que nos permitirá desarrollar aplicaciones que en un principio solo podrían concebirse para el escritorio, en aplicaciones web. Este cambio permitirá complementar aplicaciones de escritorio con aplicaciones similares con el plus de la universalidad de Internet permitiendo acceder a información de manera remota pero siempre con una interface similar y un numeroso grupo de utilidades que revalorizarán el producto convirtiendo cualquier desarrollo en algo más que una "página web".

A continuación mencionamos algunas desventajas y ventajas de utilizar AJAX en nuestras aplicaciones web:

Desventajas

- El cliente necesita un navegador que soporte javascript .Hoy por hoy la mayoría de los navegadores soporta JavaScript. Internet Explorer, Mozilla, Firefox, Safari, entre otros, ya lo traen incorporado, por lo que no se discrimina un espectro muy amplio de usuarios.
- El mal uso de ajax/javascript puede mal emplearse sobrecargando el servidor de peticiones, por ejemplo, si se realizan cada milisegundo una consulta a la base de datos.

Ventajas

- Nos permite diseñar interfaces mucho más dinámicas acercándonos a "aplicaciones de escritorio".
- La comunicación asíncrona con el servidor nos permite varias cosas que reducen el "peso de la

página", "líneas de código que el cliente tiene que descargarse".

- Campos de selección, si las opciones son muchas, no es necesario transmitir las en la primera carga de la página, podemos producir la lista de opciones cuando el cliente pulse sobre el desplegable. Otra solución es hacer una búsqueda dinámica.
- La navegación por la aplicación mediante AJAX nos permite evitar al cliente descargar la cabecera de todos los documentos y pantallas. Podemos cambiar el contenido de cualquier objeto del DOM dinámicamente ante los eventos que controlamos con javascript.
- No requiere plugins o capacidades específicas de ciertos navegadores a excepción de javascript que actualmente ya viene incorporado en la mayoría de navegadores.
- Las aplicaciones son más interactivas, responden a las interacciones del usuario más rápidamente, al estilo desktop.

- Se reduce el tamaño de la información intercambiada.
- Muchas micro-peticiones, pero el flujo de datos global es inferior.

2.3.2. Justificación del uso de la tecnología AJAX

Básicamente se decidió utilizar AJAX para mejorar la interacción de los suscriptores del portal con las diferentes pantallas que involucran los procesos de convocatoria y evaluación de artículos en los eventos que se organicen en el componente ocho.

Como se menciona en la descripción de la tecnología, su aplicación da como resultado que el usuario del sistema, no sienta la manipulación de datos ni las transacciones que se ejecutan en el interior de la aplicación.

Características como: flexibilidad, velocidad, y usabilidad, hacen que AJAX sea atractiva para aplicaciones que necesitan minimizar el tiempo de respuesta a solicitudes de usuarios al interactuar con las pantallas.

CAPITULO 3

3. ANÁLISIS DEL SISTEMA

3.1. Requerimientos funcionales

De manera general los servicios que deben proporcionar los módulos son:

- Convocatoria de artículos: los visitantes del portal podrán enviar sus trabajos de investigación, a cada evento organizado por el componente 8. Por otro lado, los administradores en esta sección, podrán publicar y administrar la convocatoria de artículos para eventos y recibir todas las publicaciones que los suscriptores envían.
- Evaluación de Artículos: los administradores del portal, podrán asignar evaluadores y revisar las evaluaciones recibidas, así

mismo tendrán disponibles pantallas para la aprobación o desaprobarción de artículos recibidos. Por otro lado, los revisores, podrán evaluar los artículos que se le asignaron en el mismo sitio web y enviar sus evaluaciones a través de pantallas disponibles en el portal, y los suscriptores que enviaron sus publicaciones podrán revisar el estado del proceso de evaluación de sus trabajos.

A continuación se detallan los requerimientos para ambos módulos:

CONVOCATORIA DE ARTÍCULOS:

- El módulo deberá permitir crear una convocatoria y asignarla a un evento específico. Para la creación de la convocatoria se deberán almacenar los siguientes datos: Referencia al evento, título de la convocatoria, fecha máxima de recepción de artículos, fecha máxima de evaluación de artículos, fecha máxima de notificación de resultados de artículos evaluados, fecha máxima de recepción de artículos corregidos, guía de referencia para la redacción de los artículos, información adicional, estado de la convocatoria: creada o finalizada.
- Debe permitir la publicación de artículos para cada convocatoria. Para la creación de artículos se deberá

almacenar la siguiente información: referencia al tema, título del artículo, referencia al autor principal, referencia al archivo del artículo, estado del Artículo: existe, evaluando, aceptado, rechazado, corregido, finalizado.

- El administrador debe poder enviar un correo electrónico de difusión de la convocatoria a una lista de correos. El correo deberá contener la información guardada de la convocatoria.
- Debe permitir la creación de temas para cada convocatoria. La información a guardar para cada tema será: un título del tema y el estado del tema: creado y finalizado.
- La funcionalidad del módulo deberá contemplar que las fechas relacionadas a la convocatoria, lógicamente no se superpongan, además, esas fechas no deben pasar la fecha inicial del evento.
- Se debe considerar que solo los suscriptores al portal pueden publicar un artículo a un evento.
- No se pueden crear convocatorias sino existe el evento previamente creado.
- No deben existir dos convocatorias asignadas a un mismo evento, solo una convocatoria por evento.
- Se debe permitir modificación y eliminación lógica de una convocatoria.

- Se debe permitir la modificación y eliminación lógica de un tema de una convocatoria.
- Cuando se crea por primera vez una convocatoria, se debe crear al menos un Tema.
- El usuario que publica un artículo podrá verificar el estado de su artículo en todo momento.
- Los suscriptores del portal podrán enviar varios artículos en un mismo evento.
- Antes de asignar artículos a revisores, el administrador puede rechazar artículos recibidos.
- Cuando se está creando la convocatoria de artículos el administrador debe poder visualizar el contenido del correo electrónico que se enviará como difusión de convocatoria.
- Los suscriptores del portal podrán modificar el trabajo enviado siempre que no haya pasado la fecha máxima de envío y que no se le haya asignado un evaluador.

EVALUACION DE ARTÍCULOS:

- Permitir asignar evaluadores a los artículos recibidos en una convocatoria, siempre que la evaluación hay sido creada.
- Se debe enviar un correo electrónico automático a cada evaluador una vez asignado el artículo, dando las indicaciones

para evaluar el artículo asignado (en el correo se debe adjuntar el archivo).

- Debe permitir crear una evaluación por cada convocatoria de artículos.
- Permitir crear secciones para la evaluación
- Permitir crear preguntas en la sección
- Permitir crear alternativas a una pregunta de selección simple o múltiple.
- Las preguntas pueden ser: de selección simple, múltiple o abiertas
- El evaluador podrá ingresar al portal y evaluar un artículo
- Cuando un usuario con rol de Evaluador ingresa al portal debe aparecerle un menú de Evaluación de artículos
- El Evaluador debe poder visualizar un listado de todos los artículos por evento que tenga que evaluar.
- El administrador debe poder visualizar un listado del estado de los artículos y en cada artículo evaluado debe poder revisar la evaluación recibida.
- Para asignar el rol de Evaluador, el usuario debe estar previamente registrado en el portal
- Debe permitir quitar el rol de Evaluador a un Usuario

- Para poder asignar algún evaluador a un artículo, previamente debe estar creada una evaluación para la convocatoria
- Permitir modificar o eliminar secciones de la evaluación, siempre que no se haya asignado un evaluador a algún artículo.
- Permitir modificar o eliminar preguntas de una sección, siempre que no se haya asignado un evaluador a algún artículo.
- Permitir modificar o eliminar alternativas de una pregunta de selección simple o múltiple, siempre que no se haya asignado un evaluador a algún artículo.
- Un Evaluador no puede evaluar un artículo si la fecha actual excede la fecha de evaluación propuesta en la convocatoria
- El administrador solo podrá aprobar un artículo cuando al menos un revisor lo haya aprobado, caso contrario se lo considera como artículo rechazado.
- Permitir que un evaluador pueda eliminar la evaluación realizada y volver a llenarla, siempre y cuando lo haga antes de la fecha máxima de evaluación de artículos.
- No se debe permitir modificar ni eliminar una evaluación si ya existe un artículo evaluado
- Cuando se haga una eliminación de una evaluación se debe realizar una eliminación lógica en todos los objetos

relacionados, eliminando las secciones, preguntas y alternativas.

- Se debe permitir crear secciones privadas entre el revisor y los organizadores del evento.
- El administrador debe poder quitar un evaluador asignado a un artículo, siempre y cuando no haya llenado una evaluación.
- El administrador debe poder visualizar un reporte por evento sobre el estado de los artículos recibidos.
- Una vez que el revisor ingresa su evaluación el sistema debe enviar un correo electrónico al correo del evento de que un artículo ya fue evaluado.
- Cuando se asignan o se quita la revisión de un artículo a un evaluador, el sistema debe enviar automáticamente un correo electrónico indicando el cambio ocurrido.
- Cuando se asigna o se quita el rol de revisor a un suscriptor del portal, el sistema debe enviar automáticamente un correo electrónico indicando el evento ocurrido.
- Cuando un artículo luego de ser evaluado es aceptado o rechazado, el sistema debe automáticamente enviar un correo electrónico al autor indicando lo sucedido.

- El administrador debe poder enviar un correo electrónico a todos aquellos revisores que tienen pendiente sus evaluaciones.
- El administrador no debe poder enviar un recordatorio a los revisores sin antes haberles asignado al menos un artículo para evaluación.
- El sistema debe permitir que un autor envíe la corrección de su artículo hasta la fecha máxima para envío de correcciones, una vez que ha sido aprobado por el comité evaluador.
- El usuario que publica un artículo podrá ver el resultado de la evaluación de su artículo, una vez que haya sido aprobado o rechazado por el comité evaluador.

3.2. Requerimientos no funcionales

Los requerimientos en cuanto a hardware son los siguientes:

- Capacidad de Memoria en el ordenador: Para un buen desempeño del sistema, una memoria de 64 MB de RAM (recomendado 128 MB o superior).
- Capacidad de Memoria en el servidor: Se recomienda un mínimo de 1GB de RAM.

- Capacidad de disco duro en el ordenador: Para un buen desempeño del sistema se recomienda un disco duro mínimo de 4 GB.
- Capacidad de disco duro en el servidor: Se recomienda un de disco duro de 80 GB como mínimo.
- Tipo de procesador en un ordenador: Pentium II de 450 MHz ó equivalente (recomendado Pentium III, equivalente ó superior).
- Tipo de procesador en servidor: Se recomienda un procesador PIII 900MHz.
- Ancho de Banda para el Servidor: Ancho de banda recomendado 128Mbits.
- Navegadores: Para un buen uso del sistema, se requiere Internet Explorer 5.0 o superior y Firefox 1.5 o superior.
- Sistema Operativo: Microsoft Windows 98 o superior o cualquier versión de Linux.

3.3. Modelo del dominio del sistema usando técnicas de MERODE

3.3.1. El gráfico de la dependencia de existencia

A continuación se muestra el gráfico de la dependencia de existencia o EDG en la figura 3.1.

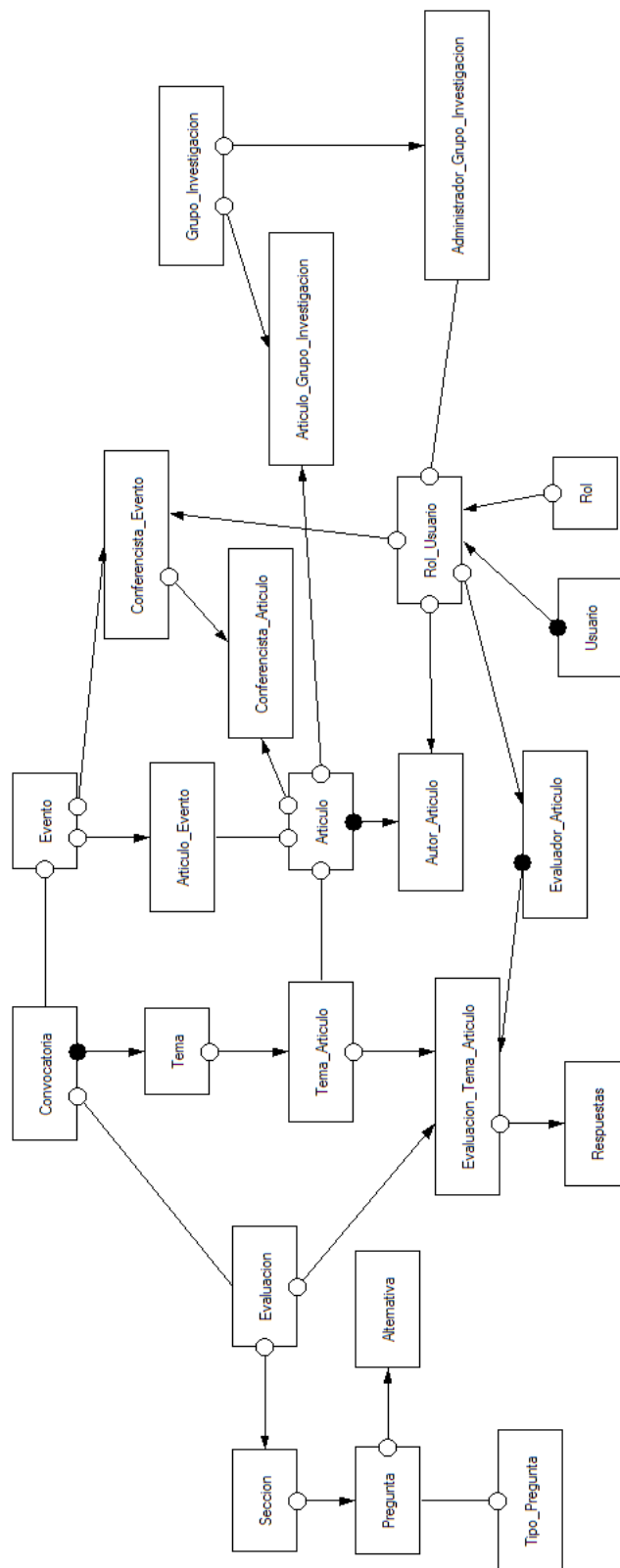


Figura 3.1 Gráfico de la dependencia de Existencia – EDG

Se necesitaron un total de 23 objetos para implementar los dos módulos adicionales al portal; de los cuales 10 objetos son adicionales al EDG original. Actualmente, en total el portal consta de 42 objetos del negocio que conforman su EDG final.

A continuación listaremos los 23 objetos del negocio que forman parte del EDG para la implementación de los módulos, cabe recalcar que los algunos de los nombres de ciertos objetos aparecerán sin tildes, debido a restricciones de la herramienta donde se diseñó el EDG.

1. Evento
2. Convocatoria
3. Tema
4. Tema_Articulo
5. Evaluacion_Tema_Articulo
6. Respuestas
7. Evaluacion
8. Seccion
9. Pregunta
- 10.Tipo_Pregunta
- 11.Alternativa
- 12.Articulo_Evento

- 13.Articulo
- 14.Conferencista_Evento
- 15.Conferencista_Articulo
- 16.Autor_Articulo
- 17.Evaluador_Articulo
- 18.Rol_Usuario
- 19.Rol
- 20.Usuario
- 21.Articulo_Grupo_Investigacion
- 22.Grupo_Investigacion
- 23.Administrador_Grupo_Investigacion

Seguidamente se explicará por qué se escogió a cada uno de los objetos listados como objetos del sistema, pero solo aquellos 23 objetos que fueron necesarios para implementar los dos módulos adicionales al portal.

Evento:

Representa los eventos que realizan los grupos de investigación ya sean estos cursos, talleres, seminarios, jornadas, etc. Está relacionado con *Conferencista_Evento* para relacionar los usuarios a los que se les asigna el rol de Conferencista Invitado

y que serán conferencistas del evento, con *Articulo_Evento* para tener registrados los artículos o publicaciones de los conferencistas que participarán en el evento, con *Convocatoria* ya que cada evento puede tener asociada una convocatoria.

Convocatoria:

Este objeto representa la llamada a presentación de artículos para algún evento. Una convocatoria tiene asociados temas en los cuales deben basarse los artículos a presentarse. Por lo tanto se relaciona con el objeto *Tema*.

Tema:

Son las áreas en las cuales debe basarse un artículo que se presenta para ser expuesto en un evento y está asociada con *Tema_Articulo* para establecer esta relación.

Tema_Articulo:

Objeto que permite relacionar un artículo que se presenta para participar en la convocatoria de un evento y el tema al cual pertenece dicho artículo.

Articulo:

Este objeto representa los artículos que expondrán los conferencistas de los eventos, los artículos que se presentan para participar de la convocatoria, y las publicaciones que realizan los grupos de investigación. Se relaciona con

Articulo_Evento para establecer la relación con los artículos de los conferencistas, con *Tema_Articulo* para establecer la relación con los artículos participantes de una convocatoria y con *Articulo_Grupo_Investigación* para establecer la relación entre las publicaciones y los grupos de investigación a los que pertenecen.

Articulo_Evento:

Objeto que permite relacionar el artículo de un conferencista a un evento.

Autor_Articulo:

Este objeto relaciona a un artículo con su autor. Es dependientemente existente de *Articulo* porque para que exista *Autor_Articulo* debe existir *Articulo* y de *Rol_Usuario* porque el usuario para ser autor debe tener asignado dicho rol.

Usuario:

Persona que se suscribe al portal y se obtiene de ella sus datos, se relaciona con *Rol_Usuario* para asignarle roles dentro del sistema, con *Empresa_Usuario* para indicar la empresa o universidad en la cual labora o estudia, y con *Registro* para tener constancia del registro a los eventos que realicen los grupos de investigación.

Rol:

Objeto que servirá para definir el rol que ocupa un usuario dentro del sistema mediante el objeto *Rol_Usuario* ya sea como suscriptor, conferencista, administrador.

Rol_Usuario:

Este objeto relaciona a un usuario del sistema y sus roles. Es dependientemente existente tanto de usuario como de rol y debe existir como mínimo un rol asociado a un usuario.

Administrador_Grupo_Investigacion:

Objeto que permite relacionar un usuario cuyo rol es de administrador a un grupo de investigación.

Conferencista_Evento:

El objeto Conferencista_Evento permite especificar que usuarios cuyo rol es Conferencista Invitado, está relacionado a un evento, además este objeto permite establecer una conexión entre el evento y el conferencista para que este último pueda subir el o los artículos a exponer.

Conferencista_Articulo:

El objeto Conferencista_Articulo permite especificar aquellos usuarios que han publicado un trabajo de investigación y han sido aprobados para que sean ponencias de un evento, además este objeto permite establecer una conexión entre el artículo y

el conferencista para saber cuántos artículos va a exponer un conferencista en determinado evento.

Grupo_Investigacion:

Es uno de los objetos principales de nuestro análisis ya que representa a cada uno de los grupos de investigación, posee administradores y publicaciones por grupos. Está relacionado con *Administrador_Grupo_Investigacion* para establecer quienes son los administradores de ese grupo, con *Articulo_Grupo_Investigacion* para definir sus publicaciones, y con *Organizador* para tener constancia de los eventos que organiza cierto grupo.

Articulo_Grupo_Investigacion:

Objeto que permite relacionar las publicaciones o artículos con algún grupo de investigación. Es dependientemente existente tanto de *Grupo_Investigacion* como de *Articulo*.

Evaluacion:

Este objeto Evaluacion servirá para almacenar las evaluaciones ingresadas para la revisión de aquellos artículos que se reciban en una convocatoria en un evento. Este objeto se relaciona con *Evaluacion_Tema_Articulo* para identificar aquellos artículos que están siendo revisados por evaluadores, quienes deben al final de su revisión llenar una evaluación determinada.

Evaluacion_Tema_Articulo:

Este objeto servirá para almacenar las relaciones que indican los artículos que están siendo revisados por determinados evaluadores de algún tema específico de una convocatoria, y se relaciona con el objeto Respuesta, el cual almacena las respuestas que envían los revisores a través de una evaluación. Además este objeto permite establecer la relación entre los objetos: Evaluacion, Tema_Articulo y Evaluador_Articulo.

Respuestas:

Son todas las respuestas que los evaluadores ingresan cuando han terminado de revisar los artículos que tienen asignados en un evento.

Evaluador_Articulo:

Son todos aquellos usuarios que tienen el rol de evaluadores de artículos y que pueden ser asignados por los organizadores de un evento cuando se haga una convocatoria de artículos. Se relaciona con el objeto Evaluacion_Tema_Articulo, el cual registra aquellos artículos que tienen asignado un evaluador para su revisión.

Seccion:

Este objeto registra todas aquellas secciones que se estén creando en una evaluación, y se relaciona con el objeto Pregunta.

Tipo_Pregunta:

El objeto Tipo_Pregunta registra los diferentes tipos de preguntas que pueden existir, para la implementación de los módulos se utilizaron preguntas de tipo: abierta, simples y múltiples, este objeto se relaciona con el objeto Pregunta.

Pregunta:

Este objeto registra todas las preguntas que se creen para las evaluaciones, y se relaciona con el objeto Alternativa.

Alternativa:

El objeto Alternativa registra todas las alternativas posibles a cada pregunta que se formule en las evaluaciones.

3.3.2. La tabla de eventos de objetos

La tabla de eventos de objetos como ya se ha descrito en el capítulo 2 es una matriz que contiene una fila por cada tipo de evento y una columna por cada tipo de objeto. Debido a que es muy extensa se ha procedido a separarla por objetos para que

sea visualizada de una mejor manera como podemos apreciar en las tablas 3.1, 3.2 y 3.3.

	Convocatoria
cr_Convocatoria	O/C
end_Convocatoria	O/E
modificar_Convocatoria	O/M
cr_Tema	A/M
end_Tema	A/M
cr_Tema_Articulo	A/M
end_Tema_Articulo	A/M
modificar_Tema	A/M

Tabla 3.1 OET – Objeto Convocatoria

	Tema
cr_Tema	O/C
end_Tema	O/E
cr_Tema_Articulo	A/M
end_Tema_Articulo	A/M
modificar_Tema	O/M

Tabla 3.2 OET – Objeto Tema

	Tema_Articulo
cr_Tema_Articulo	O/C
end_Tema_Articulo	O/E

Tabla 3.3 OET – Objeto Tema_Articulo

Las tres tablas presentadas anteriormente especifican que eventos (filas) afectan a cada uno de los objetos (columna) del modelo, respetando las reglas de propagación, regla del tipo de efecto y regla de contrato.

Debido a la regla de propagación todos los tipos de eventos en el alfabeto de un tipo de objeto dependiente son propagados a

su tipo de objeto maestro, por eso es necesario indicar el tipo de proveniencia, es decir si es: propio (O: Owned), adquirido (A: Acquired), Especializado (S: Specialised) o Heredado (I: Inherited).

Para visualizar las tablas de eventos de los otros objetos referirse al Apéndice B.

3.3.3. Máquina de estados finitos relevantes: FSM del objeto Artículo

El objeto Artículo no tiene un ciclo de vida por defecto como se puede apreciar a continuación en la figura 3.3.

Este objeto representa todos aquellos artículos que recibe el grupo de investigación en cada una de las convocatorias de los eventos que organiza y a su vez, aquellos que publiquen sus investigadores por los trabajos de investigación realizados.

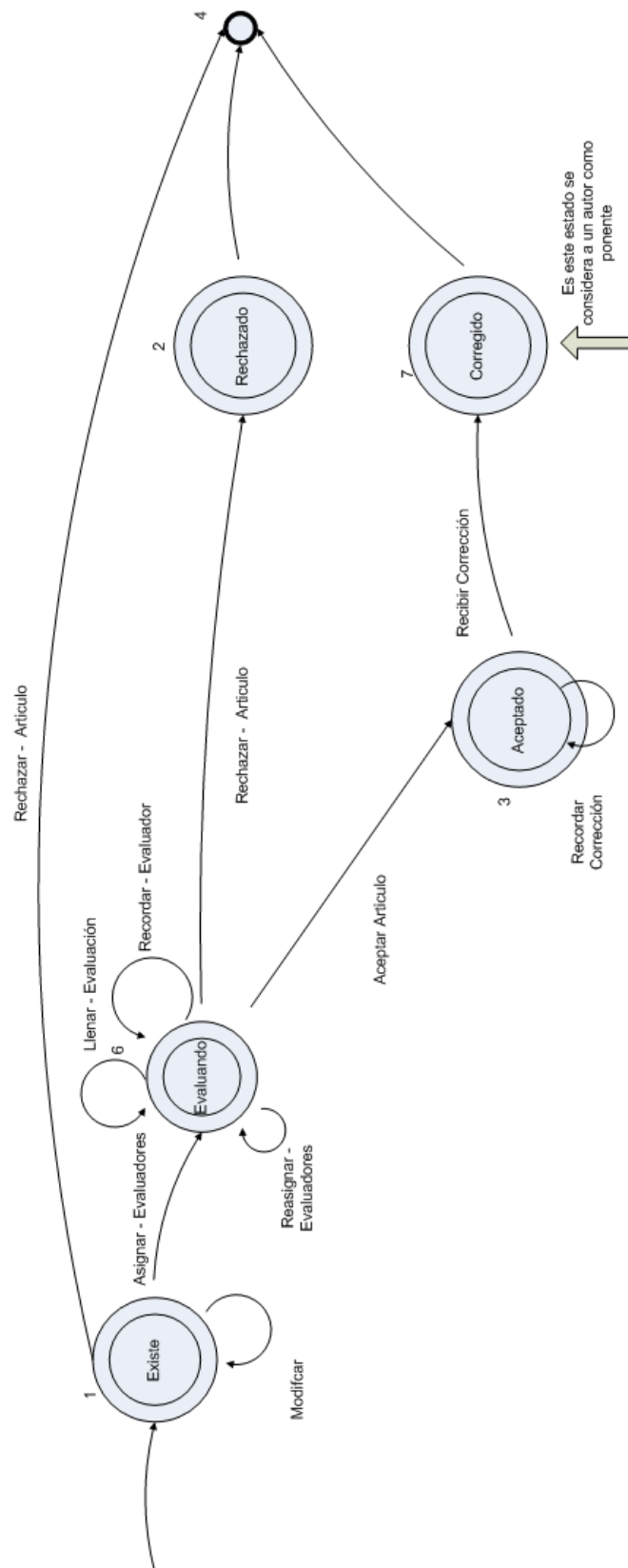


Figura 3.3 FSM del objeto Articulo

El objeto Artículo se crea mediante el evento *cr_Articulo* y su estado cambia a *existe*. Una vez creado, el objeto cambia al estado *evaluando*, una vez que los organizadores de un evento asignan al menos un evaluador al artículo.

Luego, al momento que los evaluadores asignados han realizado la revisión del artículo y han llenado la evaluación, el estado del objeto cambia a *evaluado*.

Una vez que el comité organizador recibe al menos una evaluación de un revisor, el objeto pasa al estado *aprobado* cuando se acepta al artículo como ponencia de un evento, caso contrario si el comité decide no aprobarlo, el objeto pasa al estado de *rechazado*.

Una vez que el objeto ha sido aceptado, el autor del artículo deberá enviar la corrección en base a las sugerencias de los revisores y una vez recibida, el objeto pasa al estado *corregido*, y recién en este estado el autor del artículo es considerado como ponente del evento.

3.4. Definición de roles en los módulos

Para la interacción con los módulos implementados en el portal, a continuación se especifican los roles que se han considerado indicando su función en el negocio y en el sistema. Adicionalmente, también se ha considerado los roles principales identificados en la tesis que implementó el portal.

Usuario	Función dentro del negocio	Función dentro del sistema (Rol)
Director del grupo de investigación	Encargado de coordinar todas las actividades o eventos que realiza el grupo de investigación	Administrador General
Subdirectores del grupo de investigación	Encargado de coordinar actividades o proyectos específicos	Administrador General
Ayudantes o Asistentes del grupo de investigación	Encargados de la ejecución de una actividad, evento o investigación específica.	Administrador
Suscriptor	Participa de las actividades que desarrolla el grupo de investigación (eventos) y envía sus trabajos de investigación en las convocatorias de artículos.	Suscriptor
Evaluador	Evalúan los artículos que son recibidos en las convocatorias de artículos a eventos.	Evaluador

Tabla 3.4 Usuarios del sistema

CAPITULO 4

4. DISEÑO DEL SISTEMA

4.1. Modelos de diseño

4.1.1. Modelos de Componentes

El modelo de componentes muestra gráficamente aquellos componentes que son parte de la solución para construir el sistema.

En la figura 4.1 se muestran los componentes adicionales de Convocatoria de Artículos y Evaluación de Artículos que han sido añadidos al portal.

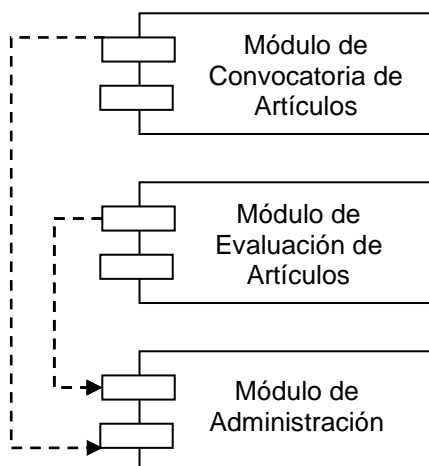


Figura 4.1 Modelo de Componentes

Cada uno de estos componentes que se muestra en la figura guarda internamente una estructura igual a la que se describe en la figura 4.2, es decir, conformados por la capa de interfaz del usuario, capa funcional, y capa de la empresa.

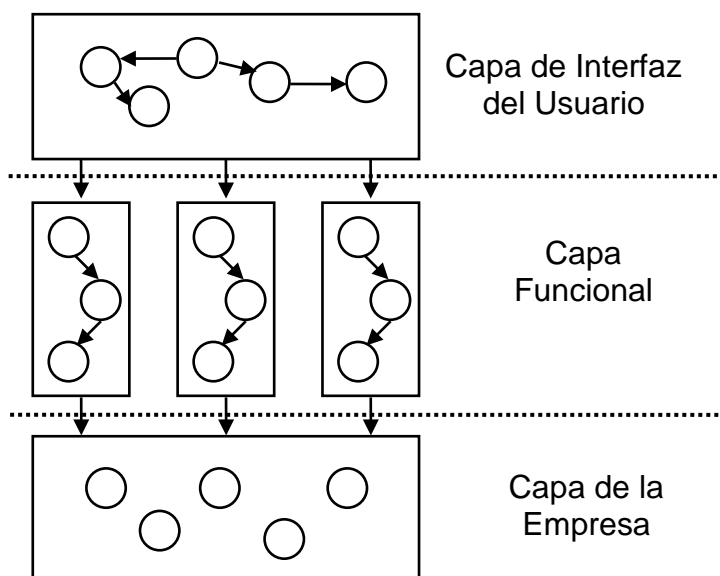


Figura 4.2 Arquitectura del Sistema de Información

MERODE

4.1.2. Arquitectura basada en patrones de diseño

4.1.2.1. Composite View

Este patrón facilita la composición de varias vistas que no necesariamente guarden relación entre sí. La finalidad es visualizar diversa información y facilitar la actualización y el mantenimiento del sistema. [6]

Como habíamos mencionado muchas de las aplicaciones Web agregan variada información en sus páginas, muchas veces poco relacionada con nuestro negocio.

El *composite view* proporciona un mecanismo para combinar modularmente, las diferentes partes con las que queremos construir una página.

A continuación en la figura 4.3 se ilustra la aplicación del patrón a la realidad del portal, tomando como referencia una de las funcionalidades del módulo de Evaluación de Artículos.

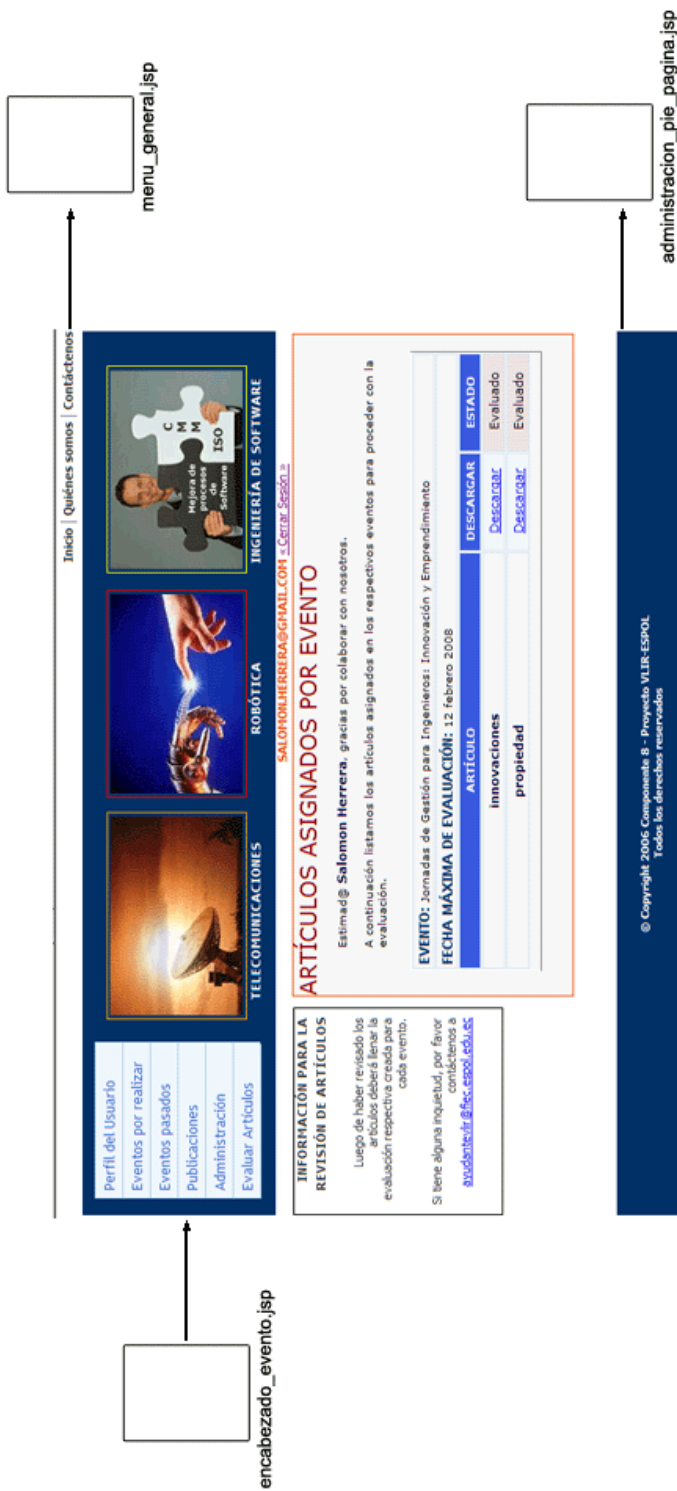


Figura 4.3 Composición de la página artículos_asignados.jsp

Como podemos notar la página artículos_asignados.jsp, está compuesta de tres páginas adicionales como son: encabezado_evento.jsp, menú_general.jsp y administración_pie_pagina.jsp; esta figura representa claramente el uso del patrón y en cada página adicional tendrá la misma composición que permitirá mantener fácilmente el sistema en caso ocurran cambios en alguna de las páginas que son parte de la composición, es decir, si se añade una opción en la página menú_general.jsp, automáticamente todas las páginas que la invoquen por composición presentarán el nuevo cambio realizado.

Para ilustrar mediante código jsp el uso del patrón, podemos apreciar en la figura 4.4 como se implementa la composición de vistas.

```

<body>
  <div id="div_body_general">
    <div class="align_right">
      <%@include file="../../../menu_general.jsp" %>
    </div>
    <%@include file="encabezado_evento.jsp" %>
    <%@include file="../../../usuario_registrado.jsp" %>
    <div id="div_body_evento">
      <div id="div_izquierdo">
        <%@include file="menu_administracion.jsp" %>
      </div>
      <div id="div_central">
        <form method="post" action="">
          <fieldset>

```

Figura 4.4 Código para armar la composición de la página artículos_asignados.jsp

Como podemos notar a través de la sentencia *include*, se invoca a cada una de las partes que conforman la página principal.

4.1.2.2. Model View Controller

Como se explicó en el capítulo II, este patrón es muy utilizado para el desarrollo de sistemas, trabaja en una clara separación de los objetos en tres categorías: *modelo* para mantenimiento de los datos, *vistas* para mostrar todo o una porción de los datos y *controladores* para administrar los eventos que afectan el modelo o las vistas. [7]

A continuación se muestra la figura 4.5 que presenta un ejemplo de la adaptación del MVC a la implementación de los módulos al portal.

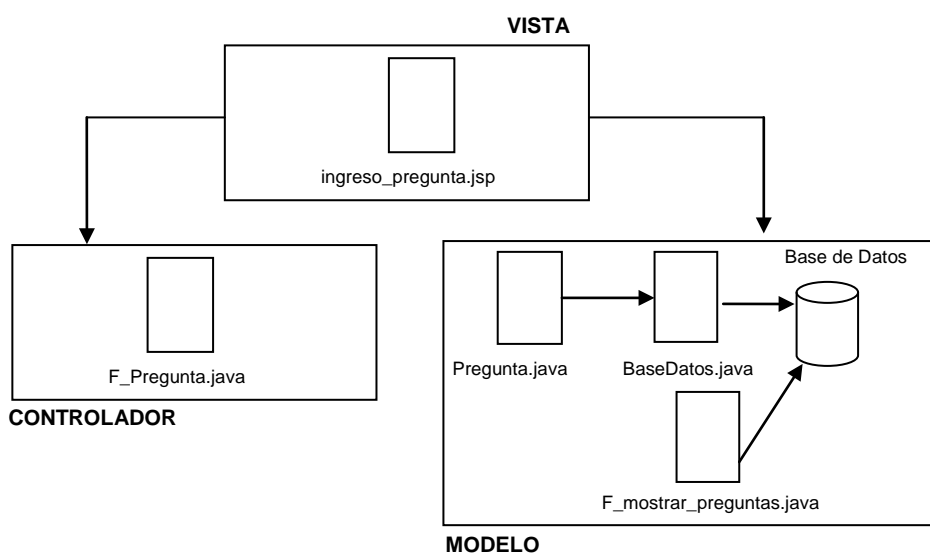


Figura 4.5 Ejemplo de aplicación del MVC en el ingreso de un pregunta en una evaluación

A través de esta figura se muestra un ejemplo del uso del patrón adaptado a la metodología MERODE, en este caso, se refiere al ingreso de una pregunta cuando se está creando una evaluación para la revisión de artículos en una convocatoria de un evento.

El usuario a través de la vista *ingreso-pregunta.jsp*, hace un ingreso de los datos de una pregunta y la vista hace una invocación al controlador *F_Pregunta.java*, quien a su vez, llama al modelo para interactuar con la clase *Pregunta.java* para poder acceder a la base de datos a través de la clase *BaseDatos.java*, finalmente para mostrar las preguntas que se han ingresado en determinada sección la clase *F_Pregunta.java* trabaja con la clase *F_mostrar_preguntas.java*, quien toma de la base de datos los registros correspondientes a través de la clase *BaseDatos.java*.

Así mismo, en la implementación se puede apreciar como en el MVC interactúan los controladores, las vistas y el modelo.

A continuación en las figuras 4.6, 4.7, 4.8 y 4.9 veremos cómo funciona toda esta interacción en el MVC para el mismo ejemplo de ingreso de una pregunta a una evaluación.

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<script language=javascript src=../../js/generales.js></script>
<script language=javascript src=../../js/admin_evaluacion.js></script>
<script language="JavaScript" type="text/javascript">
<!--
//función AJAX que envía los datos validados al servlet y que espera una respuesta
function pregunta ()
{
    var url = "/AppVlir8/F_Pregunta?operacion=" + operacion;
    // var url = "F_Seccion?operacion=" + operacion;
    url = url + "&titulo=" + document.form.txt_titulo.value + "&ref_tipo_pregunta=" + id_tipo_pregunta + "&id=" + <%=seccion.id_seccion%>;
    req = createXMLHttpRequest();
    req.open("GET", url, true);
    req.onreadystatechange = atender_pregunta;
    req.send(null);
}
}

```

Figura 4.6 Código en ingreso_pregunta.jsp cuando invoca a la clase F_Pregunta.java

```

package vlir8.funcionesentrada;

import java.io.IOException;

public class F_Pregunta extends HttpServlet
{
    public boolean OK = false;        //error por validación
    public String razon_fallo = "";   //razón de fallo por error de validación

    /* Para conexión con la base */
    public BaseDatos BD = new BaseDatos();

    /* Instanciar al objeto involucrado */
    private Seccion seccion;
    private Pregunta pregunta;
    private Alternativa alternativa;
}

```

Figura 4.7 Clase F_Pregunta.java cuando declara una instancia de la clase del modelo Pregunta.java

```

public class Pregunta
{
    /* referencias a padres (p)*/
    private Tipo_Pregunta tipo_pregunta;

    /* referencias a hijos (h)*/

    /* indicador de estado */
    public int estado;

    /* atributos adicionales */
    public int id_pregunta;
    public int ref_tipo_pregunta;
    public String titulo;

    /*más atributos, no pertenecen al objeto sino a sus padres o hijos*/

    /* Para conexión con la base */
    public BaseDatos BD;
}

```

Figura 4.8 Clase Pregunta.java cuando declara una instancia de la clase BaseDatos.java


```
*/  
public int ejecutarSP(String sp_nombre) throws SQLException  
{  
    try  
    {  
        BD = new Cconexion();  
        BD.conectar();  
        cn = BD.getConnection();  
        cn.setAutoCommit(false);  
        System.out.println(sp_nombre);  
        cs = cn.prepareCall(sp_nombre);  
        int rt = cs.executeUpdate(); //executeUpdate() utilizado  
        cn.commit();  
        return rt;  
    }  
}
```

Figura 4.9 Clase BaseDatos.java cuando invoca a la base de datos para ejecutar el procedimiento para ingresar los datos de la pregunta

CAPITULO 5

5. IMPLEMENTACIÓN Y PRUEBAS

5.1. Capas de implementación y comunicación entre capas

Como se mencionó en un inicio, nuestro objetivo era lograr implementar dos módulos que basados en el análisis con MERODE, se logren adaptar a un diseño utilizando la arquitectura J2EE.

5.1.1. Implementación basada en el análisis con MERODE y J2EE en el diseño

A través del uso de las técnicas de MERODE es posible identificar objetos del negocio y del sistema, que van a interactuar entre sí, para la comunicación, el paso de mensajes y el procesamiento de

requerimientos de quien use el sistema, para esto, la metodología distribuye los objetos en capas: Capa interna (Objetos de Negocio), capa media (Objetos de Sistema de Información) y capa externa (Interfaz con el usuario), lo cual asegura modularidad y además un fácil mantenimiento de la aplicación. Por otro lado, J2EE también divide al sistema en capas, asignando responsabilidades y permitiendo la comunicación entre objetos de cada capa para procesar información.

En nuestro caso nos hemos basado en el uso del patrón MVC de la arquitectura J2EE para la implementación de los módulos. Como habíamos indicado, el patrón está basado en la interacción entre el modelo, las vistas y controladores. Para nuestro caso, tomando como referencia el ejemplo de eliminar una *sección* en una evaluación de artículos, consideramos que en MERODE debe haber una interacción entre los objetos *Seccion*, *Pregunta* y *Alternativa*, invocando a sus métodos de *eliminación* para la comunicación entre ellos, y por otro lado, utilizando el patrón MVC la ejecución de este requerimiento es controlada por el objeto *F_Seccion*, quien hace las llamadas a los métodos de los demás objetos para su interacción.

Para este caso, utilizando MVC se ha implementado un controlador por cada objeto del negocio, así aseguramos el cumplimiento de la metodología MERODE y se asegura mayor facilidad para el mantenimiento cuando se requieran cambios en la aplicación.

5.2. Plan de Pruebas

Se ha utilizado el siguiente plan de pruebas y se ha seleccionado aquellas funcionalidades que se ejecutan con mayor frecuencia, para verificar que los módulos ejecutan los requerimientos según lo esperado.

Número	1
Nombre de la prueba	Envío de un artículo a una convocatoria
Instrucciones para la prueba	<ol style="list-style-type: none"> 1.- El Usuario ingresa al portal con el correo electrónico y clave con la que se suscribió. 2.- El usuario selecciona del menú principal la opción <i>Eventos por realizar</i>. 3.- El usuario escoge el evento del cual desea participar como ponente. 4.- El usuario elige la opción de <i>Enviar Trabajo</i> del menú del evento. 5.- El Usuario ingresa toda la información requerida para enviar su trabajo y elige la opción <i>Guardar y Cerrar</i> o <i>Guardar y Nuevo</i> en caso desee ingresar más artículos al mismo evento. 6.- El sistema verifica que la información ingresada esté correcta. 7.- El Usuario espera por una respuesta de confirmación.
Comportamiento aceptable	Si el usuario ingresa correctamente todos los datos solicitados, el sistema lo registrará como autor de un artículo o trabajo enviado a la convocatoria, mostrará un mensaje indicando que el envío ha sido exitoso y le enviará un mensaje de

	<p>confirmación por correo electrónico.</p> <p>Si el usuario ingresa datos erróneos o incompletos el sistema mostrará un mensaje indicando el o los campos que faltan por completar o que son erróneos.</p> <p>Si el usuario ingresa un título de un artículo ya existente, el sistema mostrará un mensaje de error.</p>
--	--

Número	2
Nombre de la prueba	Evaluación de un artículo
Instrucciones para la prueba	<p>1.- Un usuario suscrito al portal y con rol de revisor de artículos ingresa en la página principal el correo electrónico y la clave con la que se suscribió.</p> <p>2.- El usuario elige la opción <i>Evaluar Trabajos</i> del menú principal del portal.</p> <p>3.- El sistema muestra un listado de los artículos asignados para evaluación por evento.</p> <p>4.- El Usuario elige la opción por <i>Evaluar</i> dependiendo del artículo que vaya a evaluar.</p> <p>5.- El sistema muestra las preguntas de la evaluación que se deben llenar.</p> <p>6.- El usuario ingresa sus respuestas a cada pregunta y elige la opción <i>Enviar evaluación</i>.</p> <p>7.- El sistema verifica que la información ingresada esté correcta.</p> <p>8.- El usuario espera por una respuesta.</p>
Comportamiento aceptable	<p>Si el usuario ingresa correctamente su correo electrónico y clave, el sistema le permitirá el ingreso y le mostrará en el menú la opción de <i>Evaluar Trabajos</i>.</p> <p>Si el usuario no tiene asignado un artículo para evaluar, el sistema le mostrará un mensaje de que aún no le han asignado artículos para revisión.</p> <p>Si el usuario no ingresa todas las respuestas de selección simple y abierta que estén en la evaluación, el sistema le mostrará un mensaje de error.</p>

Número	3
Nombre de la prueba	El administrador aprueba o rechaza artículos evaluados.
Instrucciones para la prueba	1.- El administrador ingresa al sistema a través de su correo y clave, y luego pulsando el botón <i>Ingresar</i>

	<p>2.- El sistema muestra las opciones del menú a las que tiene acceso.</p> <p>3.- El administrador elige la opción Administración del menú general y luego la opción Eventos del menú administrativo</p> <p>4.- El administrador pulsa sobre el evento que desea administrar.</p> <p>5.- El sistema muestra el menú disponible para administrar el evento.</p> <p>6.- El administrador elige la opción <i>Trabajos Recibidos</i>.</p> <p>7.- El sistema muestra un listado de los trabajos que se han recibido en la convocatoria</p> <p>8.- El administrador elige la opción de <i>Ver Evaluaciones</i> correspondiente al artículo que desea aprobar o rechazar.</p> <p>9.- El sistema muestra un listado de las evaluaciones recibidas por cada evaluador asignado.</p> <p>10.- El administrador puede verificar los resultados de cada evaluación seleccionando la opción <i>Ver Evaluación</i>.</p> <p>11.- El administrador aprueba o rechaza el artículo seleccionando las opciones <i>Aceptar Artículo</i> o <i>Rechazar Artículo</i>.</p> <p>12.- El sistema procesa la solicitud del administrador.</p> <p>13.- El administrador espera por una respuesta.</p>
Comportamiento aceptable	Si el administrador seleccionó la opción <i>Aceptar Artículo</i> o <i>Rechazar Artículo</i> , el sistema enviará un correo electrónico automática al autor del artículo, informándole sobre la decisión del Comité Evaluador.

Número	4
Nombre de la prueba	El administrador envía un correo electrónico de recordatorio a todos los evaluadores que tienen pendiente la revisión de artículos en un determinado evento.
Instrucciones para la prueba	<p>1.- El administrador ingresa al sistema a través de su correo y clave, y luego pulsando el botón <i>Ingresar</i></p> <p>2.- El sistema muestra las opciones del menú a las que tiene acceso.</p> <p>3.- El administrador elige la opción <i>Administración</i> del menú general y luego la opción <i>Eventos</i> del</p>

	<p>menú administrativo</p> <p>4.- El sistema mostrará un listado de eventos que realizará el grupo de investigación al que pertenece el administrador. El administrador escoge el evento donde desea enviar el recordatorio.</p> <p>5.- Se presentará un menú de opciones, el administrador elige la opción <i>Envío de Correos</i>.</p> <p>6.- Aparecerán las opciones para enviar correos electrónicos, el administrador elige la opción <i>Recordatorios a evaluadores</i> y luego pulsa el botón de <i>Enviar mail</i>.</p> <p>7.- El sistema mostrará un mensaje que los correos han sido enviado a los evaluadores indicando en el mensaje aquellos artículos pendientes para revisión en ese evento.</p>
Comportamiento aceptable	Si el administrador no ha asignado ningún artículo para revisión, la opción de <i>Recordatorio a evaluadores</i> aparecerá deshabilitada.

Número	5
Nombre de la prueba	El administrador envía un correo electrónico de difusión de la convocatoria a una lista de contactos
Instrucciones para la prueba	<p>1.- El administrador ingresa al sistema a través de su correo y clave, y luego pulsando el botón <i>Ingresar</i></p> <p>2.- El sistema muestra las opciones del menú a las que tiene acceso.</p> <p>3.- El administrador elige la opción <i>Administración</i> del menú general y luego la opción <i>Eventos</i> del menú administrativo</p> <p>4.- El sistema mostrará un listado de eventos que realizará el grupo de investigación al que pertenece el administrador. El administrador escoge el evento donde desea enviar el recordatorio.</p> <p>5.- Se presentará un menú de opciones, el administrador elige la opción <i>Envío de Correos</i>.</p> <p>6.- Aparecerán las opciones para enviar correos electrónicos, el administrador elige la opción <i>Convocatoria a lista de contactos</i>.</p> <p>7.- El sistema muestra un caja de texto, al administrador ingresa la lista de correos electrónicos de contacto, separando cada correo con una <i>coma</i> ",".</p> <p>8.- El administrador pulsa el botón <i>Enviar mail</i>.</p> <p>9.- El sistema procesa el envío del correo a la lista de contactos.</p>

Comportamiento aceptable	Si el administrador ingresa información incorrecta, el sistema mostrará un mensaje de error.
---------------------------------	--

Número	6
Nombre de la prueba	El usuario revisa los comentarios de los evaluadores a su artículo una vez aprobado o rechazado en el evento.
Instrucciones para la prueba	<ol style="list-style-type: none"> 1.- El usuario ingresa al sistema a través de su correo y clave, y luego pulsando el botón <i>Ingresar</i> 2.- El usuario selecciona del menú principal la opción <i>Eventos por realizar</i>. 3.- El usuario escoge el evento en el que envió un trabajo. 4.- El usuario elige la opción de <i>Trabajos Enviados</i>. 5.- El sistema muestra una lista de todos los artículos que el usuario ha enviado en ese evento. 6.- El usuario revisa los comentarios de los evaluadores seleccionando la opción de <i>Aprobado</i> o <i>Rechazado</i>. 7.- El sistema muestra los comentarios de los evaluadores que revisaron el artículo.
Comportamiento aceptable	Si el comité evaluador aún no ha <i>aprobado</i> o <i>rechazado</i> un artículo, el sistema mostrará los estados <i>recibido</i> o <i>evaluado</i> según sea el caso.

Número	7
Nombre de la prueba	El administrador crea una evaluación de artículos con sus secciones preguntas y alternativas.
Instrucciones para la prueba	<ol style="list-style-type: none"> 1.- El administrador ingresa al sistema a través de su correo y clave, y luego pulsando el botón <i>Ingresar</i> 2.- El sistema muestra las opciones del menú a las que tiene acceso. 3.- El administrador elige la opción <i>Administración</i> del menú general y luego la opción <i>Eventos</i> del menú administrativo 4.- El sistema mostrará un listado de eventos que realizará el grupo de investigación al que pertenece el administrador. El administrador escoge el evento donde desea crear la evaluación. 5.- Se presentará un menú de opciones, el administrador elige la opción <i>Evaluación de Trabajo</i>. 6.- El sistema mostrará un formulario con la información que se debe ingresar.

	<p>7.- El administrador ingresa la información correspondiente a la cabecera de la evaluación: <i>mensaje al evaluador e instrucciones</i> y luego pulsa el botón <i>Guardar</i>.</p> <p>8.- El sistema verifica la información y habilita más opciones para administrar la evaluación.</p> <p>9.- El administrador pulsa el botón <i>Secciones</i> entre las nuevas opciones que habilita el sistema.</p> <p>10.- El sistema muestra un formulario con la información que se debe ingresar para crear las secciones.</p> <p>11.- El administrador ingresa la información de la sección y pulsa el botón <i>Guardar</i>.</p> <p>12.- El sistema verifica la información y actualiza la página mostrando la nueva sección creada.</p> <p>13.- El administrador pulsa sobre la opción <i>Preguntas</i> que aparece junto a cada sección que se haya ingresado.</p> <p>14.- El sistema muestra un formulario con los requerimientos para crear preguntas en una sección.</p> <p>15.- El administrador ingresa la información de la pregunta y pulsa el botón <i>Guardar</i>.</p> <p>16.- El sistema verifica la información y actualiza la página mostrando la nueva pregunta creada.</p> <p>17.- El administrador pulsa sobre la opción <i>Alternativas</i> que aparece junto a cada pregunta que se haya ingresado, si se la creó de alternativas simples o múltiples.</p> <p>18.- El sistema muestra un formulario con los requerimientos para crear alternativas en una pregunta.</p> <p>19.- El administrador ingresa la información de la alternativa y pulsa el botón <i>Guardar</i>.</p> <p>20.- El sistema verifica la información y actualiza la página mostrando la nueva alternativa creada.</p>
Comportamiento aceptable	Si el administrador a medida que ingresa la información de la evaluación, sección, pregunta o alternativa no lo hace correctamente, el sistema muestra un mensaje de error.

5.3. Diseño de métricas tomadas

Para el desarrollo de los módulos se establecieron métricas que permitan medir los avances de la implementación y el número de cambios que iban a afectar para determinar y comprobar que el uso de la metodología MERODE permite dar un mantenimiento fácil a los sistemas.

Para llevar a cabo la medición se establecieron las siguientes métricas:

- Tiempo real utilizado para el desarrollo de cada módulo (medido en horas trabajadas).
- Tiempo real vs. Tiempo estimado inicial de la codificación del proyecto en general.
- Cantidad de cambios de fondo, forma y error en cada componente.

A continuación se presenta el diseño de las plantillas

DIA	FECHA	MODULO	HORAS	ACTIVIDADES REALIZADAS

Tabla 5.1 Diseño de la plantilla utilizada para realizar la planificación del desarrollo de los módulos

En la tabla 5.1 podemos apreciar el diseño utilizado para realizar la planificación del desarrollo de los módulos basados en horas laborables por día. En el diseño encontramos el *día* en que se realiza algún trabajo, la *fecha* correspondiente, el *módulo* que se estaba trabajando, las *horas* que se planificaban para cada actividad y las *actividades* que se realizarían en ese día.

Este mismo diseño se utilizó para ir día a día llenando una bitácora del trabajo que se iba realizando.

DIA	FECHA	ESTIMADO	REAL	HORAS	MODULO

Tabla 5.2 Diseño de la plantilla utilizada para comparar las horas de trabajo real vs. trabajo estimado inicial

En la tabla 5.2 se muestra el diseño de la métrica para medir el tiempo real de desarrollo de los módulos vs. el tiempo estimado inicial. En la plantilla tenemos los siguientes campos: *día* en el que se está realizando el trabajo, la *fecha* correspondiente, las *horas que se estimaron* para ese día, las *horas reales* de trabajo, la *diferencia entre lo estimado vs. lo real* y el *módulo* que se está trabajando.

Finalmente en la tabla 5.3 se muestra el diseño de la métrica para medir el número de cambios de forma, fondo y error que se dieron durante la implementación de los módulos.

FECHA	TIPO DE CAMBIO	JUSTIFICACION

Tabla 5.3 Diseño de la plantilla utilizada para identificar los tipos de cambios que se iban solicitando

Para esta plantilla se definió plantear el campo *fecha* para saber en qué momento se solicitó el cambio, el *tipo de cambio* para saber a qué parte del proceso de implementación de los módulos se estaba afectando y la *justificación* para hacer el cambio.

5.4. Resultados de las Pruebas y Métricas tomadas

A continuación se presentarán los resultados obtenidos luego de hacer las pruebas y aplicar las métricas en el desarrollo de los módulos.

- RESULTADOS DE LAS PRUEBAS**

Prueba número:	1
Nombre de la prueba:	Envío de un artículo a una convocatoria
Número de Intento	Resultados
1	Se ingresaron datos erróneos e incompletos, y el sistema mostró mensajes de error indicando cuales campos se deben

	completar y que errores existen en los otros campos.
2	Se ingresó el título de un artículo ya recibido, y el sistema mostró un mensaje indicando que ya existe un artículo con ese título.
3	No se subió un archivo cuando se estaba ingresando la información de artículo y el sistema mostró un mensaje de error indicando que debe enviar el archivo correspondiente al artículo.

Prueba número: 2	
Nombre de la prueba: Evaluación de un artículo	
Número de Intento	Resultados
1	Se ingresó información incompleta y el sistema mostró mensajes de error de que las preguntas abiertas y de alternativas simples deben ser llenadas obligatoriamente.
2	Se llenaron todas las preguntas abiertas y de selección simple y el sistema mostró un mensaje de que la evaluación había sido ingresada sin problemas.

Prueba número: 3	
Nombre de la prueba: El administrador solicita la creación de un nuevo evento	
Número de Intento	Resultados
1	El administrador ingresó datos erróneos e incompletos, y el sistema mostró mensajes de error indicando cuales campos se deben completar y que errores existen en los otros campos.
2	El administrador ingresó correctamente la información relacionada al evento, esta información fue guardada correctamente en la base de datos y se pudo visualizar estos datos escogiendo el evento en la sección eventos por realizar

Prueba número: 3	
Nombre de la prueba: El administrador aprueba o rechaza artículos evaluados.	
Número de Intento	Resultados
1	El administrador intentó aprobar un artículo que aún no había sido evaluado, pero el sistema no muestra la opción sin que primero se haya recibido al menos una evaluación.
2	El administrador aprobó un artículo evaluado y el sistema

	mostró la página de artículos recibidos con el estado actual del artículo que fue aceptado como ponencia del evento.
--	--

Prueba número: 4	
Nombre de la prueba: El administrador envía un correo electrónico de recordatorio a todos los evaluadores que tienen pendiente la revisión de artículos en un determinado evento.	
Número de Intento	Resultados
1	El administrador intentó enviar un recordatorio a los evaluadores sin primero haber asignado artículos, y el sistema mantiene inactiva la opción hasta y muestra un mensaje de que no puede enviar hasta que al menos no asignado un artículo para revisión.
2	El administrador envió el correo de recordatorio a los evaluadores de un evento, y el sistema automáticamente armó por evaluador un correo con sus artículos pendientes por evaluar en ese evento.

Prueba número: 5	
Nombre de la prueba: El administrador envía un correo electrónico de difusión de la convocatoria a una lista de contactos	
Número de Intento	Resultados
1	El administrador envió un correo electrónico a todos los contactos que se tienen en la base de datos del Componente 8, y el sistema armó los correos automáticamente con la información de presentación de la convocatoria.
2	El administrador intentó enviar un correo de difusión a una lista de contactos externa ingresando información errónea e incompleta, el sistema mostró un mensaje de error.
3	El administrador ingresó una lista de contactos externa y envió el mail de difusión, y el sistema armó automáticamente el correo de difusión y realizó el envío.

Prueba número: 6	
Nombre de la prueba: El usuario revisa los comentarios de los evaluadores a su artículo una vez aprobado o rechazado en el evento.	
Número de Intento	Resultados

1	El usuario intentó revisar comentarios de artículos evaluados pero no aprobados, y el sistema no permite revisar los comentarios hasta que el administrador no acepte o rechaza el artículo.
2	El usuario revisó el estado de los artículos que envió a una convocatoria y de aquellos que ya están aprobados o rechazados por el comité evaluador, pudo revisar los comentarios que hicieron los evaluadores.

Prueba número: 7	
Nombre de la prueba: El administrador crea una evaluación de artículos con sus secciones preguntas y alternativas.	
Número de Intento	Resultados
1	El administrador intentó crear una evaluación sin haber creado una convocatoria previamente, y el sistema no muestra la opción sin que haya una convocatoria creada.
2	El administrador creó la cabecera de la evaluación con información incompleta, y el sistema mostró un mensaje de error.
3	El administrador creó la cabecera de la evaluación con la información requerida, y el sistema mostró un mensaje de éxito y habilitó las opciones adicionales para seguir creando la encuesta.
4	El administrador creó una sección con información incompleta, y el sistema mostró un mensaje de error.
5	El administrador creó una sección con la información requerida, y el sistema actualizó la página con la sección ingresada.
6	El administrador creó una pregunta con información incompleta, y el sistema mostró un mensaje de error.
7	El administrador creó una pregunta con la información requerida, y el sistema actualizó la página con la sección ingresada.
8	El administrador creó una alternativa con información incompleta, y el sistema mostró un mensaje de error.
9	El administrador creó una alternativa con la información requerida, y el sistema actualizó la página con la sección ingresada.

- **MÉTRICAS TOMADAS**

A continuación se presentan las métricas tomadas:

- ***Tiempo real utilizado para el desarrollo de cada módulo (medido en días laborables)***

La Tabla 5.1 presenta los componentes del sistema, el tiempo estimado, el tiempo real, la diferencia entre el tiempo real y el estimado (Tiempo A) y el porcentaje de la diferencia del tiempo A con respecto al tiempo estimado para cada componente.

Componente	Tiempo estimado (días laborables)	Tiempo real (días laborables)	Tiempo A: tiempo real – tiempo estimado (días laborables)	Porcentaje Tiempo A respecto al tiempo estimado (%)
Convocatoria de Artículos	40	73	33	83%
Evaluación de Artículos	40	49	9	23%

Tabla 5.4 Tiempos utilizados en el desarrollo de los módulos

En la tabla 5.4 podemos apreciar que el módulo que requirió mayor tiempo de implementación fue al módulo de *Convocatoria de Artículos*, muy probablemente se debió a que en principio se tenía que analizar el portal que ya había sido implementado, más la poca experiencia en el uso de la metodología MERODE y su integración a la arquitectura J2EE y además resultó ser una mala estimación del tiempo. Por otro lado, para el módulo de *Evaluación de Artículos* fue menos el porcentaje ya que se contaba con la experiencia de implementación del primer módulo y las bases para su implementación.

- ***Tiempo real vs. Tiempo estimado inicial de la codificación del proyecto en general.***

	Tiempo estimado inicial (meses)	Tiempo real (meses)	Tiempo A: tiempo real – tiempo estimado (meses)	Porcentaje Tiempo A respecto al tiempo estimado (%)
Proyecto General	4	6.1	2.1	52.5%

Tabla 5.5 Tiempos utilizados en el desarrollo del proyecto

En la tabla 5.5 podemos notar que hubo un retraso de 2.1 meses lo que representa un 52.5% más del tiempo que se estimó para la implementación de los módulos, muy probablemente como se explicó anteriormente, esto se debe a la poca experiencia en la metodología y la mala estimación de tiempo para el primer módulo.

- ***Cantidad de cambios de fondo, forma y error en cada módulo***

A través de esta métrica íbamos a medir aquellos *cambios de fondo*, es decir, aquellos en cuanto a los requerimientos que iban a afectar el EDG. Luego, todos aquellos cambios de visualización, colores, posición, léxico de las pantallas, estos serían los *cambios de forma*. Finalmente, los *cambios de error*, los cambios por errores internos del sistema al momento de la implementación.

A continuación se presenta la tabla que resume estos cambios.

Componente	Cambios		
	Fondo	Forma	Error
Convocatoria de Artículos	0	23	12

Evaluación de Artículos	2	28	14
Total de Cambios	2	51	26
Porcentaje (%)	1.27%	64.56%	32.91%

Tabla 5.6 Tabla de cambios de fondo, forma y error en los componentes

En la tabla 5.6 se observa que los cambios de forma son los que más se produjeron (64.56%), luego los cambios de error (32.91%) y finalmente los cambios de fondo con una menor porcentaje (1.27%).

Con estas mediciones hemos obtenido los resultados esperados, ya que la metodología MERODE asegura que en la capa más externa es donde ocurren más cambios, a diferencia de la capa más interna, donde se encuentran los objetos del negocio, que es en la que menos cambios ocurren.

CONCLUSIONES Y RECOMENDACIONES

Al finalizar este proyecto de tesis tenemos como resultado dos módulos añadidos al portal del componente 8 del proyecto VLIR-ESPOL que administra la convocatoria de artículos en eventos y la evaluación de los artículos recibidos para aceptarlos como ponencias en los eventos.

A continuación las conclusiones una vez finalizado el proyecto:

- Se explicó las ventajas de adoptar la metodología MERODE en el análisis de sistemas, para finalmente comprobar que el mantenimiento resulta muy fácil de administrar.
- Se utilizó MERODE como metodología de análisis y aunque en cierto punto la programación se volvió mecánica, resultó muy práctico aplicar el

patrón MVC que propone la arquitectura J2EE para la implementación de la funcionalidad, puesto que para ambos casos trabajan al sistema en capas estructuradas.

- Al finalizar la implementación de los módulos hubo una diferencia del 52.5% entre el tiempo estimado y el tiempo real para desarrollar cada módulo, lo que implica que hubo una mala estimación del tiempo para desarrollar el proyecto.
- Al iniciar el proyecto la primera dificultad fue adaptar el diseño de los nuevos módulos al diseño general del portal desarrollado en otro proyecto de tesis.
- Se demostró que el porcentaje de cambios que afectan las capas más internas en MERODE fue mucho menor que aquellos de las capas superiores, donde el porcentaje de cambios de forma y error fueron muy altos llegando 64.56% y al 32.91% respectivamente.
- Se logró integrar el análisis hecho con la metodología MERODE a los patrones de diseños propuestos para implementar los módulos bajo la arquitectura J2EE como son el Composite View y el Model View Controller.
- Se comprobó que MERODE ofrece flexibilidad al identificar en qué parte del código deberían hacerse los cambios reduciendo el costo de mantenimiento del sistema.

- Al implementar los módulos y añadirlos al portal del componente, se logró automatizar el proceso de convocatoria y evaluación de artículos en los eventos que organiza el grupo de investigación.
- Se aprendió a manejar la metodología MERODE para el análisis de proyectos reales de desarrollo de sistemas y la aplicación de patrones que aseguran una correcta implementación de funcionalidad utilizando la arquitectura J2EE.
- Se aprendió acerca de la administración y desarrollo de sistemas por capas, lo que asegura un mejor diseño y mantenimiento de las aplicaciones.
- Se logró tener un dominio y buena adaptación a sistemas ya implementados para añadir funcionalidad adicional.

Las recomendaciones que se pueden presentar con la elaboración de la presente tesis son:

- Implementar una funcionalidad que permita buscar e identificar artículos recibidos y aprobados en eventos a través de las palabras claves.
- Extender la convocatoria de artículos no solo a la publicación en eventos sino en proyectos de investigación adicionales, para publicación como un beneficio más a los usuarios que navegan en el portal.

- Implementar mayor flexibilidad para la convocatoria de artículos puesto que el subcomponente de Telecomunicaciones en primera instancia no va a trabajar con los artículos sino con los resúmenes hasta su aprobación y recién aprobados se procede a recibir los artículos completos.
- Se recomienda que si se va a añadir funcionalidad al portal se considere seguir utilizando el uso de patrones de la arquitectura J2EE adaptados al análisis realizado con MERODE pues resultó bastante semejante la utilización del patrón MVC al adaptarlo a lo analizado con la metodología.
- Brindar nuevos servicios al portal para todos los suscriptores como la implementación de una revista electrónica para publicación de trabajos de investigación que se estén desarrollando en empresas o universidades.
- Actualizar constantemente las herramientas en el servidor con nuevos parches que brinden más seguridad a la información que se está almacenando.
- Es recomendable aumentar la capacidad de procesamiento y capacidad de almacenamiento de información por la utilidad de la aplicación para el grupo de investigación.
- Es recomendable establecer otro tipos de métricas para medir los nuevos cambios que se hagan en el portal como cambios que afectan a lo ya

analizado e implementado, versiones de documentación, número de cambios en el código.

- Respecto a las pantallas del portal se recomienda mejorar la interacción en la parte administrativa pues en las pruebas se notó algo de confusión al momento de encontrar las opciones para los casos de uso.

APENDICE A

CASOS DE USO DE LOS MODULOS

LISTADOS DE CASOS DE USO:

Casos de Uso	ID
Suscriptor envía una publicación a un evento	CU-001
Suscriptor modifica la publicación enviada a un evento	CU-002
Suscriptor verifica el estado de las publicaciones enviadas a eventos	CU-003
Suscriptor elimina la publicación enviada a un evento	CU-004
Administrador crea una convocatoria	CU-005
Administrador añade nuevo tema a la convocatoria	CU-006
Administrador envía mail de difusión de la convocatoria	CU-007

Administrador revisa estado o acciones por realizar de artículos recibidos	CU-008
Administrador rechaza un artículo al que no se ha asignado un evaluador	CU-009
Administrador asigna revisores a un artículo	CU-010
Administrador revisa las evaluaciones enviadas por los revisores	CU-011
Administrador aprueba un artículo recibido en una convocatoria	CU-012
Administrador modifica la convocatoria para el evento	CU-013
Administrador modifica un tema de la convocatoria	CU-014
Administrador elimina un tema de la convocatoria	CU-015
Administrador revisa la evaluación creada	CU-016
Administrador crea una evaluación	CU-017
Administrador elimina una evaluación	CU-018
Administrador modifica una evaluación	CU-019
Administrador crea una sección a una evaluación	CU-020
Administrador modifica una sección de una evaluación	CU-021
Administrador crea una pregunta para una sección de una evaluación	CU-022
Administrador modifica una pregunta para una sección de una evaluación	CU-023
Administrador elimina una pregunta para una	CU-024

sección de una evaluación	
Administrador crea una alternativa para una pregunta para una sección de una evaluación	CU-025
Administrador modifica una alternativa para una pregunta para una sección de una evaluación	CU-026
Administrador elimina una alternativa para una pregunta para una sección de una evaluación	CU-027
Administrador asigna el rol de evaluador a un suscriptor	CU-028
Administrador quita el rol de evaluador a un suscriptor	CU-029
Evaluador revisa los artículos asignados para evaluación	CU-030
Administrador elimina una convocatoria de artículos de un evento	CU-031
Administrador elimina una sección de una evaluación	CU-032
Evaluador llena una evaluación para un artículo	CU-033
Evaluador descarga un artículo asignado para evaluación	CU-034
Administrador asigna las opciones del menú a un evento	CU-035
Iniciar sesión	CU-036
Suscribir usuarios al portal	CU-037
Administrador envía un mail de recordatorio a revisores que no han evaluado artículos	CU-038
Suscriptor envía corrección de artículo aprobado	CU-039

Administrador verifica el estado de todos los artículos	CU-040
Administrador visualiza el correo electrónico de difusión de la convocatoria mientras la va creando	CU-041
Evaluador elimina una evaluación ingresada	CU-042
Administrador rechaza un artículo que tiene evaluaciones recibidas	CU-043
Administrador quita un evaluador asignado a un artículo	CU-044
Suscriptor puede ver las evaluaciones del artículo enviado	CU-045

A continuación el detalle de cada caso de uso:

ID Caso de Uso	CU-001
Caso de Uso:	Suscriptor envía una publicación a un evento
Descripción:	Los suscriptores envían sus trabajos de investigación a un evento
Actores:	Suscriptor del portal
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber creado un evento ▪ Haber creado una convocatoria (CU-005) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elige la opción eventos por realizar del menú. 2. El sistema presenta un listado de los eventos que están por realizarse. 3. El usuario elige el evento en el que desea publicar su trabajo de investigación. 4. El sistema presenta el menú del evento y una descripción del evento. 5. El usuario elige la opción de <i>Enviar Trabajo</i> del menú del evento. 6. El sistema presenta un formulario con los siguientes campos: tema para relacionar el trabajo, título del trabajo, resumen, palabras claves y una opción para subir el trabajo al sistema. 7. El usuario llena los campos que se solicitan y sube el trabajo al 	

sistema.

8. El usuario escoge la opción que le permite enviar los datos.
9. El sistema valida la información ingresada por el usuario
10. El sistema registra el trabajo del usuario en el evento y presenta un aviso indicando que el trabajo ha sido registrado correctamente.
11. El suscriptor escoge la opción "Aceptar"

Extensiones (o Flujo Alternativo):

- 9a. El usuario ingresa datos erróneos o incompletos al sistema
 1. El sistema presenta un aviso indicando que existen datos por corregir o llenar.
 2. Volver al punto 6 del flujo normal
- 10b. El usuario escogió la opción de Guardar y Nuevo
 1. El sistema registra los datos del artículo ingresado.
 2. Volver al punto 6 del flujo normal.

Poscondiciones (Garantías de éxito):

- Los datos del artículo son almacenados en el sistema.
- El sistema envía un correo al suscriptor confirmando el registro de su artículo.

ID Caso de Uso	CU-002
Caso de Uso:	Suscriptor modifica la publicación enviada a un evento
Descripción:	El usuario modifica la publicación enviada a un evento determinado.
Actores:	Suscriptor
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elije la opción eventos por realizar del menú. 2. El sistema presenta un listado de los eventos que están por realizarse. 3. El usuario elije el evento en el que desea modificar su trabajo de investigación enviado. 4. El sistema presenta el menú del evento y una descripción del evento. 5. El usuario elije la opción de <i>Trabajos Enviados</i> del menú del evento. 6. El suscriptor elije el trabajo de investigación que va a modificar. 7. El sistema muestra un formulario con todos los campos que había ingresado el suscriptor. 8. El suscriptor modifica los datos que desee. 9. El usuario escoge la opción que le permite enviar los datos. 10. El sistema valida la información ingresada por el usuario 11. El sistema registra el trabajo del usuario en el evento y presenta un 	

aviso indicando que el trabajo ha sido registrado correctamente.

12. El suscriptor escoge la opción "Aceptar"

Extensiones (o Flujo Alternativo):

10a. El usuario ingresa datos erróneos o incompletos al sistema

1. El sistema presenta un aviso indicando que existen datos por corregir o llenar.
2. Volver al punto 8 del flujo normal

Poscondiciones (Garantías de éxito):

- Los datos del artículo son almacenados en el sistema.

ID Caso de Uso	CU-003
Caso de Uso:	Suscriptor verifica el estado de las publicaciones enviadas a un evento
Descripción:	El suscriptor podrá verificar en qué estado se encuentran las publicaciones que ha enviado a un evento.
Actores:	Suscriptor
Precondiciones:	<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036)
Escenario principal de éxito (Flujo normal):	<ol style="list-style-type: none"> 1. El usuario elige la opción eventos por realizar del menú. 2. El sistema presenta un listado de los eventos que están por realizarse. 3. El usuario elige el evento en el que desea verificar el estado de sus trabajos de investigación enviados. 4. El sistema presenta el menú del evento y una descripción del evento. 5. El usuario elige la opción de Trabajos Enviados del menú del evento. 6. El sistema muestra un listado de todos los trabajos que el suscriptor ha enviado indicando el estado en el que se encuentran.
Extensiones (o Flujo Alternativo):	<p>6a. El suscriptor no ha enviado un artículo a la convocatoria</p> <ol style="list-style-type: none"> 1. Se muestra un mensaje de que aún no se ha enviado un artículo. 2. Refiérase al Caso de Uso CU-001. 3. Volver al punto 5 del flujo normal
Poscondiciones (Garantías de éxito):	

ID Caso de Uso	CU-004
Caso de Uso:	Suscriptor elimina la publicación enviada a un evento

Descripción:	El suscriptor requiere eliminar la publicación enviada a un evento
Actores:	Suscriptor
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber enviado una publicación al evento (CU-001). <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elije la opción eventos por realizar del menú. 2. El sistema presenta un listado de los eventos que están por realizarse. 3. El usuario elije el evento en el que desea eliminar su trabajo de investigación enviado. 4. El sistema presenta el menú del evento y una descripción del evento. 5. El usuario elije la opción de Artículos Enviados del menú del evento. 6. El suscriptor elije el trabajo de investigación que va a eliminar. 7. El sistema muestra un formulario con todos los datos ingresados del artículo. 8. El usuario escoge la opción que le permite eliminar el artículo enviado. 9. El sistema muestra un mensaje de confirmación de querer eliminar el artículo. 10. El suscriptor escoge la opción "Aceptar". 11. El sistema muestra un mensaje de éxito. <p>Extensiones (o Flujo Alternativo):</p> <ol style="list-style-type: none"> 6a. El suscriptor no ha enviado un artículo a la convocatoria <ol style="list-style-type: none"> 1. Se muestra un mensaje de no haber publicado una artículo. 2. Refiérase al Caso de Uso CU-001. 3. Volver al punto 5 del flujo normal <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ Se actualiza el estado del artículo eliminado. 	

ID Caso de Uso	CU-005
Caso de Uso:	Administrador crea una convocatoria
Descripción:	El administrador crea la convocatoria con sus respectivos datos para invitar a concurso de artículos a ser presentados en el evento.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Que el evento haya sido creado ▪ Que el evento no tenga una convocatoria asignada. ▪ Estar en la pantalla de administración del portal 	

Escenario principal de éxito (Flujo normal):

1. El usuario elige la opción Eventos del menú administrativo.
2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro.
3. El usuario elige el evento al que le creará la convocatoria
4. El usuario elige la opción convocatoria del menú de opciones que aparece
5. El sistema presenta un formulario con los siguientes campos: descripción de la convocatoria, fecha de recepción del artículo, fecha de evaluación del artículo, fecha de notificación de aceptación del artículo, y fecha del envío de la versión final en el caso de ser aceptado el artículo, además de permitir subir el archivo que contiene el formato de presentación de los artículos, y el primer tema que posee la convocatoria.
6. El usuario llena los datos solicitados
7. El usuario escoge la opción guardar
8. El sistema valida la información ingresada por el usuario
9. El sistema registra la convocatoria del evento y presenta un aviso indicando que ha sido creada correctamente
10. El suscriptor escoge la opción "Aceptar"

Extensiones (o Flujo Alternativo):

- 4a. El menú del evento no tiene la opción convocatoria
 1. Refiérase al Caso de Uso CU-035: Administrador asigna las opciones del menú a un evento
 2. Luego volver al punto 4 del flujo normal
- 9a. El usuario ingresa datos erróneos o incompletos al sistema, por ejemplo, las fechas que se ingresan no deben superponerse.
 1. El sistema presenta un aviso indicando que existen datos por corregir o llenar.
 2. Volver al punto 6 del flujo normal

Poscondiciones (Garantías de éxito):

- La convocatoria y el primer tema de la convocatoria son almacenados en el sistema.
- Se puede visualizar el mail de la convocatoria CU-041.

ID Caso de Uso	CU-006
Caso de Uso:	Administrador añade un nuevo tema a la convocatoria
Descripción:	El administrador adiciona un tema que podrá ser tratado en la invitación a presentación de artículos o convocatoria de un evento específico.
Actores:	Administrador
Precondiciones:	

- Haber iniciado sesión (CU-036)
- Haber escogido la opción administración del menú general.
- Haber creado la convocatoria para el evento.

Escenario principal de éxito (Flujo normal):

1. El usuario elige la opción Eventos del menú administrativo.
2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro.
3. El usuario elige el evento al cual se le adicionarán temas a su convocatoria
4. El sistema presenta todas las opciones del menú de ese evento
5. El usuario elige la opción convocatoria del menú de opciones que aparece
6. El usuario elige la opción añadir nuevo tema a convocatoria
7. El sistema presenta un formulario donde el usuario debe llenar el nombre del nuevo tema para la convocatoria.
8. El usuario llena los campos que se solicitan
9. El usuario escoge la opción guardar y cerrar
10. El sistema valida la información ingresada por el usuario
11. El sistema muestra un aviso indicando que el tema ha sido creado correctamente.
12. El usuario escoge la opción "Aceptar"

Extensiones (o Flujo Alternativo):

- 9a.** El usuario desea ingresar otro nuevo tema para la convocatoria
1. El usuario elige la opción guardar y nuevo
 2. El sistema valida la información ingresada por el usuario
 3. El sistema muestra un aviso indicando que el tema ha sido creado correctamente.
 4. El usuario escoge la opción "Aceptar"
 5. Volver al punto 7 del flujo normal
- 10ª** El usuario ingresa datos erróneos o incompletos al sistema
1. El sistema presenta un aviso indicando que existen datos por corregir o llenar.
 2. Volver al punto 8 del flujo normal

Poscondiciones (Garantías de éxito):

- Un nuevo tema de la convocatoria es creado en el sistema

ID Caso de Uso	CU-007
Caso de Uso:	Administrador envía mail de difusión de la convocatoria
Descripción:	El administrador requiere realizar una difusión de la convocatoria
Actores:	Administrador
Precondiciones:	

- Haber iniciado sesión (CU-036)
- Haber escogido la opción administración del menú general

Escenario principal de éxito (Flujo normal):

1. El usuario elige la opción Eventos del menú administrativo.
2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro.
3. El usuario elige el evento del cual se hará una difusión de la convocatoria.
4. El sistema presenta todas las opciones del menú de ese evento
5. El usuario elige la opción *Envío de Correo* del menú de opciones que aparece.
6. El sistema muestra tres opciones: enviar un recordatorio a los revisores que tienen pendiente la evaluación de trabajos y hacer la difusión para todos los suscriptores del portal o para una lista de correos.
7. El usuario elige la opción que necesita y luego escoge la opción para enviar la difusión de la convocatoria.
8. El sistema verifica la información y envía la cadena de correos electrónicos.

Extensiones (o Flujo Alternativo):

- 8a.** El usuario ingresa datos erróneos o incompletos al sistema cuando elige la opción de envío a una cadena de correos.
1. El sistema presenta un aviso indicando que existen datos por corregir o llenar.
 2. Volver al punto 6 del flujo normal

Poscondiciones (Garantías de éxito):

ID Caso de Uso	CU-008
Caso de Uso:	Administrador revisa estado o acciones por realizar de artículos recibidos
Descripción:	El administrador de un grupo de investigación específico ingresa un nuevo evento
Actores:	Administrador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario selecciona el evento que desea revisar. 4. El sistema muestra el menú de opciones del evento. 	

5. El usuario selecciona la opción de *Estados Artículos* del menú de opciones del evento.
6. El sistema mostrará un listado de los artículos asignados y sus respectivos estados o acciones por realizar.

Extensiones (o Flujo Alternativo):

No existe flujo alternativo

Poscondiciones (Garantías de éxito):

No existen poscondiciones

ID Caso de Uso	CU-009
Caso de Uso:	Administrador rechaza un artículo al que no se le han asignado evaluadores
Descripción:	El administrador requiere rechazar un artículo recibido en un evento.
Actores:	Administrador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario selecciona el evento que desea revisar. 4. El sistema muestra el menú de opciones del evento. 5. El usuario selecciona la opción de <i>Trabajos Recibidos</i> del menú de opciones del evento. 6. El sistema mostrará un listado de los artículos asignados y sus respectivos estados o acciones por realizar. 7. El usuario elige la opción rechazar. 8. El sistema mostrará la información del mail que se enviará al autor. 9. El usuario ingresa los comentarios y escoge la opción de <i>Rechazar Artículo</i> 10. El sistema procesa la petición del usuario. 	
Extensiones (o Flujo Alternativo):	
No existe flujo alternativo	
Poscondiciones (Garantías de éxito):	
<ul style="list-style-type: none"> ▪ El sistema actualiza el estado del artículo a rechazado. ▪ Se envía un mail al autor con los comentarios ingresados por el administrador. 	

ID Caso de Uso	CU-010
Caso de Uso:	Administrador asigna revisores a un artículo

Descripción:	El administrador requiere asignar un revisor a un artículo para su evaluación.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación de artículos (CU-017) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario selecciona el evento que desea revisar. 4. El sistema muestra el menú de opciones del evento. 5. El usuario selecciona la opción de <i>Trabajos Recibidos</i> del menú de opciones del evento. 6. El sistema mostrará un listado de los artículos asignados y sus respectivos estados o acciones por realizar. 7. El usuario elige la opción asignar evaluador. 8. El sistema muestra la información del artículo y una lista de revisores ya asignados en caso de que ya se haya designado alguna revisión. 9. El usuario elige la opción <i>Añadir más evaluadores</i>. 10. El sistema muestra una lista de evaluadores que aún no tienen asignado un artículo en ese evento. 11. El usuario elige la opción <i>Asignar</i> en cada evaluador. <p>Extensiones (o Flujo Alternativo): No hay flujo alternativo</p> <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema envió un correo electrónico a cada revisor indicando que le fue asignado un trabajo para evaluación. ▪ Cambia el estado del artículo a <i>evaluando</i>. 	

ID Caso de Uso	CU-011
Caso de Uso:	Administrador revisa las evaluaciones enviadas por los revisores
Descripción:	El administrador puede revisar las evaluaciones que han enviado los revisores asignados a un artículo.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber asignado un evaluador al artículo (CU-010) ▪ Haber llenado al menos una evaluación de calificación del artículo revisado (CU-037) 	

Escenario principal de éxito (Flujo normal):

1. El administrador elige la opción *Administración* del menú.
2. El sistema presenta el ambiente de administración del portal.
3. El administrador elige la opción *Eventos* del menú administrativo.
4. El sistema presenta una lista de todos los eventos activos.
5. El administrador elige el evento que desea revisar.
6. El sistema presenta el menú de opciones para administrar el evento.
7. El administrador elige la opción *Trabajos recibidos*.
8. El sistema muestra un listado de todos los artículos que se han recibido.
9. El administrador identifica el artículo que desea analizar y luego selecciona la opción *Ver Evaluaciones*.
10. El sistema muestra un listado de todos los revisores que se han asignado para revisión del artículo.
11. El administrador escoge la opción *Ver Resultado* sobre el revisor que haya enviado su evaluación.
12. El sistema muestra las respuestas ingresadas por el revisor cuando evaluó el artículo.

Extensiones (o Flujo Alternativo):

- 12a. El usuario desea revisar los resultados de las demás evaluaciones
1. Volver al punto 10 del flujo normal.

Poscondiciones (Garantías de éxito):

ID Caso de Uso	CU-012
Caso de Uso:	Administrador aprueba un artículo recibido en una convocatoria
Descripción:	El administrador aprueba los artículos que recibe en una convocatoria y que fueron evaluados por revisores asignados.
Actores:	Administrador
Precondiciones:	<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber recibido al menos una evaluación (CU-038)
Escenario principal de éxito (Flujo normal):	<ol style="list-style-type: none"> 1. El administrador elige la opción <i>Administración</i> del menú principal. 2. El sistema presenta el ambiente de administración del portal. 3. El administrador elige la opción <i>Eventos</i> del menú administrativo. 4. El sistema presenta una lista de todos los eventos activos. 5. El administrador elige el evento que desea revisar. 6. El sistema presenta el menú de opciones para administrar el evento. 7. El administrador elige la opción <i>Trabajos recibidos</i>. 8. El sistema muestra un listado de todos los artículos que se han

recibido.

9. El administrador identifica el artículo que desea analizar y luego selecciona la opción *Ver Evaluaciones*.
10. El sistema muestra un listado de todos los revisores que se han asignado para revisión del artículo.
11. El administrador elige la opción *Aceptar Artículo*.
12. El sistema actualiza el estado del artículo, envía un correo electrónico al autor de que su artículo ha sido aceptado en el evento y presenta la página con la lista de todos los artículos y el estado actualizado.

Extensiones (o Flujo Alternativo):

- 10a.** La fecha máxima de aceptación ha finalizado
1. El sistema muestra un mensaje de que ya no es posible aprobar o rechazar un artículo.
 2. El administrador puede modificar las fechas de la convocatoria CU-013
 3. Volver al punto 3 del flujo normal

Poscondiciones (Garantías de éxito):

- Los datos del artículo son almacenados en el sistema.
- El sistema envía un correo al suscriptor confirmando que su artículo fue aceptado.

ID Caso de Uso	CU-013
Caso de Uso:	Administrador modifica la convocatoria para el evento
Descripción:	El administrador modifica los datos de la convocatoria creada para un evento.
Actores:	Administrador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Que se haya creado la convocatoria CU-005 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento al que le modificará la convocatoria 4. El usuario elige la opción convocatoria del menú de opciones que aparece 5. El sistema presenta un formulario con los siguientes campos que corresponden a la información de la convocatoria: descripción de la convocatoria, fecha de recepción del artículo, fecha de evaluación del artículo, fecha de notificación de aceptación del artículo, y fecha 	

del envío de la versión final en el caso de ser aceptado el artículo, el archivo que contiene el formato de presentación de los artículos.

6. El usuario modifica y llena los campos que cree conveniente
7. El usuario escoge la opción guardar modificación
8. El sistema valida la información ingresada por el usuario
9. El sistema modifica la convocatoria del evento y presenta un aviso indicando que ha sido modificada correctamente
10. El suscriptor escoge la opción "Aceptar"

Extensiones (o Flujo Alternativo):

- 9a. El usuario ingresa datos erróneos o incompletos al sistema
 1. Las fechas que se ingresan en la convocatoria no deben superponerse.
 2. El sistema presenta un aviso indicando que existen datos por corregir o llenar.
 3. Volver al punto 6 del flujo normal

Poscondiciones (Garantías de éxito):

- La convocatoria de un evento específico es modificada en el sistema

ID Caso de Uso	CU-014
Caso de Uso:	Administrador modifica un tema de la convocatoria
Descripción:	El administrador modifica un tema que será tratado en la invitación a presentación de artículos o convocatoria de un evento específico.
Actores:	Administrador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la convocatoria (CU-005) ▪ Haber creado el tema a modificar (CU-006) 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento al cual le modificará un tema de la convocatoria 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción convocatoria del menú de opciones que aparece 6. El sistema presenta los datos de la convocatoria y un listado de los temas 7. El usuario elige el tema que desea modificar 8. El sistema muestra un formulario con el nombre del tema 9. El usuario modifica los campos 	

<p>10. El usuario escoge la opción guardar modificación</p> <p>11. El sistema valida la información ingresada por el usuario</p> <p>12. El sistema muestra un aviso indicando que el tema ha sido creado correctamente.</p> <p>13. El usuario escoge la opción "Aceptar"</p> <p>Extensiones (o Flujo Alternativo):</p> <p>13a El usuario ingresa datos erróneos o incompletos al sistema</p> <ol style="list-style-type: none"> 1. El sistema presenta un aviso indicando que existen datos por corregir o llenar. 2. Volver al punto 9 del flujo normal <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El tema de la convocatoria es modificado en el sistema
--

ID Caso de Uso	CU-015
Caso de Uso:	Administrador elimina un tema de la convocatoria
Descripción:	El administrador elimina un tema de la convocatoria de un evento específico.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la convocatoria (CU-005) ▪ Haber creado el tema a eliminar (CU-006) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento al cual se le eliminará un tema de la convocatoria 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción convocatoria del menú de opciones que aparece 6. El sistema presenta los datos de la convocatoria y un listado de los temas 7. El usuario elige el tema que desea eliminar 8. El sistema muestra un formulario con el nombre del tema 9. El usuario escoge la opción eliminar 10. El sistema muestra un aviso ratificando el deseo de eliminar el tema de la convocatoria 11. El usuario acepta 12. El sistema valida el requerimiento 13. El sistema muestra un aviso indicando que el tema ha sido eliminado correctamente. 	

<p>14. El usuario escoge la opción "Aceptar"</p> <p>Extensiones (o Flujo Alternativo):</p> <p>6a. Si solo existe un tema en la convocatoria no es posible eliminarlo.</p> <p>9a. El usuario cancela la eliminación del tema de la convocatoria.</p> <ol style="list-style-type: none"> 1. El usuario elige cancelar 2. Se anula el proceso de eliminación del tema <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El tema de la convocatoria es eliminado del sistema ▪ El estado del tema cambia a finalizado.
--

ID Caso de Uso	CU-016
Caso de Uso:	Administrador revisa la evaluación creada
Descripción:	El administrador desea visualizar la evaluación a medida que va ingresando las secciones, preguntas o alternativas.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación (CU-017) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El administrador elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El sistema presenta todas las opciones del menú de ese evento 4. El administrador elige la opción <i>Evaluación de Trabajos</i> o <i>Ver Evaluación de Trabajos</i> en el caso que ya se haya asignado un artículo a revisión. 5. Si la opción es Evaluación de Trabajos: <ol style="list-style-type: none"> a. El administrador dentro de la opción debe seleccionar la opción Ver Evaluación b. El sistema muestra las preguntas de la evaluación. 6. Si la opción es Ver Evaluación de Trabajos: <ol style="list-style-type: none"> a. El administrador escoge la opción b. El sistema muestra las preguntas de la evaluación <p>Extensiones (o Flujo Alternativo):</p> <p>5a. Si el administrador está ingresando secciones:</p> <ol style="list-style-type: none"> 1. El administrador puede elegir la opción Ver Evaluación directamente en esta página. 2. Volver al punto 6. <p>5a. Si el administrador está ingresando preguntas:</p> <ol style="list-style-type: none"> 1. El administrador puede elegir la opción Ver Evaluación directamente en esta página. 	

<p>2. Volver al punto 6.</p> <p>5a. Si el administrador está ingresando alternativas:</p> <ol style="list-style-type: none"> 1. El administrador puede elegir la opción Ver Evaluación directamente en esta página. 2. Volver al punto 6. <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema muestra la evaluación que se está creando.
--

ID Caso de Uso	CU-017
Caso de Uso:	Administrador crea una evaluación
Descripción:	El administrador requiere crear una evaluación para revisión de artículos en un evento.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la convocatoria (CU-005) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento en el que se creará la evaluación. 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción <i>Evaluación de Trabajos</i> del menú de opciones que aparece 6. El sistema presenta los campos: mensaje y descripción. 7. El usuario ingresa la información requerida. 8. El usuario escoge la opción para guardar una evaluación. 9. El sistema verifica la información registrada. 10. El sistema crea la nueva evaluación y muestra un mensaje que la evaluación se creó con éxito. <p>Extensiones (o Flujo Alternativo):</p> <p>5a. La opción de crear evaluación no se muestra en el menú.</p> <ol style="list-style-type: none"> 1. Referirse al CU-035. 2. Volver al punto 5 del flujo normal <p>9a. Si no se ingresa la información exigida se muestra un mensaje de error</p> <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema crea la nueva evaluación al evento asociado. 	

ID Caso de Uso	CU-018
Caso de Uso:	Administrador elimina una evaluación
Descripción:	El administrador requiere eliminar una evaluación para revisión de artículos en un evento.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación (CU-017) ▪ No debe haber ni un artículo asignado para evaluación <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento en el que se creará la evaluación. 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción <i>Evaluación de Trabajos</i> del menú de opciones que aparece 6. El usuario elige la opción eliminar evaluación. 7. El sistema muestra un mensaje de confirmación si desea eliminar la evaluación. 8. El usuario acepta la eliminación 9. El sistema elimina la evaluación y muestra un mensaje que la evaluación se eliminó con éxito. <p>Extensiones (o Flujo Alternativo):</p> <p>Poscondiciones (Garantías de éxito):</p>	

ID Caso de Uso	CU-019
Caso de Uso:	Administrador modifica una evaluación
Descripción:	El administrador requiere modificar una evaluación para revisión de artículos en un evento.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación (CU-017) ▪ No debe haber ni un artículo asignado para evaluación. <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 10. El usuario elige la opción Eventos del menú administrativo. 11. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 	

12. El usuario elige el evento en el que se creará la evaluación.
13. El sistema presenta todas las opciones del menú de ese evento
14. El usuario elige la opción *Evaluación de Trabajos* del menú de opciones que aparece
15. El sistema presenta los campos: mensaje y descripción con la información que había sido ingresada.
16. El usuario modifica la información requerida.
17. El usuario escoge la opción para guardar.
18. El sistema verifica la información registrada.
19. El sistema modifica la evaluación y muestra un mensaje confirmando que la evaluación fue modificada sin problemas.
20. El usuario escoge aceptar.

Extensiones (o Flujo Alternativo):

- 7a. El usuario ingresó datos incorrectos.
 1. El sistema muestra un mensaje de error
 2. Volver al punto 6 del flujo normal

Poscondiciones (Garantías de éxito):

- El sistema modifica la evaluación al evento asociado.

ID Caso de Uso	CU-020
Caso de Uso:	Administrador crea una sección en una evaluación
Descripción:	El administrador requiere crear una sección en una evaluación.
Actores:	Administrador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación (CU-017) 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento en el que se creará la evaluación. 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción Evaluación de Artículos del menú de opciones que aparece 6. El sistema presenta la información de la evaluación. 7. El usuario escoge la opción <i>Secciones</i>. 8. El sistema presenta los campos que se requieren ingresar de las secciones. 9. El usuario ingresa la información de la sección y escoge la opción <i>Guardar</i>. 10. El sistema modifica la evaluación. 	

Extensiones (o Flujo Alternativo):

- 9a. El usuario ingresó datos incorrectos.
1. El sistema muestra un mensaje de error
 2. Volver al punto 8 del flujo normal

Poscondiciones (Garantías de éxito):

- El sistema crea la sección en la evaluación.

ID Caso de Uso	CU-021
Caso de Uso:	Administrador modifica una sección de una evaluación
Descripción:	El administrador requiere modificar una sección en una evaluación.
Actores:	Administrador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación (CU-017) ▪ Haber creado una sección (CU-020) ▪ No tener asignado un evaluador para esta evaluación. 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción <i>Eventos</i> del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento en el que se creará la evaluación. 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción <i>Evaluación de Artículos</i> del menú de opciones que aparece 6. El sistema presenta los datos de la evaluación. 7. El usuario escoge la opción <i>Secciones</i>. 8. El sistema muestra todas las secciones que se han creado en la evaluación. 9. El usuario pulsa sobre la sección que desea modificar. 10. El sistema presenta los campos con la información ingresada de la sección. 11. El usuario modifica los datos de la sección y escoge la opción para guardar. 12. El sistema verifica la información registrada. 13. El sistema modifica la sección y muestra un mensaje confirmando que la sección fue modificada con sin problemas. 14. El usuario escoge aceptar. 	
Extensiones (o Flujo Alternativo):	
<p>11a. El usuario ingresó datos incorrectos.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error 	

2. Volver al punto del flujo normal
Poscondiciones (Garantías de éxito):
<ul style="list-style-type: none"> ▪ El sistema modifica la sección de la evaluación.

ID Caso de Uso	CU-022
Caso de Uso:	Administrador crea una pregunta para una sección
Descripción:	El administrador requiere crear una pregunta en una sección.
Actores:	Administrador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación (CU-017) ▪ Haber creado una sección (CU-020) 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento en el que se creará la evaluación. 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción Evaluación de Artículos del menú de opciones que aparece 6. El sistema presenta la información de la evaluación. 7. El usuario escoge la opción <i>Secciones</i>. 8. El sistema presenta todas las secciones que se han ingresado. 9. El usuario escoge la opción <i>Preguntas</i> que se encuentra junto a cada sección. 10. El sistema muestra los campos que se deben llenar para crear la pregunta. 11. El usuario ingresa la información de la pregunta y escoge la opción <i>Guardar</i>. 12. El sistema crea la pregunta en la sección. 	
Extensiones (o Flujo Alternativo):	
11a. El usuario ingresó datos incorrectos. <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error 2. Volver al punto 11 del flujo normal 	
Poscondiciones (Garantías de éxito):	
<ul style="list-style-type: none"> ▪ El sistema crea la pregunta en la sección. 	

ID Caso de Uso	CU-023
Caso de Uso:	Administrador modifica una pregunta de una sección

Descripción:	El administrador requiere modificar una pregunta de una sección.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación (CU-017) ▪ Haber creado una sección (CU-020) ▪ Haber creado una pregunta (CU-022) ▪ No tener asignado un evaluador para esta evaluación <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elige la opción <i>Eventos</i> del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento en el que se creará la evaluación. 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción <i>Evaluación de Artículos</i> del menú de opciones que aparece 6. El sistema presenta los datos de la evaluación. 7. El usuario escoge la opción <i>Secciones</i>. 8. El sistema presenta todas las secciones que se han ingresado. 9. El usuario escoge la opción <i>Preguntas</i> que se encuentra junto a cada sección. 10. El sistema muestra todas las preguntas que se han creado para la sección. 11. El usuario escoge la pulsa sobre la pregunta que desea modificar. 12. El sistema muestra los campos con la información inicial que se había creado de la pregunta. 13. El usuario modifica los datos de la pregunta. 14. El sistema verifica la información registrada. 15. El sistema modifica la pregunta y muestra un mensaje que la pregunta fue modificada con éxito 16. El usuario escoge aceptar. <p>Extensiones (o Flujo Alternativo):</p> <p>13a. El usuario ingresó datos incorrectos.</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error 2. Volver al punto 9 del flujo normal <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema modifica la pregunta de la sección. 	

ID Caso de Uso	CU-024
Caso de Uso:	Administrador elimina una pregunta de una sección

Descripción:	El administrador requiere eliminar una pregunta de una sección.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado la evaluación (CU-017) ▪ Haber creado una sección (CU-020) ▪ Haber creado una pregunta (CU-022) ▪ No tener asignado un evaluador para esta evaluación <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elige la opción <i>Eventos</i> del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento en el que se creará la evaluación. 4. El sistema presenta todas las opciones del menú de ese evento 5. El usuario elige la opción <i>Evaluación de Artículos</i> del menú de opciones que aparece 6. El sistema presenta los datos de la evaluación. 7. El usuario escoge la opción <i>Secciones</i>. 8. El sistema presenta todas las secciones que se han ingresado. 9. El usuario escoge la opción <i>Preguntas</i> que se encuentra junto a cada sección. 10. El sistema muestra todas las preguntas que se han creado para la sección. 11. El usuario escoge la opción <i>Eliminar</i> que se encuentra junto a cada pregunta. 12. El sistema muestra un mensaje de confirmación. 13. El usuario escoge aceptar. 14. El sistema elimina la pregunta y muestra un mensaje que la pregunta fue eliminada con éxito. <p>Extensiones (o Flujo Alternativo):</p> <p>13a. El usuario ingresó datos incorrectos.</p> <ol style="list-style-type: none"> 3. El sistema muestra un mensaje de error 4. Volver al punto 9 del flujo normal <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema modifica la pregunta de la sección. 	

ID Caso de Uso	CU-025
Caso de Uso:	Administrador crea una alternativa a una pregunta
Descripción:	El administrador requiere crear una alternativa a una pregunta de alternativas simple o múltiple.
Actores:	Administrador

Precondiciones:

- Haber iniciado sesión (CU-036)
- Haber escogido la opción administración del menú general
- Haber creado la evaluación (CU-017)
- Haber creado una sección (CU-020)
- Haber creado una pregunta de alternativas simple o múltiples (CU-022)

Escenario principal de éxito (Flujo normal):

1. El usuario elige la opción Eventos del menú administrativo.
2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro.
3. El usuario elige el evento en el que se creará la evaluación.
4. El sistema presenta todas las opciones del menú de ese evento
5. El usuario elige la opción Evaluación de Artículos del menú de opciones que aparece
6. El sistema presenta la información de la evaluación.
7. El usuario escoge la opción *Secciones*.
8. El sistema presenta todas las secciones que se han ingresado.
9. El usuario escoge la opción *Preguntas* que se encuentra junto a cada sección.
10. El sistema muestra todas las preguntas creadas para la sección.
11. El usuario escoge la opción Alternativas que se encuentra junto a cada pregunta
12. El sistema muestra los campos que se deben llenar para crear la alternativa.
13. El usuario ingresa la información de la alternativa y escoge la opción *Guardar*.
14. El sistema crea la alternativa en la pregunta.

Extensiones (o Flujo Alternativo):

- 13a. El usuario ingresó datos incorrectos.
1. El sistema muestra un mensaje de error
 2. Volver al punto 13 del flujo normal

Poscondiciones (Garantías de éxito):

- El sistema crea la alternativa en la pregunta.

ID Caso de Uso	CU-026
Caso de Uso:	Administrador modifica una alternativa a una pregunta
Descripción:	El administrador requiere modificar una alternativa a una pregunta de alternativas simple o múltiple.
Actores:	Administrador
Precondiciones:	<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036)

- Haber escogido la opción administración del menú general
- Haber creado la evaluación (CU-017)
- Haber creado una sección (CU-020)
- Haber creado una pregunta de alternativas simple o múltiples (CU-022)
- Haber creado una alternativa (CU-025)
- No tener asignado un evaluador para esta evaluación

Escenario principal de éxito (Flujo normal):

1. El usuario elige la opción Eventos del menú administrativo.
2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro.
3. El usuario elige el evento en el que se creará la evaluación.
4. El sistema presenta todas las opciones del menú de ese evento
5. El usuario elige la opción Evaluación de Artículos del menú de opciones que aparece
6. El sistema presenta la información de la evaluación.
7. El usuario escoge la opción *Secciones*.
8. El sistema presenta todas las secciones que se han ingresado.
9. El usuario escoge la opción *Preguntas* que se encuentra junto a cada sección.
10. El sistema muestra todas las preguntas creadas para la sección.
11. El usuario escoge la opción Alternativas que se encuentra junto a cada pregunta
12. El sistema muestra las alternativas creadas para la pregunta
13. El usuario pulsa sobre la alternativa que desea modificar
14. El sistema muestra los campos con la información que se ingresó de la alternativa.
15. El usuario modifica la información de la alternativa y escoge la opción *Guardar*.
16. El sistema modifica la alternativa en la pregunta.

Extensiones (o Flujo Alternativo):

- 15a. El usuario ingresó datos incorrectos.
1. El sistema muestra un mensaje de error
 2. Volver al punto 13 del flujo normal

Poscondiciones (Garantías de éxito):

- El sistema modifica la alternativa en la pregunta.

ID Caso de Uso	CU-027
Caso de Uso:	Administrador elimina una alternativa a una pregunta
Descripción:	El administrador requiere eliminar una alternativa a una pregunta de alternativas simple o múltiple.
Actores:	Administrador

Precondiciones:

- Haber iniciado sesión (CU-036)
- Haber escogido la opción administración del menú general
- Haber creado la evaluación (CU-017)
- Haber creado una sección (CU-020)
- Haber creado una pregunta de alternativas simple o múltiples (CU-022)
- Haber creado una alternativa (CU-025)
- No tener asignado un evaluador para esta evaluación

Escenario principal de éxito (Flujo normal):

1. El usuario elige la opción Eventos del menú administrativo.
2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro.
3. El usuario elige el evento en el que se creará la evaluación.
4. El sistema presenta todas las opciones del menú de ese evento
5. El usuario elige la opción *Evaluación de Trabajos* del menú de opciones que aparece
6. El sistema presenta la información de la evaluación.
7. El usuario escoge la opción *Secciones*.
8. El sistema presenta todas las secciones que se han ingresado.
9. El usuario escoge la opción *Preguntas* que se encuentra junto a cada sección.
10. El sistema muestra todas las preguntas creadas para la sección.
11. El usuario escoge la opción Alternativas que se encuentra junto a cada pregunta
12. El sistema muestra las alternativas creadas para la pregunta
13. El usuario escoge la opción *Eliminar* que se encuentra junto a cada alternativa.
14. El sistema muestra un mensaje de confirmación para eliminar la alternativa.
15. El usuario escoge aceptar.
16. El sistema elimina la alternativa en la pregunta y muestra un mensaje confirmando la eliminación de la alternativa.

Extensiones (o Flujo Alternativo):**Poscondiciones (Garantías de éxito):**

- El sistema elimina la alternativa en la pregunta.

ID Caso de Uso	CU-028
Caso de Uso:	Administrador asigna el rol de evaluador a un suscriptor
Descripción:	El administrador requiere asignar el rol de evaluador a un suscriptor del portal.
Actores:	Administrador

Precondiciones:

- Haber iniciado sesión (CU-036)

Escenario principal de éxito (Flujo normal):

1. El usuario elige la opción *Administración* del menú principal.
2. El sistema muestra el ambiente de administración.
3. El usuario elige la opción *Usuarios* del menú administrativo.
4. El sistema muestra las diferentes opciones para administrar a los usuarios del portal.
5. El usuario escoge la opción *Asignar rol administrativo a Usuarios*.
6. El sistema presenta las opciones para buscar al usuario.
7. El administrador ingresa la información del usuario y escoge una de las opciones de búsqueda.
8. El sistema muestra la información de la búsqueda.
9. El usuario selecciona el suscriptor al que desea asignar el rol de revisor.
10. El sistema muestra la opción para seleccionar un rol administrativo.
11. El usuario escoge la opción *Revisor* del listbox y pulsa sobre el botón *Aceptar*.
12. El sistema muestra un mensaje para confirmar si desea asignar el rol de *Revisor*.
13. El usuario escoge *Aceptar*.
14. El sistema muestra un mensaje confirmando que el rol fue asignado.

Extensiones (o Flujo Alternativo):

- 13a El sistema muestra un mensaje de que el suscriptor ya tiene el rol de revisor
1. El usuario escoge *Aceptar*.
 2. Volver al punto 7 del flujo normal.

Poscondiciones (Garantías de éxito):

- El sistema asigna el rol de revisor al suscriptor.
- El sistema envía un correo electrónico al suscriptor indicando que el rol de revisor le fue asignado.

ID Caso de Uso	CU-029
Caso de Uso:	Administrador quita el rol de evaluador a un suscriptor
Descripción:	El administrador requiere quitar el rol de evaluador a un suscriptor del portal.
Actores:	Administrador
Precondiciones:	<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber asignado el rol de revisor a un suscriptor (CU-028)
Escenario principal de éxito (Flujo normal):	<ol style="list-style-type: none"> 1. El usuario elige la opción <i>Administración</i> del menú principal.

2. El sistema muestra el ambiente de administración.
3. El usuario elige la opción *Usuarios* del menú administrativo.
4. El sistema muestra las diferentes opciones para administrar a los usuarios del portal.
5. El usuario escoge la opción *Quitar rol administrativo a Usuarios*.
6. El sistema presenta las opciones para buscar al usuario.
7. El administrador ingresa la información del usuario y escoge una de las opciones de búsqueda.
8. El sistema muestra la información de la búsqueda.
9. El usuario selecciona el suscriptor al que desea quitar el rol de revisor.
10. El sistema muestra los tipos de roles que tiene asignado el suscriptor.
11. El usuario escoge la opción de *Quitar rol de Revisor*.
12. El sistema muestra un mensaje para confirmar si desea quitar el rol de *Revisor*.
13. El usuario escoge *Aceptar*.
14. El sistema muestra un mensaje confirmando que se quitó el rol de revisor.

Extensiones (o Flujo Alternativo):

No existen flujos alternativos en este caso.

Poscondiciones (Garantías de éxito):

- El sistema quita el rol de revisor al suscriptor.
- El sistema envía un correo electrónico al suscriptor indicando que el rol de revisor le fue quitado.

ID Caso de Uso	CU-030
Caso de Uso:	Evaluador revisa los artículos asignados para evaluar
Descripción:	El evaluador requiere revisar aquellos artículos que le han sido asignados para revisión.
Actores:	Evaluador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber asignado el rol de revisor a un suscriptor (CU-028) ▪ Haber asignado un artículo al revisor para evaluación (CU-010) 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción <i>Evaluar Trabajos</i> del menú principal. 2. El sistema muestra todos los artículos clasificados por eventos que le han sido asignados para revisión. 	
Extensiones (o Flujo Alternativo):	
No existen flujos alternativos en este caso.	
Poscondiciones (Garantías de éxito):	

- No existen poscondiciones en este caso.

ID Caso de Uso	CU-031
Caso de Uso:	Administrador elimina una convocatoria de artículos a un evento
Descripción:	El administrador requiere eliminar una convocatoria de artículos.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber creado una convocatoria (CU-005) ▪ No haber recibido un trabajo <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario elige la opción <i>Administración</i> del menú principal. 2. El sistema muestra el ambiente de administración del portal. 3. El usuario escoge la opción <i>Eventos</i> del menú administrativo. 4. El sistema muestra una lista de los eventos activos. 5. El usuario escoge el evento en el que desea eliminar la convocatoria. 6. El sistema muestra las opciones de administración del evento. 7. El usuario escoge la opción <i>Convocatoria</i>. 8. El sistema muestra la información de la convocatoria. 9. El usuario escoge la opción <i>Eliminar</i>. 10. El sistema muestra un mensaje para confirmar que se desea eliminar la convocatoria. 11. El usuario escoge <i>Aceptar</i>. 12. El sistema muestra un mensaje confirmando que se eliminó la convocatoria. 13. El usuario escoge <i>Aceptar</i>. <p>Extensiones (o Flujo Alternativo): No existen flujos alternativos en este caso.</p> <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema elimina la convocatoria. 	

ID Caso de Uso	CU-032
Caso de Uso:	Administrador elimina una sección de una evaluación
Descripción:	El administrador requiere eliminar una sección de una evaluación.
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) 	

- Haber escogido la opción administración del menú general
- Haber creado la evaluación (CU-017)
- Haber creado una sección (CU-020)
- No tener asignado un evaluador para esta evaluación

Escenario principal de éxito (Flujo normal):

1. El usuario elige la opción Eventos del menú administrativo.
2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro.
3. El usuario elige el evento en el que se creará la evaluación.
4. El sistema presenta todas las opciones del menú de ese evento
5. El usuario elige la opción Evaluación de Artículos del menú de opciones que aparece
6. El sistema presenta la información de la evaluación.
7. El usuario escoge la opción *Secciones*.
8. El sistema presenta todas las secciones que se han ingresado.
9. El usuario escoge la opción *Eliminar* que se encuentra junto a cada sección.
10. El sistema muestra un mensaje de confirmación para eliminar la sección.
11. El usuario escoge aceptar.
12. El sistema elimina la sección en la evaluación y muestra un mensaje confirmando la eliminación de la alternativa.

Extensiones (o Flujo Alternativo):

Poscondiciones (Garantías de éxito):

- El sistema elimina la sección de la evaluación.

ID Caso de Uso	CU-033
Caso de Uso:	Evaluador llena una evaluación para un artículo
Descripción:	El evaluador requiere llenar una evaluación para un artículo.
Actores:	Evaluador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber creado la evaluación (CU-017) ▪ Haber creado una sección (CU-020) ▪ Haber creado una pregunta de alternativas simple o múltiples (CU-022) ▪ Haber asignado al menos un artículo al evaluador (CU-010) 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción <i>Evaluar Trabajos</i> del menú administrativo. 2. El sistema presenta un listado de todos los artículos por evento que le han sido asignados para revisión. 3. El usuario escoge la opción <i>Por Evaluar</i> que se encuentra junto al 	

artículo que vaya a evaluar.

4. El sistema muestra todas las preguntas de la evaluación que debe llenar.
5. El usuario ingresa las respuestas a cada pregunta y pulsa el botón *Enviar Evaluación*.
6. El sistema valida la información y muestra un mensaje que la evaluación fue ingresada exitosamente.
7. El usuario escoge *Aceptar*.
8. El sistema regresa a la página que muestra la lista de artículos asignados para revisión.

Extensiones (o Flujo Alternativo):

- 3a La fecha máxima de evaluación ha finalizado
1. El sistema muestra un mensaje en el artículo que *Terminó el plazo* para evaluarlo.
 2. El usuario solo puede descargar el artículo.
- 6a El usuario ingresó información incompleta:
1. El sistema muestra un mensaje de error indicando que faltan preguntas por contestar.
 2. El usuario escoge aceptar
 3. Volver al punto 5 del flujo normal

Poscondiciones (Garantías de éxito):

- El sistema guarda la información de las respuestas de la evaluación.
- El sistema actualiza el estado del artículo.
- El sistema envía un correo electrónico al correo del evento indicando que el artículo ya fue evaluado.

ID Caso de Uso	CU-034
Caso de Uso:	Evaluador descarga un artículo asignado para evaluación
Descripción:	El evaluador requiere descargar un artículo asignado para revisión.
Actores:	Evaluador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber asignado al menos un artículo al evaluador (CU-010) 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción <i>Evaluar Trabajos</i> del menú administrativo. 2. El sistema presenta un listado de todos los artículos por evento que le han sido asignados para revisión. 3. El usuario escoge la opción <i>Descargar</i> que se encuentra junto al artículo que ya está evaluado. 	
Extensiones (o Flujo Alternativo):	
No existe flujo alternativo para este caso	

Poscondiciones (Garantías de éxito):

- No existen poscondiciones para este caso.

ID Caso de Uso	CU-035
Caso de Uso:	Administrador asigna las opciones del menú a un evento
Descripción:	El administrador asigna a un evento las opciones que estarán disponibles para administrar o para ser visualizadas al presentar el evento.
Actores:	Administrador
Precondiciones: <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber creado el evento ▪ Haber escogido la opción administración del menú general Escenario principal de éxito (Flujo normal): <ol style="list-style-type: none"> 1. El usuario elige la opción Eventos del menú administrativo. 2. El sistema presenta un listado de todos los eventos del grupo de investigación que se realizarán en el futuro. 3. El usuario elige el evento al que desea asignarle opciones de menú. 4. El usuario escoge la opción menú de opciones 5. El sistema presenta un listado de opciones para el evento. 6. El usuario escoge de entre este listado las opciones que desea mostrar y administrar para el evento 7. El usuario elige la opción grabar y cerrar Extensiones (o Flujo Alternativo): No existen flujos alternativos en este caso.	
Poscondiciones (Garantías de éxito): <ul style="list-style-type: none"> ▪ Se asigna al evento un grupo de opciones. 	

ID Caso de Uso	CU-036
Caso de Uso:	Iniciar sesión
Descripción:	El usuario requiere iniciar su sesión para acceder a los distintos servicios que el sistema le ofrece.
Actores:	Suscriptor, administrador, conferencista invitado
Precondiciones: <ul style="list-style-type: none"> ▪ Haberse suscrito en el portal. (Caso de uso CU-001) Escenario principal de éxito (Flujo normal): <ol style="list-style-type: none"> 1. El suscriptor, administrador o conferencista invitado ingresa su nombre de usuario (correo electrónico) y su contraseña 2. El suscriptor, administrador o conferencista invitado elige la opción "Ingresar". 3. El sistema verifica que el usuario existe en la base de datos y que 	

los datos ingresados son correctos

4. El sistema muestra las diferentes opciones a las que tiene acceso el usuario ya sea como suscriptor, administrador o conferencista invitado

Extensiones (o Flujo Alternativo):

- 4a. El suscriptor, administrador o conferencista invitado ingresa datos erróneos o incompletos
 1. El sistema presenta un aviso indicando que los datos ingresados son inválidos
 2. Volver al punto 1 del flujo normal

Poscondiciones (Garantías de éxito):

- Ingresar al portal con acceso a ciertas opciones que dependen de el o los roles que tiene dentro del sistema.

ID Caso de Uso	CU-037
Caso de Uso:	Suscribir usuarios al Portal
Descripción:	Suscribir visitantes al portal pidiendo sus datos
Actores:	Visitante del portal
Precondiciones:	
No hay precondición	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario elige la opción que le permitirá suscribirse. 2. El sistema presenta un formulario con los siguientes campos: número de cédula o pasaporte, correo electrónico, contraseña, nombres, apellidos, género ya sea masculino o femenino, dirección del domicilio, teléfono del domicilio, celular, fecha de nacimiento, país y ciudad a la que pertenece, carrera o título, la empresa para la cual trabaja, el cargo que desempeña dentro de esa empresa, el fax y teléfono de la oficina. 3. El usuario llena los campos que se solicitan 4. El usuario escoge la opción que le permite enviar los datos. 5. El sistema valida la información ingresada por el usuario 6. El sistema registra al usuario como suscriptor del portal y presenta un aviso indicando que ha sido registrado correctamente 7. El suscriptor escoge la opción "Aceptar" 	
Extensiones (o Flujo Alternativo):	
<ol style="list-style-type: none"> 3b. El usuario ingresa los datos solicitados pero la empresa o universidad a la que pertenece el usuario no se encuentra entre las que se muestran en la lista <ol style="list-style-type: none"> 1. Usuario ingresa una nueva empresa. 2. Volver al punto 4 del flujo normal 6a. El usuario ingresa datos erróneos o incompletos al sistema <ol style="list-style-type: none"> 1. El sistema presenta un aviso indicando que existen datos 	

<p>por corregir o llenar.</p> <p>2. Volver al punto 3 del flujo normal</p> <p>6b. El número de cédula o correo electrónico ingresados por el usuario ya están registrados en el sistema.</p> <p>1. El sistema presenta un aviso indicando que ya existe un suscriptor con el mismo número de cédula o con el mismo correo electrónico</p> <p>a. El usuario cambia dichos datos y vuelve al punto 4 del flujo normal</p> <p>b. El usuario no cambia los datos requeridos con lo que se anula el proceso de suscripción.</p> <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ Los datos del usuario son almacenados en el sistema. ▪ El sistema envía un correo al suscriptor confirmando su registro.
--

ID Caso de Uso	CU-038
Caso de Uso:	Administrador envía un correo electrónico a revisores que no han evaluado artículos
Descripción:	El administrador requiere enviar un correo electrónico a revisores que aun no han evaluado artículos asignados
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber asignado al menos un evaluador a un artículo (CU-010) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario escoge la opción <i>Eventos</i> del menú administrativo. 2. El sistema muestra una lista de todos los eventos activos. 3. El usuario escoge el evento. 4. El sistema muestra las opciones de administración del evento. 5. El usuario escoge la opción de <i>Envío de correos</i>. 6. El usuario elige la opción de <i>Recordatorio a Evaluadores</i> y pulsa sobre <i>Enviar mail</i>. 7. El sistema muestra un mensaje de que el correo fue enviado a todos los revisores que tienen pendiente una revisión. 8. El usuario escoge <i>Aceptar</i>. <p>Extensiones (o Flujo Alternativo):</p> <p>No existe flujo alternativo en este caso</p> <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema envía automáticamente un correo a cada evaluador que no ha revisado artículos indicándole los artículos que tienen pendiente por evaluar. 	

ID Caso de Uso	CU-039
Caso de Uso:	Suscriptor envía la corrección de su artículo aprobado
Descripción:	El suscriptor requiere enviar la corrección del artículo que fue aceptado como ponencia en el evento
Actores:	Suscriptor
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario escoge la opción <i>Eventos por realizar</i> del menú principal. 2. El sistema muestra una lista de todos los eventos que se están organizando. 3. El usuario escoge el evento. 4. El sistema muestra la información general del evento. 5. El usuario escoge la opción <i>Trabajos Enviados</i> del menú del evento. 6. El sistema muestra una lista de todos los artículos que el suscriptor ha enviado en ese evento. 7. El usuario pulsa sobre el título del artículo que ha sido aceptado. 8. El sistema muestra los campos del artículo con la información que fue inicialmente enviada. 9. El usuario actualiza la información enviando la corrección del artículo y pulsa sobre la opción <i>Guardar</i>. 10. El sistema verifica la información. 11. El sistema muestra un mensaje de que la información del artículo fue actualizada. <p>Extensiones (o Flujo Alternativo):</p> <p>6a Si la fecha máxima para enviar la corrección ha finalizado:</p> <ol style="list-style-type: none"> 1. El sistema no permite modificar el artículo. <p>10a El usuario ingresó información incompleta:</p> <ol style="list-style-type: none"> 1. El sistema muestra un mensaje de error 2. Volver al punto 9 del flujo normal. <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema actualiza la información del artículo. 	

ID Caso de Uso	CU-040
Caso de Uso:	Administrador verifica el estado de todos los artículos
Descripción:	El administrador requiere verificar el estado de todos los artículos
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general 	

<p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario escoge la opción <i>Eventos</i> del menú administrativo. 2. El sistema muestra una lista de todos los eventos que se están organizando. 3. El usuario escoge el evento. 4. El sistema muestra la información general del evento. 5. El usuario escoge la opción <i>Estados Artículos</i> del menú del evento. 6. El sistema muestra una lista de todos los artículos recibidos indicando todos los estados que puede tener un artículo y un resumen general de los artículos recibidos. <p>Extensiones (o Flujo Alternativo): No existe flujo alternativo para este caso</p> <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ No existe poscondiciones en este caso

ID Caso de Uso	CU-041
Caso de Uso:	Administrador visualiza el correo electrónico de difusión de la convocatoria mientras la va creando
Descripción:	El administrador requiere revisar la información del correo para difundir la convocatoria cuando la está creando
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber escogido la opción administración del menú general ▪ Haber creado una convocatoria (CU-005) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario escoge la opción <i>Eventos</i> del menú administrativo. 2. El sistema muestra una lista de todos los eventos que se están organizando. 3. El usuario escoge el evento. 4. El sistema muestra la información general del evento. 5. El usuario escoge la opción <i>Convocatoria</i> del menú del evento. 6. El sistema muestra la información de la convocatoria. 7. El usuario pulsa sobre el mensaje que indica revisar la información del correo para difundir la convocatoria. 8. El sistema muestra la información del correo <p>Extensiones (o Flujo Alternativo): No existe flujo alternativo para este caso</p> <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ No existe poscondiciones en este caso 	
ID Caso de Uso	CU-042
Caso de Uso:	Evaluador elimina una evaluación ingresada

Descripción:	El evaluador requiere eliminar una evaluación ingresada
Actores:	Evaluador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber evaluado un artículo (CU-033) <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario escoge la opción <i>Evaluar trabajos</i> del menú principal. 2. El sistema muestra todos los artículos por evento que le han sido asignado. 3. El evaluador escoge la opción <i>Eliminar</i> que se encuentra junto a cada artículo. 4. El sistema muestra un mensaje de confirmación para eliminar la evaluación. 5. El usuario escoge Aceptar. 6. El sistema actualiza la página con el cambio ocurrido. <p>Extensiones (o Flujo Alternativo):</p> <p>2a Si la fecha máxima para evaluar artículos ha finalizado:</p> <ol style="list-style-type: none"> 1. El sistema no permite al usuario hacer una eliminación de su evaluación. <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ El sistema elimina la evaluación 	

ID Caso de Uso	CU-043
Caso de Uso:	Administrador rechaza un artículo que tiene evaluaciones recibidas
Descripción:	El administrador requiere rechazar artículos que tienen evaluaciones recibidas
Actores:	Administrador
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber asignado al menos un evaluador al artículo (CU-010) ▪ Haber recibido al menos una evaluación <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario escoge la opción <i>Eventos</i> del menú administrativo. 2. El sistema muestra una lista de todos los eventos que se están organizando. 3. El usuario escoge el evento. 4. El sistema muestra la información general del evento. 5. El usuario escoge la opción de <i>Trabajos recibidos</i> 6. El sistema muestra un listado de todos los artículos recibidos. 7. El usuario escoge la opción <i>Ver Evaluaciones</i> 8. El sistema muestra un listado con las evaluaciones recibidas por 	

evaluador.

9. El usuario escoge la opción de rechazar artículo.

Extensiones (o Flujo Alternativo):

- 8ª Si la fecha máxima para aceptar artículos ha finalizado:

1. El sistema no permite rechazar el artículo.

Poscondiciones (Garantías de éxito):

- El sistema actualiza el estado del artículo a *rechazado*.
- El sistema envía un correo al autor del artículo indicando que el artículo fue rechazado.

ID Caso de Uso	CU-044
Caso de Uso:	Administrador quita un evaluador asignado a un artículo
Descripción:	El administrador requiere quitar a un evaluador asignado a un artículo
Actores:	Administrador
Precondiciones:	
<ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber asignado al menos un evaluador al artículo (CU-010) ▪ No haber recibido alguna evaluación de uno de los evaluadores 	
Escenario principal de éxito (Flujo normal):	
<ol style="list-style-type: none"> 1. El usuario escoge la opción <i>Eventos</i> del menú administrativo. 2. El sistema muestra una lista de todos los eventos que se están organizando. 3. El usuario escoge el evento. 4. El sistema muestra la información general del evento. 5. El usuario escoge la opción de <i>Trabajos recibidos</i> 6. El sistema muestra un listado de todos los artículos recibidos. 7. El usuario escoge la opción <i>Asignar Evaluador</i> 8. El sistema muestra una lista de todos los evaluadores asignados. 9. El usuario escoge la opción "X" que está en cada evaluador 10. El sistema muestra un mensaje de que el evaluador ha sido quitado para evaluación de ese artículo. 	
Extensiones (o Flujo Alternativo):	
Poscondiciones (Garantías de éxito):	
<ul style="list-style-type: none"> ▪ El sistema envía un correo al evaluador indicando que ha sido quitado para revisión del artículo. 	

ID Caso de Uso	CU-045
Caso de Uso:	Suscriptor puede ver las evaluaciones del artículo

	enviado
Descripción:	El suscriptor requiere revisar los comentarios que le han hecho los evaluadores a su artículo
Actores:	Suscriptor
<p>Precondiciones:</p> <ul style="list-style-type: none"> ▪ Haber iniciado sesión (CU-036) ▪ Haber aprobado o rechazado un artículo <p>Escenario principal de éxito (Flujo normal):</p> <ol style="list-style-type: none"> 1. El usuario escoge la opción <i>Eventos por realizar</i> del menú principal. 2. El sistema muestra una lista de todos los eventos que se están organizando. 3. El usuario escoge el evento. 4. El sistema muestra la información general del evento. 5. El usuario escoge la opción <i>Trabajos Enviados</i> del menú del evento. 6. El sistema muestra una lista de todos los artículos que el suscriptor ha enviado en ese evento. 7. El usuario escoge la opción <i>Rechazado o Aceptado</i> 8. El sistema muestra los comentarios que hicieron los evaluadores de ese artículo. <p>Extensiones (o Flujo Alternativo):</p> <p>Poscondiciones (Garantías de éxito):</p> <ul style="list-style-type: none"> ▪ No hay poscondiciones 	

APENDICE B

TABLAS DE EVENTOS DE OBJETOS

MERODE representa a la tabla de eventos de objetos como una matriz cuyas filas representan los diferentes tipos de evento y las columnas los diferentes tipos de objetos. Cada tabla especifica que eventos (filas) afectan a cada uno de los objetos (columna) del modelo.

A continuación se describe por objetos los tipos de eventos que los afectan:

	Evaluacion
cr_Evaluacion	O/C
end_Evaluacion	O/E
cr_Seccion	A/M
end_Seccion	A/M
cr_Pregunta	A/M
end_Pregunta	A/M
cr_Alternativa	A/M
end_Alternativa	A/M
cr_Evaluacion_Tema_Articulo	A/M
end_Evaluacion_Tema_Articulo	A/M
cr_Respuestas	A/M

end_Respuestas	A/M
modificar_evaluacion	O/M
modificar_seccion	A/M
modificar_pregunta	A/M
modificar_alternativa	A/M

OET – Objeto Evaluacion

	Seccion
cr_Seccion	O/C
end_Seccion	O/E
cr_Pregunta	A/M
end_Pregunta	A/M
cr_Alternativa	A/M
end_Alternativa	A/M
modificar_seccion	O/M
modificar_pregunta	A/M
modificar_alternativa	A/M

OET – Objeto Seccion

	Pregunta
cr_Pregunta	O/C
end_Pregunta	O/E
cr_Alternativa	A/M
end_Alternativa	A/M
modificar_pregunta	O/M
modificar_alternativa	A/M

OET – Objeto Pregunta

	Alternativa
cr_Alternativa	O/C
end_Alternativa	O/E
modificar_alternativa	O/M

OET – Objeto Alternativa

	Tipo_Pregunta
cr_Pregunta	A/M
end_Pregunta	A/M
cr_Alternativa	A/M
end_Alternativa	A/M
cr_Tipo_Pregunta	O/C

end_Tipo_Pregunta	O/E
cr_Administrador_Grupo_Investigacion	O/C
modificar_pregunta	A/M
modificar_alternativa	A/M

OET – Objeto Tipo_Pregunta

	Convocatoria
cr_Evaluacion	A/M
end_Evaluacion	A/M
cr_Seccion	A/M
end_Seccion	A/M
cr_Pregunta	A/M
end_Pregunta	A/M
cr_Alternativa	A/M
end_Alternativa	A/M
cr_Convocatoria	O/C
end_Convocatoria	O/E
cr_Tema	A/M
end_Tema	A/M
cr_Tema_Articulo	A/M
end_Tema_Articulo	A/M
cr_Evaluacion_Tema_Articulo	A/M, A/M
end_Evaluacion_Tema_Articulo	A/M, A/M
cr_Respuestas	A/M, A/M
end_Respuestas	A/M, A/M
modificar_evaluacion	A/M
modificar_seccion	A/M
modificar_pregunta	A/M
modificar_alternativa	A/M
modificar_convocatoria	O/M
modificar_tema	A/M

OET – Objeto Convocatoria

	Tema
cr_Tema	O/C
end_Tema	O/E
cr_Tema_Articulo	A/M
end_Tema_Articulo	A/M
cr_Evaluacion_Tema_Articulo	A/M
end_Evaluacion_Tema_Articulo	A/M

cr_Respuestas	A/M
end_Respuestas	A/M
modificar_tema	O/M

OET – Objeto Tema

	Tema_Articulo
cr_Tema_Articulo	O/C
end_Tema_Articulo	O/E
cr_Evaluacion_Tema_Articulo	A/M
end_Evaluacion_Tema_Articulo	A/M
cr_Respuestas	A/M
end_Respuestas	A/M

OET – Objeto Tema_Articulo

	Evaluacion_Tema_Articulo
cr_Evaluacion_Tema_Articulo	O/C
end_Evaluacion_Tema_Articulo	O/E
cr_Respuestas	A/M
end_Respuestas	A/M

OET – Objeto Evaluacion_Tema_Articulo

	Respuestas
cr_Respuestas	O/C
end_Respuestas	O/E

OET – Objeto Respuestas

	Evento
cr_Evaluacion	A/M
end_Evaluacion	A/M
cr_Seccion	A/M
end_Seccion	A/M
cr_Pregunta	A/M
end_Pregunta	A/M
cr_Alternativa	A/M
end_Alternativa	A/M
cr_Convocatoria	A/M
end_Convocatoria	A/M
cr_Tema	A/M
end_Tema	A/M
cr_Tema_Articulo	A/M

end_Tema_Articulo	A/M
cr_Evaluacion_Tema_Articulo	A/M, A/M
end_Evaluacion_Tema_Articulo	A/M, A/M
cr_Respuestas	A/M, A/M
end_Respuestas	A/M, A/M
cr_Evento	O/C
end_Evento	O/E
cr_Articulo_Evento	A/M
end_Articulo_Evento	A/M
cr_Conferencista_Evento	A/M
end_Conferencista_Evento	A/M
cr_Conferencista_Articulo	A/M
end_Conferencista_Articulo	A/M
modificar_evaluacion	A/M
modificar_seccion	A/M
modificar_pregunta	A/M
modificar_alternativa	A/M
modificar_convocatoria	A/M
modificar_tema	A/M

OET – Objeto Evento

	Articulo_Evento
cr_Articulo_Evento	O/C
end_Articulo_Evento	O/E

OET – Objeto Articulo_Evento

	Articulo
cr_Tema_Articulo	A/M
end_Tema_Articulo	A/M
cr_Evaluacion_Tema_Articulo	A/M
end_Evaluacion_Tema_Articulo	A/M
cr_Respuestas	A/M
end_Respuestas	A/M
cr_Articulo_Evento	A/M
end_Articulo_Evento	A/M
cr_Articulo	O/C

end_Articulo	O/E
cr_Autor_Articulo	A/M
end_Autor_Articulo	A/M
cr_Conferencista_Articulo	A/M
end_Conferencista_Articulo	A/M
cr_Articulo_Grupo_Investigacion	A/M
end_Articulo_Grupo_Investigacion	A/M
aprobar_articulo	O/M
modificar_articulo	O/M
evaluar_articulo	O/M
corregir_articulo	O/M
rechazar_articulo	O/M

OET – Objeto Articulo

	Autor_Articulo
cr_Autor_Articulo	O/C
end_Autor_Articulo	O/E

OET – Objeto Autor_Articulo

	Evaluador_Articulo
cr_Evaluacion_Tema_Articulo	A/M
end_Evaluacion_Tema_Articulo	A/M
cr_Respuestas	A/M
end_Respuestas	A/M
cr_Evaluador_Articulo	O/C
end_Evaluador_Articulo	O/E

OET – Objeto Evaluador_Articulo

	Conferencista_Evento
cr_Conferencista_Evento	O/C
end_Conferencista_Evento	O/E
cr_Conferencista_Articulo	A/M
end_Conferencista_Articulo	A/M

OET – Objeto Conferencista_Evento

	Conferencista_Articulo
cr_Conferencista_Articulo	O/C
end_Conferencista_Articulo	O/E

OET – Objeto Conferencista_Articulo

	Rol_Usuario
cr_Evaluacion_Tema_Articulo	A/M
end_Evaluacion_Tema_Articulo	A/M
cr_Respuestas	A/M
end_Respuestas	A/M
cr_Autor_Articulo	A/M
end_Autor_Articulo	A/M
cr_Evaluador_Articulo	A/M
end_Evaluador_Articulo	A/M
cr_Conferencista_Evento	A/M
end_Conferencista_Evento	A/M
cr_Conferencista_Articulo	A/M
end_Conferencista_Articulo	A/M
cr_Rol_Usuario	O/C
end_Rol_Usuario	O/E
cr_Administrador_Grupo_Investigacion	A/M
end_Administrador_Grupo_Investigacion	A/M

OET – Objeto Rol_Usuario

	Usuario
cr_Evaluacion_Tema_Articulo	A/M
end_Evaluacion_Tema_Articulo	A/M
cr_Respuestas	A/M
end_Respuestas	A/M
cr_Autor_Articulo	A/M
end_Autor_Articulo	A/M
cr_Evaluador_Articulo	A/M
end_Evaluador_Articulo	A/M
cr_Conferencista_Evento	A/M
end_Conferencista_Evento	A/M
cr_Conferencista_Articulo	A/M
end_Conferencista_Articulo	A/M
cr_Rol_Usuario	A/M
end_Rol_Usuario	A/M
cr_usuario	O/C
end_usuario	O/E
cr_Administrador_Grupo_Investigacion	A/M
end_Administrador_Grupo_Investigacion	A/M

OET – Objeto Usuario

	Rol
cr_Evaluacion_Tema_Articulo	A/M
end_Evaluacion_Tema_Articulo	A/M
cr_Respuestas	A/M
end_Respuestas	A/M
cr_Autor_Articulo	A/M
end_Autor_Articulo	A/M
cr_Evaluador_Articulo	A/M
end_Evaluador_Articulo	A/M
cr_Conferencista_Evento	A/M
end_Conferencista_Evento	A/M
cr_Conferencista_Articulo	A/M
end_Conferencista_Articulo	A/M
cr_Rol_Usuario	A/M
end_Rol_Usuario	A/M
cr_Rol	O/C
end_Rol	O/E
cr_Administrador_Grupo_Investigacion	A/M
end_Administrador_Grupo_Investigacion	A/M

OET – Objeto Rol

	Articulo_Grupo_Investigacion
cr_Articulo_Grupo_Investigacion	O/C
end_Articulo_Grupo_Investigacion	O/E

OET – Objeto Articulo_Grupo_Investigacion

	Grupo_Investigacion
cr_Articulo_Grupo_Investigacion	A/M
end_Articulo_Grupo_Investigacion	A/M
cr_Grupo_Investigacion	O/C
end_Grupo_Investigacion	O/E
cr_Administrador_Grupo_Investigacion	A/M
end_Administrador_Grupo_Investigacion	A/M

OET – Objeto Grupo_Investigacion

	Administrador_Grupo_Investigacion
cr_Administrador_Grupo_Investigacion	O/C
end_Administrador_Grupo_Investigacion	O/E

OET – Objeto Administrador_Grupo_Investigacion

APENDICE C

MODELO LÓGICO DE LA BASE DE DATOS

A continuación se mostrará el modelo lógico de la base de datos separado en 3 partes de tal forma que se puedan apreciar cada una de las tablas que lo conforman:

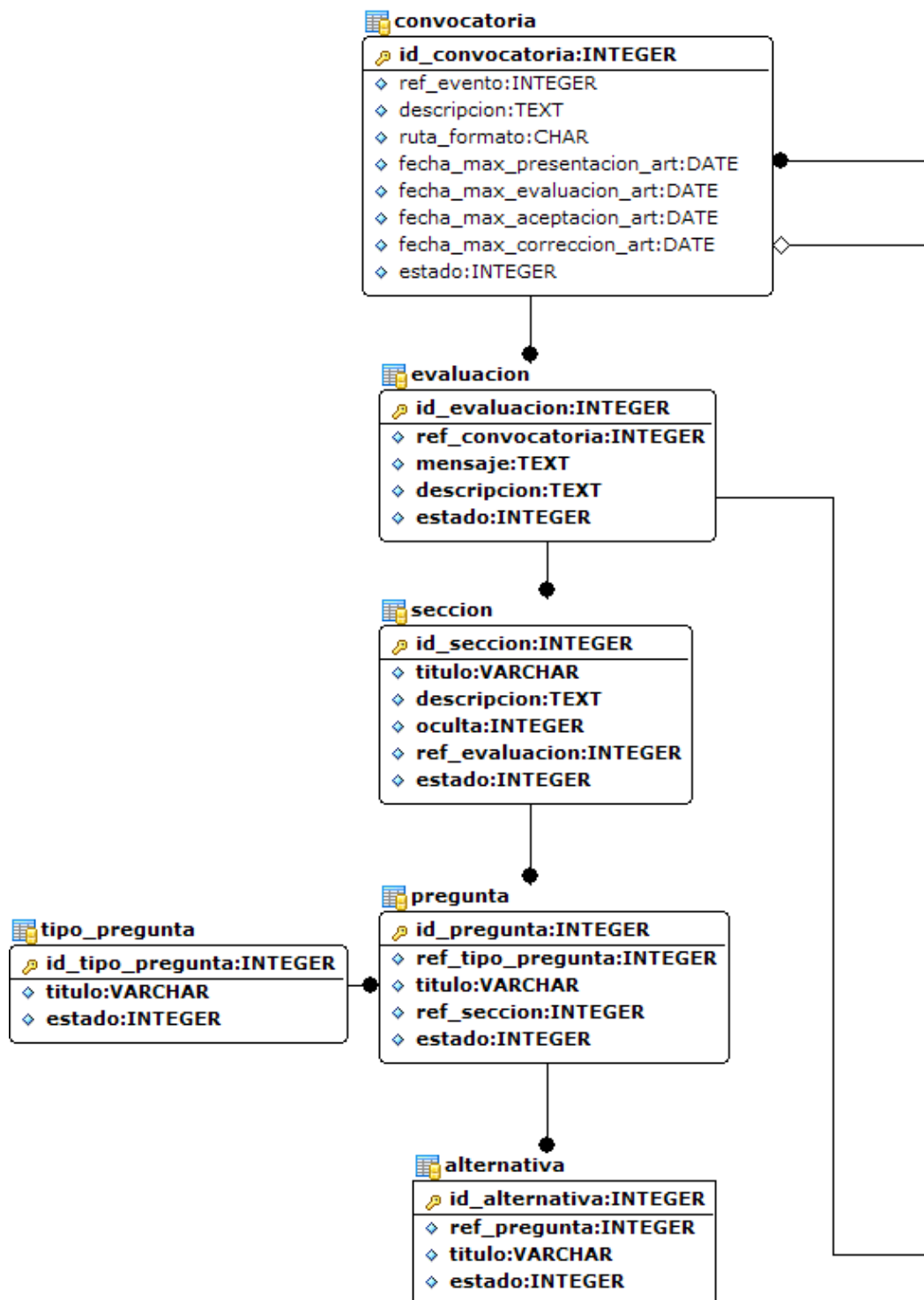


Figura 1. Modelo lógico de la base de datos – Parte 1

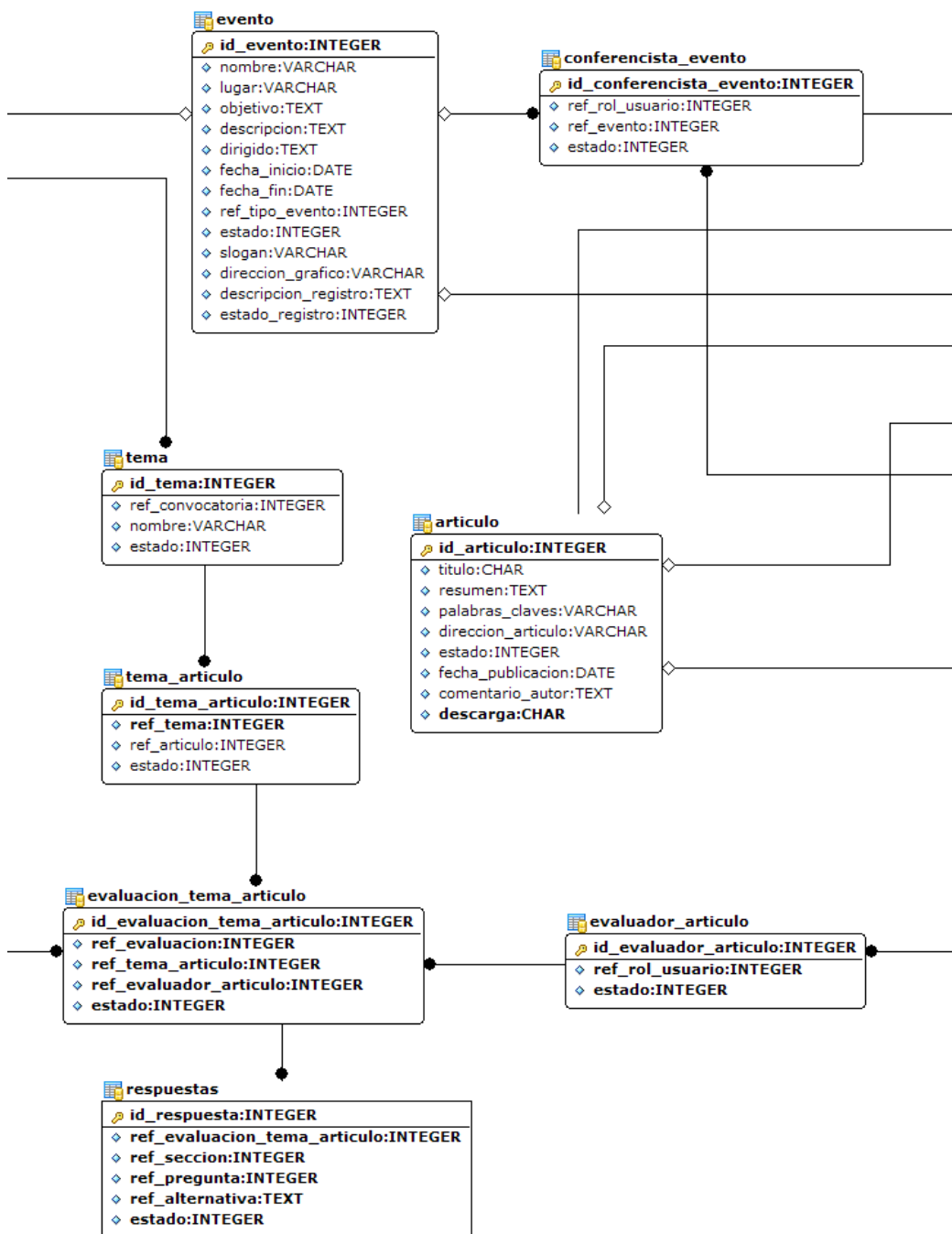


Figura 2. Modelo lógico de la base de datos – Parte 2

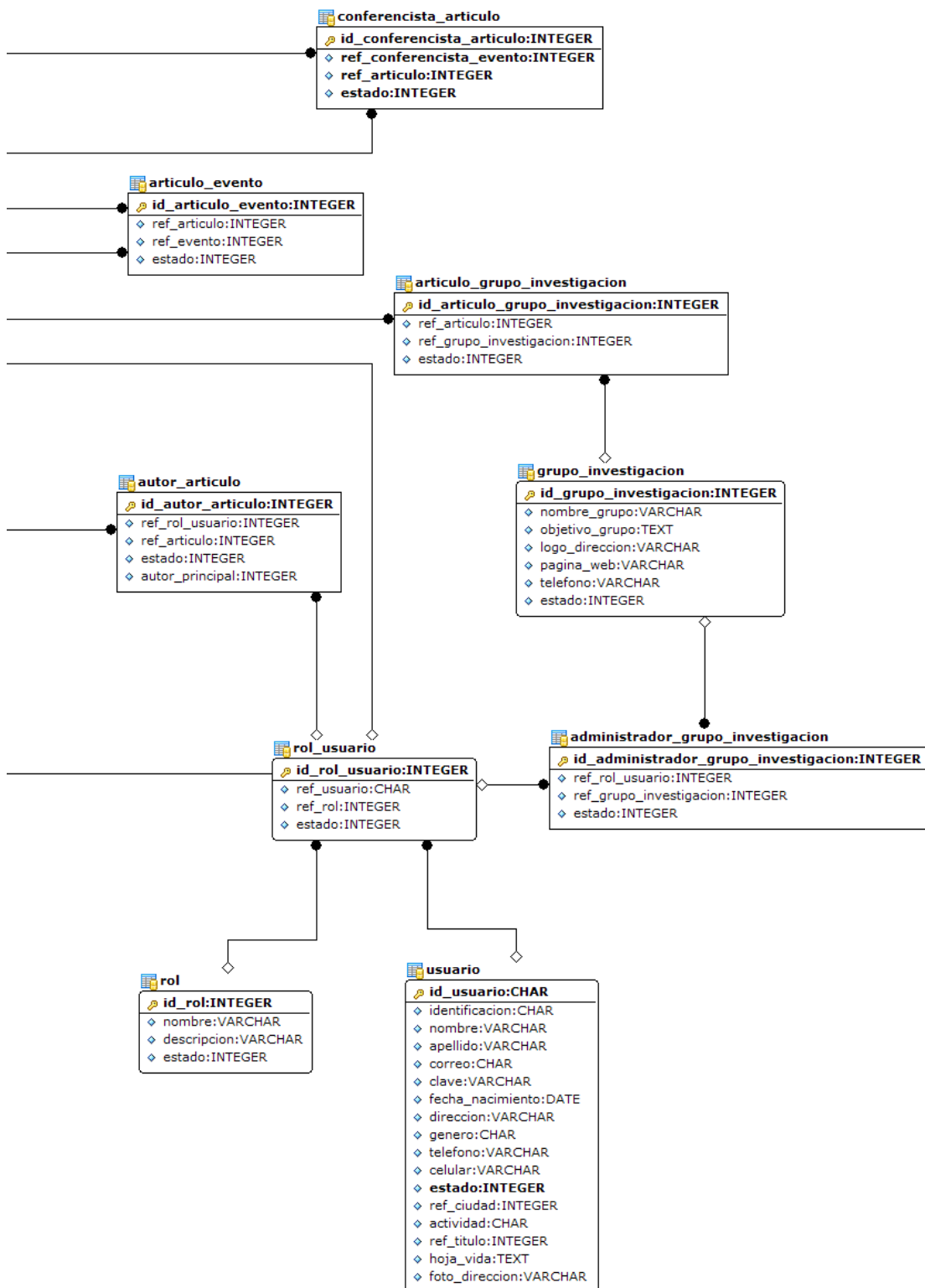


Figura 3. Modelo lógico de la base de datos – Parte 3

APENDICE E

DICCIONARIO DE DATOS

A continuación se detallará el diccionario de datos correspondiente al modelo lógico de la base de datos.

administrador_grupo_investigacion.- Tabla que almacenará los administradores de cada grupo de investigación, y que relaciona al grupo de investigación con el rol de administrador asignado al usuario administrador.

Esta tabla tendrá los siguientes campos:

- **id_administrador_grupo_investigacion.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla administrador_grupo_investigacion.
- **ref_rol_usuario.-** Campo no nulo de tipo INTEGER que almacenará el identificador del rol_usuario asociado.

- **ref_grupo_investigacion.-** Campo no nulo de tipo INTEGER que almacenará el identificador del grupo de investigación asociado.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento (existe o finalizado).

articulo.- Tabla que almacenará los artículos que se expondrán en los eventos y las publicaciones que realiza cada grupo de investigación.

- **id_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador del artículo.
- **titulo.-** Campo no nulo de tipo CHAR con una longitud máxima de 250 caracteres que almacenará el nombre o título del artículo.
- **fecha_publicacion.-** Campo de tipo DATE que almacenará la fecha en que se publicó por primera vez el artículo.
- **resumen.-** Campo no nulo de tipo TEXT que almacenará el resumen correspondiente al artículo, es decir un abstracto de lo que se tratará en el mismo.
- **palabras_claves.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 100 caracteres que almacenará las palabras principales que hacen referencia al artículo. Estas palabras servirán para realizar búsquedas de artículos.

- **direccion_articulo.-** Campo de tipo VARCHAR con una longitud máxima de 255 caracteres que almacenará el nombre del archivo que contendrá el artículo.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el artículo (existe, aprobado o finalizado).
- **comentario_autor:** Campo de tipo TEXT que almacenará los comentarios que haga el autor cuando esté enviando la corrección de su artículo.
- **descargar:** Campo de tipo CHAR que almacenará la condición que habilita o no que un artículo publicado por el grupo de investigación sea descargado o no del portal.

articulo_evento.- Tabla que almacenará los artículos relacionados a un evento. Estos artículos pertenecen a los conferencistas y no están relacionados con los temas de la convocatoria del evento. Esta tabla tendrá los siguientes campos:

- **id_articulo_evento.-** Campo no nulo de tipo INTEGER que almacenará el identificador.
- **ref_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador del artículo asociado.
- **ref_evento.-** Campo no nulo de tipo INTEGER que almacenará el identificador del evento asociado.

- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento (existe o finalizado).

articulo_grupo_investigacion.- Tabla que almacenará las publicaciones de los grupos de investigación. Esta tabla tendrá los siguientes campos:

- **id_articulo_grupo_investigacion.-** Campo no nulo de tipo INTEGER que almacenará el identificador.
- **ref_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador del artículo asociado.
- **ref_grupo_investigacion.-** Campo no nulo de tipo INTEGER que almacenará el identificador del grupo de investigación asociado.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la tabla (existe o finalizado).

autor_articulo.- Tabla que almacenará los autores de un artículo a través de la relación entre el artículo y el rol de autor asignado a un usuario. Esta tabla tendrá los siguientes campos:

- **id_autor_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla autor_articulo.
- **ref_rol_usuario.-** Campo no nulo de tipo INTEGER que almacenará el identificador del rol_usuario asociado.

- **ref_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador del artículo asociado.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el autor_articulo (existe o finalizado).
- **autor_principal.-** Campo de tipo INTEGER que indicará si el autor es el autor principal del artículo (1), o un coautor (0).

conferencista_evento.- Tabla que almacenará los conferencistas que intervendrán en los eventos, a través del rol de Conferencista Invitado asignado al usuario conferencista. Esta tabla tendrá los siguientes campos:

- **id_conferencista_evento.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla conferencista_evento.
- **ref_rol_usuario.-** Campo no nulo de tipo INTEGER que almacenará el identificador del rol_usuario asociado.
- **ref_evento.-** Campo no nulo de tipo INTEGER que almacenará el identificador del evento asociado.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la relación conferencista_evento (existe o finalizado).

convocatoria.- Tabla que almacenará los datos de la convocatoria o llamada para presentación de artículos para los eventos. Esta tabla tendrá los siguientes campos:

- **id_convocatoria.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla convocatoria.
- **ref_evento.-** Campo no nulo de tipo INTEGER que almacenará el identificador del evento al cual pertenece la convocatoria.
- **descripción.-** Campo no nulo de tipo TEXT que almacenará la descripción de la convocatoria que se presentará a los visitantes del portal.
- **ruta_formato.-** Campo de tipo CHAR con una longitud máxima de 100 caracteres que almacenará el nombre del archivo que contendrá el formato que deberán seguir los artículos a presentarse en la convocatoria.
- **fecha_max_presentacion_art.-** Campo no nulo de tipo DATE que almacenará la fecha máxima para la presentación del artículo para la convocatoria.
- **fecha_max_evaluacion_art.-** Campo no nulo de tipo DATE que almacenará la fecha máxima en la que se evaluarán los artículos presentados en la convocatoria.
- **fecha_max_aceptacion_art.-** Campo no nulo de tipo DATE que almacenará la fecha máxima en la que se presentarán los resultados

de las evaluaciones y se mostrará los artículos que han sido aceptados.

- **fecha_max_correccion_art.-** Campo no nulo de tipo DATE que almacenará la fecha máxima en la que los autores de los artículos aceptados presentarán las correcciones hechas en las evaluaciones.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la convocatoria (existe o finalizado).

evento.- Tabla que almacenará los eventos que realizan los grupos de investigación. Esta tabla tendrá los siguientes campos:

- **id_evento.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla evento.
- **nombre.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 50 caracteres que almacenará el nombre del evento.
- **lugar.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 100 caracteres que almacenará el lugar donde se realizará el evento.
- **objetivo.-** Campo no nulo de tipo TEXT que almacenará el objetivo que se persigue al realizar el evento.
- **descripción.-** Campo no nulo de tipo TEXT que almacenará una descripción general del evento.
- **dirigido.-** Campo no nulo de tipo TEXT que almacenará a quien va dirigida la realización del evento.

- **fecha_inicio.-** Campo no nulo de tipo DATE que almacenará la fecha en la cual iniciará el evento.
- **fecha_fin.-** Campo no nulo de tipo DATE que almacenará la fecha en la cual finalizará el evento.
- **ref_tipo_evento.-** Campo no nulo de tipo INTEGER que almacenará el identificador del tipo de evento al que pertenece el evento.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el evento (existe, inactivo o finalizado).
- **slogan.-** Campo de tipo VARCHAR con una longitud máxima de 100 caracteres que almacenará el slogan o frase que identifique al evento.
- **direccion_grafico.-** Campo de tipo VARCHAR con una longitud máxima de 50 caracteres que almacenará el nombre del archivo que contendrá el gráfico del evento que se mostrará en la página principal del mismo.
- **descripción_registro.-** Campo de tipo TEXT que almacenará una descripción acerca del procedimiento que debe realizarse para los registros al evento.
- **estado_registro.-** Campo de tipo INTEGER que almacenará el estado en el que se encuentran los registros al evento parados (0) o activados (1).

grupo_investigacion.- Tabla que almacenará los grupos de investigación que son parte fundamental del negocio. Esta tabla tendrá los siguientes campos:

- **id_grupo_investigacion.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla grupo_investigacion.
- **nombre_grupo.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 50 caracteres que almacenará el nombre del grupo de investigación.
- **objetivo_grupo.-** Campo no nulo de tipo TEXT que almacenará el objetivo que persigue el grupo de investigación.
- **logo_direccion.-** Campo de tipo VARCHAR con una longitud máxima de 100 caracteres que almacenará el nombre del archivo que contendrá el logo del grupo de investigación.
- **pagina_web.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 100 caracteres que almacenará la dirección de la página Web del grupo de investigación.
- **telefono.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 16 caracteres que almacenará el teléfono del grupo de investigación.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el grupo de investigación (existe o finalizado).

rol.- Tabla que almacenará los roles que se asignarán a los usuarios dentro del sistema. Esta tabla tendrá los siguientes campos:

- **id_rol.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla rol.
- **nombre.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 40 caracteres que almacenará el nombre del rol.
- **descripción.-** Campo de tipo VARCHAR con una longitud máxima de 100 caracteres que almacenará una descripción del rol.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el rol (existe o finalizado).

rol_usuario.- Tabla que almacenará que roles se les asigna a los usuarios. Esta tabla tendrá los siguientes campos:

- **id_rol_usuario.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla rol_usuario.
- **ref_rol.-** Campo no nulo de tipo INTEGER que almacenará el identificador del rol asociado.
- **ref_usuario.-** Campo no nulo de tipo CHAR con una longitud máxima de 7 caracteres que almacenará el identificador del usuario al que se le asignará los roles.

- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el rol asignado a un usuario (existe o finalizado).

tema.- Tabla que almacenará los temas que conforman y sobre los cuales tratará una convocatoria. Esta tabla tendrá los siguientes campos:

- **id_tema.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla tema.
- **ref_convocatoria.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la convocatoria a la que pertenece el tema.
- **nombre.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 100 caracteres que almacenará el nombre del tema.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el tema (existe o finalizado).

tema_articulo.- Tabla que representa la relación que existe entre los artículos que participan en un tema específico de una convocatoria. Esta tabla tendrá los siguientes campos:

- **id_tema_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla tema_articulo.
- **ref_tema.-** Campo no nulo de tipo INTEGER que almacenará el identificador del tema asociado.

- **ref_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador del artículo asociado.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el artículo asignado a un tema (existe o finalizado).

usuario.- Tabla que almacenará a los usuarios que se encuentran registrados en el portal. Esta tabla tendrá los siguientes campos:

- **id_usuario.-** Campo no nulo de tipo CHAR con una longitud máxima de 7 caracteres que almacenará el identificador de la tabla usuario. Este identificador consta de dos partes los primeros 2 dígitos corresponden al año en el que el usuario se suscribe al portal y los siguientes 5 dígitos son números consecutivos asignados.
- **identificación.-** Campo no nulo de tipo CHAR con una longitud máxima de 10 caracteres que almacenará el número de cédula del usuario o su número de pasaporte.
- **nombre.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 30 caracteres que almacenará el nombre del usuario.
- **apellido.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 30 caracteres que almacenará el apellido del usuario.

- **correo.-** Campo no nulo de tipo CHAR con una longitud máxima de 50 caracteres que almacenará el correo del usuario, que le servirá para ingresar al portal.
- **clave.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 15 caracteres que almacenará la clave para el ingreso al portal.
- **fecha_nacimiento.-** Campo no nulo de tipo DATE que almacenará la fecha de nacimiento del usuario.
- **direccion.-** Campo no nulo de tipo VARCHAR con una longitud máxima de 50 caracteres que almacenará la dirección domiciliaria del usuario.
- **genero.-** Campo no nulo de tipo CHAR con una longitud máxima de 1 caracter que almacenará el sexo del usuario. F si es femenino y m si es masculino.
- **telefono.-** Campo de tipo VARCHAR con una longitud máxima de 16 caracteres que almacenará el número de teléfono convencional del usuario.
- **celular.-** Campo de tipo VARCHAR con una longitud máxima de 16 caracteres que almacenará el número celular del usuario.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento usuario (existe o finalizado).
- **ref_ciudad.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la ciudad a la que pertenece el usuario.

- **ref_titulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador del título que posee el usuario.
- **hoja_vida.-** Campo de tipo TEXT que almacenará la hoja de vida del usuario en caso de ser un conferencista invitado a un evento.
- **foto_direccion.-** Campo de tipo VARCHAR con una longitud máxima de 50 caracteres que almacenará el nombre del archivo que contendrá la foto del usuario en el caso de ser un conferencista invitado a un evento.

evaluación_tema_articulo.- Tabla que almacenará los revisores que han sido asignados para evaluar un artículo en una convocatoria.. Esta tabla tendrá los siguientes campos:

- **id_evaluacion_tema_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla evaluación_tema_articulo.
- **ref_evaluacion.-** Campo no nulo de tipo INTEGER que almacenará la referencia hacia la evaluación creada en el evento para evaluar artículos.
- **ref_tema_articulo.-** Campo no nulo de tipo INTEGER que almacenará la referencia hacia la tabla tema_articulo.
- **ref_evaluador_articulo.-** Campo no nulo de tipo INTEGER que almacenará la referencia hacia la tabla evaluador_articulo.

- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la evaluación_tema_articulo (existe o finalizado).

evaluador_ articulo.- Tabla que almacenará los suscriptores que fueron asignados con el rol de revisores de artículos para el portal. Esta tabla tendrá los siguientes campos:

- **id_evaluador_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla evaluador_articulo.
- **ref_rol_usuario.-** Campo no nulo de tipo INTEGER que almacenará la referencia al rol_usuario para identificar que usuario fue asignado como revisor.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el evaluador_articulo (existe o finalizado).

respuestas.- Tabla que almacenará las respuestas que los evaluadores ingresan cuando llenan la evaluación de un artículo. Esta tabla tendrá los siguientes campos:

- **id_respuestas.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla respuestas.
- **ref_evaluacion_tema_articulo.-** Campo no nulo de tipo INTEGER que almacenará la referencia hacia la tabla evaluación_tema_articulo.

- **ref_seccion.-** Campo no nulo de tipo INTEGER que almacenará la referencia a la tabla sección para identificar la sección de la evaluación.
- **ref_pregunta.-** Campo no nulo de tipo INTEGER que almacenará la referencia a la tabla pregunta para identificar la pregunta que ha llenado el revisor.
- **ref_alternativa.-** Campo no nulo de tipo TEXT que almacenará la alternativa o respuesta ingresada por el revisor. Si el dato es un número entonces existe una alternativa y es una referencia a la tabla y si es un texto significa que viene de una pregunta abierta.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la respuesta (existe o finalizado).

conferencista_articulo.- Tabla que almacenará la relación entre el conferencista del evento y el artículo de su conferencia. Esta tabla tendrá los siguientes campos:

- **id_conferencista_articulo.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla conferencista_articulo.
- **ref_conferencista_evento.-** Campo no nulo de tipo INTEGER que almacenará la referencia a la tabla conferencista_evento.
- **ref_articulo.-** Campo no nulo de tipo INTEGER que almacenará la referencia al artículo.

- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento al conferencista_articulo (existe o finalizado)

evaluacion.- Tabla que almacenará la cabecera de la evaluación que se crea en un evento para revisión de artículos. Esta tabla tendrá los siguientes campos:

- **id_evaluacion.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla evaluacion.
- **ref_convocatoria.-** Campo no nulo de tipo INTEGER que almacenará la referencia la convocatoria de artículos.
- **mensaje.-** Campo no nulo de tipo TEXT que almacenará el mensaje que los administradores quieren dar a los evaluadores.
- **descripcion.-** Campo no nulo de tipo TEXT que almacenará la descripción de la evaluación.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la evaluación (existe o finalizado)

seccion.- Tabla que almacenará las secciones de las evaluaciones que se estén creando. Esta tabla tendrá los siguientes campos:

- **id_seccion.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla seccion.

- **titulo.-** Campo no nulo de tipo VARCHAR de longitud máxima de 250 que almacenará el título de las sección creadas en una evaluación.
- **descripcion.-** Campo no nulo de tipo TEXT que almacenará una descripción de cada sección.
- **oculta.-** Campo no nulo de tipo INTEGER que almacenará la información para determinar si una sección dentro de una evaluación puede o no ser vista por el autor.
- **ref_evaluacion.-** Campo no nulo de tipo INTEGER que almacenará la referencia a la evaluación.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la seccion (existe o finalizado)

pregunta.- Tabla que almacenará las preguntas de las secciones que se estén creando. Esta tabla tendrá los siguientes campos:

- **id_pregunta.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla pregunta.
- **ref_tipo_pregunta.-** Campo no nulo de tipo INTEGER que almacenará la referencia a la tabla tipo_pregunta.
- **titulo.-** Campo no nulo de tipo VARCHAR de longitud máxima de 250 almacenará el título de cada pregunta.
- **ref_seccion.-** Campos no nulo de tipo INTEGER que al almacenará la referencia a la tabla seccion.

- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la seccion (existe o finalizado).

tipo_pregunta.- Tabla que almacenará los tipos de preguntas que pueden existir. Esta tabla tendrá los siguientes campos:

- **id_tipo_pregunta.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla tipo_pregunta.
- **titulo.-** Campo no nulo de tipo VARCHAR de longitud máxima de 250 almacenará el título del tipo de pregunta.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento el tipo_poregunta (existe o finalizado).

alternativa.- Tabla que almacenará las alternativas de las preguntas. Esta tabla tendrá los siguientes campos:

- **id_alternativa.-** Campo no nulo de tipo INTEGER que almacenará el identificador de la tabla alternativa.
- **ref_pregunta.-** Campo no nulo de tipo INTEGER que almacenará la referencia a la tabla pregunta.
- **titulo.-** Campo no nulo de tipo VARCHAR de longitud máxima de 250 almacenará el título de la alternativa.
- **estado.-** Campo de tipo INTEGER que almacenará el estado que posee en ese momento la alternativa (existe o finalizado).

BIBLIOGRAFIA

- [1] Snoeck, M., Dedene, G., Verhelst, M., Depuydt, A-M., *Object-Oriented Enterprise Modelling with MERODE*, Leuven University Press, 1999.
- [2] Página de referencia del MVC,
<http://www.enode.com/x/markup/tutorial/mvc.html>
- [3] Core J2EE Patterns Best Practices and Design Strategies,
<http://www.corej2eepatterns.com/Patterns2ndEd/CompositeView.htm>
- [4] Core J2EE Patterns Best Practices and Design Strategies,
<http://www.corej2eepatterns.com/Patterns2ndEd/DataAccessObject.htm>
- [5] Java 2 Platform, Enterprise Edition (J2EE) Overview,
<http://java.sun.com/j2ee/overview.html>
- [6] Ejemplo del uso de Ajax en Java, <http://www.tutorial-lab.com/tutoriales-Java/ID153-Ejemplo-de-AJAX-con-Java.aspx>
- [7] Ajax hispano, <http://www.ajaxhispano.com/>