

**ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO DEL PORTAL  
WEB, ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE NUEVOS  
SERVICIOS PARA EL PORTAL WEB DE LA ESCUELA DE INGENIERIA  
MECÁNICA.**

**LAURA MARCELA RUEDA TRISTANCHO**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA**

**2014**

**ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO DEL PORTAL  
WEB, ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE NUEVOS  
SERVICIOS PARA EL PORTAL WEB DE LA ESCUELA DE INGENIERIA  
MECÁNICA.**

**LAURA MARCELA RUEDA TRISTANCHO**

**Trabajo de grado para optar al título de Ingeniera de Sistemas**

**Director**

**Msc. LUIS IGNACIO GONZÁLEZ RAMÍREZ**

**Magíster en Informática**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER  
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS  
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA  
BUCARAMANGA**

**2014**

## Agradecimientos

Agradezco a Dios por permitirme alcanzar este gran logro en mi vida, por llenarme de fuerza en los momentos difíciles y por darme la oportunidad de compartir este triunfo con las personas que más amo. A mis padres Jesús y Blanca porque sin ellos esto no hubiese sido posible, gracias por todo su apoyo, comprensión y amor, gracias por no perderse un solo momento de mi vida son mi más grande ejemplo a seguir. A mis hermanos Oscar, María y Nicolás por estar siempre a mi lado, a Oscar por enseñarme que las dificultades son más llevaderas si nos unimos en familia, a María y Nicolás porque por ellos tengo la responsabilidad de ser cada día mejor para ser su ejemplo. A mi novio Andrés por hacerme entender que las cosas buenas toman tiempo y que con amor y dedicación todo se logra, por enseñarme cuanto valgo, por estar presente aun cuando no lo he notado y por tener fe en mí. Al profesor Luis Ignacio por permitirme pertenecer al grupo, por su interés en dirigir mi trabajo de grado y por su apoyo en todo mi proceso de aprendizaje. Quiero agradecer especialmente al grupo calumet por acogerme y enseñarme todo lo necesario, por sus explicaciones y la paciencia a la hora de enseñarme.

Laura Marcela Rueda Tristancho.

## CONTENIDO

<b>INTRODUCCIÓN .....</b>	<b>14</b>
<b>1. PRESENTACIÓN DEL PROYECTO.....</b>	<b>15</b>
1.1 ORIENTACIÓN SOBRE EL CONTENIDO DEL INFORME.....	15
1.2 ANTECEDENTES .....	16
1.3 DEFINICIÓN DEL PROBLEMA .....	16
1.4 OBJETIVOS .....	18
1.4.1 Objetivo General.....	18
1.4.2Objetivos Específicos .....	18
1.5 JUSTIFICACIÓN .....	21
1.6 ALCANCES Y LIMITACIONES.....	22
<b>2. MARCO TEÓRICO .....</b>	<b>24</b>
2.1 ARQUITECTURA CLIENTE/SERVIDOR .....	24
2.1.1 Arquitectura Cliente/Servidor aplicada .....	25
2.1.2Ventajas del esquema Cliente/Servidor.....	25
2.1.3Desventajas del esquema Cliente/Servidor .....	26
2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS .....	26
2.2.1 Tecnología aplicada .....	27
2.3 BASES DE DATOS .....	29
2.3.1 MySQL .....	29
2.4 NETBEANS.....	30
2.5 SISTEMA DE CONTROL DE VERSIONES .....	31

2.5.1 Subversión .....	32
2.6 PROGRAMACIÓN UTILIZADA .....	33
2.6.1 Clases .....	33
2.6.2 Objetos .....	33
2.6.3 Atributos .....	34
2.6.4 Métodos.....	34
2.6.5 Herencia .....	34
2.6.6 Beneficios de la Programación Orientada a Objetos .....	34
2.6.7 Java y JDK (Java Development Kit) .....	35
2.7 SERVIDORES WEB.....	35
2.7.1 Servidor Jakarta Tomcat .....	36
<b>3 MARCO METODOLÓGICO .....</b>	<b>37</b>
3.1 PROTOTIPO EVOLUTIVO.....	37
3.2 LENGUAJE DE MODELADO UNIFICADO.....	39
3.2.1 Diagramas de UML.....	39
3.3 ESTÁNDARES DE PROGRAMACIÓN.....	41
3.3.1 Modelo de datos .....	41
3.3.2 Nombres de las tablas.....	41
3.3.3 Clases .....	42
3.3.4 Páginas JSP .....	42
3.3.5 Organización de Directorios .....	43
<b>4 DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO.....</b>	<b>44</b>
4.1 PROTOTIPO ESPERADO .....	44
4.1.1 Diagramas de Casos de Uso.....	53
4.1.2 Documentación de Casos de Uso del Sistema .....	58
4.1.3 Diseño y Análisis. ....	63
Diagramas Entidad/Relación de los servicios desarrollados.....	63

4.1.4	Modelo de procesos del sistema.....	67
4.1.5	Implementación, Implantación y Pruebas Generales .....	67
4.2	<b>MANTENIMIENTO Y ADMINISTRACIÓN .....</b>	<b>68</b>
4.2.1	Actividades de Mantenimiento.....	68
4.2.2	Actividades de Soporte a Usuarios.....	72
4.2.3	Actividades de Administración.....	73
<b>5</b>	<b>PRUEBAS DEL SISTEMA.....</b>	<b>75</b>
5.1	PRUEBAS DE VERIFICACIÓN .....	75
5.1.1	Pruebas por componente .....	75
5.2	PRUEBAS DE INTEGRACIÓN.....	76
5.3	PRUEBAS DE VALIDACIÓN.....	76
<b>6</b>	<b>CONCLUSIONES .....</b>	<b>77</b>
<b>7</b>	<b>RECOMENDACIONES .....</b>	<b>78</b>
	<b>BIBLIOGRAFIA.....</b>	<b>76</b>
	<b>ANEXOS.....</b>	<b>77</b>

## LISTA DE TABLAS

Tabla 1. Caso de Uso: Proponer eventos.....	58
Tabla 2. Caso de Uso: Eventos propuestos.....	58
Tabla 3. Caso de Uso: Portal escuela.....	59
Tabla 4. Caso de Uso: Portal Producción Intelectual.....	59
Tabla 5. Caso de Uso: Servicio archivos. Portal escuela.....	60
Tabla 6. Caso de Uso: Servicio archivos. ....	60
Tabla 7. Caso de Uso: Asignar consejo.....	61
Tabla 8. Caso de Uso: Cronograma. ....	61
Tabla 9. Caso de Uso: Mantenimiento y Administración del portal EIMWEB. ....	62
Tabla 10. Caso de Uso: Descripción de las Entidades .....	66
Tabla 11. Caso de Uso: Pruebas Realizadas: Proponer eventos. ....	105
Tabla 12. Caso de Uso: Pruebas Realizadas: Eventos propuestos.....	106
Tabla 13. Caso de Uso: Portal Escuela. ....	106
Tabla 14. Caso de Uso: Portal Producción Intelectual.....	107
Tabla 15. Caso de Uso: Servicios Archivos. ....	108
Tabla 16. Caso de Uso: Portal grupos.....	108
Tabla 17. Caso de Uso: Archivos aula.....	109
Tabla 18. Caso de Uso: Asignar consejo de escuela y de facultad. ....	109
Tabla 19. Caso de Uso: Asignar consejo de escuela y de facultad. ....	110
Tabla 20. Caso de Uso: Cronograma. ....	111

## LISTA DE FIGURAS

Figura 1. Modelo Cliente/Servidor.....	24
Figura 2. Modelo de acceso a JSP .....	28
Figura 3. Prototipo Evolutivo.....	37
Figura 4. Diagrama de Casos de Uso: Proponer eventos.....	53
Figura 5. Diagrama de Casos de Uso: Eventos propuestos. ....	53
Figura 6. Diagrama de Casos de Uso: Producción intelectual.....	54
Figura 7. Diagrama de Casos de Uso: Portal Producción intelectual.....	54
Figura 8. Diagrama de Casos de Uso: Servicios-Archivos. Portal escuela.....	55
Figura 9. Diagrama de Casos de Uso: Servicios-Archivos. Portal grupos. ....	55
Figura 10. Diagrama de Casos de Uso: Asignar consejo. ....	56
Figura 11. Diagrama de Casos de Uso: Cronograma.....	56
Figura 12 Diagrama de Casos de Uso: Fin de sesión.....	57
Figura 13 Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración. ....	57
Figura 14 Diagrama E/R: Eventos comunidad UIS.....	63
Figura 15 Diagrama E/R: Producción Intelectual. ....	64
Figura 16 Diagrama E/R: Crear, modificar o eliminar consejo escuela.....	65
Figura 17 Diagrama E/R: Crear, modificar o eliminar consejo de facultad.....	65
Figura 18 Arquitectura Cliente/Servidor de Dos Capas. ....	83
Figura 19 Arquitectura Cliente/Servidor de Tres Capas.....	84
Figura 20 Función del DBMS.....	91
Figura 21 Diagramas de Casos de Uso. ....	92
Figura 22 Proponer eventos.....	94
Figura 23 Eventos propuestos. ....	96
Figura 24 Producción intelectual.....	98
Figura 25 Portal Producción intelectual. ....	100
Figura 26 Portal Producción intelectual. ....	101
Figura 27 Asignar consejo de escuela y facultad.....	103
Figura 28 Asignar consejo de escuela y facultad.....	104

## LISTA DE ANEXOS

Anexo A. ARQUITECTURA CLIENTE/SERVIDOR .....	81
Anexo B. TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS .	85
Anexo C. BASES DE DATOS .....	87
Anexo D. LEGUAJE DE MODELADO UNIFICADO (UML) .....	92
Anexo E. PRUEBAS DEL SISTEMA.....	105

## RESUMEN

**TÍTULO:** ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO DEL PORTAL WEB, ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE NUEVOS SERVICIOS PARA EL PORTAL WEB DE LA ESCUELA DE INGENIERIA MECÁNICA<sup>i</sup>.

**AUTOR**<sup>ii</sup>: Laura Marcela Rueda Tristancho.

**PALABRAS CLAVE:** Portal Web, Consejo de Escuela, EIMWEB (Escuela Ingeniería Mecánica).

## DESCRIPCIÓN

La escuela de Ingeniería Mecánica cuenta con un sistema de información orientado a la Web, que se encarga de la administración y control de las actividades tanto académicas como administrativas que se realizan dentro de la escuela, así como del control de usuarios y servicios que se les proporcionan. Estos servicios ofrecen soluciones a los problemas y necesidades de los usuarios, por esto las labores de mantenimiento y actualización son indispensables.

En el portal de las escuelas existe el módulo de Propuesta Eventos comunidad escuela en el los usuarios proponen eventos, permitiendo correos informativos, comentarios y la opción de unirse, si lo desean, a la propuesta. Consistente con nuestro enfoque comunidad UIS, se hace necesario crear el módulo de propuesta de eventos que permita la creación de estos, por parte de un miembro, dirigida a usuarios de otras escuelas.

Dando respuesta a las necesidades de la comunidad, se decidió crear un Portal de Producción Intelectual. Para esto, se conforman cuatro interfaces diferentes para presentar la información de la comunidad, ellas son: Portal de la Escuela, Portal de los Grupos, Portal del Profesor, y por último, Portal Producción Intelectual. Este último mostrará los productos propios de la comunidad como: artículos, ponencias, libros, software, etc.

El portal permite variadas formas de compartir documentos. No obstante, estos servicios, en su mayoría, permiten realizar esta labor sobre archivos tipo Zip, donde surge la necesidad de poder subir otros tipos como por ejemplo: rar, pdf.

Actualmente el grupo Calumet desarrolla el módulo de soporte a los consejos de escuela y facultad. En este módulo es necesario crear el servicio que permita eliminar, agregar o modificar miembros de los consejos. A su vez, es necesario permitir definir las fechas de inicio y fin para inscribir solicitudes dirigidas a los consejos por los usuarios de las escuelas.

---

<sup>i</sup> Trabajo de grado Modalidad Trabajo de Investigación.

<sup>ii</sup> Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería de Sistemas e Informática.  
Director: Luis Ignacio González Ramírez.

## ABSTRACT

**TITLE:** ADMINISTRATION, SUPPORT USERS WEB SITE MAINTENANCE, ANALYSIS, DEVELOPMENT AND IMPLEMENTATION OF NEW SERVICES FOR SCHOOL WEB PORTAL OF MECHANICAL ENGINEERING<sup>i</sup>.

**AUTHORS<sup>ii</sup>:** Laura Marcela Rueda Tristancho.

**KEYWORDS:** Web Portal, School Council, EIMWEB (Mechanical Engineering School).

## DESCRIPTION

Mechanical Engineering School has an information -oriented Web , which is responsible for the administration and control of both academic and administrative activities that take place within the school as well as the control of users and services are provided. These services offer solutions to the problems and needs of users, so the maintenance and updating are essential.

In the schools portal exist the Motion Events community school module in this module the users provide events allowing informative emails, comments and the option to join if they wish, to the proposal. Consistent with our approach UIS community, it is necessary to create the given event module that allows the creation of these, by a member, addressed to users from other schools.

Giving response to the needs of the community, decided to establish an Intellectual Production Portal. For this, four different interfaces conform to present information in the community, they are: the School Portal, Portal Groups , Teacher Portal , and finally , Intellectual Production Portal . The latter show the products themselves in the community as articles, presentations, books, software, etc. . . .

The portal allows various forms of sharing documents. However, these services mostly allow this type work on Zip files, where the need to upload other types such as arises: rar, pdf.

Currently the group is developing Calumet module support school councils and faculty. This module is necessary to create a service that allows delete, add or modify members of the councils. In turn, it must be possible to define the start and end dates to register requests to councils by users' schools.

---

<sup>i</sup> Degree Work: Research Paper.

<sup>ii</sup> Physicomechanical Engineering Faculty, System Engineering School.  
Director: M.Sc. Luis Ignacio González Ramírez

## **INTRODUCCIÓN**

El portal EIMWEB perteneciente a la Escuela de Ingeniería Mecánica, se ha convertido en uno de los principales canales de comunicación e integración entre los miembros de la comunidad de la escuela. Gracias a la aceptación y utilización por parte de los usuarios surge el objetivo fundamental de fortalecer y mejorar los servicios que se ofrecen, además de la creación de nuevos servicios para suplir las necesidades crecientes de los miembros de la escuela.

El grupo de desarrollo de software Calumet se encarga de desarrollar, administrar y mantener el portal EIMWEB al cual se le realizan continuos cambios y actualizaciones para hacer de este una herramienta cada vez más útil, eficiente, agradable y de fácil uso para la comunidad. Con el objeto de llevar a cabo esta labor y proporcionar páginas con contenido dinámico se ha contado con herramientas software de libre distribución como JSP, Java, JavaScript, jQuery y MySQL.

El continuo soporte ofrecido al portal EIMWEB por el grupo Calumet se hace de manera integral permitiendo el mejoramiento continuo de los servicios ofrecidos.

Para satisfacer las nuevas necesidades de los usuarios se hace necesaria la creación de nuevos servicios, tales como eventos comunidad UIS, nuevo portal de Producción intelectual, permitir al administrador eliminar, agregar o modificar los miembros del consejo de escuela y facultad y la subida de diferente tipos de archivos.

En este documento se presenta el soporte teórico, metodológico y técnico del desarrollo web de los servicios previamente mencionados, así como de las labores de administración, mantenimiento y soporte a usuarios desempeñadas durante el desarrollo del proyecto.

## **PRESENTACIÓN DEL PROYECTO**

### **1.1 ORIENTACIÓN SOBRE EL CONTENIDO DEL INFORME**

Este documento presenta en detalle las etapas llevadas en el desarrollo del proyecto: ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO DEL PORTAL WEB, ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE NUEVOS SERVICIOS PARA EL PORTAL WEB DE LA ESCUELA DE INGENIERIA MECÁNICA; la información está distribuida de la siguiente forma:

**CAPITULO 1.** Presentación del Proyecto: Se hace un análisis de los antecedentes del proyecto, definición del problema, objetivos generales, específicos, justificación, alcances y limitaciones del proyecto.

**CAPITULO 2.** Marco Teórico: Se presentan los conceptos utilizados en el desarrollo técnico del proyecto.

**CAPITULO 3.** Marco Metodológico: Se menciona el procedimiento metodológico que se siguió para la elaboración del proyecto y las razones por la que fue elegida dicha metodología.

**CAPITULO 4.** Desarrollo de la herramienta: Se presenta el análisis de requisitos y el diseño de la herramienta.

**CAPITULO 5.** Documento de pruebas del sistema: se presenta un informe de las pruebas realizadas a la herramienta desarrollada y los resultados obtenidos.

**CAPITULO 6.** Conclusiones del trabajo realizado.

**CAPITULO 7.** Recomendaciones y sugerencias a tener en cuenta en la elaboración de futuros proyectos.

## **1.2 ANTECEDENTES**

Debido al crecimiento de la comunidad de la Universidad Industrial de Santander y en particular la comunidad de la Escuela de Ingeniería Mecánica se hace necesario un canal de información y control de la misma. Es así como surge la idea de crear una herramienta que en primer lugar, facilite la interacción entre los miembros de la escuela, desarrollo llevado a cabo en el año 2009 por el grupo de desarrollo de software Calumet el cual es conformado y liderado por estudiantes de pregrado de Ingeniería de Sistemas.

Debido a la labor de administración, mantenimiento y actualización que realizan sus integrantes, el grupo de desarrollo de software Calumet ha logrado la implementación de su portal web en las escuelas pertenecientes a las facultades de Ingenierías Fisicomecánicas y Fisicoquímicas, así como en sus respectivas decanaturas.

## **1.3 DEFINICIÓN DEL PROBLEMA**

En la actualidad la escuela de Ingeniería Mecánica cuenta con un sistema de información orientado a la Web que se encarga de la administración y control de las diferentes actividades tanto académicas como administrativas que se realizan dentro de la escuela, así como del control de usuarios y servicios que se le proporciona. Los servicios de los portales Web de las escuelas deben mejorar constantemente y adaptarse a los cambios que se presenten en su entorno, a su vez deben dar solución a los problemas y necesidades que surjan por parte de los usuarios del sistema para incrementar su tiempo de vida útil y no llegar a convertirse en un software obsoleto, razón por la cual las labores de mantenimiento y actualización se hacen indispensables.

Motivados por brindarle a los profesores, grupos de investigación, y a la comunidad en general un mejor servicio y un sistema robusto capaz de suplir todas las necesidades se hace necesaria la implantación de nuevos servicios para la gestión, control y ejecución de proyectos de diferente índole.

En este momento existe el módulo de Propuesta de Eventos comunidad escuela en el que los usuarios proponen eventos dirigidos a parte de la comunidad, o a toda ella permitiendo correos informativos, comentarios y adherirse, si lo desean, a la propuesta. Consistente con nuestro enfoque comunidad UIS, se hace necesario crear el módulo de propuesta de eventos que permita la creación de eventos, por parte de un miembro, y dirigida a usuarios de diferente tipo de otras escuelas. Allí los usuarios, de diferentes escuelas, podrán opinar y adherirse a la propuesta. Correos a los usuarios interesados acompañarán la propuesta y las intervenciones.

Una de las necesidades de la comunidad, es la creación de un Portal de Producción Intelectual de los miembros de las escuelas. De esta manera, se conforman cuatro interfaces diferentes para presentar la información de la comunidad de las escuelas, ellas son: Portal de la Escuela, Portal de los Grupos, Portal del Profesor, y por último, Portal de Producción Intelectual.

El portal permite a la comunidad variadas formas de compartir documentos, hojas de cálculo y archivos en general. No obstante, estos servicios, en su mayoría, solo permiten realizar esta labor sobre archivos tipo Zip, de donde surge la necesidad de la comunidad de poder subir otros tipos como por ejemplo: rar, pdf...

Actualmente el grupo Calumet desarrolla el módulo de soporte a los consejos de escuela y facultad. Dentro de este módulo es necesario crear el servicio que permita al administrador eliminar, agregar o modificar los miembros del consejo de escuela y facultad. A su vez, es necesario permitir al administrador definir las

fechas de inicio y fin de los lapsos de tiempo en el semestre para inscribir solicitudes dirigidas al consejo de escuela y facultad por los usuarios de las escuelas. Estas fechas varían de acuerdo al tipo de solicitud.

## **1.4 OBJETIVOS**

### **1.4.1 Objetivo General**

Realizar las funciones de Administración, Mantenimiento, Análisis, Diseño y Desarrollo de nuevos servicios para el portal Web existente en la escuela de Ingeniería Mecánica para hacer óptimos, eficientes y sostenibles los trámites dentro de la escuela y el acceso a la información de la misma.

### **1.4.2 Objetivos Específicos**

**Realizar labores de administración del portal web de la Escuela de Ingeniería Mecánica teniendo en cuenta principalmente:**

- Generar Backups (copias de seguridad) diariamente de las bases de datos.
- Hacer seguimiento del uso que hacen los usuarios a los servicios de Foros, Eventos Propuestos, Cartelera, Archivos y Mi perfil dentro del portal EIMWEB, para detectar usos indebidos o incorrectos por parte de éstos.
- Actualizar periódicamente las bases de datos con el objeto de mantener al día la información referente a matrículas, horarios, estados y categorías de los usuarios.

- Atender las consultas y sugerencias que los usuarios hagan para proponerlas como mejoramiento y ofrecimiento de nuevos servicios en la próxima versión del portal web.
- Realizar una revisión constante de los archivos que se suben al sitio, eliminando los que no son necesarios para evitar saturación del portal.

**Llevar a cabo labores de mantenimiento al portal EIMWEB, en las que podemos encontrar:**

- Hacer el seguimiento del funcionamiento del portal para corregir posibles defectos generados por errores en el código fuente que se puedan presentar.
- Revisar y depurar la estructura de directorios y archivos del portal web.

**Análisis, diseño, desarrollo e implementación de nuevos servicios o reingeniería de servicios ya existentes, tales como:**

- Crear el servicio de Propuesta de Eventos comunidad UIS donde los usuarios al crear la propuesta podrán dirigirla a aquellas escuelas que consideren es de su interés. Allí, los usuarios de las diferentes escuelas, podrán opinar y sumarse a ella. El usuario creador podrá promover su evento enviando un correo a los interesados.
- Crear el Portal de Producción Intelectual de las escuelas donde se muestren los productos y la demás información relacionada con

ellos. Estos productos estarán enlazados con los profesores y grupos que los realizaron y permitirá, este Portal, navegar al Portal de esos profesores y de esos grupos. Así mismo, se debe realizar los ajustes necesarios para que se pueda navegar entre los cuatro portales: Escuela, Profesor, Grupo y Producción intelectual manteniendo las sesiones y dando acceso sencillo a toda la información. Se debe mejorar la presentación de los perfiles de los profesores en su portal.

- El portal permite la subida de archivos en diferentes servicios condicionando, en la mayoría de casos, a que sean tipo zip. Se debe realizar una reingeniería que permita subir diferentes tipos de archivos, entre estos: pdf, rar, Zip,... entre otros, dando flexibilidad y mejorando los servicios del portal.
- Crear el servicio para eliminar, agregar y cambiar miembros de los consejos de escuela en el módulo de soporte a los consejos de escuela y facultad.
- Dentro del módulo de soporte al Consejo de Escuela y Facultad es necesario crear el servicio que permita definir las fechas de inicio y fin de inscripción de solicitudes a los consejos por parte de los miembros de la escuela. Estas fechas varían de acuerdo al tipo de solicitud, y deben ser visibles a toda la comunidad.

**Desempeñar labores de soporte a los usuarios del portal EIMWEB, brindando así solución a sus diferentes necesidades o situaciones que se puedan presentar. Entre estas labores se destacan:**

- Capacitar usuarios y estudiantes del primer nivel en el uso de servicios dentro del Portal EIMWEB, promoviendo así su utilización.

- Atender usuarios por olvido de la contraseña, creación de grupos, solicitudes de propuestas de eventos o de cartelera en el índice y creación de agendas con eventos y foros con sus respectivas conversaciones.
- Crear usuarios de forma manual, para personas no pertenecientes a la escuela y que por algún motivo necesitan registrarse en el sitio.
- Modificar los estados de los usuarios de acuerdo a la relación con la Escuela (Activo, Inactivo, Suspendido).

**Capacitar a los estudiantes que relevaran las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro del portal EIMWEB, en cuanto a:**

- Implantación del sitio local para la creación de nuevos servicios y realización de pruebas.
- Realizar inducción en cuanto al manejo y utilización de los JSP, Beans, base de datos, JavaScript y jQuery.
- Llevar a cabo la familiarización con el entorno del portal web.

## **1.5 JUSTIFICACIÓN**

Los servicios del portal de las escuelas deben mejorar constantemente y proporcionalmente a los cambios que se presenten en su entorno, a su vez deben dar correcta solución a los problemas y necesidades que surjan por parte de los

usuarios, razón por la cual las labores de mantenimiento y actualización son indispensables.

Motivados por brindarle a los profesores, estudiantes, grupos de investigación, y a la comunidad en general un mejor servicio y un sistema robusto capaz de suplir todas las necesidades se hace necesaria la implantación del servicio de Eventos comunidad UIS.

El portal permite a la comunidad variadas formas de compartir documentos, hojas de cálculo y archivos en general. No obstante, estos servicios, en su mayoría, solo permiten realizar esta labor sobre archivos tipo Zip, de donde surge la necesidad de la comunidad de poder subir otros tipos como son: rar, pdf,...

Dando respuesta a las necesidades de la comunidad, se ha decidido la creación de un Portal de Producción Intelectual de los miembros de las escuelas en donde se pueda acceder a toda la información sobre las diferentes producciones de los grupos, de las escuelas o de los profesores.

Actualmente el grupo Calumet desarrolla el módulo de soporte a los consejos de escuela y facultad. Dentro de este módulo es necesario crear el servicio que permita al administrador eliminar, agregar o modificar los miembros del consejo de escuela y facultad.

## **1.6 ALCANCES Y LIMITACIONES**

Actualmente el grupo Calumet desarrolla el módulo de soporte a los consejos de escuela y facultad. Dentro de este módulo es necesario crear el servicio que permita al administrador eliminar, agregar o modificar los miembros del consejo de escuela y facultad. A su vez, es necesario permitir al administrador definir las

fechas de inicio y fin de los lapsos de tiempo en el semestre para inscribir solicitudes dirigidas al consejo de escuela y facultad por los usuarios de las escuelas. Estas fechas varían de acuerdo al tipo de solicitud.

Dando respuesta a las necesidades de la comunidad, se ha decidido la creación de un Portal de Producción Intelectual de los miembros de las escuelas. De esta manera, se conforman cuatro interfaces diferentes para presentar la información de la comunidad de las escuelas, ellas son: Portal de la Escuela, Portal de los Grupos, Portal del Profesor, y por último, Portal de Producción Intelectual. Este último mostrará los productos fruto del resultado de la labor propia de la comunidad como son: artículos, ponencias, libros, software, proyectos, manuales, congresos, y otros. Este Portal contendrá toda la información sobre el producto, así como los docentes y grupos relacionados con ellos.

Además, relacionado con lo anterior, se hace necesario permitir la navegación, de una manera sencilla entre los cuatro tipos de portales manteniendo la sesión de los usuarios.

## 2. MARCO TEÓRICO

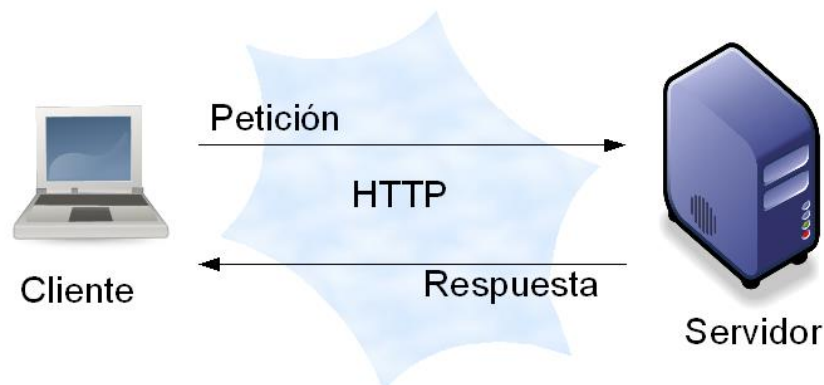
### 2.1 ARQUITECTURA CLIENTE/SERVIDOR

Se define como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma.

En el modelo C/S, el cliente envía un mensaje solicitando un servicio a un servidor (hacer una petición), y este envía uno o varios mensajes con la respuesta (provee el servicio) (ver figura 1). En un sistema distribuido cada máquina puede cumplir el rol de servidor para algunas tareas y el rol de cliente para otras.

La arquitectura C/S es una extensión de programación modular en la que la base fundamental es separar una gran pieza de software en módulos con el fin de hacer más fácil el desarrollo y mejorar su mantenimiento.

**Figura 1. Modelo Cliente/Servidor**



Fuente: Internet. <http://grupo-701-marce-nestore.wikispaces.com/Cliente-Servidor>

La arquitectura C/S presenta unas características y clasificaciones que permiten la aplicación correcta de este modelo. (Ver Anexo A).

### **2.1.1 Arquitectura Cliente/Servidor aplicada**

En el desarrollo de este proyecto se recurre a arquitectura de tres capas, debido a las ventajas ofrecidas como: Escalabilidad, fácil mantenimiento y el manejo de un mayor número de usuarios que la ofrecida por la arquitectura C/S de dos capas. La arquitectura es aplicada de la siguiente forma:

- Capa de Cliente: Interfaz con el usuario, se usa un navegador web.
- Capa Intermedia: Para los servicios del negocio se utiliza un computador configurado como servidor web, el cual almacena el portal web conformado por páginas JSP y JavaBeans. Allí se realizan los procesos complejos, y se solicitan los servicios del servidor de datos cuando es necesario acceder a la información almacenada en la base de datos.
- Capa de Servidor: Se utiliza el motor de bases de datos MySQL, el cual se encuentra en el mismo servidor web.

### **2.1.2 Ventajas del esquema Cliente/Servidor**

- La existencia de plataformas de software y hardware de varios fabricantes y cada vez más a económicas contribuye a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones.
- Este esquema facilita la integración entre sistemas heterogéneos y comparte información permitiendo que las maquinas existentes puedan ser utilizadas con interfaces amigables al usuario, de esta forma integrar los

computadores con sistemas medianos y grandes, sin necesidad de que todos tengan que utilizar el mismo sistema operacional.

- Facilita a los diferentes departamentos de una organización soluciones locales, permitiendo la integración de la información principal totalmente.

### **2.1.3 Desventajas del esquema Cliente/Servidor**

- El mantenimiento de los sistemas es complejo pues implica la interacción de diferentes partes hardware y software de diferentes proveedores, lo cual dificulta el diagnóstico de fallas.
- Se cuenta con escasas herramientas para la administración y ajuste del desempeño de los sistemas, además se deben tener estrategias para el manejo de errores y para salvaguardar la consistencia de los datos.
- La seguridad del esquema C/S es preocupante, un ejemplo: las validaciones y verificaciones que se deben hacer tanto en el cliente como en el servidor.
- El desempeño es un aspecto a tener en cuenta en el esquema C/S, problemas de este estilo pueden presentarse por congestión en la red.

## **2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS**

Las páginas dinámicas aportan grandes beneficios porque permiten entrar a bases de datos para extraer información que pueda presentarse al usuario, dependiendo de algunos permisos y de la misma forma para almacenar información.

Existen diferentes tecnologías para el desarrollo de páginas dinámicas, las cuales se pueden agrupar en Código del Lado del Cliente (Client Side Scripts) y Código del Lado del Servidor (Server Side Scripts). (Ver Anexo B).

### **2.2.1 Tecnología aplicada**

La tecnología aplicada para la creación del portal web fue JSP, por lo tanto los nuevos servicios son desarrollados con esta misma tecnología, ya que permite producir aplicaciones independientes de la plataforma y portables a otros sistemas operativos y servidores web.

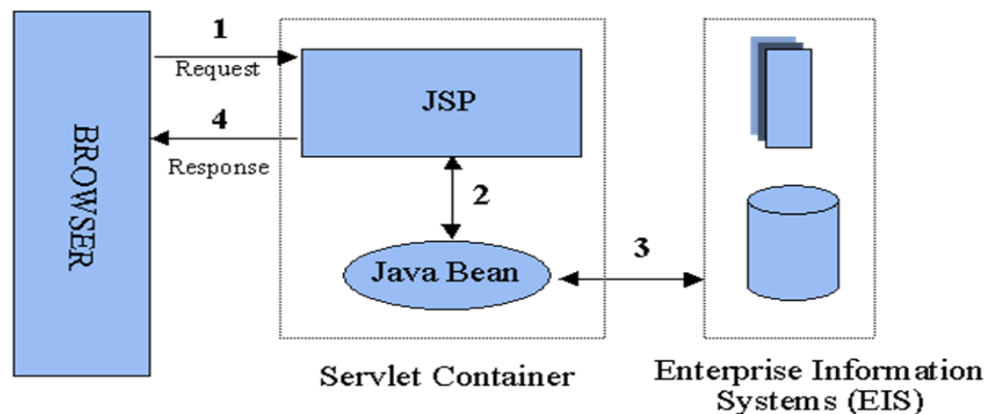
Las paginas JSP y servlets se ejecutan en la Máquina Virtual de Java, lo cual permite que se puedan usar en cualquier tipo de computador, siempre y cuando este instalada la Máquina Virtual de Java. Cada JSP se ejecuta en su propio contexto (llamado también hilo o hebra); pero no se comienza a ejecutar cada vez que recibe una petición, sino que persiste de una petición a la siguiente, de forma que no se pierde tiempo en invocarlo. Su persistencia permite hacer cosas de forma más eficiente como la conexión a bases de datos y manejo de sesiones.

Una página JSP se compila a una aplicación Java la primera vez que se invoca, y de esta aplicación Java se crea una clase que empieza a ejecutarse en el servidor como un servlet. Un JSP es una página web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa que recibe peticiones y genera a partir de ellas una página web.

## Modelo de acceso a jsp

1. Un usuario en su navegador web cliente hace una petición que es enviada a un archivo JSP. Este archivo accede a componentes del servidor que generan contenido dinámico y lo presentan en el navegador.
2. Después de recibir la petición del cliente, el archivo JSP pide información de un JavaBean si es necesario.
3. El JavaBean en turnos puede pedir información de otro JavaBean o de una base de datos.
4. Una vez el JavaBean genera el contenido, el archivo JSP puede consultar y presentar el contenido del JavaBean al navegador.

Figura 2. Modelo de acceso a JSP



Fuente: Internet: <http://geneura.ugr.es/~jmerelo/JSP/modelo1.gif>

La primera vez que un archivo JSP es invocado, este es compilado en un objeto, la respuesta del objeto es HTML estándar, el cual es interpretado por el navegador

para ser presentado al usuario. Después de la compilación, el objeto de la página es almacenado en la memoria del servidor. En las peticiones posteriores a esta página, el servidor revisa si el archivo JSP ha cambiado. Si no ha cambiado, el servidor utiliza el objeto de la página compilada guardado en memoria para generar la respuesta al cliente, en caso contrario el servidor automáticamente compila el archivo de la página y reemplaza el objeto en la memoria.

## **2.3 BASES DE DATOS**

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso, con una redundancia controlada y una estructura que refleja las interrelaciones y restricciones existentes en el mundo real. En la base de datos se almacena información considerada necesaria para una determinada organización o negocio.

Dentro de las bases de datos existen diferentes modelos, entre los cuales encontramos bases de datos jerárquicas, de red y relacionales. Así mismo para tener conexión a estos modelos es necesario utilizar conectores los más comunes son ODBC, MDB y JDBC. (Ver Anexo C).

Existen diferentes manejadores de bases de datos como MySQL, ORACLE, FoxPro, Microsoft Access o PowerBuilder, en el desarrollo de los portales FIFMEWeb y EIMWeb se utiliza MySQL.

### **2.3.1 MySQL**

Es un sistema de base de datos operacional considerado uno de los más importantes y utilizados por usuarios del medio para el diseño y programación de base de datos de tipo relacional. MySQL se usa como servidor a través del cual

pueden conectarse múltiples usuarios y utilizarlo al mismo tiempo. La característica más interesante de MySQL es que permite recurrir a las bases de datos multiusuario a través de la web y en diferentes lenguajes de programación y diferentes plataformas que se adaptan a diferentes necesidades y requerimientos, además MySQL es conocida por desarrollar alta velocidad de búsqueda de datos e información, a diferencia de sistemas anteriores.

### **Ventajas de MySQL**

- El MySQL es un Open Source, o sea código abierto que puede ser usado y modificado.
- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consume puede ser ejecutado en una maquina con escasos recursos sin ningún problema.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Su conectividad, velocidad, y seguridad hacen de MySQL altamente apropiado para acceder a bases de datos en internet.

### **2.4 NETBEANS**

Es un entorno de desarrollo integrado (IDE), siendo una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas escritos

en JAVA, pero puede servir para cualquier otro tipo lenguaje de programación. Netbeans es un producto libre y gratuito sin restricciones de uso.

- El Netbeans es un entorno de desarrollo integrado de código abierto escrito completamente en Java usando la plataforma Netbeans, soporta desarrollo de todos los tipos de aplicación Java (J2SE, web, EJB y aplicaciones móviles).
- La versión actual es NetBeans IDE 7.1.2. Desde NetBeans IDE 6.5 se extienden las características existentes del Java EE (incluyendo Soporte a Persistencia, EEJB 3 y JAX-WS). Adicionalmente, el Netbeans Enterprise Pack soporta el desarrollo de Aplicaciones empresariales java EE 5, incluyendo herramientas de desarrollo visuales de SOA, herramientas de esquemas XML, orientación a web servicios (for BPEL), y modelado UML. El NetBeans C/C++ Pack soporta proyectos de C/C++.
- Modularidad. Todas las funciones del IDE son provistas por módulos. Cada Módulo provee una función bien definida, tales como el soporte de Java, edición, o soporte para el sistema de control de versiones. Netbeans contiene todos los módulos necesarios para el desarrollo de aplicaciones Java en una sola descarga, permitiéndole al usuario comenzar a trabajar inmediatamente.

## **2.5 SISTEMA DE CONTROL DE VERSIONES**

- Un sistema de control de versiones es un software que administra el acceso a un conjunto de ficheros, y mantiene un historial de cambios realizados. El control de versiones es útil para guardar cualquier documento que cambie con frecuencia, o el código fuente de un programa.

- Normalmente consiste en una copia maestra en un repositorio central, y un programa cliente con el que cada usuario sincroniza su copia local. Además, el repositorio guarda registro de los cambios realizados por cada usuario, y permite volver a un estado anterior en caso de necesidad.
- Existen multitud de sistemas de control de versiones, pero sin duda, el más popular es CVS (Concurrent Versions System). CVS tuvo el mérito de ser el primer sistema usado por el movimiento de código abierto para que los programadores colaboran remotamente mediante el envío de parches. Es de uso gratuito, código abierto, y emplea fusión de cambios.
- Subversión se creó para igualar y mejorar la funcionalidad de CVS, preservando su filosofía de desarrollo.

### **2.5.1 Subversión**

Sistema de control de versiones iniciado por CollabNet Inc. Emplea licencia Apache/BSD. Se usa para mantener versiones actuales e históricas y los cambios de archivos tales como los de código fuente, páginas web y/o documentación. Esto permite recuperar versiones antiguas de los datos o examinar cómo han ido evolucionando esto. Su objetivo es ser un sucesor prácticamente compatible del ampliamente usado Concurrent Version system (CVS).

Subversión puede trabajar a través de redes, lo que permite que las personas que estén en diferentes computadores puedan usarlo, con la posibilidad de que varias personas modifiquen y gestionen el mismo conjunto de datos desde sus sitios promueve la colaboración, y como el trabajo está versionado, ya que si se produce algún cambio incorrecto de los datos, sólo hace falta deshacerlo.

## **2.6 PROGRAMACIÓN UTILIZADA**

Para el desarrollo de este proyecto se usó la Programación Orientada a Objetos (P.O.O.). La P.O.O. es una de las formas más populares de programas que usa objetos y sus interacciones para diseñar aplicaciones y programas de computador, intenta simular el mundo real a través del significado de objetos que contienen características y funciones; abstrae algunas características de sistemas naturales complejos como son:

- Atributos: estado del objeto.
- Métodos: comportamiento del objeto.
- Herencia: comportamientos comunes entre objetos relacionados para hallar relaciones de especialización y generalización de comportamientos.

### **2.6.1 Clases**

Definición de todos los elementos de que esta hecho un objeto. Cuando se programa un objeto y se definen sus características y funcionalidades, realmente se programa una clase. Por lo tanto para realizar la abstracción de sistemas naturales, observamos y analizamos un grupo de cosas con características comunes, el resultado de esta abstracción será válido para todas estas cosas.

### **2.6.2 Objetos**

Cualquier cosa real o abstracta, que posee atributos y un conjunto de operaciones que manipulan esos atributos que da un comportamiento particular. Un objeto es una instancia de una clase, el estado del objeto se determina por el estado (valor) de sus propiedades o características (atributos).

### **2.6.3 Atributos**

Características de un objeto siendo un conjunto de datos (valores) y calificadores para aquellos datos. Estos atributos pueden ser desde tipos de datos simples (enteros, caracteres, cadenas de texto) hasta otros objetos.

### **2.6.4 Métodos**

Son funciones o procedimientos propios de la clase que pueden tener acceso a los atributos de la misma para realizar las operaciones para los que son programados.

### **2.6.5 Herencia**

Se fundamenta en usar una clase ya creada para tomar sus características en clases más especializadas o derivadas de ésta para reutilizar el código que sea común con la clase base, y solamente definir nuevos métodos o redefinir algunos de los existentes para ajustarse al comportamiento particular de esta subclase.

### **2.6.6 Beneficios de la Programación Orientada a Objetos**

- Permite obtener aplicaciones modificables y fácilmente extensibles a partir de componentes reutilizables.
- Disminución en el tiempo de desarrollo gracias a la reutilización del código.
- El desarrollo del software es más intuitivo porque las personas piensan naturalmente en términos de objetos más que en términos de algoritmos de software.

A continuación se presenta una breve descripción de Java, el lenguaje de programación orientado a objetos que se usó en el desarrollo de este proyecto.

### 2.6.7 Java y JDK (Java Development Kit)

Java es un lenguaje desarrollado por Sun Microsystems, en el año 2009 fue adquirida por la compañía Oracle.

Permite escribir aplicaciones que puedan ejecutarse en casi cualquier plataforma. El lenguaje toma parte de la sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. Además, cuenta con una característica denominada “recolección de basura”, que examina la memoria y libera cualquier variable u objeto que no esté siendo usado. El JDK es un software que provee herramientas de desarrollo para la creación de programas en java.

Para trabajar con Java se necesita un kit de desarrollo que proporciona:

- Un compilador: *javac*
- Un intérprete: *java*
- Un generador de documentación: *javadoc*
- Un visor de applet para generar sus vistas previas, ya que un applet carece de método main y no se puede ejecutar con el programa java: Appletviewer.

## 2.7 SERVIDORES WEB

Es un tipo de software que se encuentra a la espera de una petición hecha por una aplicación cliente y da respuesta a dicha petición a través de una página web. Para cada transacción el servidor debe realizar dos acciones básicas: integrar todos los componentes de la página (texto, imágenes, vídeo, scripts, etc.) y

enviarla rápidamente al usuario. A continuación se describe el servidor Web que se ajusta a la tecnología escogida para el proyecto.

### **2.7.1 Servidor Jakarta Tomcat**

Servidor de aplicaciones Java basado en los estándares definidos por Sun Microsystems. Tomcat es desarrollado como parte del proyecto de código abierto Jakarta de la fundación de software Apache y es uno de los servidores de aplicaciones Java más utilizados, en especial porque es liviano, cumple con todos los estándares, sencillo de instalar, tiene muy buena documentación y es gratuito, además por ser escrito en Java funciona en cualquier sistema operativo que disponga de la Máquina Virtual de Java (JVM).

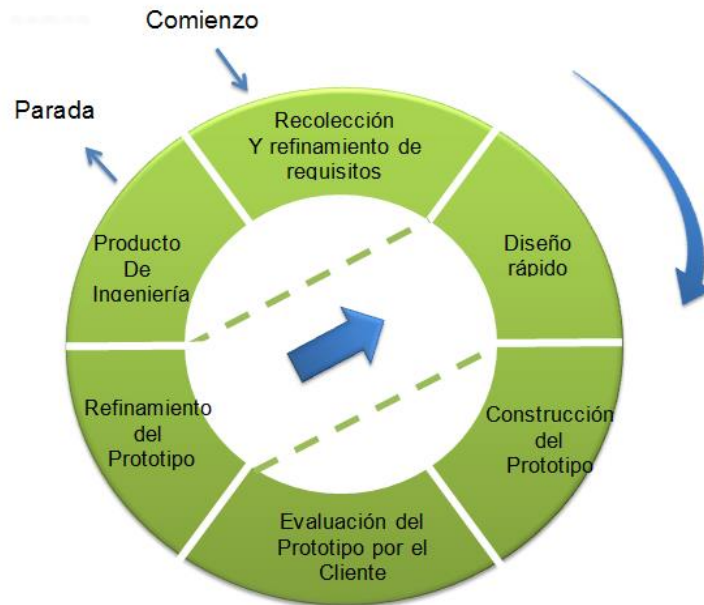
Es posible ejecutarlo desde la línea de comandos (consola o terminal), después de configurar algunas variables de entorno, sin embargo configurar cada variable de entorno y seguir los parámetros de las líneas de comando usados por Tomcat es tedioso y expuesto a errores, en su lugar se proporciona código existente para arrancar y detener el servicio.

### 3 MARCO METODOLÓGICO

#### 3.1 PROTOTIPO EVOLUTIVO

Para realizar los nuevos servicios para el Portal de la de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander se propone como metodología de desarrollo el prototipo evolutivo.

**Figura 3. Prototipo Evolutivo.**



Fuente: Autor.

La elección de esta metodología se debe a las siguientes razones:

- El portal EIMWEB presenta constantemente necesidades por parte de los usuarios, por lo tanto se requiere el desarrollo de nuevos servicios y así mismo la mejora de los ya existentes, debido a que el sistema no es un producto final, el mismo se considera en constante reconstrucción.

- El desarrollo de los aspectos visuales del sistema puede estar sujeto a cambios por parte de los usuarios del portal EIMWEB durante el desarrollo del proyecto.
- Durante la primera etapa de los nuevos servicios es necesario tener una clara interpretación de las especificaciones dadas por los usuarios y escuelas, pero muchas veces los usuarios no tienen una idea clara de lo que necesitan. Por esto la construcción de prototipos brinda la posibilidad de efectuar refinamientos de los requerimientos en forma sucesiva a fin de acercarse al producto deseado.
- Otra ventaja es tener la posibilidad de realizar cambios en etapas tempranas y crear varios prototipos evaluables durante el desarrollo, obteniéndose de este modo una metodología integral para el proceso de evaluación del programa.
- Esta metodología favorece la autocrítica al sistema, lo que conlleva a que se produzca muchas pruebas antes de dar un nuevo prototipo, así como mejoras rápidas a problemas que puedan surgir durante su uso.

**Procedimiento a seguir para la metodología planteada:**

- Se toma nota de los requisitos para la construcción de los prototipos.
- Se especifican en detalle los objetivos globales del software a realizar, mediante una reunión entre el desarrollador y el usuario, en la cual se identifican los requisitos básicos y se concluyen las áreas donde se necesita mayor definición.

- Posteriormente se presenta al usuario el diseño de un prototipo que se enfoca en los aspectos visuales del software, métodos de entrada y formatos de salida, para proceder a la construcción.
- El prototipo es evaluado por el usuario y se utiliza para filtrar los requisitos del software a desarrollar.
- Se produce un proceso interactivo en el que el prototipo es depurado para satisfacer necesidades del usuario, de igual forma el desarrollador obtiene una mejor comprensión de lo que hay que hacer para la entrega del producto final de ingeniería requerido por el usuario.

## **3.2 LENGUAJE DE MODELADO UNIFICADO**

El Lenguaje de Modelado Unificado o Unified Modeling Language (UML), es el más utilizado en la actualidad. Es un lenguaje gráfico estándar para visualizar, especificar, construir y documentar un sistema para describir un modelo del sistema, incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

### **3.2.1 Diagramas de UML**

Los diagramas UML utilizados en el desarrollo de este proyecto fueron: diagramas de casos de uso y diagramas de secuencias. Las principales razones por las cuales se prefirió UML como el lenguaje de modelado son:

- UML tiene una notación gráfica muy expresiva que permite representar todas las fases de un proyecto informático: desde el análisis con casos de uso, el diseño con diagramas de clases, objetos, etc.
- UML facilita el entendimiento de la información, la función y el comportamiento de un sistema, haciendo fácil el análisis de los requerimientos, ya que sirve de apoyo en los procesos de análisis de un problema.
- UML permite a los creadores de sistemas realizar diseños que faciliten la comunicación a otras personas de manera convencional.
- UML permite generar un punto de comparación entre lo logrado y lo planificado.

### **Diagramas de casos de uso**

Representación gráfica del entorno del sistema (actores) y su funcionalidad principal. Describe lo que hace el sistema desde el punto de vista de un observador externo, concentrándose en expresar lo que hace el sistema y no en dar respuesta de cómo lograr su comportamiento. (Ver Anexo D).

### **Diagramas de secuencias**

Es aquel que muestra la forma en que los objetos interactúan entre sí al transcurrir el tiempo. Consta de objetos que se representan del modo usual: rectángulos con nombre (subrayado), mensajes representados por líneas continuas con una punta de flecha y el tiempo representado como una progresión vertical. (Ver Anexo D).

### 3.3 ESTÁNDARES DE PROGRAMACIÓN

#### 3.3.1 Modelo de datos

Es un lenguaje utilizado para la descripción de una base de datos, por lo general permite describir estructuras de datos de la base de datos (el tipo de datos que incluye la base y la forma en que se relacionan), las restricciones de integridad (las condiciones que los datos deben cumplir para reflejar correctamente la realidad deseada) y las operaciones de manipulación de los datos (agregar, borrar, modificar).

#### 3.3.2 Nombres de las tablas

Los nombres de los campos, así como de las tablas de la base de datos, se escriben en minúsculas, exceptuando la primera letra de cada palabra que conforme su nombre; si es un nombre compuesto por dos o más palabras, los nombres tendrán en mayúscula la primera letra de cada palabra que la forma.

Se han definido tres categorías para las diferentes tablas que conforman la base de datos. Dada la categoría de la tabla, se antepondrá un prefijo a su nombre que permita conocer la categoría a la que pertenece. Las categorías son:

- *Tabla básica:* Aquella cuyos registros son necesarios para el correcto funcionamiento de la base de datos. Estas tablas no experimentan muchos cambios en los datos. El prefijo a anteponer a los nombres de estas tablas es “TB\_”, es decir la tabla que almacena el nombre del cargo de los miembros del consejo de escuela es llamada “TB\_NombreCargo”.
- *Tabla de Relación:* Surge de la relación muchos a muchos de una o dos tablas cualquiera. Los nombres de las tablas de relación deberán ser descriptivos

para cada relación. El prefijo a anteponer a los nombres de estas tablas es “TR\_”, es decir la tabla de “MiembrosConsejo” es conocida como “TR\_MiembrosConsejo”.

- *Tabla Principal:* Aquella cuya población de registros tiende a crecer en gran cantidad y que además no es posible clasificar como tabla básica o de relación. Un ejemplo de tabla principal es la tabla que almacena los usuarios del portal EIMWeb. El prefijo a anteponer a los nombres de estas tablas es “TP\_”, es decir la tabla “Usuarios”, es conocida como “TP\_Usuarios”.

### **3.3.3 Clases**

Los nombres de las clases deben ser sustantivos en plural, la primera letra de cada palabra debe ser mayúscula. Estos deben ser simples, descriptivos como por ejemplo: Usuarios.java.

### **3.3.4 Páginas JSP**

Los nombres de las paginas JSP que componen el portal EIMWEB serán escritos inicializando en mayúscula precedido de letras minúsculas, en caso de que el nombre del JSP compuesto por dos o más palabras, entonces la primera de cada palabra interna debe ir en mayúscula , por ejemplo: InscribirConsejoEscuela.jsp, PropuestasActuales.jsp

### 3.3.5 Organización de Directorios

Los directorios del sitio están organizados de tal manera que los archivos que se almacenen en ellos correspondan a lo que describe el nombre del directorio. Por ejemplo:

- El sitio cuenta con un directorio llamado “images”; en éste se encuentran almacenados todos los archivos .jpg, .gif, .png.
- Para el desarrollo de este proyecto se creó el directorio “EventosUis”, el cual almacena los archivos jsp correspondientes para el funcionamiento de dichos servicios.
- El sitio cuenta con un directorio llamado Script, donde están todos los archivos de JavaScript “.js” necesarios para el buen funcionamiento del portal.

## **4 DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO**

Como se mencionó anteriormente para el desarrollo de este proyecto se siguió la metodología de prototipo evolutivo.

Al iniciar el proyecto se elaboró un primer prototipo durante la fase de requerimientos, el cual fue mejorado con la inclusión de nuevos requerimientos surgidos en la fase de desarrollo, a medida que se generaba un prototipo, el mismo era sometido a pruebas de funcionamiento y se le realizaban los refinamientos pertinentes a partir del resultado de dichas pruebas.

### **4.1 PROTOTIPO ESPERADO**

Al inicio el proyecto no se tenía una concepción clara de cómo sería el producto final, sin embargo durante el desarrollo y evolución de los prototipos, las pruebas y análisis del sistema se pudo comprobar que se estaba acercando a los requerimientos iniciales, esto con el fin de enfocar exitosamente el desarrollo a la solución de las necesidades de los usuarios.

El objetivo específico inicial y los requisitos que surgieron se dieron gracias a la realización de prototipos y la realimentación con el cliente. Para cada objetivo se listan los requerimientos detallados de este, los cuales se cumplieron para el prototipo final.

- 1. Permitir la propuesta de eventos a la comunidad UIS de tal forma que puedan opinar y agregarse usuarios de diferentes escuelas, los eventos creados se guardaran como eventos propuestos permitiendo que los usuarios puedan participar en ellos ya sea comentando o uniéndose a la propuesta. Se enviaron los correos necesarios a los usuarios interesados en las propuestas.**

Objetivo Inicial:

- Permitir la propuesta de eventos a la comunidad UIS de tal forma que puedan opinar y agregarse usuarios de diferentes escuelas.

Requisitos finales del objetivo:

- Permitir la creación de eventos en la comunidad UIS
- Crear la opción de unirse a las propuestas vigentes.
- Los usuarios pueden comentar las propuestas que les sea de su interés
- Se enviaron correo a los interesados, este correo podrá ser a cualquier escuela.

- 2. El sistema de portales de las escuelas debe permitir enlazar los diferentes portales manteniendo la sesión (si la hay). El portal de la escuela debe enlazar a los profesores, a los grupos y al portal de P.I (Producción Intelectual). De manera sencilla se deben enlazar todos los portales para navegar por grupos, profesores y producción intelectual. Se debe mejorar la presentación de los perfiles de los profesores en su portal.**

Objetivo inicial:

- Creación del Portal de Producción Intelectual P.I y enlaces entre Portal Escuela, Grupos, Profesores y P.I (Producción Intelectual).

Requisitos finales del objetivo:

- Se debe crear un nuevo portal de Producción Intelectual en donde se puedan ver las Producciones de las diferentes escuelas, los grupos y los profesores.
- Se podrá acceder al portal de Producción Intelectual desde los otros portales sin perder la sesión. Estarán enlazados.

**3. El portal permite la subida de archivos en diferentes servicios condicionando, en la mayoría de casos, a que sean tipo .Zip.**

**Se debe realizar una reingeniería que permita subir diferentes tipos de archivos, entre estos: pdf, rar, Zip,... entre otros, dando flexibilidad y mejorando los servicios del portal.**

Objetivo inicial:

- Se debe permitir la subida de diferente tipo de archivos, dentro de ellos pdf, rar, otros, en los diferentes servicios del portal en los que se permita la subida de archivos.

Requisitos finales del objetivo:

- Se permite la subida de diferente tipo de archivos en los diferentes servicios del portal en donde se requiere la subida de archivos.
- Se permite la subida de archivos de diferente tipo como .zip, .rar, .xls, xlsx, doc, docx, ppt, pptx, pdf, y algunos de linux como .odt, .odp, .ods.

**4. Crear el servicio para eliminar, agregar y cambiar miembros del consejo. En el consejo de escuela y de facultad se debe crear un servicio que permita agregar nuevos usuarios que serán miembros del consejo, también eliminar un miembro existente o modificarlo.**

Objetivo inicial:

Crear el servicio para eliminar, agregar y cambiar miembros de los consejos de escuela en el módulo de soporte a los consejos de escuela y facultad.

Requisitos finales del objetivo:

- Este servicio permite la creación de los consejos de escuela y de facultad dando como opciones los diferentes usuarios que puedan hacer parte de los consejos dependiendo de sus cargos.
- Se permite la opción de eliminar los miembros actuales de los consejos de escuela y de facultad.
- Se puede modificar un consejo existente ya sea en su totalidad o diferentes miembros.
- Para el consejo de facultad se da la opción de crear manualmente al miembro del consejo, este tendrá un cargo específico.

**5. Dentro del módulo de soporte al Consejo de Escuela y Facultad es necesario crear el servicio que permita definir las fechas de inicio y fin de inscripción de solicitudes a los consejos por parte de los miembros de la escuela. Estas fechas varían de acuerdo al tipo de solicitud, y deben ser visibles a toda la comunidad.**

Objetivo inicial:

- Se debe crear el servicio que permita definir las fechas de inicio y fin, de acuerdo al tipo, para poder inscribir solicitudes a los consejos. Por fuera de estas fechas se le avisará al usuario de la imposibilidad para su realización. También es necesario crear el servicio para hacer visibles estas fechas.

Requisitos finales del objetivo:

- Se puede escoger el tipo de solicitud a la que se le desea modificar las fechas de inicio y fin.
- Se muestra a que punto del consejo en específico pertenece dicha solicitud.
- La disponibilidad de una solicitud podrá ser parcial o total, si es total no se asignan fechas de inicio ni de fin de la solicitud.

Para aquellos servicios que lo requieran, se debe incluir la rutina de login de usuario y verificación de sesión, requerimiento de seguridad intrínseco de los sitios web a los que presta sus servicios el grupo CALUMET.

### **Administración.**

Objetivos iniciales:

- Efectuar labores de administración del portal web de la Escuela de Ingeniería Mecánica; teniendo en cuenta entre otras:
  - Generar Backups (copias de respaldo) diariamente de la Base de Datos.
  - Hacer seguimiento del uso que hacen los usuarios en cuanto a los servicios de Foros, Eventos Propuestos, Cartelera, Archivos y Mi perfil dentro del portal EIMWeb para detectar usos indebidos o incorrectos por parte de éstos.
  - Actualizar periódicamente las Bases de Datos con el objeto de mantener al día la información referente a matrículas, horarios, estados y categorías de los usuarios.

- Atender consultas y sugerencias que los usuarios hagan para proponerlas como mejoramiento y ofrecimiento de nuevos servicios en la próxima versión del portal web.
- Realizar una revisión constante de los archivos que se suben al sitio, eliminando los que no son necesarios para evitar saturación del portal.

Requisitos finales del objetivo:

Para usuarios con perfil de administrador:

- Corregir cada uno de los fallos que a diario se presentan en los portales web.
- Mantenimiento a la bases de datos para eliminar tablas que ya no se usan o crear nuevas que se necesitan para que los nuevos servicios funcionen.
- Actualizar el diagrama Entidad/Relación de la base de datos Diamante y subirlo al portal EISIWeb, en el espacio del grupo Calumet; para realizar futuras actualizaciones sobre estos últimos.
- Revisar la estructura de directorios del portal, borrar los archivos que ya no se usan y agregar nuevos servicios implantados en el portal.
- Realizar limpiezas en cuanto a conversaciones y datos que ya no sean necesarios dentro de las Bases de datos.
- Mantener los JavaBeans actualizados y subirlos al portal EISIWeb, en el espacio del grupo Calumet; para realizar futuras modificaciones sobre estos últimos.

Cada una de las actividades mencionadas en el objetivo inicial se llevó a cabo durante la práctica y desarrollo del proyecto.

## **Mantenimiento.**

Objetivos iniciales:

- Llevar a cabo labores de mantenimiento al portal EIMWeb en las que podemos encontrar:
  - Hacer el seguimiento del funcionamiento de los portales para corregir posibles defectos generados por errores en el código fuente que se puedan presentar y dañar el funcionamiento del sistema.
  - Revisar y depurar la estructura de directorios y archivos del portal web.

Requisitos finales del objetivo:

Para usuarios con perfil de administrador:

- Revisar regularmente el historial de cambios para supervisar el uso correcto que los usuarios le dan a los servicios.
- Realizar copias periódicas de las bases de datos Division y Diamante y de los directorios del sitio EIMWEB.
- Eliminar de la base de datos diamante información que no se usa, ya sean archivos o conversaciones.
- Hacer la actualización periódica de la base de datos Diamante.

Cada una de las actividades mencionadas en el objetivo inicial se llevó a cabo durante la práctica.

## **Soporte a Usuarios.**

Objetivo inicial:

- Desempeñar labores de soporte a los usuarios del portal EIMWeb, brindando así solución a sus diferentes necesidades, conflictos o situaciones que se puedan presentar. Entre estas labores se destacan:
  - Capacitar usuarios y estudiantes del primer nivel en el uso de servicios dentro del portal EIMWeb, promoviendo así su utilización.
  - Atender usuarios por olvido de la contraseña, creación de grupos, solicitudes de propuestas de eventos, solicitudes de cartelera y creación de agendas con eventos y foros.
  - Crear usuarios de forma manual para personas no pertenecientes a EIMWeb y que por algún motivo necesitan registrarse en el sitio web correspondiente.
  - Modificar los estados de los usuarios de acuerdo a la relación con la escuela (activo, inactivo, suspendido).

Requisitos finales del objetivo:

Para usuarios con perfil de administrador:

- Conocer el total funcionamiento del portal EIMWEB, a fin de dar solución a las posibles dificultades que se presenten.
- Investigar las soluciones a nuevas dificultades que se presenten.

Cada una de las actividades mencionadas en el objetivo anterior se llevó a cabo durante la práctica en repetidas ocasiones. Además, se dio orientación a los usuarios sobre el uso de determinados servicios.

## **Capacitación a nuevos integrantes del grupo Calumet.**

Objetivo inicial:

- Capacitar a los estudiantes que relevaran las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro del portal EIMWEB en cuanto a:
  - Implantación del sitio local para la creación de nuevos servicios y realización de pruebas.
  - Realizar inducción en cuanto al manejo y utilización de los JSP, JavaBeans y Base de Datos.
  - Llevar a cabo la familiarización con el entorno de los portales web.

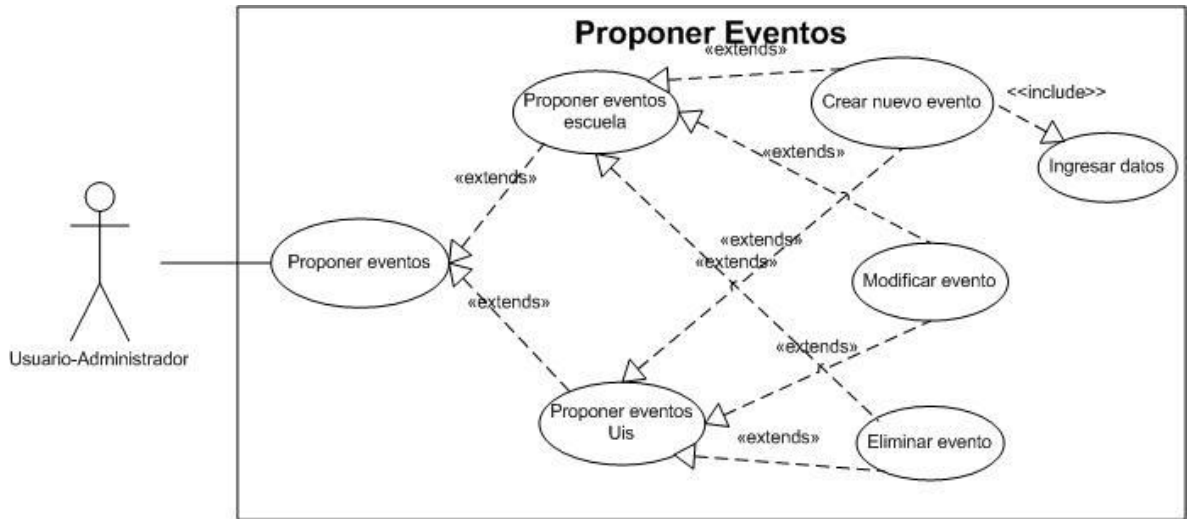
Requisitos finales del objetivo:

- Para usuarios con perfil de administrador.
- Dar orientación a los nuevos integrantes del grupo sobre posibles errores en la instalación del sitio local.
- Dar orientación básica a los nuevos integrantes del grupo sobre la programación en JSP, uso de los JavaBeans y funcionalidad de cada tabla dentro de la base de datos.
- Dar capacitación a los nuevos integrantes del grupo sobre los estándares que se siguen para la programación en cuanto a nombres y estilos.
- Orientar a los nuevos integrantes sobre el procedimiento para crear y habilitar nuevos servicios en el sitio.

## 4.1.1 Diagramas de Casos de Uso

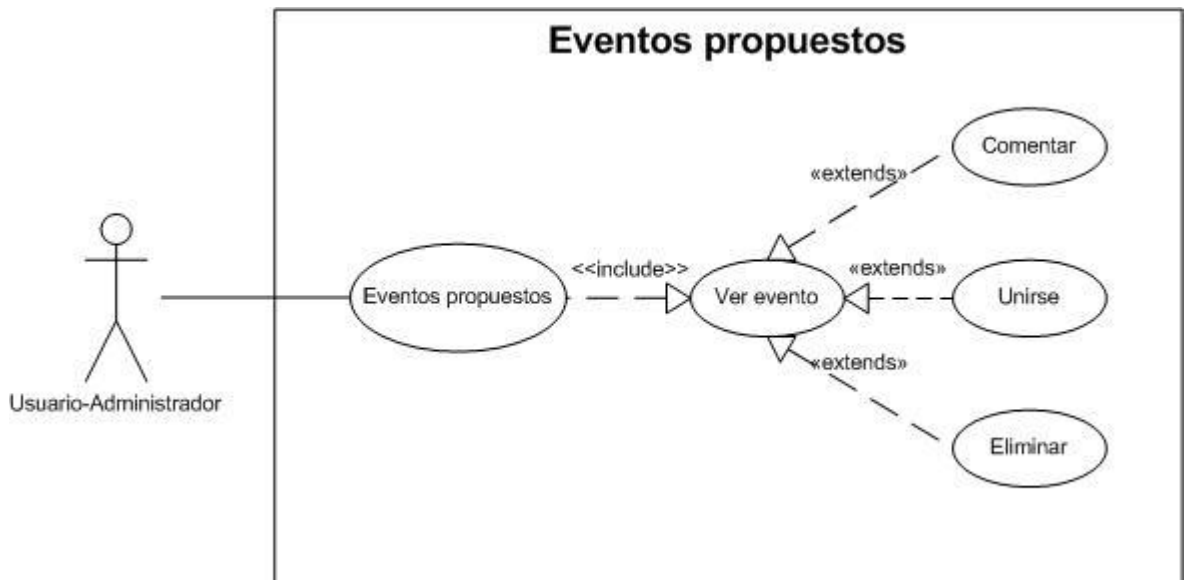
### 4.1.1.1 Eventos comunidad Uis

Figura 4. Diagrama de Casos de Uso: Proponer eventos.



Fuente: Autor.

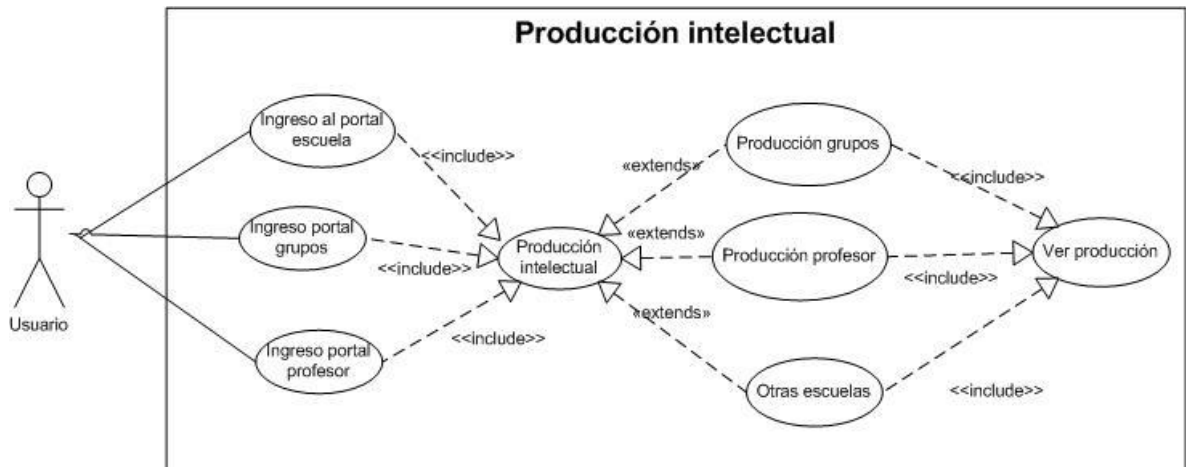
Figura 5. Diagrama de Casos de Uso: Eventos propuestos.



Fuente: Autor.

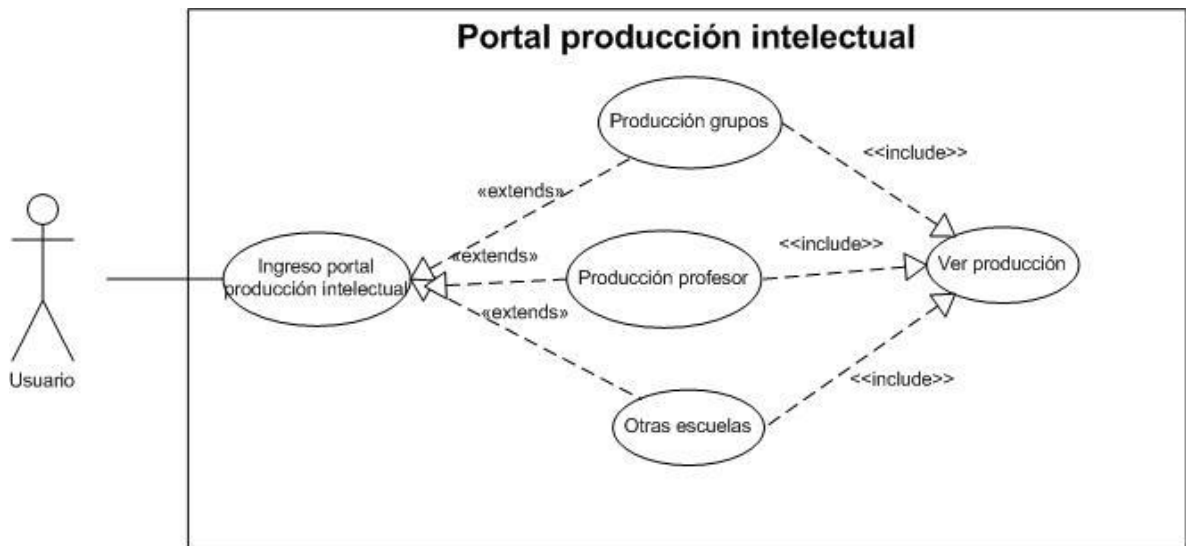
#### 4.1.1.2 Casos de uso de Portal de producción intelectual PI

Figura 6. Diagrama de Casos de Uso: Producción intelectual.



Fuente: Autor.

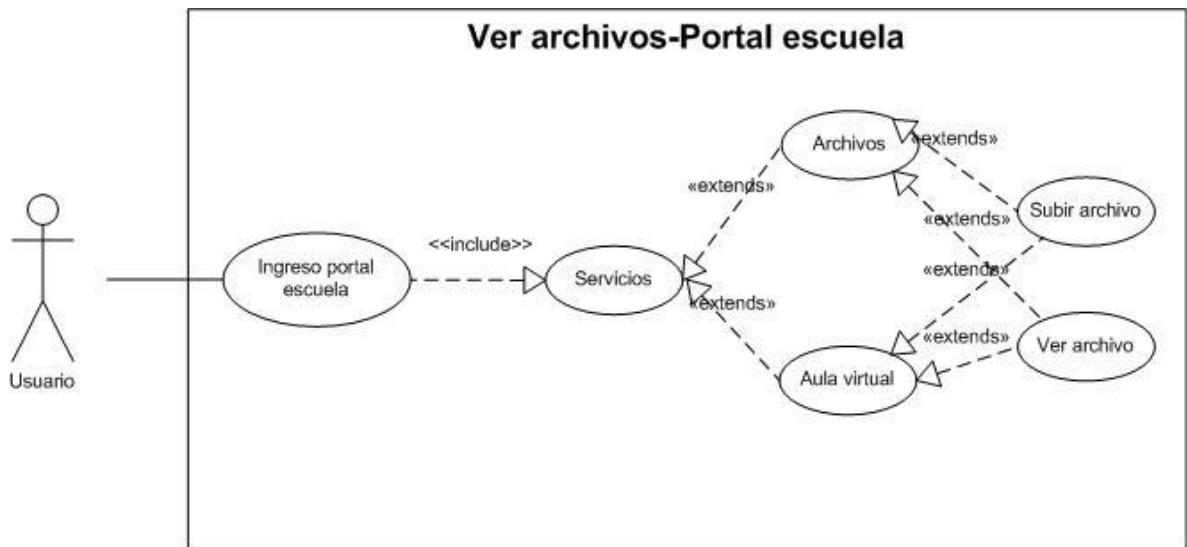
Figura 7. Diagrama de Casos de Uso: Portal Producción intelectual.



Fuente: Autor.

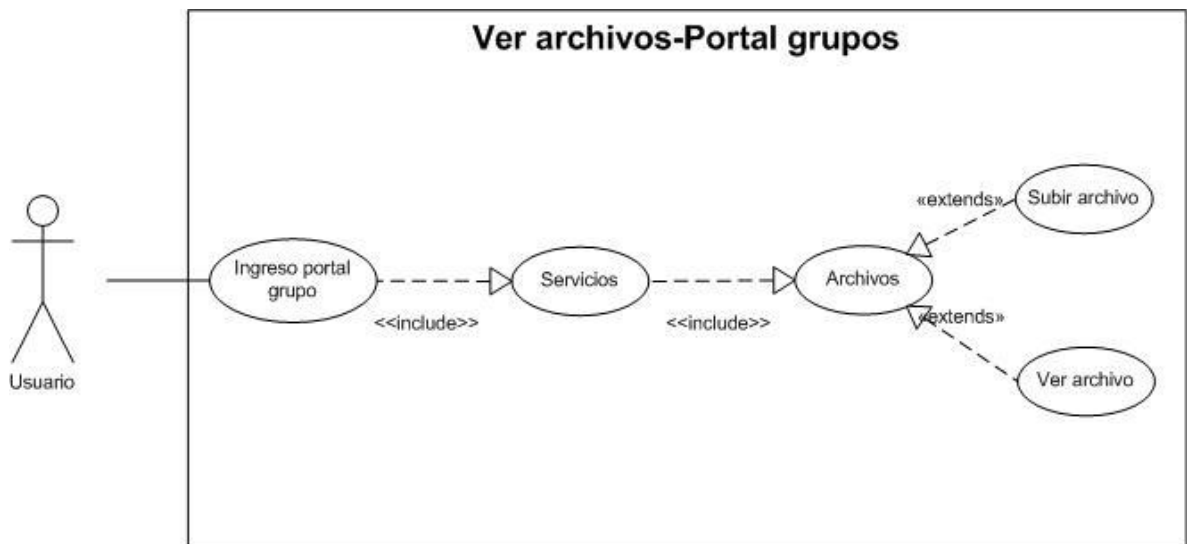
#### 4.1.1.3 Casos de uso: Subida de archivos de diferente tipo.

Figura 8. Diagrama de Casos de Uso: Servicios-Archivos. Portal escuela.



Fuente: Autor.

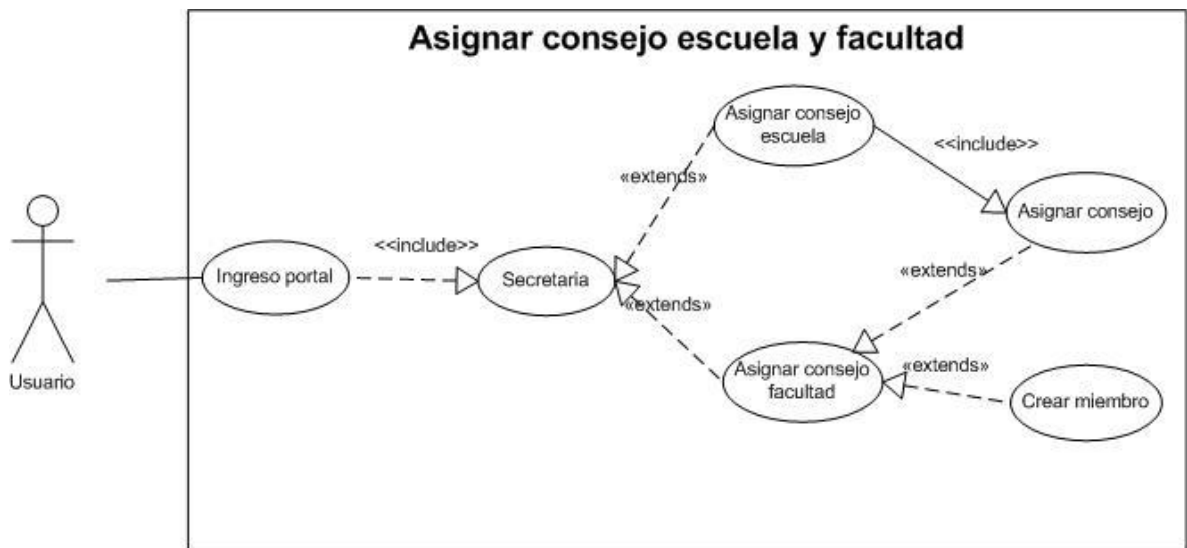
Figura 9. Diagrama de Casos de Uso: Servicios-Archivos. Portal grupos.



Fuente: Autor.

#### 4.1.1.4 Casos de uso: Crear, modificar o eliminar consejo escuela y de facultad.

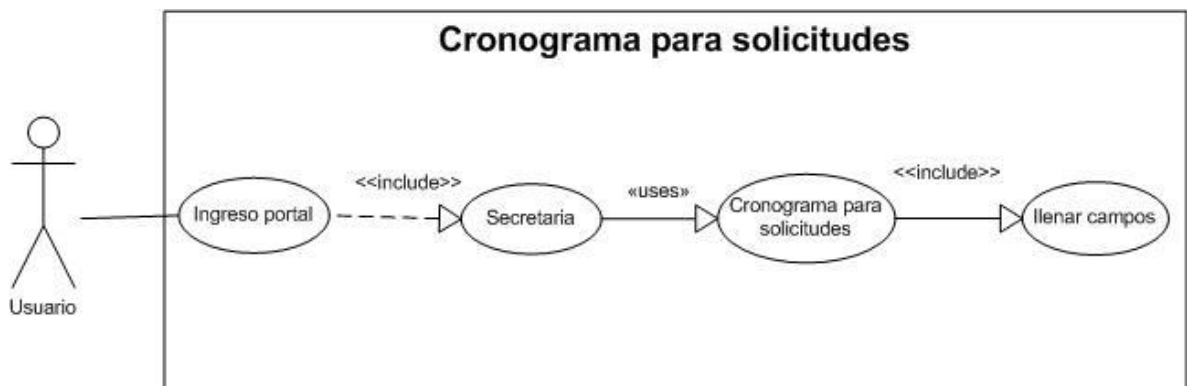
Figura 10. Diagrama de Casos de Uso: Asignar consejo.



Fuente: Autor.

#### 4.1.1.5 Casos de uso: Cronograma de solicitudes.

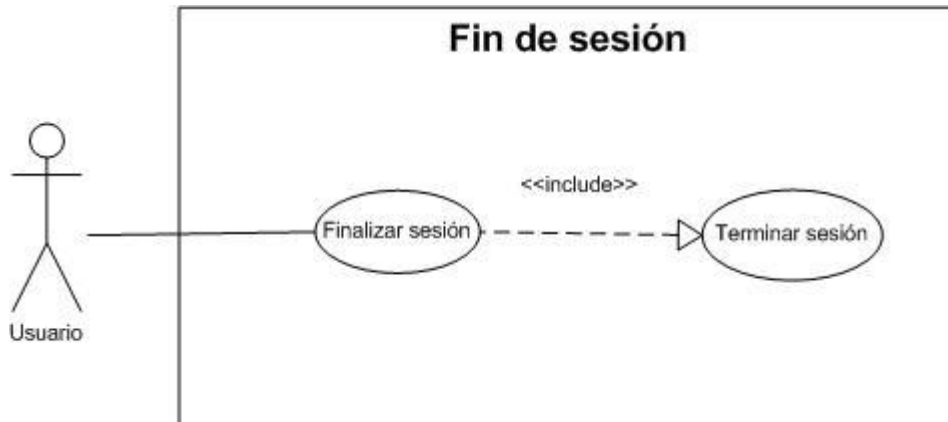
Figura 11. Diagrama de Casos de Uso: Cronograma.



Fuente: Autor.

#### 4.1.1.6 Casos de uso: Fin de sesión.

Figura 12 Diagrama de Casos de Uso: Fin de sesión.



Fuente: Autor.

#### 4.1.1.7 Casos de uso: Mantenimiento y Administración del portal EIMWEB.

Figura 13 Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.



Fuente: Autor.

#### 4.1.2 Documentación de Casos de Uso del Sistema

##### 4.1.2.1 Eventos comunidad UIS

**Tabla 1. Caso de Uso: Proponer eventos.**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Proponer eventos	Proponer eventos dirigidos a la escuela o a la comunidad UIS.
Proponer eventos escuela	Se listan los eventos propuestos para la escuela.
Proponer eventos UIS	Se listan los eventos propuestos a la comunidad UIS.
Crear nuevo evento	Se crea el evento.
Modificar	Se modifica un evento creado.
Eliminar Evento	Se elimina un evento creado.

**Tabla 2. Caso de Uso: Eventos propuestos.**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Eventos propuestos.	Acceder a los eventos que se encuentran propuestos por diferentes usuarios.
Ver evento	Se ve toda la información del evento creado.
Comentar	Se hacen comentarios sobre un evento específico
Unirse	Se une el usuario a una propuesta que sea de su interés.

#### 4.1.2.2 Producción Intelectual.

**Tabla 3. Caso de Uso: Portal escuela.**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Ingreso al portal escuela	Permite ingresar al portal de la escuela.
Ingreso al portal grupos	Permite ingresar al portal de los grupos.
Ingreso al portal profesor	Permite ingresar al portal del profesor.
Producción intelectual	Se accede a las producciones intelectuales.
Producción grupos	Se accede a las producciones intelectuales del grupo.
Producción profesor	Se accede a las producciones intelectuales del profesor.
Otras escuelas	Se accede a las producciones intelectuales de diferentes escuelas.
Ver producción	Se ve en detalle la información de la producción escogida.

**Tabla 4. Caso de Uso: Portal Producción Intelectual.**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Ingreso portal producción intelectual	Permite ingresar al portal de producción intelectual.
Producción grupos	Se accede a las producciones intelectuales de los diferentes grupos.
Producción profesor	Se accede a las producciones intelectuales de los profesores.

Otras escuelas	Se accede al portal de producción intelectual de otras escuelas.
Ver producción	Se ve en detalle la información de la producción escogida.

#### 4.1.2.3 Subida de archivos de diferente tipo.

**Tabla 5. Caso de Uso: Servicio archivos. Portal escuela.**

TÍTULO	FUNCIONES PRIMARIAS
Ingreso al portal escuela	Permite ingresar al portal de la escuela abriendo la sesión del usuario.
Servicios	Menú para acceder a los diferentes servicios.
Archivos	Se ven las opciones de subir o ver archivos.
Aula virtual	Se accede al aula virtual escogida para poder subir o ver los archivos.
Subir archivos	Se subirá un archivo con las especificaciones y datos requeridos.
Ver archivos	Se consultan y descargan archivos.

**Tabla 6. Caso de Uso: Servicio archivos.**

TÍTULO	FUNCIONES PRIMARIAS
Ingreso al portal grupo	Permite ingresar al portal del grupo.
Servicios	Menú para acceder a los diferentes servicios.
Archivos	Se ven las opciones de subir o ver archivos.
Subir archivos	Se subirá un archivo con las especificaciones y datos requeridos.

Ver archivos	Se consultan y descargan archivos.
--------------	------------------------------------

#### 4.1.2.4 Crear, modificar o eliminar consejo escuela y de facultad.

**Tabla 7. Caso de Uso: Asignar consejo.**

TÍTULO	FUNCIONES PRIMARIAS
Ingreso al portal	Permite al usuario iniciar sesión para poder acceder a este servicio.
Secretaria	Menú para acceder a los diferentes servicios.
Consejo escuela	Se accede a las opciones del consejo de escuela.
Consejo facultad	Se accede a las opciones del consejo de facultad.
Asignar consejo	Se asignan los miembros del consejo.
Crear miembro	Se crea un nuevo miembro para el consejo de escuela.

#### 4.1.2.5 Cronograma de solicitudes.

**Tabla 8. Caso de Uso: Cronograma.**

TÍTULO	FUNCIONES PRIMARIAS
Ingreso al portal	Permite ingresar al portal del usuario.
Secretaría	Menú para acceder a los diferentes servicios.
Cronograma para solicitudes	Se accede al servicio en donde se define el cronograma para las diferentes solicitudes.
Llenar campos	Llena los campos necesarios para hacer valido el servicio.

#### 4.1.2.6 Mantenimiento y Administración del portal EIMWEB.

**Tabla 9. Caso de Uso: Mantenimiento y Administración del portal EIMWEB.**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>
Administrar Servicios	Crear, modificar y eliminar servicios del portal. Establecer parámetros.
Administrar Sugerencias	Revisar todas las sugerencias para atender o eliminar.
Administrar Varios	Ejecutar labores de mantenimiento que no clasifican en otras categorías. Llevar el contador de visitas a cero. Insertar, cambiar y eliminar banner.
Administrar Foros	Ver información y eliminar conversaciones de los usuarios en foros.
Administrar Usuarios	Crear usuarios manualmente. Ver información de usuarios. Cambiar parámetros. Cambiar clave. Autorizar servicios a categorías y perfiles. Crear, modificar, eliminar categorías y perfiles. Enviar Correos.
Administrar Ayudas	Crear, modificar y eliminar las ayudas en el portal de ayudas de las escuelas.
Administrar Parámetros	Crear, modificar y eliminar los diferentes parámetros del portal.
Backups	Generar copias de la base de datos División y Diamante.
Administrar Archivos	Ver información, descargar y eliminar archivos cargados por los usuarios.

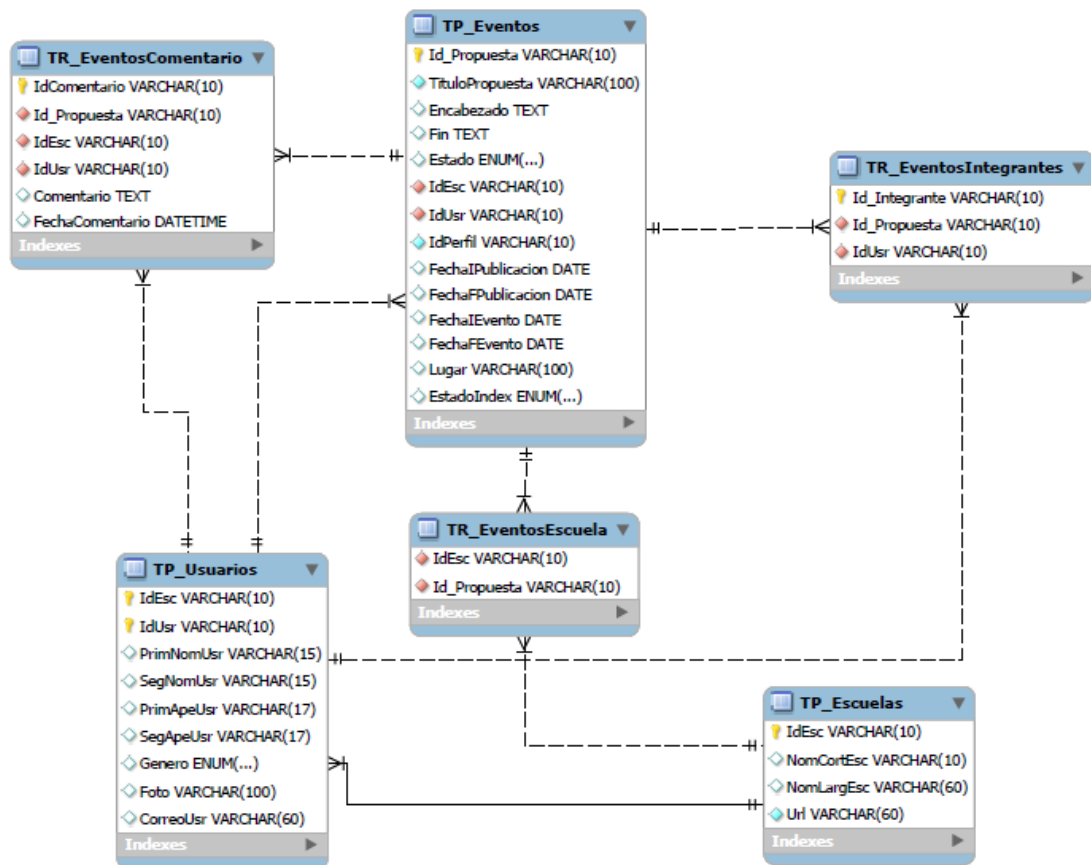
Administrar Horarios	Cambiar, restaurar, ver incoherencias y cambios de horarios. Crear grupo de materia.
Administrar Actualizaciones	Ver historial y estadísticas de cambios del portal web.
Actualizar Base de Datos	Actualizar tablas de datos de División y Diamante. Ver contadores de registros.

Fuente: Autor.

### 4.1.3 Diseño y Análisis.

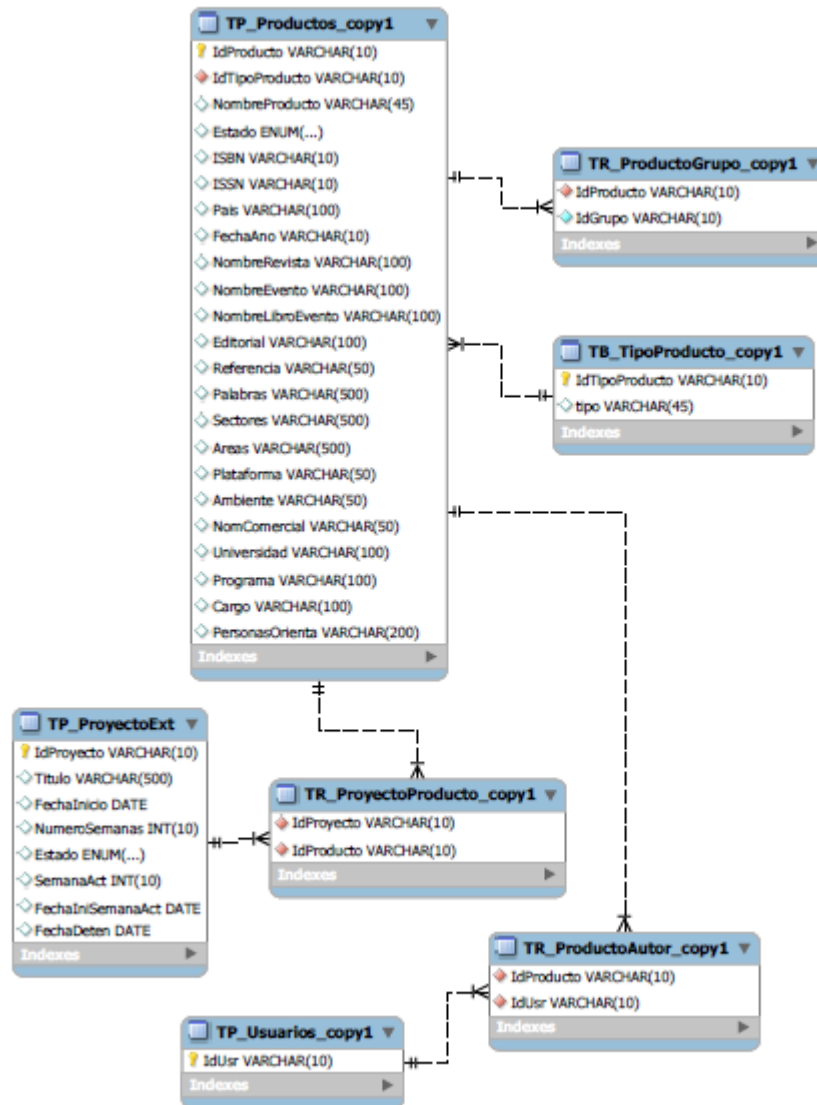
#### Diagramas Entidad/Relación de los servicios desarrollados

Figura 14 Diagrama E/R: Eventos comunidad UIS.



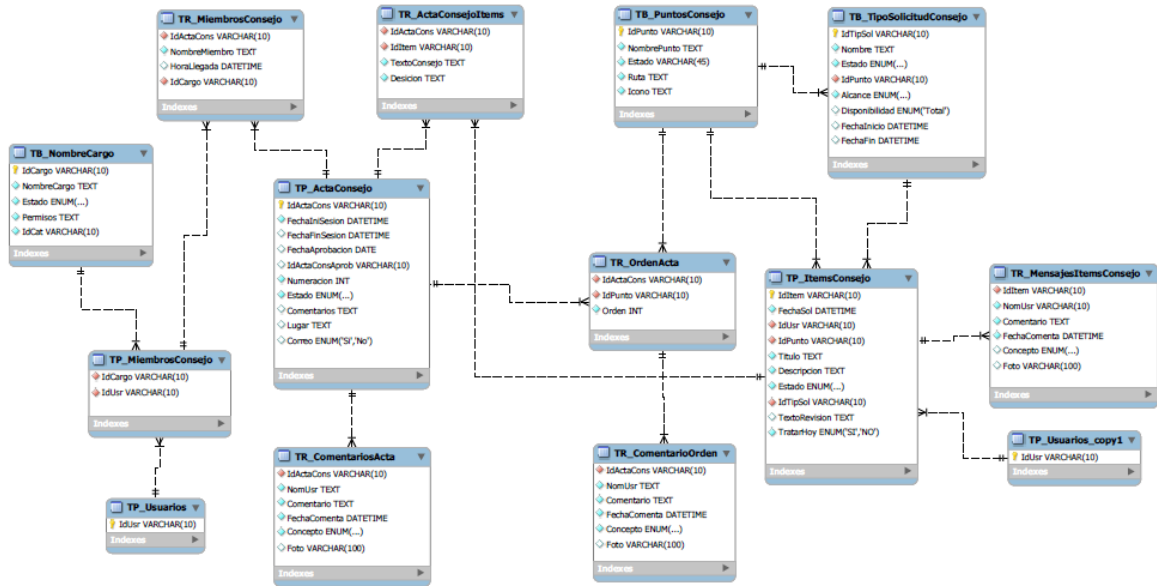
Fuente: Autor.

Figura 15 Diagrama E/R: Producción Intelectual.



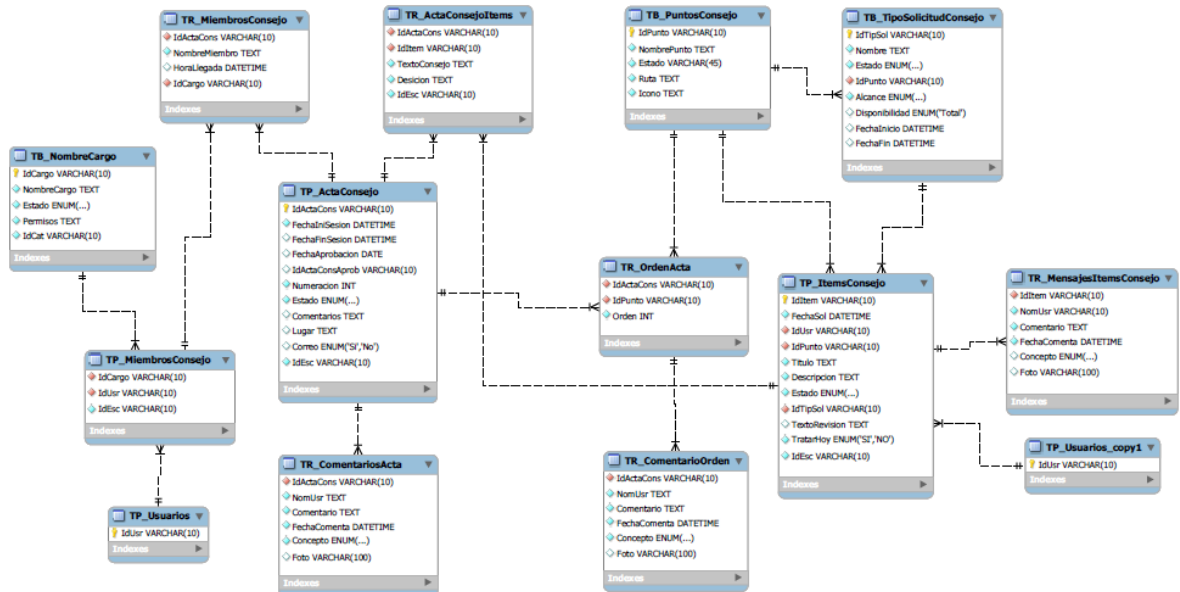
Fuente: Autor.

Figura 16 Diagrama E/R: Crear, modificar o eliminar consejo escuela.



Fuente: Autor

Figura 17 Diagrama E/R: Crear, modificar o eliminar consejo de facultad.



Fuente: Autor

## Descripción de las Entidades

Definición de las entidades utilizadas y creadas en la base de datos Poseidón ubicada en el servidor de la escuela de Ingeniería Civil y de la base de datos Diamante ubicada en el servidor de los portales de las escuelas de ingeniería mecánica y de sistemas.

**Tabla 10. Caso de Uso: Descripción de las Entidades**

ENTIDAD	DESCRIPCIÓN
TP_Usuarios	Contiene toda la información de los usuarios registrados.
TP_Escuelas	Contiene toda la información necesaria de las escuelas.
TP_Eventos	Contiene toda la información del evento creado.
TR_EventosComentario	Contiene los comentarios hechos en un evento específico.
TR_EventosIntegrantes	Contiene la información de los usuarios integrantes de un evento.
TR_EventosEscuela	Relaciona los eventos con las escuelas.
TR_CorreoDatosBusqueda	Almacena la información necesaria para envío de correos.
TP_Productos	Contiene toda la información de un producto.
TB_TipoProducto	Contiene el tipo de producto.
TR_ProductoGrupo	Contiene la información necesaria para saber a qué grupo pertenece una producción.

TR_ProductoAutor	Contiene la información necesaria para saber a qué usuario pertenece una producción.
TB_NombreCargo	Contiene los nombres de los cargos de los miembros del consejo.
TP_MiembrosConsejo	Contiene los miembros del consejo.
TR_MiembrosConsejo	Relaciona los miembros del consejo con su participación en un acta.

Fuente: Autor.

#### **4.1.4 Modelo de procesos del sistema.**

Para una mejor interpretación de los modelos de procesos del sistema, se realizaron los diagramas de secuencias necesarios para cada caso de uso en los que se explica con detalle los pasos para el funcionamiento de cada uno de los servicios. (Ver Anexo D)

#### **4.1.5 Implementación, Implantación y Pruebas Generales**

Para la implementación de los servicios se utilizaron las siguientes herramientas:

- Lenguaje de marcado para la elaboración de páginas web, HTML.
- Lenguaje de programación orientado a la web, JSP.
- Lenguaje de programación interpretado orientado a objetos, JavaScript.
- Hojas de estilo en cascada, CSS.
- Lenguaje Java.
- NetBeans, IDE para desarrollar las clases de Java y los archivos JSP.
- Servidor Jakarta Tomcat.
- Manejador de Base de datos, MySQL 5.0.

Haciendo uso de las anteriores herramientas, y con la asesoría y seguimiento del director de proyecto se diseñaron los nuevos servicios y se estructuraron los datos para el primer prototipo. También se tomaron en cuenta sugerencias recibidas por miembros del Grupo Calumet, para realizar un posterior refinamiento a las interfaces.

Para el desarrollo se trabajó con la base de datos "Diamante" ubicada en todos los servidores de los portales de las escuelas, y con la base de datos centralizada "Poseidón" ubicada en el servidor del portal web de la Escuela de Ingeniería Civil. En las cuales se crearon las tablas necesarias para el funcionamiento de los servicios.

Se llevaron a cabo pruebas para cada subsistema propuesto, verificando que los resultados correspondieran con lo esperado, de esta forma, se evidenció el correcto funcionamiento en la captura de datos, selección de ítems y almacenamiento de información.

La implantación de los servicios se realizó en primer lugar en el portal web de pruebas del Grupo Calumet, donde se realizan las pruebas a todos los componentes. Finalmente se implementan en todos los portales web a los cuales el Grupo Calumet presta soporte.

## **4.2 MANTENIMIENTO Y ADMINISTRACIÓN**

### **4.2.1 Actividades de Mantenimiento**

Dentro de las funciones que se realizan en la administración y mantenimiento de los servicios de los portales web se encuentra la tarea de corregir errores, las cuales se denominan incidencias y que se corrigen durante la primera fase como

integrantes del Grupo Calumet. Las siguientes son las incidencias que se realizaron paralelamente al desarrollo de este proyecto.

- **Ventana emergente al actualizar la BD Diamante**

#### **Funcionalidad:**

El administrador del portal puede realizar labores de actualización de las BD, que consiste en actualizar la categoría y el estado de los usuarios, por una parte, y por otra actualizar la matrícula de los estudiantes y los horarios de los profesores.

#### **Desarrollo del problema:**

Se mejoró el código para que el administrador pueda realizar sus actualizaciones de la base de datos Diamante indicando con un mensaje cuando termino dicho proceso, incluyendo un reporte de la actualización.

#### **Problemática:**

Estas actualizaciones requieren de una ventana emergente que informe sobre la finalización del proceso y los posibles cambios ocurridos.

#### **Descripción:**

Se trabajó sobre el archivo `/eisi/web/Administrador/adminActEisi.jsp` Se hizo una consulta del método `GetContadoresBDdiamante` que se creó en el java llamado `Materias.java`. Se incluyó un alert que va a indicar cuando la actualización se termina y enseguida de este con consultas a las BD sale un reporte de la actualización.

- **Enviar correo a los directores informando del evaluador y/o calificadores asignados.**

#### **Funcionalidad:**

El comité de trabajos de grado, en su sesión, asigna evaluadores y calificadores a los proyectos. La solicitud de evaluador o calificadores la realizan los directores de proyectos. Esta información puede ser consultada al día siguiente en el Acta Actual, además se envían correos a los usuarios implicados en las decisiones del comité.

#### **Desarrollo del Problema:**

Cuando se reúne el comité de trabajos de grado y toma alguna decisión sobre calificadores y/o evaluadores no se le estaba enviando un correo a los directores. Ahora cada vez que se ingresa al acta actual se envían los mensajes respectivos.

#### **Problemática:**

El sistema no incluye enviar correos a los directores de trabajos de grado informando del evaluador o calificadores asignados por el comité. Deben agregarse estos correos al sistema para que los directores se enteren al respecto.

#### **Descripción:**

Se trabajó sobre el archivo `/eisi/web/TrabajoGrado/Consultas/ConsultarActaActual.jsp`. Se hicieron las consultas necesarias a las BD para obtener el correo del director y poder enviarle el mensaje. Se utilizó el método `enviarcorreo` de `Email.java` para enviar el mensaje en donde se le indica al director quien o quienes son los calificadores o evaluador.

- **Cambiar nombre de archivos en el Libro de Aula**

#### **Funcionalidad:**

El Aula virtual permite la creación de un libro que desarrolla el contenido de las materias por unidad y temática. A cada temática se pueden subir diferentes tipos de archivos.

#### **Desarrollo del Problema:**

Se cambió el nombre de los archivos al subirse al libro del aula. Ahora sale la fecha, la hora y un texto de máximo 15 caracteres el cual no contiene tildes, ñ y demás.

#### **Problemática:**

Se debe cambiar el nombre de los archivos, por uno que contenga la fecha hora de la subida y un texto sobre el archivo

#### **Descripción:**

Para solucionar esta incidencia se hicieron cambios en listalmagenes.jsp y verServidor.jsp de varias carpetas llamadas animación, imagen, pdf, presentación, simulador, video; estas carpetas se encuentran en OVA/AdminArchivos. Para reemplazar el texto y quitar las tildes y demás caracteres que no se puedan subir se hizo una función JavaScript llamada replaceAll en todos los verservidor.jsp.

- **Agregar número fecha de última descarga de un archivo del aula**

**Funcionalidad:**

El aula permite compartir archivos a sus usuarios de manera sencilla, para esto existen los servicios de subir y ver archivos dentro de ella.

**Desarrollo del Problema:**

En el aula se pueden compartir archivos. Se modificó para que aparezca en la información del archivo la fecha de la última descarga y la cantidad de veces que se ha descargado ese archivo.

**Problemática:**

Se hace necesario llevar el número de veces que ha sido descargado un archivo, y la última vez que se descargó.

**Descripción:**

Se trabajó sobre los jsp llamados ArchivosVerClasesGrupo, ArchivosVerClasesGuion y ArchivosVerClasesProfesor que están localizados en la carpeta Aula/ArchivosAula. Se modificó una consulta para hacer lectura de dos campos que son la última descarga y la fecha de esta. Se agregó al final de los jsp la función ajax para que actualice la fecha y el número de descargas.

**4.2.2 Actividades de Soporte a Usuarios**

Las escuelas cuentan en sus portales con un servicio de consultas y sugerencias, a través del cual se responden preguntas y se resuelven problemas de los

usuarios. Diariamente se reciben consultas de estudiantes y profesores, que necesitan asesoría en el funcionamiento de servicios o soporte en el manejo de su usuario y contraseña. También se resuelven problemas de manera presencial, donde el usuario acude directamente a las oficinas del Grupo Calumet y es atendido directamente por alguno de sus miembros, que le prestan la asesoría necesaria. Algunas de las labores realizadas comúnmente son:

- Restablecimiento de contraseña a usuarios.
- Dar respuesta y solución a las inquietudes y problemas que tengan los usuarios de los portales respecto al uso de los servicios.
- Facilitar orientación a los estudiantes nuevos de las escuelas a las cuales presta sus servicios el grupo Calumet, en cuanto al registro en el portal y el uso de sus principales servicios.

#### **4.2.3 Actividades de Administración**

Dentro de las actividades realizadas por los miembros del Grupo Calumet, se encuentra la tarea de administrar los servidores de los portales de las escuelas, cada semestre se asignan nuevos administradores, los cuales cuentan con un usuario dentro de uno de los portales a los cuales presta soporte el grupo. Como administrador las tareas que se realizan comúnmente son:

- Realizar periódicamente copias de las bases de datos Diamante y División.
- Realizar copias de seguridad diarias de las bases de datos Diamante y División.

- Dar aval a las solicitudes de publicación de los usuarios en la cartelera para que puedan ser accedidas por la comunidad.
- Atender las sugerencias hechas por los usuarios del sistema a través del servicio de consultas y sugerencias.
- Actualizar periódicamente las bases de datos con respecto a la información que ofrece la División de Servicios de Información de la Universidad Industrial de Santander, para que el portal cuente con información actualizada.

## **5 PRUEBAS DEL SISTEMA**

Para garantizar el correcto desarrollo de los servicios realizados, se realizaron las siguientes pruebas:

### **5.1 PRUEBAS DE VERIFICACIÓN**

Esta prueba es una de las más prácticas y más utilizadas mediante la cual se aplican diferentes técnicas para detectar errores en el sistema antes de ser utilizado.

Se efectúa elaborando actividades para cada servicio desarrollado, realizando pruebas como: verificar que los campos que contienen datos obligatorios no queden vacíos, o examinar que los campos contengan el tipo de dato que le corresponde.

#### **5.1.1 Pruebas por componente**

Esta prueba se realizó para cada caso de uso de cada servicio desarrollado y descrito anteriormente. De igual manera en las mejoras a los servicios ya existentes.

Los resultados obtenidos con la realización de las pruebas, evidencian el cumplimiento de los objetivos propuestos, en la medida en que los servicios y mejoras funcionan correctamente. (Ver anexo E).

## **5.2 PRUEBAS DE INTEGRACIÓN**

Una vez creados todos los servicios se verificó que cada uno funcionara correctamente y no interfirieran con el funcionamiento del portal. Se verificó que las consultas a la base de datos se hicieran de manera correcta, que en el sistema se visualizaran correctamente los servicios y que cada tipo de usuario tuviera acceso a ellos dependiendo del perfil y categoría al que está asociado.

## **5.3 PRUEBAS DE VALIDACIÓN**

Con el fin de probar el funcionamiento de cada servicio implementado se realizaron pruebas de validación elaboradas por integrantes del grupo Calumet, administrativos de las escuelas y el director de proyecto. Para algunos de los servicios observamos irregularidades las cuales fueron corregidas durante la marcha hasta obtener los resultados requeridos.

## 6 CONCLUSIONES

- La creación del servicio eventos comunidad UIS, hace posible la interacción de las diferentes escuelas en diferentes actividades. Se crea un espacio que le permite al usuario acceder a todos los eventos que se puedan estar llevando a cabo en las diferentes escuelas así como crear nuevos eventos, además podrá opinar o unirse a la propuesta si así lo desea.
- Con la creación del nuevo portal de producción intelectual se da la posibilidad de dar a conocer los diferentes proyectos ya sean personales como los de los profesores o de las escuelas y sus diferentes grupos.
- En diferentes servicios del portal se permite a los usuarios subir archivos de diferente tipo de extensión tales como: zip, .rar, .xls, xlsx, doc, docx, ppt, pptx, pdf, y algunos de linux como .odt, .odp, .ods. Esto amplía la posibilidad de participación de los usuarios permitiendo una interacción más amplia.
- La creación del servicio para asignar los consejos de escuela y de facultad, facilita la labor a la hora de asignar los miembros. El usuario podrá modificar, agregar o eliminar los diferentes miembros del consejo cuando lo desee.
- Con el cronograma para solicitudes se le asigna una fecha de inicio y de fin a los diferentes tipos de solicitudes que tenga determinado punto del consejo.
- Los servicios web ofrecidos surgen como apoyo a la política de Gobierno denominada “Cero Papel”.

## 7 RECOMENDACIONES

- Para mantener una buena comunicación con los usuarios de los portales, es fundamental dar pronta y precisa respuesta a las sugerencias de los mismos. De esta manera se logra dar soporte oportuno a sus inquietudes.
- Se hace necesaria la divulgación y capacitación necesaria sobre los nuevos servicios para que se les dé un uso adecuado y en su totalidad evitando así que se conviertan en servicios obsoletos o de poco uso.
- Se recomienda continuar con la optimización del servicio de producción intelectual siendo este de gran importancia.

## BIBLIOGRAFÍA

1. JOHNSON, James. Bases de datos: Modelos lenguajes y diseño. 1ª ed. Oxford, 2000. Presenta temas de teoría de bases de datos: modelos y métodos de acceso, administración, diseño de aplicaciones.
2. PIATTINI, Mario, CALVO-MANZANO, José A., CERVERA, Joaquín, FERNANDEZ, Luis. Análisis y diseño detallado de Aplicaciones Informáticas de Gestión. Alfaomega, 2000. Este libro contiene información sobre técnicas para el buen modelado de aplicaciones informáticas.
3. SCHMULLER, JOSEPH. Aprendiendo UML en 24 horas. 1ª ed. México: Alhambra Mexicana S.A, 2000. En este libro se encuentra una guía muy práctica que permite conocer y entender sobre UML.
4. STALLINGS, William. Sistemas Operativos: Aspectos Internos y Principios de Diseño. Madrid: Pearson Prentice Hall, 2005. Este libro se ocupa de los conceptos completos de las características de los sistemas operativos.
5. STUMPF, Robert, TEAGUE, Lavette. Object-Oriented Systems Analysis and Design with UML. Prentice Hall. 2004. Este libro introduce los conceptos y métodos del análisis y diseño de sistemas orientados a objetos.
6. WEITZENFELD, Alfredo. Ingeniería de Software Orientada a Objetos con UML, JAVA e Internet. Thomson International, 2003. En este libro se encuentra información sobre desarrollo de software orientado a objetos.
7. <http://api.jquery.com/>. Api de la librería jQuery.
8. <http://cormoran.uis.edu.co/eisi/Estandar>. Documentación de una librería del

grupo para el diseño de interfaces.

9. <http://docs.oracle.com/javase/docs/api/>. Api de Java.
10. <http://g701giadar.wikispaces.com/Arquitctura+Cliente+Servidor>. Información acerca de la Arquitectura Cliente Servidor.
11. <http://manuales.dgsca.unam.mx/jsp>. Manuales básicos de JSP.
12. <http://www.htmlquick.com/es/tutorials.html>, tutoriales de cómo programar en html de gran ayuda en detalles básicos y sencillos.
13. <http://www.mysql.com>. Sitio oficial de MySQL, que permite participar en capacitaciones en línea y tener las ultimas noticias acerca del producto para su mayor aprovechamiento.
14. <http://www.programacion.com/java/tutorial>. En este sitio se encuentran una gran variedad de tutoriales relacionados con: Los APIS, las herramientas y los servicios que proporciona los Servlets y las Java Server Pages (JSP), los tópicos necesarios para la programación de acceso a bases de datos en Java.

## ANEXOS

### Anexo A. ARQUITECTURA CLIENTE/SERVIDOR

#### 1. Características de la arquitectura Cliente/Servidor

Las características básicas de una arquitectura Cliente / Servidor son:

- El proceso del cliente da la interface entre usuarios y el resto del sistema (Interacción con usuarios y el proceso del servidor), maneja recursos compartidos tales como bases de datos, impresoras, módems, etc.
- El cliente y el servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades independientes.
- Las tareas del cliente y el servidor tienen diferentes requerimientos como: velocidad del procesador, memoria o capacidad del disco, por tanto la plataforma de hardware y el sistema operativo del cliente y del servidor no son siempre la misma y eso se conoce como ambiente heterogéneo.
- La escalabilidad horizontal permite agregar más estaciones de trabajo activas sin afectar el rendimiento y la escalabilidad vertical permite mejorar las características del servidor o agregar múltiples servidores. Se puede realizar independientemente cambios en las plataformas de los clientes o de los servidores, ya sea actualización o reemplazo tecnológico, de manera transparente para el usuario final.

## 2. Clasificación de las arquitecturas Cliente/Servidor

Los sistemas Cliente / Servidor se clasifican de acuerdo al nivel de abstracción del servicio que se ofrece. Se distinguen tres componentes básicos de software:

- Presentación: Presentación de resultados al usuario de forma comprensible.
- Lógica de aplicación: Esta capa es la responsable del procesamiento de la información que tiene lugar en la aplicación.
- Base de datos: Está compuesta por los archivos que contienen los datos persistentes de la aplicación.

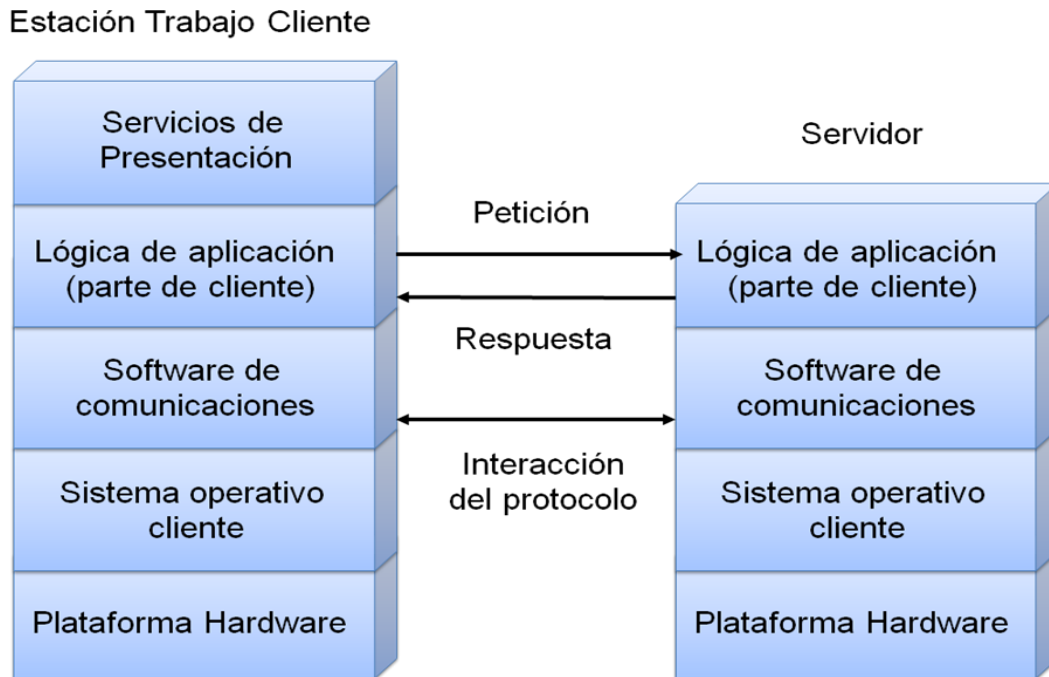
A continuación se muestra la clasificación de los sistemas Cliente / Servidor:

### 1.1. ARQUITECTURA CLIENTE/SERVIDOR DE DOS CAPAS

Consiste en una capa de presentación y lógica de la aplicación; y otra de la base de datos, cuando el cliente solicita recursos entonces el servidor responde directamente a la solicitud con sus propios recursos.

- Normalmente esta arquitectura es utilizada en las siguientes situaciones:
  - Cuando se requiera poco procesamiento de datos en la organización.
  - Cuando se tiene una base de datos centralizada en un solo servidor.
  - Cuando la base de datos es relativamente estática.
  - Cuando se requiere un mantenimiento mínimo.

**Figura 18 Arquitectura Cliente/Servidor de Dos Capas.**



**Fuente: Autor.**

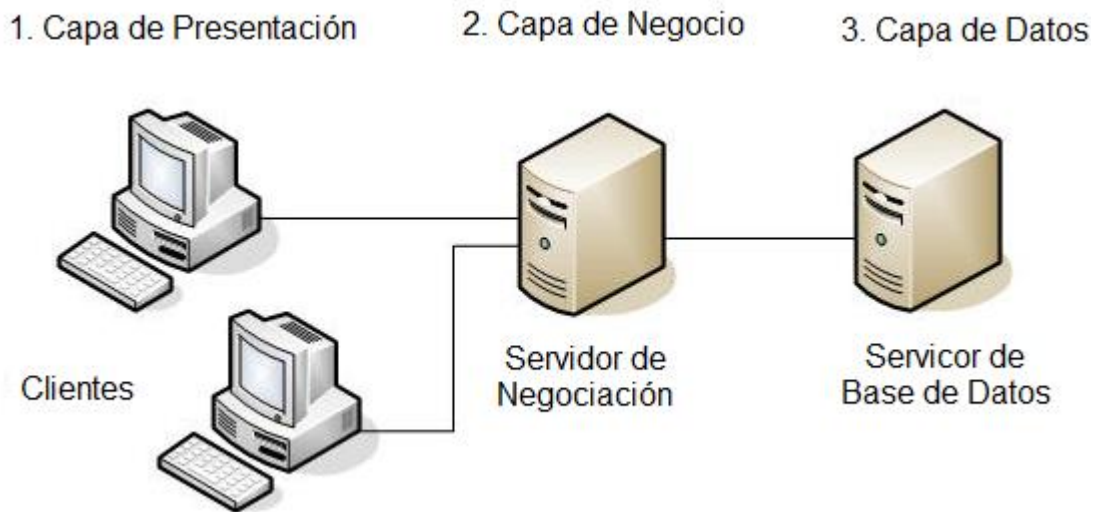
## **1.2. ARQUITECTURA CLIENTE/SERVIDOR DE TRES CAPAS**

Define como organizar el modelo de diseño en capas, que pueden estar físicamente distribuidas, es decir que los componentes de una capa solo pueden hacer referencia a componentes en capas inferiores. Este patrón es importante porque simplifica la comprensión y la organización del desarrollo de sistemas complejos, reduciendo las dependencias de forma que las capas más bajas no conscientes de ningún detalle o interfaz de las superiores, está compuesta de:

- Un equipo cliente con una interfaz de usuario (habitualmente se utiliza un navegador web), que solicita los recursos.

- El servidor de aplicaciones (o software intermedio), cuya tarea es prestar los recursos solicitados, pero que requiere de otro servidor para hacerlo.
- El servidor de datos que almacena y proporciona al servidor de aplicaciones los datos que requiere.

**Figura 19 Arquitectura Cliente/Servidor de Tres Capas**



Fuente: Internet. [http://www.ecured.cu/index.php/Archivo:Tres\\_capas.JPG](http://www.ecured.cu/index.php/Archivo:Tres_capas.JPG)

## **Anexo B. TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS**

### **1. Código del Lado del Cliente (Client Side Scripts)**

Código ejecutado por los navegadores, el cual los computadores clientes tienen instalados. Las tecnologías más comunes de este tipo son:

- *JavaScript*: Lenguaje de programación interpretado, es decir, que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C. Permite, crear ventanas, mostrar texto en movimiento y verificar las entradas a un formulario.
- *Controles Activos*: tecnología Microsoft que permite la creación de aplicaciones Windows, como pueden ser Visual Basic Script o Visual C. Es la respuesta de Microsoft a los Applets de Java.
- *Java Applets*: Programas escritos en lenguaje de programación Java, se incrustan en HTML y se ejecutan en el navegador gracias a la Máquina Virtual de Java (JVM) que lleva éste incorporado.

### **2. Código del Lado del Servidor (Server Side Scripts)**

Código que se ejecuta en el servidor. Para su actividad el programa ejecuta y procesa los datos o peticiones que el usuario envía desde su navegador, para luego enviar los resultados del programa en una página HTML que el usuario verá normalmente en su navegador. Los más usados son:

- ASP (Active Server Pages): Permite crear dinámicamente páginas Web mediante HTML, scripts, y componentes de servidor ActiveX reutilizables, requiere de un computador configurado como Servidor Web de Microsoft (Microsoft Web Server), el navegador del cliente es indiferente pues el trabajo se realiza del lado del servidor. De gran uso en la gestión de Bases de Datos ya que puede conectarse a SQL, Access, Oracle u otras.
- PHP (PHP Hypertext Pre-processor): Lenguaje de programación interpretado, diseñado para la creación de páginas web dinámicas. Es un lenguaje de código abierto (Open Source) y gratuito. Su gran potencia se encuentra en la interacción con los motores de bases de datos como Oracle y MySQL.
- JSP (Java Server Pages): tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo. Permiten la utilización de código Java mediante scripts.

## **Anexo C. BASES DE DATOS**

### **1. Modelos de Bases De Datos**

Este modelo describe la estructura de una base de datos, donde se incluye conceptos como: entidades, atributos y relaciones, la mayoría de los modelos de datos poseen un conjunto de operaciones básicas para especificar consultar y actualizaciones. Las bases de datos se pueden clasificar de acuerdo a su modelo de administración de datos. Algunos modelos utilizados con frecuencia son:

#### **1.1. Base de Datos Jerárquica**

Estas bases de datos almacenan su información en una estructura escalonada, organizando los datos en forma similar a un árbol (visto al revés), en donde un nodo padre de información puede tener varios hijos, el nodo que no tiene padres es llamado raíz, y a los nodos que no tienen hijos se les conoce como hojas. Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento. Esta limitado por su incapacidad de representar eficientemente la redundancia de datos.

#### **1.2. Base de Datos de Red**

En este modelo se permite que un mismo nodo tenga varios padres. Ofrece una solución eficiente al problema de redundancia de datos; sin embargo, la dificultad para administrar los datos en una base de datos de red ha conllevado a que sea un modelo usado más por programadores que por usuarios finales.

### **1.3. Base de Datos Relacional**

Es el más utilizado para modelar problemas reales y administrar datos dinámicamente. Su fundamento es el uso de "relaciones". Estas relaciones podrían considerarse en forma lógica como conjuntos de datos, también llamados tuplas. Cada relación es una tabla que está compuesta por registros (las filas de una tabla), que representan las tuplas, y campos (las columnas de una tabla). Los datos pueden ser recuperados o almacenados mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información. El lenguaje más habitual para construir las consultas a bases de datos relacionales es el Lenguaje Estructurado de Consultas (Structured Query Language, SQL), un estándar implementado por los principales manejadores de bases de datos relacionales.

## **2. Acceso a Base de Datos**

Para desarrollar aplicaciones que conecten bases de datos, se utilizan interfaces y programas estándar que envían demandas escritas en SQL, y procesan los resultados. Para conectarse a un motor de bases de datos determinado, se necesita un driver o una interfaz estándar que medie entre la aplicación y la base de datos.

## **3. Conectores más utilizados**

- ODBC (Open DataBase Connectivity): Estándar de acceso a Bases de datos, su objetivo es hacer posible el acceder a cualquier dato desde cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos (DBMS por sus siglas en inglés) almacene los datos, logra esto al insertar una capa intermedia llamada manejador de Bases de Datos, entre la aplicación y el DBMS, el propósito de esta capa es traducir las consultas de

datos de la aplicación en comandos que el DBMS entienda. Para que esto funcione tanto la aplicación como el DBMS deben ser compatibles con ODBC.

- MDB: Esta aplicación permite trabajar con tablas de base de datos creadas en Access 97/2000. Es posible abrir tablas en SQL, visualizarlas, navegar, crear y borrar índices, fijar relaciones, copiar, etc.
- JDBC (Java Database Connectivity): Interfaz de programación de aplicaciones que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java, independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede, utilizando el dialecto SQL del modelo de base de datos que se utilice.

Para el desarrollo de los portales y cada uno de sus módulos se empleó el conector JDBC. Uno de sus mayores beneficios es su capacidad para crear aplicaciones cuya programación sea independiente de la base de datos, es decir, pueden ser migradas a otro servidor de bases de datos sin complicaciones. Sin embargo, dos elementos siguen estando ligados a una base de datos en particular, el nombre de la clase que se usa para cargar el controlador JDBC y la dirección URL (Universal Resource Locator) para acceder a la base de datos.

Los servlets y las páginas JSP usan JDBC habitualmente de la misma manera que cualquier otra aplicación en Java, típicamente los datos del controlador JDBC, la cadena de conexión, el nombre de usuario y contraseña para conectarse a la base de datos son codificados dentro del programa.

Las operaciones primordiales realizadas durante la ejecución del JDBC son:

- Cargar un controlador JDBC.
- Utilizar el controlador para abrir una conexión con la base de datos.

- Formular instrucciones SQL a través de la conexión.
- Procesar los conjuntos de resultados devueltos por las operaciones SQL.

#### **4. Manejadores o Gestores de Bases de Datos**

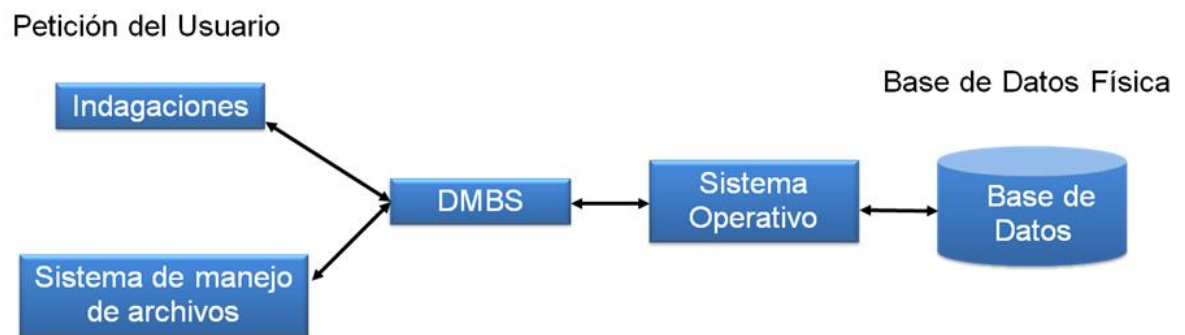
El sistema manejador de bases de datos es la porción más importante del software de un sistema de base de datos. Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica.

Las funciones principales de un DBMS son:

- Crear y organizar la Base de Datos.
- Establecer y mantener las trayectorias de acceso a la base de datos de tal forma que los datos puedan ser capturados rápidamente.
- Manejar los datos de acuerdo a las peticiones de los usuarios.
- Registrar el uso de las bases de datos.
- Interacción con el manejador de archivos a través de las sentencias en Lenguaje Manipulador de Datos (Data Manipulation Language, DML) al comando del sistema de archivos.
- Respaldo y recuperación: Consiste en contar con mecanismos implantados que permitan la recuperación fácilmente de los datos en caso de ocurrir fallas en el sistema de base de datos.

- Control de concurrencia: consiste en controlar la interacción entre los usuarios concurrentes para preservar la consistencia de los datos.
- Seguridad e Integridad: consiste en contar con mecanismos que permitan el control de la consistencia de los datos evitando que estos se vean perjudicados por cambios no autorizados o previstos.

**Figura 20 Función del DBMS.**



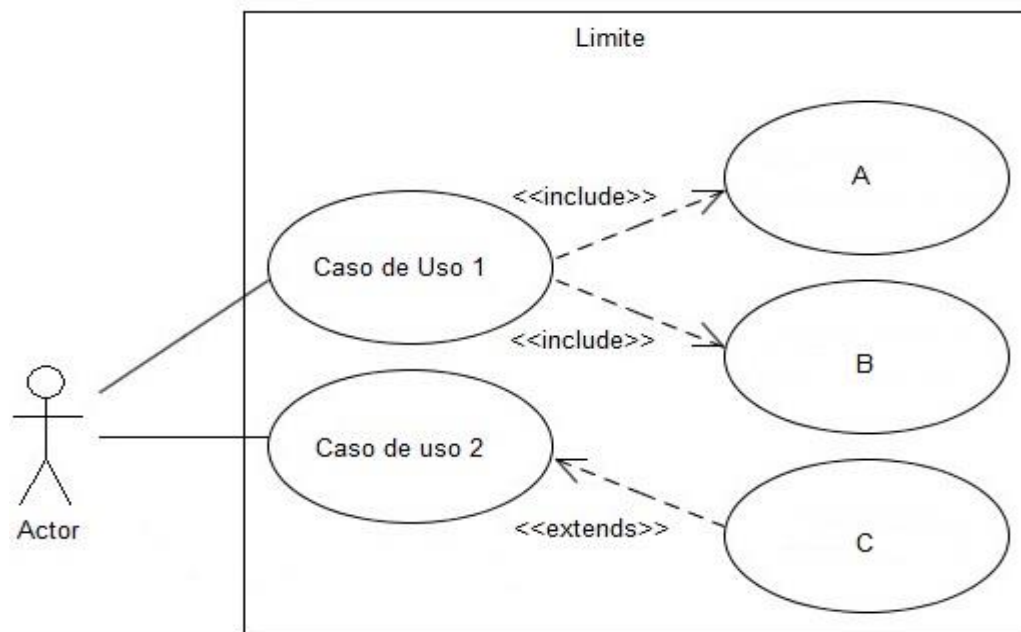
Fuente: Autor.

La figura muestra el DBMS como interface entre la base de datos física y las peticiones del usuario. El DBMS interpreta las peticiones de entrada/salida del usuario y las manda al sistema operativo para la transferencia de datos entre la unidad de memoria secundaria y la memoria principal. Es decir el DBMS es el corazón de la base de datos ya que se encarga del control total de los posibles aspectos que la puedan afectar.

## Anexo D. LEGUAJE DE MODELADO UNIFICADO (UML)

### 1. Diagramas de casos de uso

Figura 21 Diagramas de Casos de Uso.



Fuente: Autor.

**Actores:** Un actor en un caso de uso representa un rol, que alguien o algo puede desempeñar dentro un sistema y no un alguien o algo específico.

En este proyecto se destacan dos clases de actores:

- **Administradores:** Son usuarios que además de pertenecer a la categoría de usuarios tienen un perfil de administrador, con el cual pueden desempeñar ciertas labores diferentes a un usuario normal dentro del sitio. Estos son: los Auxiliares de administración del portal, profesores, secretaria con ciertos

privilegios. Dentro de esta categoría se incluye también el súper administrador.

- Usuario EIMWEB: Es el tipo de usuario común del portal EIMWEB y a quien van dirigidos los servicios. Este usuario solo tiene control sobre sus privilegios.

**Inclusión (include–uses):** Es una forma de interacción, un caso de uso dado puede "incluir" otro. Una inclusión es utilizada para indicar que un caso de uso depende de otro, es decir, la funcionalidad de determinado caso de uso se requiere para realizar las tareas de otro. En la figura 7 el caso de uso "Caso de uso 1" depende de los casos de uso "A" y "B".

**Extensión (extend):** Es otra forma de interacción, una extensión representa una variación de un caso de uso a otro, es decir, una dependencia específica entre los casos de uso, a través de la cual un caso de uso puede extender a otro.

## 2. Diagramas de secuencias

**Objetos:** Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de cada objeto será una línea discontinua conocida como la *línea de vida* de un objeto. Con la línea de vida se encuentra un pequeño rectángulo conocido como *activación*, el cual representa la ejecución de una operación que realiza el objeto.

**Mensaje:** Un mensaje que va de un objeto a otro pasa la línea de vida de un objeto a otro. Un objeto puede enviarse un mensaje a sí mismo. Un mensaje puede ser simple, sincrónico o asincrónico.

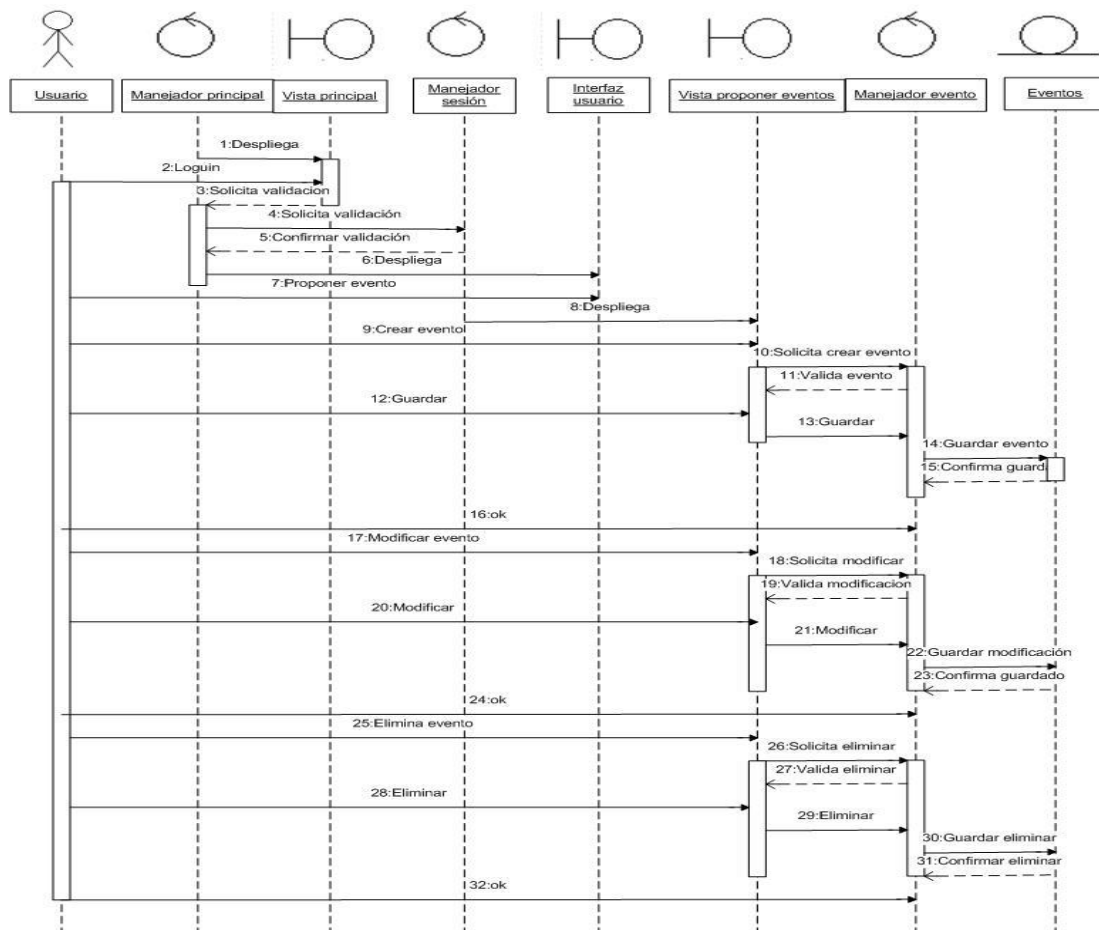
**Tiempo:** El diagrama representa al tiempo en dirección vertical. Inicia en la parte superior y avanza hacia la parte inferior. Un mensaje que esté más cerca de la parte superior ocurrirá antes que uno que esté cerca de la parte inferior.

**GUI:** (Siglas en Ingles) La interfaz gráfica de usuario; es la interfaz de interacción del usuario y en la que más interactividades se presentan con otros objetos.

### Modelo de Procesos del Sistema.

#### Eventos Uis

Figura 22 Proponer eventos.

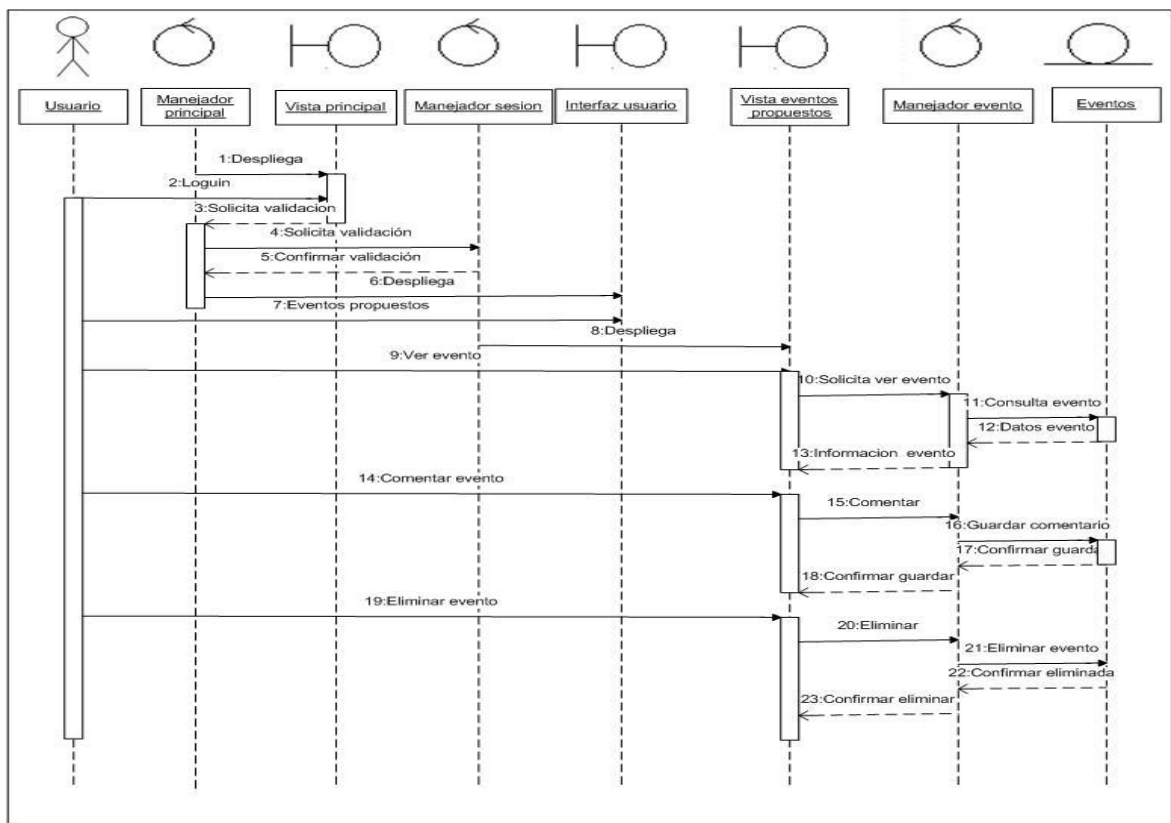


Fuente: Autor.

1. El Manejador principal despliega la Vista Principal.
2. El usuario ingresa sus datos de registro para acceder al servicio deseado.
3. La vista principal solicita al manejador principal la validación de los datos ingresados por el usuario en el login.
4. El manejador principal solicita al manejador de sesión la validación de los datos del usuario.
5. El manejador de sesión le confirma al manejador principal que los datos del usuario son correctos.
6. El manejador principal despliega la interfaz del usuario.
7. El usuario en su interfaz accede al servicio proponer eventos.
8. El manejador de sesión despliega la vista de proponer eventos.
9. El usuario crea un evento accediendo al servicio en la vista de proponer eventos.
10. La vista proponer eventos solicita al manejador de eventos crear el evento con los datos ingresados por el usuario.
11. El manejador de evento valida el evento.
12. El usuario guarda los datos del evento
13. La vista de proponer eventos le pide al manejador de evento que guarde los datos.
14. El manejador de evento guarda el evento en la base de datos.
15. La base de datos confirma el guardado.
16. El usuario acepta.
17. El usuario en la vista proponer eventos accede a la opción modificar evento.
18. La vista proponer eventos solicita al manejador de eventos modificar el evento con los datos ingresados por el usuario.
19. El manejador de evento valida la modificación del evento.
20. El usuario guarda los datos de la modificación del evento
21. La vista de proponer eventos le pide al manejador de evento que guarde los datos de la modificación.
22. El manejador de evento guarda los cambios en la base de datos.

23. La base de datos confirma el guardado.
24. El usuario acepta.
25. El usuario en la vista proponer eventos va a eliminar un evento.
26. La vista proponer eventos solicita al manejador de eventos eliminar el evento.
27. El manejador de evento valida la eliminacion del evento.
28. El usuario elimina el evento
29. La vista de proponer eventos le pide al manejador de evento que elimine los datos del evento.
30. El manejador de evento guarda los cambios en la base de datos.
31. La base de datos confirma el guardado.
32. El usuario acepta.

**Figura 23 Eventos propuestos.**



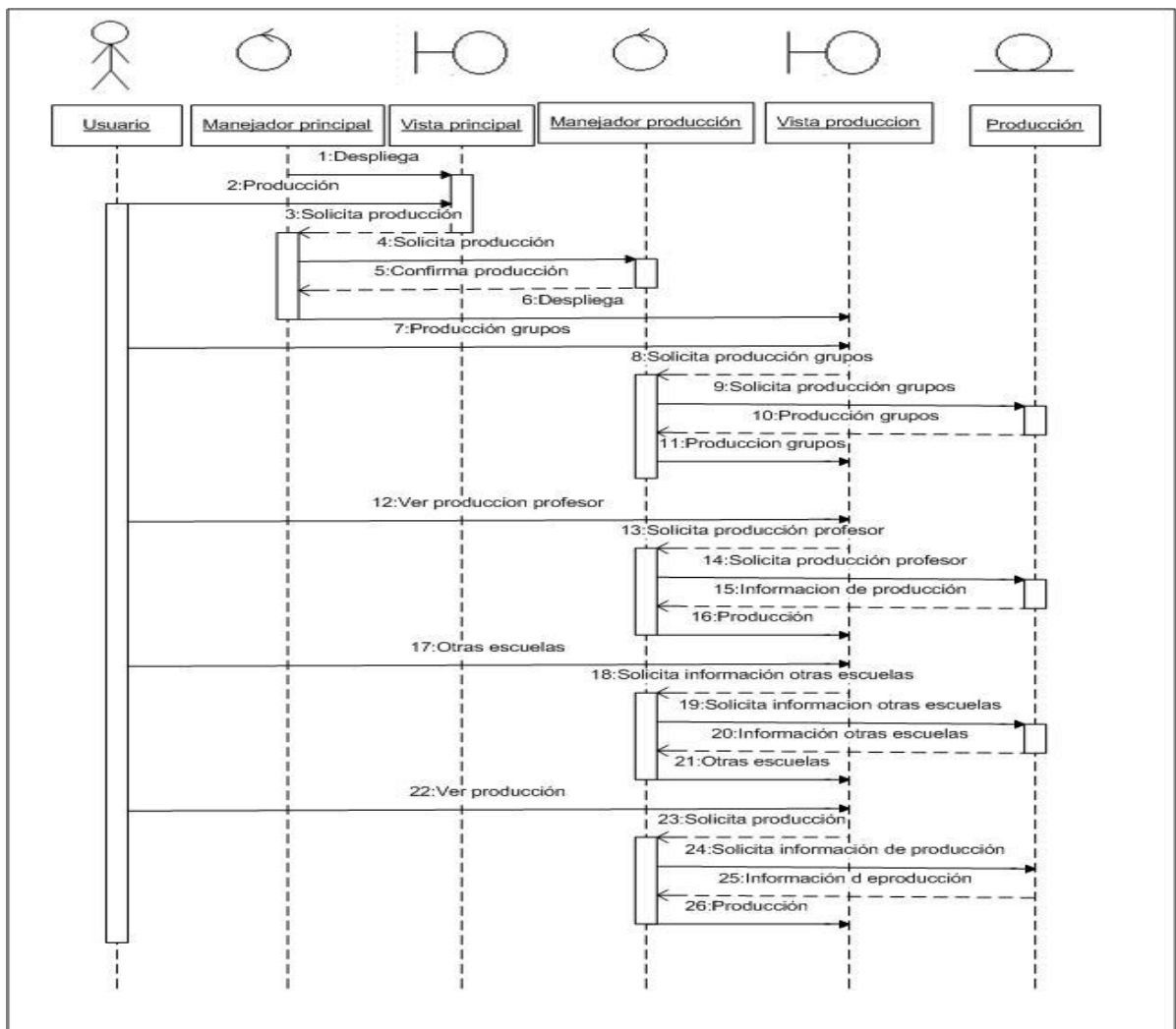
Fuente: Autor.

1. El Manejador principal despliega la Vista Principal.
2. El usuario ingresa sus datos de registro para acceder al servicio deseado.
3. La vista principal solicita al manejador principal la validación de los datos ingresados por el usuario en el loguin.
4. El manejador principal solicita al manejador de sesión la validación de los datos del usuario.
5. El manejador de sesión le confirma al manejador principal que los datos del usuario son correctos.
6. El manejador principal despliega la interfaz del usuario.
7. El usuario en su interfaz accede al servicio eventos propuestos.
8. El manejador de sesión despliega la vista de eventos propuestos.
9. El usuario en la vista eventos propuestos accede a la opción ver evento.
10. La vista eventos propuestos solicita al manejador de eventos los datos del evento.
11. El manejador de evento consulta la información del evento en la base de datos.
12. La base de datos envía los datos del evento.
13. El manejador de eventos envía los datos del evento a la vista de eventos propuestos.
14. El usuario en la vista eventos propuestos comenta un evento que está consultando.
15. La vista de eventos propuestos envía al manejador de eventos el comentario realizado por el usuario.
16. El manejador de evento guarda el comentario en la base de datos.
17. La base de datos confirma el guardado.
18. El manejador de evento confirma que se ha guardado el comentario con éxito a la interfaz de eventos propuestos.
19. El usuario en la vista eventos propuestos elimina un evento.

20. La vista de eventos propuestos envía al manejador de eventos la acción de eliminar.
21. El manejador de eventos solicita eliminar los datos del evento de la base de datos.
22. La base de datos confirma la eliminada del evento.
23. El manejador de evento confirma que se ha eliminado con éxito a la interfaz de eventos propuestos.

## Producción Intelectual

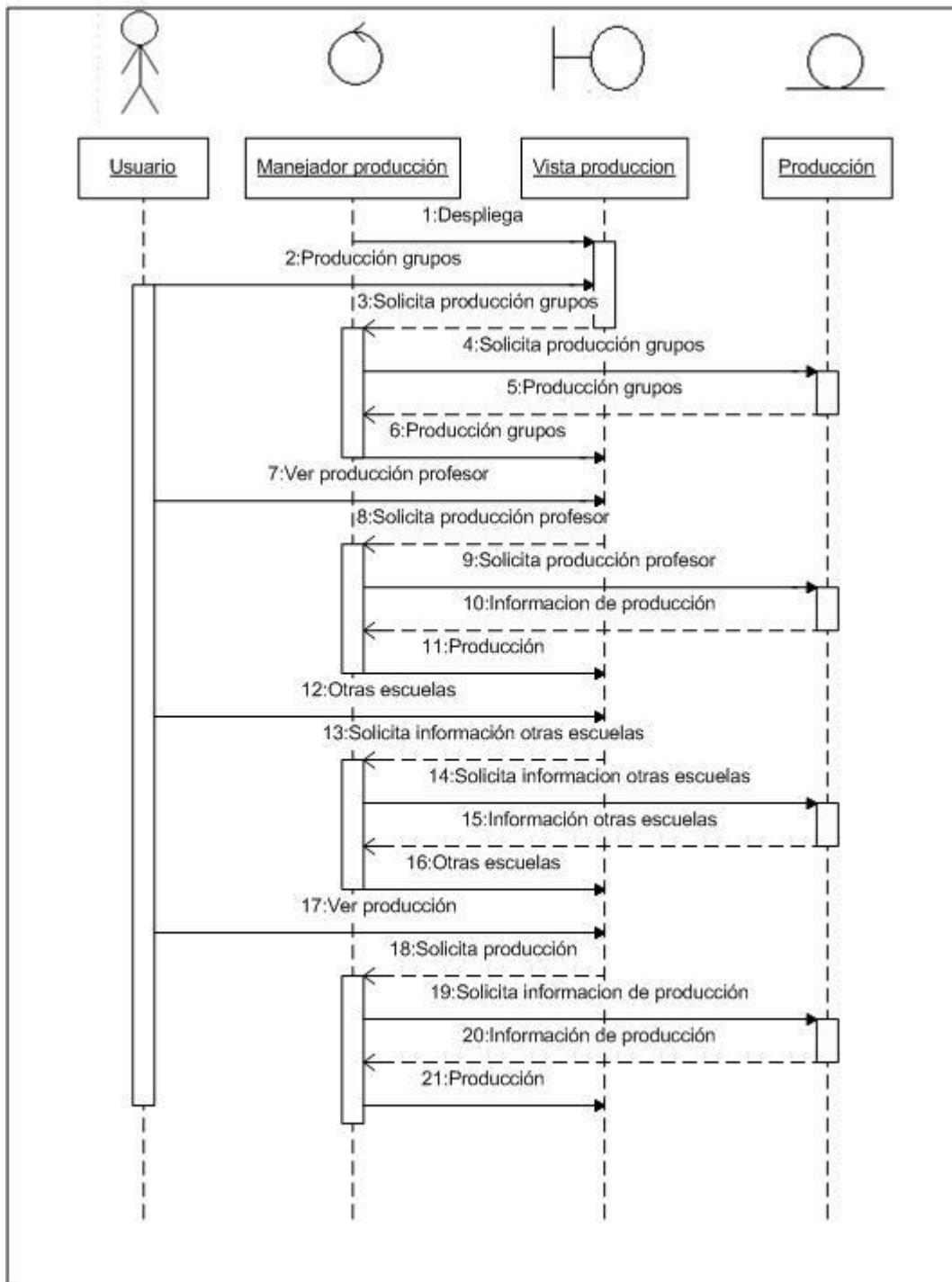
Figura 24 Producción intelectual.



Fuente: Autor.

1. El manejador principal despliega la vista principal. Ya sea el portal de la escuela, el portal de grupos o el portal de profesor.
2. El usuario solicita el servicio de producción intelectual.
3. La vista principal solicita al manejador principal la producción.
4. El manejador principal solicita al manejador de producción la validación.
5. El manejador de producción confirma al manejador principal la validación del servicio.
6. El manejador principal despliega la vista de producción intelectual.
7. El usuario en la vista de producción intelectual accede a la producción de los grupos.
8. La vista principal solicita al manejador de producción la producción de los grupos.
9. El manejador de producción solicita la información a la base de datos.
10. La base de datos envía la información de la producción de grupos al manejador de producción.
11. El manejador de producción envía la información de los grupos a la vista de producción.
12. De acá en adelante se repiten los pasos de los numerales 7, 8, 9, 10 y 11 para la producción de los profesores, la producción de otras escuelas y para ver una producción intelectual en específico.

Figura 25 Portal Producción intelectual.

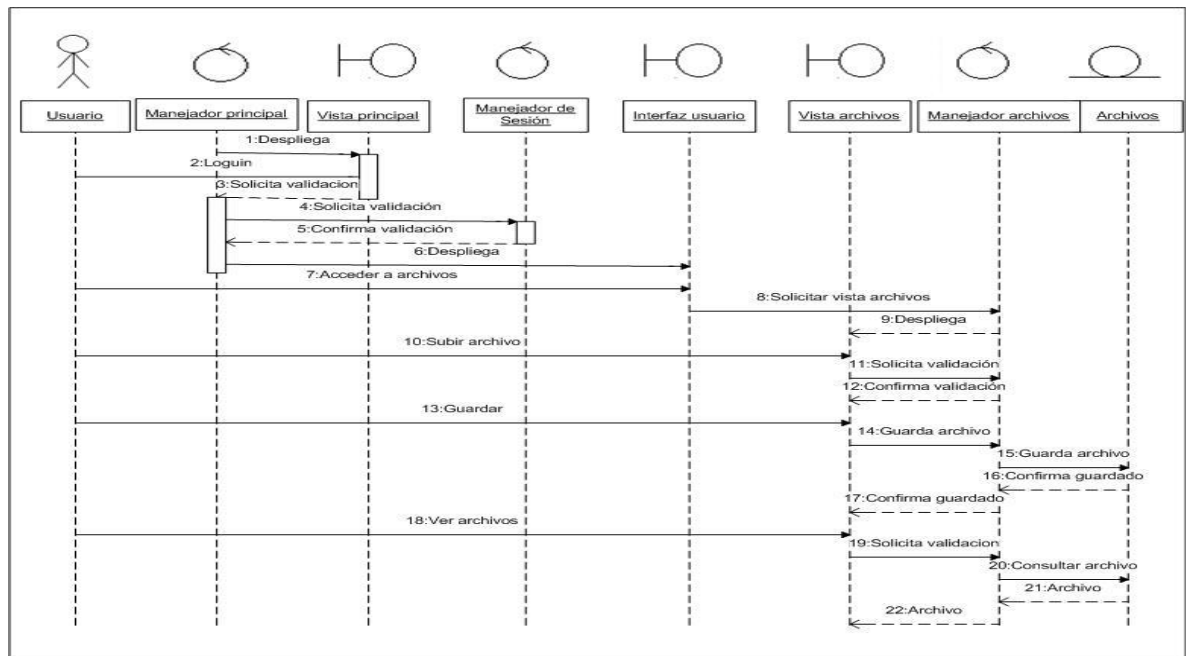


Fuente: Autor.

1. El manejador producción despliega la vista de producción intelectual.
2. El usuario en la vista de producción intelectual accede a las producciones de los grupos.
3. La vista de producción solicita al manejador de producción la información de la producción de los grupos.
4. El manejador de producción solicita la información de la producción de los grupos a la base de datos.
5. La base de datos envía la información de la producción de los grupos al manejador de producción.
6. El manejador de producción muestra la producción de los grupos en la vista de producción.
7. De acá en adelante se repiten los pasos para la producción de profesor, de otras escuelas y para una producción específica.

## Archivos

**Figura 26 Portal Producción intelectual.**

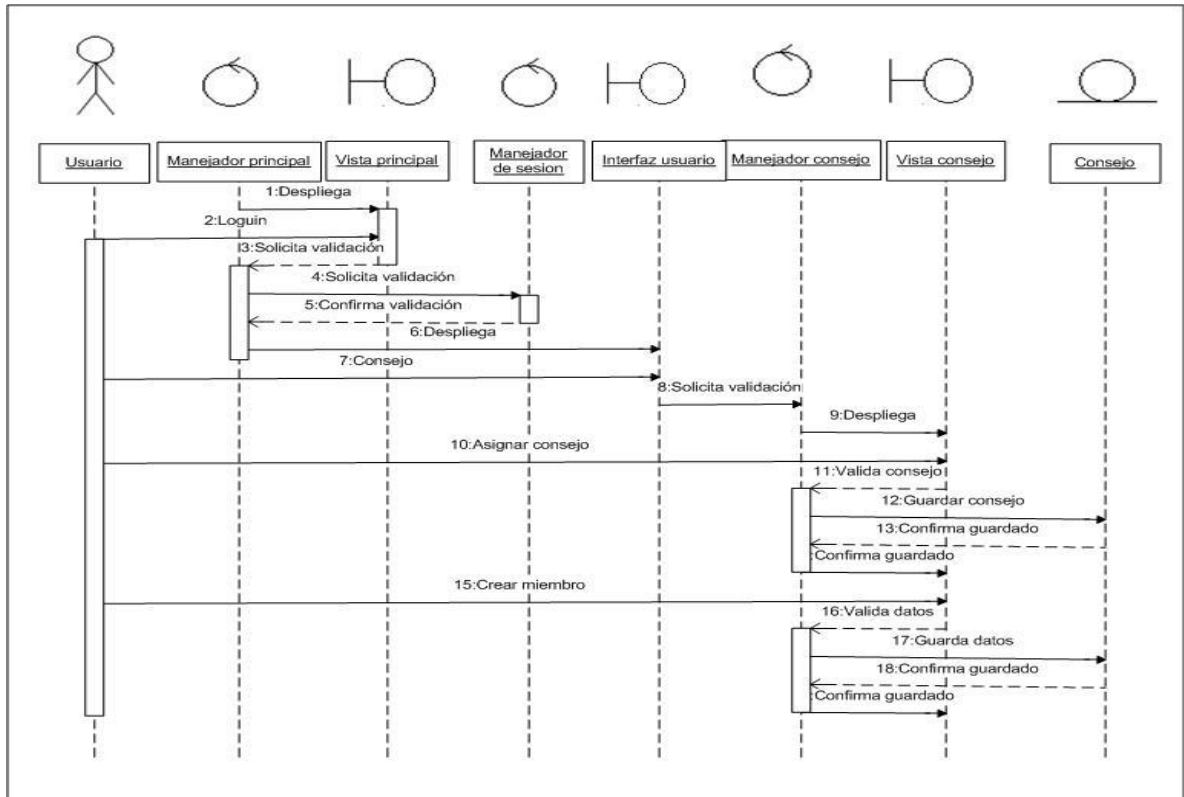


Fuente: Autor.

1. El Manejador principal despliega la Vista Principal.
2. El usuario ingresa sus datos de registro para acceder al servicio deseado.
3. La vista principal solicita al manejador principal la validación de los datos ingresados por el usuario en el login.
4. El manejador principal solicita al manejador de sesión la validación de los datos del usuario.
5. El manejador de sesión le confirma al manejador principal que los datos del usuario son correctos.
6. El manejador principal despliega la interfaz del usuario.
7. El usuario en la interfaz de usuario accede al servicio de archivos.
8. La interfaz de usuario solicita la vista de archivos al manejador de archivos.
9. El manejador de archivos despliega la vista de archivos.
10. El usuario solicita subir un archivo.
11. La vista de archivos solicita validación al manejador de archivos.
12. El manejador de archivos confirma la validación del archivo.
13. El usuario le da guardar al archivo.
14. La vista de archivos solicita guardar el archivo al manejador.
15. El manejador guarda el archivo en la base de datos.
16. La base de datos confirma el guardado al manejador de archivos.
17. El manejador de archivos confirma el guardado a la interfaz de archivos.
18. El usuario solicita en la interfaz de archivo ver un archivo en específico.
19. La vista de archivos solicita al manejador de archivos la validación.
20. El manejador de archivos consulta en la base de datos la información del archivo.
21. La base de datos devuelve al manejador la información del archivo.
22. El manejador de archivos muestra en la interfaz de archivos la información.

## Consejo

Figura 27 Asignar consejo de escuela y facultad.



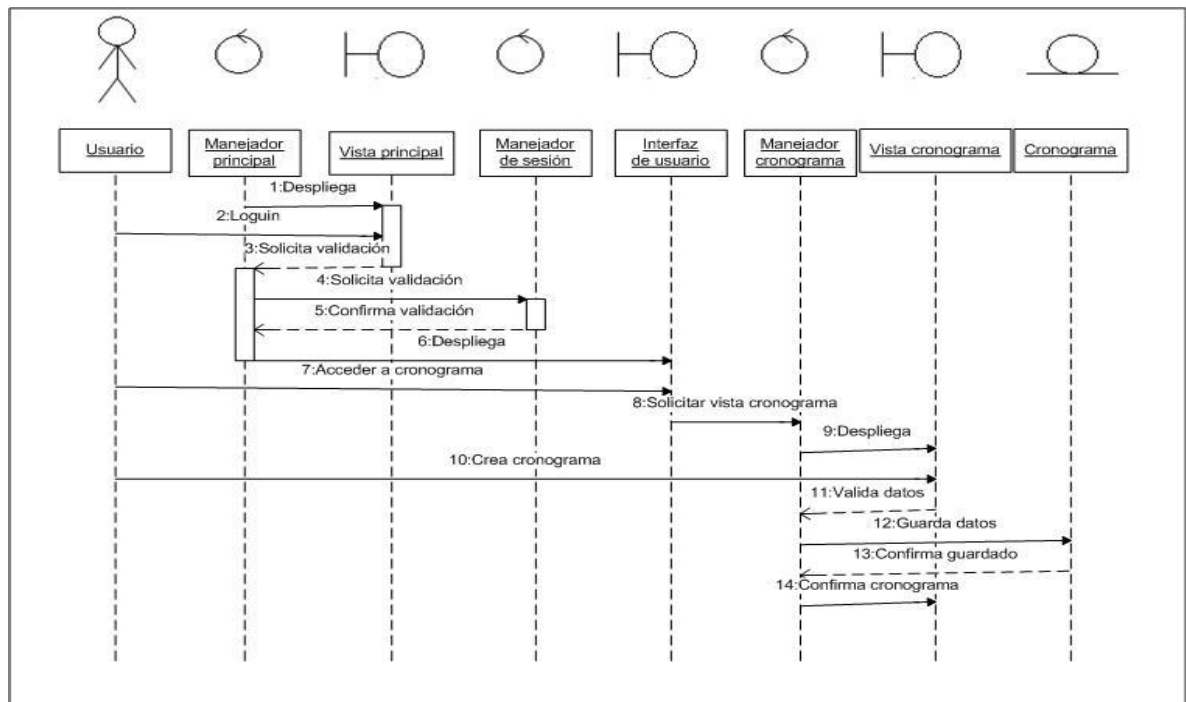
Fuente: Autor.

1. El Manejador principal despliega la Vista Principal.
2. El usuario ingresa sus datos de registro para acceder al servicio deseado.
3. La vista principal solicita al manejador principal la validación de los datos ingresados por el usuario en el loguin.
4. El manejador principal solicita al manejador de sesión la validación de los datos del usuario.
5. El manejador de sesión le confirma al manejador principal que los datos del usuario son correctos.
6. El manejador principal despliega la interfaz del usuario.
7. El usuario ingresa al servicio consejo en la interfaz del usuario.

8. La interfaz de usuario solicita al manejador de consejo la validación.
9. El manejador de consejo despliega la vista consejo.
10. El usuario en la vista consejo accede al servicio asignar consejo.
11. La vista consejo pide validación de la información al manejador de consejo.
12. El manejador de consejo guarda la información en la base de datos.
13. La base de datos confirma el guardado de la información al manejador de consejo.
14. El manejador de consejo confirma guardado a la interfaz de consejo.
15. El usuario accede al servicio crear miembro en la interfaz de consejo.
16. La vista de consejo le pide la validación de los datos al manejador de consejo.
17. El manejador de consejo guarda los datos en la base de datos.
18. La base de datos confirma guardado de la información al manejador de consejo.
19. El manejador de consejo confirma el guardado de la información a la interfaz de consejo.

## Cronograma.

**Figura 28 Asignar consejo de escuela y facultad.**



Fuente: Autor.

## Anexo E. PRUEBAS DEL SISTEMA

### 1. Eventos UIS.

**Tabla 11. Caso de Uso: Pruebas Realizadas: Proponer eventos.**

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Proponer eventos	Acceder a la interfaz para proponer un evento nuevo.	✓
Proponer eventos escuela	Acceder a la interfaz para proponer un evento nuevo en la escuela	✓
Proponer eventos UIS	Acceder a la interfaz para proponer un evento nuevo en la UIS	✓
Nuevo evento	Crear un nuevo evento comunidad UIS	✓
Modificar	Modificar un evento de la comunidad UIS verificando que se modifiquen también los campos necesarios en las diferentes tablas de la base de datos.	✓
Eliminar	Eliminar un evento de la comunidad UIS verificando que se eliminen también los campos necesarios en las diferentes tablas de la base de datos.	✓

**Tabla 12. Caso de Uso: Pruebas Realizadas: Eventos propuestos.**

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Eventos propuestos	Acceder a la interfaz para ver los eventos propuestos.	✓
Eventos propuestos escuela	Acceder a la interfaz para ver los eventos que están propuestos en la escuela.	✓
Eventos propuestos UIS	Acceder a la interfaz para ver los eventos que están propuestos en la comunidad UIS	✓
Ver evento	Verificar que se muestre de manera correcta toda la información necesaria sobre un evento.	✓
Comentar	Hacer un comentario sobre determinado evento.	✓
Unirse	Unirse a determinado evento propuesto.	✓

## **2. Producción Intelectual.**

**Tabla 13. Caso de Uso: Portal Escuela.**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>	<b>RESULTADO</b>
Ingreso al portal escuela	Ingresar al portal de la escuela.	✓
Ingreso al portal grupos	Ingresar al portal de un grupo	✓

Ingreso al portal profesor	Ingresar al portal de un profesor.	✓
Producción intelectual	Acceder al servicio de producción intelectual.	✓
Producción grupos	Acceder a la producción de los grupos.	✓
Producción profesor	Acceder a la producción de los profesores.	✓
Otras escuelas	Acceder a las producciones intelectuales de diferentes escuelas.	✓
Ver producción	Ver en detalle la información de una producción.	✓

**Tabla 14. Caso de Uso: Portal Producción Intelectual.**

<b>TÍTULO</b>	<b>FUNCIONES PRIMARIAS</b>	<b>RESULTADO</b>
Ingreso portal producción intelectual	Ingresar al portal de producción intelectual.	✓
Producción grupos	Acceder a las producciones intelectuales de los diferentes grupos.	✓
Producción profesor	Acceder a las producciones intelectuales de los profesores.	✓
Otras escuelas	Acceder al portal de producción intelectual de otras escuelas.	✓
Ver producción	Ver en detalle la información de una producción.	✓

### 3. Subida de archivos de diferente tipo.

**Tabla 15. Caso de Uso: Servicios Archivos.**

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Ingreso al portal	Iniciar sesión con un login y contraseña.	✓
Servicios	Acceder al menú servicios y en este al submenú archivos	✓
Subir archivos	Escoger un archivo de cada uno de los diferentes tipos de extensiones y comprobar que suba correctamente.	✓
Ver archivos	Ver correctamente los archivos que se encuentran en el portal.	✓

**Tabla 16. Caso de Uso: Portal grupos.**

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Ingreso al portal grupos	Iniciar sesión con un login y contraseña.	✓
Servicios grupo	Acceder al menú servicios grupo y en este al submenú archivos	✓

Ver todos los archivos	Ver correctamente los archivos que se encuentran en el portal.	✓
Subir archivo	Escoger un archivo de cada uno de los diferentes tipos de extensiones y comprobar que suba correctamente.	✓

**Tabla 17. Caso de Uso: Archivos aula.**

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Servicios	Acceder al menú servicios grupo y en este al submenú Aula virtual.	✓
Acceder al aula	Verificar que se tenga acceso a las diferentes aulas a las que pertenecen los usuarios.	✓
Subir archivo	Escoger un archivo de cada uno de los diferentes tipos de extensiones y comprobar que suba correctamente.	✓
Ver archivos	Ver correctamente los archivos que se encuentran en el Aula.	✓

#### 4. Consejo

**Tabla 18. Caso de Uso: Asignar consejo de escuela y de facultad.**

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Ingreso al portal	Iniciar sesión con un login y contraseña.	✓

Secretaría	Verificar que se tenga acceso al menú secretaria.	✓
Consejo escuela	Ingresar al servicio donde se podrá asignar el consejo de escuela.	✓
Consejo facultad	Ingresar al servicio donde se podrá asignar el consejo de facultad.	✓

**Tabla 19. Caso de Uso: Asignar consejo de escuela y de facultad.**

<b>CASO DE USO</b>	<b>PRUEBA REALIZADA</b>	<b>RESULTADO</b>
Consejo escuela	Ingresar al servicio donde se podrá asignar el consejo de escuela.	✓
Consejo facultad	Ingresar al servicio donde se podrá asignar el consejo de facultad.	✓
Asignar consejo	Escoger los miembros que van a conformar el consejo verificando que queden guardados correctamente todos los campos en la base de datos.	✓
Crear miembro	Crear un nuevo miembro que hará parte del consejo de facultad. Verificar que el usuario nuevo quedo guardado en la base de datos.	✓

## 5. Cronograma.

Tabla 20. Caso de Uso: Cronograma.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Ingreso al portal	Iniciar sesión con un login y contraseña.	✓
Secretaría		✓
Cronograma para solicitudes	Escoger el punto del consejo y el tipo de solicitud a la que se le va a signar las fechas de inicio y de fin. Asignar las diferentes fechas y verificar que los campos quedaron correctamente guardados en la base de datos.	✓