

**ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO DEL PORTAL
WEB, ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE NUEVOS
SERVICIOS PARA EL PORTAL WEB DE LA ESCUELA DE METALÚRGICA Y
LA FACULTAD DE INGENIERÍAS FISICOMECÁNICAS.**

DANAY FAISURY QUINTERO CASTRO

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2015**

**ADMINISTRACIÓN, SOPORTE A USUARIOS, MANTENIMIENTO DEL PORTAL
WEB, ANÁLISIS, DESARROLLO E IMPLEMENTACIÓN DE NUEVOS
SERVICIOS PARA EL PORTAL WEB DE LA ESCUELA DE METALÚRGICA Y
LA FACULTAD DE INGENIERÍAS FISICOMECÁNICAS.**

DANAY FAISURY QUINTERO CASTRO

**Trabajo de grado presentado como requisito para optar al título de
Ingeniera de Sistemas**

Director

LUIS IGNACIO GONZÁLEZ RAMÍREZ

Ingeniero de Sistemas, Magíster en Informática

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

2015

Agradecimientos

*Agradezco a **Dios** y a la **Virgen María** por estar conmigo siempre, por llenarme de fortaleza para afrontar cada reto, de motivación para no desfallecer, por guiarme en el buen camino, por colocar personas tan maravillosas en mi vida que me han acompañado y enseñado valores, conocimientos en esta gran etapa.*

*A mi mamá **Ana María** que ha sido una mujer ejemplar, mi mejor amiga, mi confidente, mi apoyo incondicional, por ser tan maravillosa, por creer en mí, por guiarme con su dulzura y tenacidad, por hacer posible este gran sueño, por ayudarme a alcanzar este gran logro.*

*A mi mejor amiga, mi apoyo incondicional, mi motor de vida, mi hermana **María Fernanda**, que es una niña increíblemente maravillosa, que ha sido capaz de invertir mis momentos difíciles y en los cuales desistía con palabras de aliento, con su gran amor, admiración, madurez, ternura, e inteligencia; me hace reír a carcajadas, olvidar y superar las adversidades. A ella le debo momentos inolvidables llenos de alegría, de risas, de amor.*

A ella le agradezco por ser la mejor hermana que una persona pueda tener y por ayudarme a culminar esta gran etapa de mi vida.

*A mi director de proyecto el Ingeniero **Luis Ignacio** por darme la oportunidad de pertenecer al grupo **Calumet**, por sus enseñanzas, sus consejos y su apoyo.*

*Al grupo **Calumet**, que es una gran familia, gracias por hacer parte de este gran logro, por sus enseñanzas, su paciencia y carisma. Por el humor que hicieron que los días largos de trabajo se convirtieran en momentos alegres y llenos de risa. Gracias por el conocimiento transmitido*

*A mis **profesores**, por su gran labor de orientarme, de enseñarme, y mostrarme las diversas oportunidades que tengo para seguir avanzando.*

*A mis **amigos** y **compañeros**, con los que he compartido gratos momentos y que hicieron de esta etapa algo increíble.*

Danay Faisury Quintero Castro.

CONTENIDO

| | |
|--|-----------|
| INTRODUCCIÓN | 17 |
| 1. DEFINICIÓN DEL PROBLEMA | 20 |
| 2. JUSTIFICACIÓN | 22 |
| 3. OBJETIVOS | 24 |
| 3.1 OBJETIVO GENERAL | 24 |
| 3.2 OBJETIVOS ESPECÍFICOS | 24 |
| 4. ALCANCES Y LIMITACIONES | 28 |
| 5. MARCO TEÓRICO | 29 |
| 5.1 ARQUITECTURA CLIENTE/SERVIDOR | 29 |
| 5.1.1 Arquitectura Cliente/Servidor aplicada | 30 |
| 5.1.2 Ventajas del esquema Cliente/Servidor | 30 |
| 5.1.3 Desventajas del esquema Cliente/Servidor | 31 |
| 5.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS | 31 |
| 5.2.1 Tecnología aplicada | 32 |
| 5.3 BASES DE DATOS | 34 |
| 5.3.1 MySQL | 35 |
| 5.4 NETBEANS | 36 |
| 5.5 SISTEMA DE CONTROL DE VERSIONES | 36 |
| 5.5.1 Subversión | 37 |
| 5.6 PROGRAMACIÓN UTILIZADA | 38 |
| 5.6.1 Clases | 38 |
| 5.6.2 Objetos | 38 |

| | | |
|-----------|---|-----------|
| 5.6.3 | Atributos..... | 39 |
| 5.6.4 | Métodos | 39 |
| 5.6.5 | Herencia..... | 39 |
| 5.6.6 | Beneficios de la Programación Orientada a Objetos | 39 |
| 5.6.7 | Java y JDK (Java Development Kit)..... | 40 |
| 5.7 | SERVIDORES WEB..... | 40 |
| 5.7.1 | Servidor Jakarta Tomcat..... | 41 |
| 6. | DISEÑO METODOLÓGICO..... | 42 |
| 6.1 | PROTOTIPO EVOLUTIVO | 42 |
| 6.2 | LENGUAJE DE MODELADO UNIFICADO..... | 44 |
| 6.2.1 | Diagramas de UML | 44 |
| 6.3 | ESTÁNDARES DE PROGRAMACIÓN | 45 |
| 6.3.1 | Modelo de datos | 45 |
| 6.3.2 | Nombres de las tablas | 46 |
| 6.3.3 | Clases..... | 47 |
| 6.3.4 | Páginas JSP | 47 |
| 6.3.5 | Organización de Directorios..... | 47 |
| 7. | DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y | |
| | MANTENIMIENTO..... | 48 |
| 7.1 | PROTOTIPO ESPERADO..... | 48 |
| 7.1.1 | Diagramas de Casos de Uso | 58 |
| 7.1.2 | Documentación de casos de uso del sistema | 68 |
| 7.1.3 | Diseño y Análisis..... | 68 |
| 7.1.4 | Modelo de Procesos del Sistema..... | 76 |
| 7.1.5 | Implementación, Implantación y Pruebas Generales..... | 76 |
| 7.2 | MANTENIMIENTO Y ADMINISTRACIÓN | 77 |
| 7.2.1 | Actividades de Mantenimiento | 77 |
| 7.2.2 | Actividades de Soporte a Usuarios | 83 |

| | | |
|------------|-------------------------------------|-----------|
| 7.2.3 | Actividades de Administración | 84 |
| 8. | PRUEBAS DEL SISTEMA..... | 85 |
| 8.1 | PRUEBAS DE VERIFICACIÓN | 85 |
| 8.1.1 | Pruebas por componente..... | 85 |
| 8.2 | PRUEBAS DE INTEGRACIÓN..... | 85 |
| 8.3 | PRUEBAS DE VALIDACIÓN | 86 |
| 9. | CONCLUSIONES | 87 |
| 10. | RECOMENDACIONES | 88 |
| | BIBLIOGRAFÍA | 89 |
| | ANEXOS..... | 93 |

LISTA DE TABLAS

| | |
|--|-----|
| Tabla 1. Descripción de las Entidades..... | 73 |
| Tabla 2. Casos de uso: Sistema de notificaciones..... | 108 |
| Tabla 3. Caso de Uso: Enviar correos por nivel del semestre. | 108 |
| Tabla 4. Casos de uso: Correos por nivel de semestre. | 108 |
| Tabla 5. Casos de uso: Correos UIS. | 109 |
| Tabla 6. Casos de uso: Menú del profesor. | 110 |
| Tabla 7. Caso de Uso: Producción intelectual. | 110 |
| Tabla 8. Caso de Uso: Mantenimiento y Administración de los portales EIMTWEB y FIFMEWEB. | 111 |
| Tabla 9. Pruebas Realizadas: Enviar correo de notificación de foros a nivel de comunidad UIS. | 134 |
| Tabla 10. Pruebas Realizadas: Enviar correo de notificación de foros comunidad Escuela..... | 135 |
| Tabla 11. Pruebas Realizadas: Enviar correo de notificación de eventos comunidad UIS. | 136 |
| Tabla 12. Pruebas Realizadas: Enviar correo de notificación de eventos comunidad Escuela..... | 137 |
| Tabla 13. Pruebas Realizadas: Correos por nivel, selección programas académicos..... | 139 |
| Tabla 14. Pruebas Realizadas: Visualizar correos creados..... | 141 |
| Tabla 15. Pruebas Realizadas: Crear correos UIS. | 141 |
| Tabla 16. Pruebas Realizadas: Visualizar y eliminar destinatarios del correo UIS. | 143 |
| Tabla 17. Pruebas Realizadas: Eliminar correo UIS..... | 144 |
| Tabla 18. Pruebas Realizadas: Crear menú para el portal del profesor. | 144 |
| Tabla 19. Pruebas Realizadas: Visualizar menú actual/ Actualizar rutas de acceso. | 145 |
| Tabla 20. Pruebas Realizadas: Modificar menú actual..... | 147 |

| | |
|--|-----|
| Tabla 21. Pruebas Realizadas: Visualizar productos intelectuales..... | 148 |
| Tabla 22. Pruebas Realizadas: Ingresar producto intelectual..... | 149 |
| Tabla 23. Pruebas Realizadas: Modificar producto intelectual. | 150 |
| Tabla 24. Pruebas Realizadas: Consultar producción intelectual públicamente. . | 150 |
| Tabla 25. Pruebas Realizadas: Consultar producción intelectual de un grupo específico..... | 151 |
| Tabla 26. Pruebas Realizadas: Rechazar o aceptar inclusión en un producto intelectual..... | 152 |

LISTA DE FIGURAS

| | |
|---|----|
| Figura 1. Modelo Cliente/Servidor..... | 29 |
| Figura 2. Modelo de acceso a JSP | 33 |
| Figura 3. Prototipo Evolutivo..... | 42 |
| Figura 4. Diagrama de Casos de Uso: Enviar correo de notificación de foros comunidad UIS. | 58 |
| Figura 5. Diagrama de Casos de Uso: Enviar correo de notificación de foros comunidad Escuela..... | 59 |
| Figura 6. Diagrama de Casos de Uso: Enviar correo de notificación de eventos comunidad UIS. | 59 |
| Figura 7. Diagrama de Casos de Uso: Enviar correo de notificación de eventos comunidad Escuela..... | 59 |
| Figura 8. Diagrama de Casos de Uso: Selección de criterio por programa académico para envío de un correo..... | 60 |
| Figura 9. Diagrama de Casos de Uso: Selección de niveles para los programas académicos..... | 60 |
| Figura 10. Diagrama de Casos de Uso: Ver correos. | 61 |
| Figura 11. Diagrama de Casos de Uso: Crear correos UIS..... | 61 |
| Figura 12. Diagrama de Casos de Uso: Visualizar y eliminar destinatarios del correo UIS..... | 62 |
| Figura 13. Diagrama de Casos de Uso: Eliminar correo UIS..... | 62 |
| Figura 14. Diagrama de Casos de Uso: Visualizar menú actual del portal del profesor..... | 63 |
| Figura 15. Diagrama de Casos de Uso: Modificar menú actual. | 63 |
| Figura 16. Diagrama de Casos de Uso: Crear menú para el portal del profesor. .. | 64 |
| Figura 17. Diagrama de Casos de Uso: Visualizar productos intelectuales. | 64 |
| Figura 18. Diagrama de Casos de Uso: Ingresar producto intelectual..... | 65 |
| Figura 19. Diagrama de Casos de Uso: Modificar producto intelectual. | 65 |

| | |
|---|-----|
| Figura 20. Diagrama de Casos de Uso: Consultar producción intelectual públicamente..... | 66 |
| Figura 21. Diagrama de Casos de Uso: Consultar producción intelectual de un grupo específico..... | 66 |
| Figura 22. Diagrama de Casos de Uso: Rechazar o aceptar inclusión en un producto intelectual..... | 67 |
| Figura 23. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración. | 67 |
| Figura 24. Diagrama E/R: Módulo eventos comunidad escuela. | 68 |
| Figura 25. Diagrama E/R: Módulo eventos comunidad UIS..... | 69 |
| Figura 26. Diagrama E/R: Módulo foros comunidad escuela..... | 70 |
| Figura 27. Diagrama E/R: Módulo foros comunidad UIS | 70 |
| Figura 28. Diagrama E/R: Correos por nivel en Base de datos Diamante | 71 |
| Figura 29. Diagrama E/R: Correos por nivel en Base de datos Poseidon | 71 |
| Figura 30. Diagrama E/R: Correo UIS | 72 |
| Figura 31. Diagrama E/R: Módulo de producción intelectual | 73 |
| Figura 32. Arquitectura Cliente/Servidor de Dos Capas | 95 |
| Figura 33. Arquitectura Cliente/Servidor de Tres Capas..... | 96 |
| Figura 34. Función del DBMS..... | 103 |
| Figura 35. Diagramas de Casos de Uso | 104 |
| Figura 36. Diagrama de Secuencias..... | 107 |
| Figura 37. Diagrama de Secuencia: Envío de notificaciones de Eventos/foros. ... | 114 |
| Figura 38. Diagrama de Secuencia: Envío correo por programa académico-nivel. | 115 |
| Figura 39. Diagrama de Secuencia: Crear correo..... | 117 |
| Figura 40. Diagrama de Secuencia: Ver correo/Ver destinatarios/ Eliminar todo el correo..... | 119 |
| Figura 41. Diagrama de Secuencia: Eliminar destinatarios/ Agilizar envío del correo..... | 121 |
| Figura 42. Diagrama de Secuencia: Crear Menú/ Actualizar rutas. | 122 |

| | |
|---|-----|
| Figura 43. Diagrama de Secuencia: Modificar Menú. | 124 |
| Figura 44. Diagrama de Secuencia: Producción intelectual..... | 125 |
| Figura 45. Diagrama de Secuencia: Portal producción intelectual..... | 127 |
| Figura 46. Diagrama de Secuencia: Ingresar un producto intelectual/ Modificar un producto intelectual/ Eliminar un producto intelectual. | 129 |
| Figura 47. Diagrama de Secuencia: Gestionar Productos intelectuales | 132 |
| Figura 48. Pruebas Realizadas: Correos por nivel, selección de nivel | 139 |
| Figura 49. Pruebas Realizadas: Envío de correo por programa-nivel..... | 140 |

LISTA DE ANEXOS

| | |
|---|-----|
| Anexo A. ARQUITECTURA CLIENTE/SERVIDOR | 93 |
| Anexo B. TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS | 97 |
| Anexo C. BASES DE DATOS | 99 |
| Anexo D. LEGUAJE DE MODELADO UNIFICADO (UML) | 104 |
| Anexo E. DOCUMENTACIÓN DE CASOS DE USO DEL SISTEMA | 108 |
| Anexo F. MODELO DE PROCESOS DEL SISTEMA | 114 |
| Anexo G. PRUEBAS DEL SISTEMA | 134 |

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 METALÚRGICA Y LA FACULTAD DE INGENIERÍAS FISICOMECAÑICAS.ⁱ.

AUTORESⁱⁱ: Danay Faisury Quintero Castro

PALABRAS CLAVE: Producción Intelectual, Correo UIS, Eventos Comunidad UIS, Eventos Comunidad Escuela, Foros Comunidad UIS, Foros Comunidad Escuela, EIMTWEB (Escuela de Metalúrgica), FIFMEWEB (Facultad de Ingenierías FísicoMecánicas).

DESCRIPCIÓN

En la actualidad, la escuela de Metalúrgica y facultad de Ingenierías Fisicomecánicas, cuentan 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 las escuelas, 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 esta razón las labores de mantenimiento y actualización son indispensables.

El portal web de las Escuelas permite a sus usuarios crear conversaciones o proponer eventos que se pueden dirigir a la comunidad de la escuela o a las comunidades de varias escuelas. Consistente con nuestro enfoque comunidad UIS, se creó un sistema de notificaciones que permite informar lo acontecido en dichos eventos y foros a los usuarios relacionados con ellos.

El servicio Correos Comunidad UIS permite el envío de mensajes a los miembros de otras escuelas. Se creó un módulo para correos UIS en el cual el usuario puede hacer seguimiento de los correos y puede acelerar el envío de los mismos. También se agregó la opción de envío por programa académico y nivel que cursan los estudiantes.

Se creó un Portal de Producción Intelectual para presentar la información de la comunidad el cual muestra los productos propios de la comunidad como: artículos, ponencias, libros, software, etc. Así mismo para facilitar la gestión de la información se creó un servicio para ingresar, visualizar, editar o eliminar un producto intelectual.

Dentro de las funciones de los administradores de los Portales web de las diferentes Escuelas y Facultades, se encuentra la de generar y actualizar los menús que se mostrarán en el Portal de Profesor; esta labor se realizaba manualmente, por lo cual se automatizó esta tarea.

ⁱ Trabajo de grado: Modalidad Trabajo de Investigación.

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

ABSTRACT

TITLE: ADMINISTRATION, USERS SUPPORT, WEBSITE MAINTENANCE, ANALYSIS, DEVELOPMENT AND IMPLEMENTATION OF NEW SERVICES FOR THE WEB PORTAL OF METALLURGICAL SCHOOL AND FACULTY OF PHYSICAL MECHANICAL ENGINEERINGⁱ.

AUTHORSⁱⁱ: Danay Faisury Quintero Castro.

KEYWORDS: Intellectual Production, Mail UIS, Community forums UIS, Community forums school, Community events UIS, Community events school, EIMTWEB (Metallurgical School), FIFMEWEB (Faculty of Physical Mechanical Engineering).

DESCRIPTION

Nowadays the Metallurgical School and the Faculty of Physical Mechanical Engineering have a web oriented information system that handles the administration and control of the different academic and administrative activities that take place in the schools, as well as user control and services that are provided. These services provide solutions to the problems and needs of the users, for this reason maintenances and updates are indispensable.

The Web portal of each school allow the users to create forums or to propose events, those could be addressed to the community of the school, or the community of several schools. Consistent with our focus as UIS community, a notification system was created allowing to inform the things that happened after the creation of the forum or event to the users related with them.

The UIS postal community service allows to send mails to the members of other schools. Because of that, a mailing module for institutional mails was created, letting the user to follow up the e-mails and even speed up the sending of them. Also the "send by academic program" and "level that students are coursing" option were added.

An intellectual production portal was developed to present the information to the community, displaying their own products, such as articles, papers and software. Likewise, a service to log in, view, edit or delete an intellectual product was implemented, to make easier the management of information.

Among the functions of administrators of web portals of each schools and faculty, is found the generate and update the displayed menus on the teacher's portal, this work was done by hand, for this reason this work was automated.

ⁱ Degree Work: Research Paper.

ⁱⁱ Faculty of Physical Mechanical Engineering, School of System Engineering and Computing Science.

Director: M.Sc. Luis Ignacio González Ramírez

INTRODUCCIÓN

Debido al crecimiento de la comunidad de la Universidad Industrial de Santander e inicialmente la comunidad de la escuela de Ingeniería de Sistemas, se hace necesario un canal de información y control dentro 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 comunidad, desarrollo llevado a cabo en el año 2004 por el grupo de desarrollo de software Calumet, el cual, conformado y liderado por estudiantes de pregrado de Ingeniería de Sistemas, da vida a la primera versión del portal web de la escuela de Ingeniería de Sistemas EISIWeb.

Dado el crecimiento y la acogida del portal web EISIWeb, apoyado en 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.

Los portales EIMTWEB y FIFMEWEB pertenecientes a la escuela de Metalúrgica y a la facultad de Ingenierías Fisicomecánicas, se han convertido en uno de los principales canales de comunicación e integración entre los miembros de la comunidad de la escuela y la facultad. 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 y la facultad.

El grupo de desarrollo de software Calumet se encarga de desarrollar, administrar y mantener los portales EIMTWEB y FIFMEWEB a los cuales se les realizan continuos cambios y actualizaciones para hacer de estos 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 portal web de las Escuelas, en línea con su objetivo de aportar lugares para la comunidad, permite a sus usuarios crear conversaciones o proponer eventos que se pueden dirigir a la comunidad de la escuela o a las comunidades de varias escuelas (comunidad Uis). La creación de conversaciones y eventos permiten el envío de correos a las personas de la comunidad con el fin de invitarlas a la participación de diferentes formas. Consistente con nuestro enfoque comunidad UIS, se hace necesario generar mensajes de notificación que permitan al creador, como a los demás participantes enterarse de lo acontecido a partir de su creación.

Dentro de las herramientas que ofrece el Portal Web de las Escuelas para acercar a la comunidad se encuentra el servicio de Correos Comunidad UIS que permite el envío de mensajes a los miembros de otras escuelas. Desde este, se pueden programar mensajes por diferentes opciones a diferentes usuarios de las diferentes escuelas y decanatos. Una vez enviado el mensaje con sus destinos, al usuario remitente solo se le informa que los datos han sido guardados en bases de datos para su envío, y un mensaje a su correo personal le entregará un informe al finalizar el envío. Surge la necesidad de crear un servicio que permita el usuario hacer seguimiento al envío, así como la posibilidad de eliminar destinatarios e incluso todo el mensaje. Permitirá observar cómo se van enviando los mensajes e incluso acelerar el envío de los de mayor prioridad. En el servicio de Correos Comunidad UIS se hace necesario agregar la opción de envío por escuela, programa y nivel que cursan los estudiantes de una determinada carrera.

Brindando respuesta a las necesidades de la comunidad, se creó el Portal de Producción Intelectual de los miembros de las escuelas. De esta manera, se conformaron cuatro interfaces diferentes para presentar la información de la

comunidad de las escuelas, estas son: Portal de la Escuela, Portal de los Grupos, Portal del Profesor y, por último, Portal de Producción Intelectual. Este último muestra 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 contiene toda la información sobre el producto, así como los docentes y grupos relacionados con ellos. Surge la necesidad de crear el servicio inicial de modificar, eliminar e ingresar productos intelectuales, así mismo crear los enlaces desde el portal Escuela, Portal de profesor, Portal de grupos y Portal de producción intelectual.

Dentro de las funciones de los administradores de los Portales web de las diferentes Escuelas y Facultades, se encuentra la de generar y actualizar los menús que se mostrarán en el Portal de Profesor. Esta labor se realiza por un servicio creado para este fin. Para actualizar los menús del portal de profesor en este momento, es necesario eliminar manualmente de la Base de Datos los registros en la tabla correspondiente, para de esta forma volver a crearlos todos. Por consiguiente, surge la necesidad de crear un servicio que consulte la tabla, muestre los menús vigentes, y actualice rutas del menú actual, de tal forma que sea sencillo actualizarlos, reubicarlos, eliminarlos o agregar uno nuevo.

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.

1. DEFINICIÓN DEL PROBLEMA

En la actualidad la escuela de Metalúrgica y la facultad de Ingenierías Fisicomecánicas cuentan 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 estas, así como del control de usuarios y servicios que se les 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.

Actualmente existen los servicios de Eventos y Foros, los cuales tienen la función de integrar a la comunidad y motivarla a participar en temas referentes a los planteados, o en un tema que desee proponer. Consistente con nuestro enfoque Comunidad UIS, surge la necesidad de crear un sistema de notificaciones a nivel comunidad escuela y a las comunidades de varias escuelas (Comunidad UIS), que permita informar a los usuarios que participaron y al creador, de lo acontecido a partir de su creación.

El módulo Correos UIS que presenta los portales de las escuelas permite de manera sencilla crear un correo. Sin embargo el módulo no cuenta con un canal de comunicación entre el creador y lo que sucede con el correo, por lo cual surge la necesidad de crear un módulo que permita al usuario hacer seguimiento al envío, así como la posibilidad de eliminar destinatarios e incluso todo el mensaje, y poder visualizar el contenido de dicho mensaje, a su vez que el usuario pueda acelerar el envío de los correos con mayor prioridad.

Actualmente existe un servicio de producción intelectual, el cual presenta una interfaz antigua y limitada, se ve la necesidad de crear un nuevo módulo con una mejor interfaz e incorporar más funcionalidades que permita al usuario crear, modificar, y eliminar los diferentes productos intelectuales que haya creado. También se hace necesario realizar una reingeniería al portal de Producción Intelectual, mejorando su interfaz y funcionalidad. Incorporando enlaces entre los diferentes portales, los cuales son: portal de grupos, portal de profesores y portal de la escuela.

Dentro de las funciones de los administradores de los Portales web de las diferentes Escuelas y Facultades, se encuentra la de generar y actualizar los menús que se mostrarán en el Portal de Profesor. Esta labor se realiza por un servicio creado para este fin. Sin embargo solo permite crear dicho Menú; por lo cual es necesario crear un módulo que permita visualizar el menú actual, crear, modificar, eliminar el menú, y actualizar las rutas del menú actual.

2. 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 implantó un sistema de notificaciones para los servicios de eventos y foros, a nivel comunidad escuela y UIS, con el fin de ofrecer un canal de comunicación entre los usuarios que participan en ellos, y con ello informar a los participantes lo acontecido con dicha propuesta.

Dando respuesta a las necesidades de la comunidad se implementó un módulo de Correos UIS, el cual permite llevar un control de los correos creados, y el estado de los mismos, a su vez, permite agilizar el envío, y visualizar los resultados en tiempo real. También se incorporó la opción de envío por escuela, programa y nivel que cursan los estudiantes de una determinada carrera.

Respondiendo a las necesidades de los usuarios se desarrolló e implemento un servicio de Producción Intelectual, el cual permite visualizar, modificar, eliminar, los diferentes productos intelectuales que haya creado el usuario. Con el fin de que el usuario tome el control de los productos realizados y pueda disponer de ellos cuando lo requiera, y no como antes, que después de haberlo creado, solo podía visualizarlo desde el Portal de Producción Intelectual.

Se realizó reingeniería al portal de Producción intelectual, el cual consistió en realizar una interface amigable para el usuario, y ampliar la navegabilidad entre los

diferentes portales que ofrece el grupo Calumet a la comunidad, los cuales son: portal de grupos, portal del profesor, y portal de la escuela.

Con el fin de optimizar los procesos realizados por los administradores de los diferentes portales con los que cuenta Calumet, se creó un nuevo servicio para mejorar la creación del Menú del Profesor, con el cual se logró optimizar los procesos de creación, modificación, eliminación y actualización; debido a que antes se realizaba manualmente esta labor.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Realizar las funciones de administración, soporte a usuarios, mantenimiento del portal web, análisis, desarrollo e implementación de nuevos servicios para el portal web de la escuela de Metalúrgica y la Facultad de Ingenierías Fisicomecánicas para hacer óptimos, eficientes y sostenibles los trámites dentro de las escuelas y el acceso a la información de las mismas.

3.2 OBJETIVOS ESPECÍFICOS

Efectuar labores de Administración del Portal web de la Escuela de Metalúrgica y Facultad de Ingenierías Fisicomecánicas 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 de los portales EIMTWEB y FIFMEWEB, 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 de los portales web.
- Efectuar una revisión constante de los archivos que se suben al sitio, eliminando los que no son necesarios para evitar saturación de los portales.

Llevar a cabo labores de mantenimiento a los portales EIMTWEB y FIFMEWEB, 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.
- Revisar y depurar la estructura de directorios y archivos de los portales web.

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

- Correos de notificación diarios de las actividades realizadas el día anterior en el Portal por los usuarios cuando crean y/o participan en foros y eventos.
- Envío de correos por nivel que cursan los estudiantes de un programa
- Modificar el servicio de correos comunidad UIS para que el usuario pueda, una vez enviado el mensaje, mediante una interface amigable revisar el asunto y contenido del correo, así como observar el avance de su envío pudiendo eliminar destinatarios e incluso todo el correo si lo considera necesario.
- Para el Portal de Producción Intelectual, que se viene desarrollando por el grupo, crear el servicio que permita el ingreso de un producto nuevo y la

modificación o eliminación de uno ya existente. Además, se debe crear la posibilidad de navegar entre los Portales de Profesor, Producción Intelectual, Grupos y Escuela.

- Para los Portales de Profesor se hace necesario mejorar el servicio que permite a los administradores agregar, eliminar, cambiar el orden de los menús que se presentan a estudiantes y profesores a través de ellos.

Desempeñar labores de soporte a los usuarios de los portales EIMTWEB y FIFMEWEB, 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 los portales EIMTWEB y FIFMEWEB, 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 o la facultad y que por algún motivo necesitan registrarse en el sitio.
- Modificar los estados de los usuarios de acuerdo a la relación con la Escuela o la Facultad (Activo, Inactivo, Suspendido).

Capacitar a los estudiantes que recibirán las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro de los portales EIMTWEB y FIFMEWEB 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.

4. ALCANCES Y LIMITACIONES

El sistema de notificaciones de eventos y foros a nivel de Comunidad escuela y UIS, brinda información a los participantes y al creador de dicha propuesta y/o foro, de los diferentes acontecimientos después de haber sido creado.

El módulo de Correos UIS, mediante una interface amigable permite la visualización de los correos que ha creado el usuario, con la información del asunto y los destinatarios del mismo, a su vez, da la opción de revisar el estado del envío en tiempo real, y poder agilizar el envío de sus correos. También cuenta con la funcionalidad de crear un correo, y poder enviarlo a estudiantes por el nivel que cursan.

En el módulo de producción intelectual mediante una interface amigable se puede visualizar los diferentes productos intelectuales ingresados y categorizados por el tipo del mismo. A su vez permite el ingreso de un producto nuevo, la modificación o eliminación de uno ya existente.

En el portal de Producción intelectual se realizó una reingeniería a la interface, convirtiéndola más amigable al usuario, a su vez se incorporó la posibilidad de navegar entre los portales de profesor, producción intelectual, grupos y escuela, con eso facilitar la navegabilidad entre los mismos.

En el portal de profesores se mejoró el servicio que permite a los administradores crear un menú del profesor. Ahora cuenta con nuevas funcionalidades, en las cuales se encuentran, la posibilidad de crear un menú, agregar, eliminar, actualizar las rutas o cambiar el orden de los menús que se presentan a estudiantes y profesores a través de ellos. Con lo cual se consiguió optimizar la labor del administrador.

5. MARCO TEÓRICO

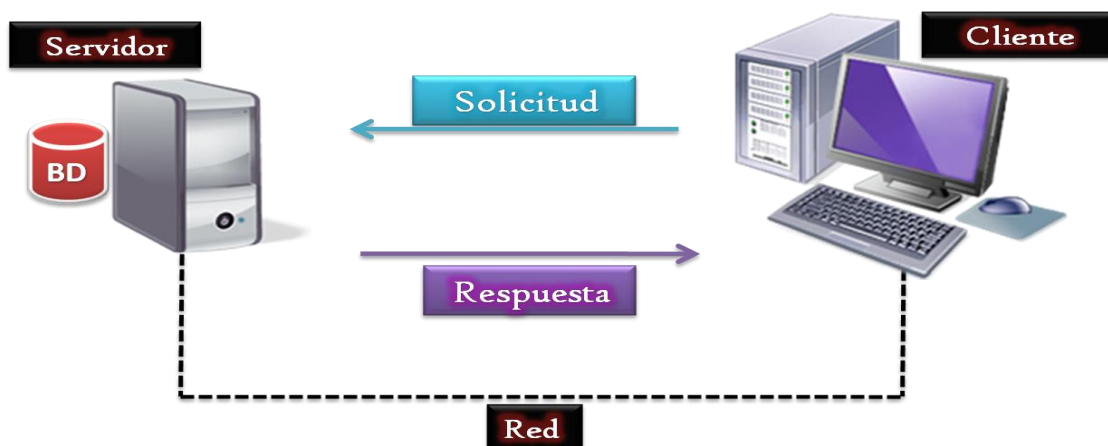
5.1 ARQUITECTURA CLIENTE/SERVIDOR

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: Arquitectura cliente servidor. [en línea].

<http://ulyla003.wordpress.com/page/4/>[citado el 01 Mayo de 2014]

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

5.1.1 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.

5.1.2 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.

5.1.3 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 diagnostico 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.

5.2 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).

5.2.1 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 páginas 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.

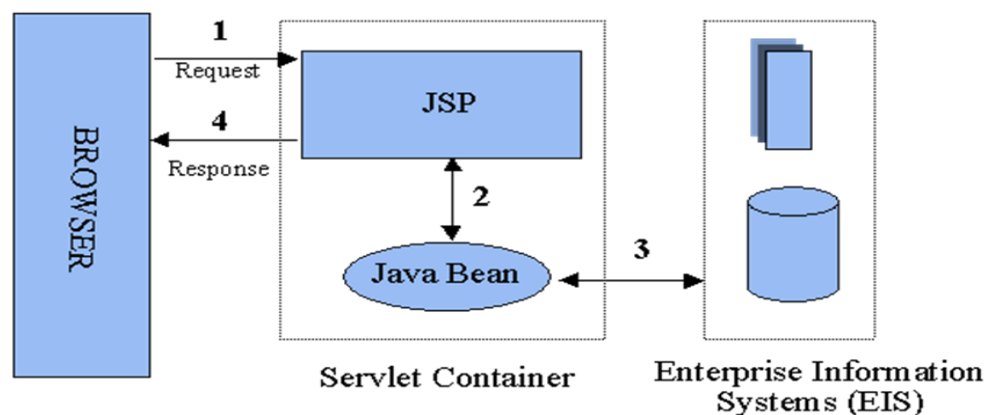
Una página JSP se compila a una aplicación Java la primera vez que se invoca, y de esta aplicación Java se crea una clase que empieza a ejecutarse en el servidor como un servlet. Un JSP es una página web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa que recibe 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: Modelo JSP. [en línea]. <http://geneura.ugr.es/~jmerelo/JSP/modelo1.gif> [citado el 05 Mayo de 2014]

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.

5.3 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 GEOWEB y FIFMEWEB se utiliza MySQL.

5.3.1 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.

5.4 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.

5.5 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.

5.5.1 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.

5.6 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.

5.6.1 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.

5.6.2 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).

5.6.3 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.

5.6.4 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.

5.6.5 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.

5.6.6 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.

5.6.7 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.

5.7 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.

5.7.1 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 la 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.

6. DISEÑO METODOLÓGICO

6.1 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.



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

- Los portales EIMTWEB y FIFMEWEB presentan 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 de los portales EIMTWEB y FIFMEWEB 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.

6.2 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.

6.2.1 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).

6.3 ESTÁNDARES DE PROGRAMACIÓN

6.3.1 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).

6.3.2 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 los puntos que se tratan en un consejo de escuela es llamada “TB_PuntosConsejo”.
- *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

EIMTWEB. El prefijo a anteponer a los nombres de estas tablas es “TP_”, es decir la tabla “Usuarios”, es conocida como “TP_ Usuarios”.

6.3.3 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: ProductoIntelectual.java, PortalProfesor.java.

6.3.4 Páginas JSP: Los nombres de las paginas JSP que componen los portales EIMTWEB y FIFMEWEB 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: IngresarProductoIntelectual.jsp, CorreosPorPoseidon.jsp.

6.3.5 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.
- El sitio cuenta con un directorio llamado Script, donde están todos los archivos de JavaScript “.js” necesarios para el buen funcionamiento del portal.

7. 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.

7.1 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. Crear un sistema de notificación para los eventos y foros a nivel de comunidad escuela y UIS, generando correos diarios con un informe a los participantes y al creador. En el informe debe explicar que usuarios han**

participado y/o unido a la propuesta o foro respectivo. Si no hay notificaciones del evento o el foro, no se debe enviar ningún correo.

Objetivo Inicial:

- Correos de notificación diarios de las actividades realizadas el día anterior en el Portal por los usuarios cuando crean y/o participan en foros y eventos.

Requisitos finales del objetivo:

- Los correos de notificación generados, deben enviarse a los usuarios involucrados en dicha propuesta o foro, y al creador del mismo.
- Para los eventos a nivel de comunidad escuela, el correo generado debe especificar el nombre de los usuarios que han participado o se han unido a la propuesta.
- Para los eventos a nivel de comunidad UIS, el correo generado debe especificar el nombre de los usuarios y la escuela a la que pertenecen, los participantes o nuevos miembros de la propuesta.
- Referente a los foros a nivel de comunidad escuela, en el correo de notificación debe especificar el nombre del usuario que ha participado.
- Referente a los foros a nivel de comunidad UIS, en el correo de notificación debe especificar el nombre y la escuela a la cual pertenece, el usuario que ha participado.
- Generar correos de notificación en horas no laborales, para evitar el tráfico en el envío de datos.

2. Implementar la posibilidad de enviar Correos a los estudiantes por el nivel de semestre en el que se encuentran.

Objetivo inicial:

- Envío de correos por el nivel que cursan los estudiantes de un programa.

Requisitos finales del objetivo:

- Permitir enviar un correo a diferentes programas académicos.
- Permitir seleccionar escuelas y habilitarle al usuario los programas académicos relacionados a dichas escuelas
- Permitir enviar un correo a uno o varios niveles de semestre de uno o varios programas académicos con el mismo tiempo de duración.
- Permitir enviar un correo a todos los niveles de semestre, si los programas académicos seleccionados son de diferente duración.

3. Implementar un módulo de Correos UIS, que permita la visualización de los correos creados por el usuario, la opción de crear un nuevo correo, visualizar el asunto del correo, y el estado del envío. Especificando la fecha respectiva del envío, si se ha enviado, o si se encuentra en espera. También se requiere la posibilidad de que el usuario pueda agilizar el envío de los correos que tengan mayor prioridad de ser entregados. A su vez permitir la eliminación del correo si y solo si no se haya empezado a enviar los correos a los destinatarios.

Objetivo inicial:

- Modificar el servicio de correos comunidad UIS para que el usuario pueda, una vez enviado el mensaje, mediante una interface amigable revisar el asunto y

contenido del correo, así como observar el avance de su envío pudiendo eliminar destinatarios e incluso todo el correo si lo considera necesario.

Requisitos finales del objetivo:

- Permitir al usuario visualizar los correos que ha creado mediante una interfaz amigable, en la cual pueda también visualizar el estado del correo, la fecha en la cual se creó el correo, y el asunto del correo.
 - Permitir en la interfaz anterior descrita eliminar un correo si y solo si no se ha empezado a enviar ningún correo a los destinatarios.
 - El usuario podrá crear un nuevo correo.
 - Permitir a los usuarios poder visualizar el contenido de cada correo, como el asunto, y el mensaje.
 - Permitir al usuario poder visualizar los destinatarios del respectivo correo, mostrándole el estado de envío de cada correo del destinatario.
 - Permitir al usuario agilizar el envío de los correos, dándole la opción de enviar el correo en tiempo real, si y solo si las cuentas de correo para cada escuela se encuentran disponibles para enviar correos.
 - El usuario podrá eliminar destinatarios de su correo, si y solo si, dicho mensaje no se ha empezado a enviar.
- 4. Modificar el servicio de Producción Intelectual que permita visualizar los productos intelectuales que haya digitado el usuario, ingresar un nuevo producto intelectual, modificar y eliminar uno ya existente. A su vez implementar en el portal de Producción intelectual la posibilidad de navegar entre los portales de Profesor, Producción Intelectual, Grupos y Escuela.**

Objetivo inicial:

- Para el Portal de Producción Intelectual, que se viene desarrollando por el grupo, se requiere modificar su interfaz, crear el servicio que permita el ingreso de un producto nuevo y la modificación o eliminación de uno ya existente. Además, se debe crear la posibilidad de navegar entre los Portales de Profesor, Producción Intelectual, Grupos y Escuela.

Requisitos finales del objetivo:

- Diseñar una interfaz amigable para el usuario, en la cual encuentre fácilmente información acerca de los diferentes productos intelectuales.
- Permitir al usuario ingresar un nuevo producto intelectual.
- Permitir al usuario modificar un producto intelectual ya existente.
- Permitir al usuario eliminar un producto intelectual ya existente.
- Permitir al usuario llevar un control de todos los productos intelectuales que ha ingresado, por medio de una interfaz amigable, en la cual pueda visualizar el nombre, el estado, y el año de publicación de dicho producto.
- Permitir al usuario visualizar los diferentes productos intelectuales en los cuales lo han incluido. Dar la posibilidad al usuario de aceptar o rechazar la inclusión en dicho producto.
- Permitir al administrador de cada grupo de investigación o de desarrollo, conocer cuando han incluido al grupo en un determinado producto. Dar la posibilidad de aceptar o rechazar dicha inclusión.
- Permitir al público (usuarios no logueados en el portal) visualizar la producción intelectual que poseen los diferentes profesores catedra y planta de determinada escuela, a su vez permitir visualizar la producción que ha realizado cada grupo de dicha escuela.
- Facilitar al usuario el desplazamiento entre los portales de Producción Intelectual, Profesor, Grupos y Escuela.

- Realizar reingeniería en el portal de producción intelectual, mejorando la interfaz e implementando nuevos servicios.
- 5. Crear un servicio que permita visualizar el menú actual del profesor, si existe uno, y permitir modificarlo, como agregar, eliminar o cambiar de posición los menús actuales o nuevos menús. A su vez permitir la creación del menú del profesor, si no hay uno actual**

Objetivo inicial:

- Para los Portales de Profesor se hace necesario mejorar el servicio que permite a los administradores agregar, eliminar, cambiar el orden de los menús que se presentan a estudiantes y profesores a través de ellos

Requisitos finales del objetivo:

- Permitir visualizar el menú del profesor actual.
- Permitir actualizar rutas de los diferentes servicios que se encuentran en el menú actual del profesor.
- Permitir modificar el menú actual del profesor, agregando nuevos menús, eliminando menús existentes o cambiar de ubicación los menús existentes en el menú actual.
- Permitir crear un menú del profesor, si no hay uno existente.

Para aquellos servicios que lo requieran, se debe incluir la rutina de logueo 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 de los portales web de la escuela de Metalúrgica y la facultad de Ingenierías Fisicomecánicas; 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 de los portales EIMTWEB y FIFMEWEB, 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 a los portales EIMTWEB y FIFMEWEB 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, Diamante y de los directorios de los sitios EIMTWEB y FIFMEWEB
- 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 de los portales EIMTWEB y FIFMEWEB, 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 de los portales EIMTWEB y FIFMEWEB, 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 los portales EIMTWEB y FIFMEWEB 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 de los portales EIMTWEB y FIFMEWEB, 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 de los portales EIMTWEB y FIFMEWEB 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.

7.1.1 Diagramas de Casos de Uso

- **Sistema de notificaciones para Eventos y Foros**

Figura 4. Diagrama de Casos de Uso: Enviar correo de notificación de foros comunidad UIS.



Figura 5. Diagrama de Casos de Uso: Enviar correo de notificación de foros comunidad Escuela.



Figura 6. Diagrama de Casos de Uso: Enviar correo de notificación de eventos comunidad UIS.

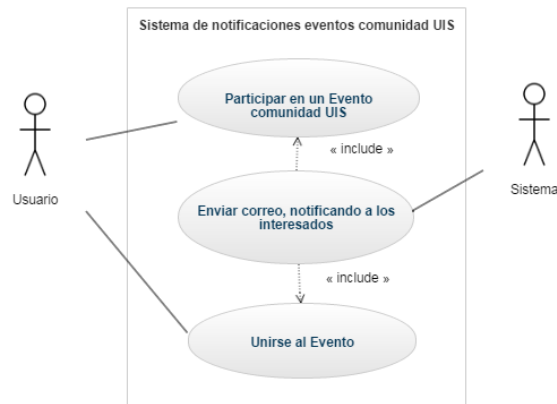
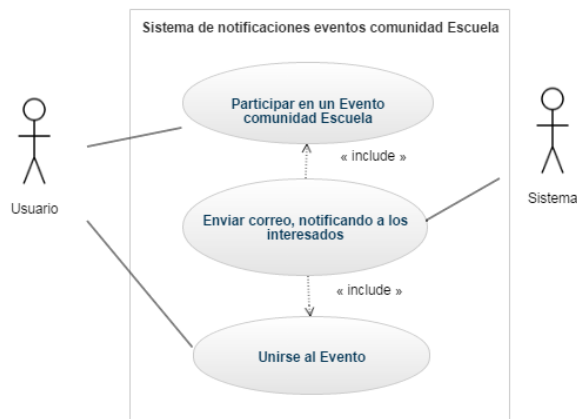


Figura 7. Diagrama de Casos de Uso: Enviar correo de notificación de eventos comunidad Escuela.



- **Correos por nivel**

Figura 8. Diagrama de Casos de Uso: Selección de criterio por programa académico para envío de un correo.

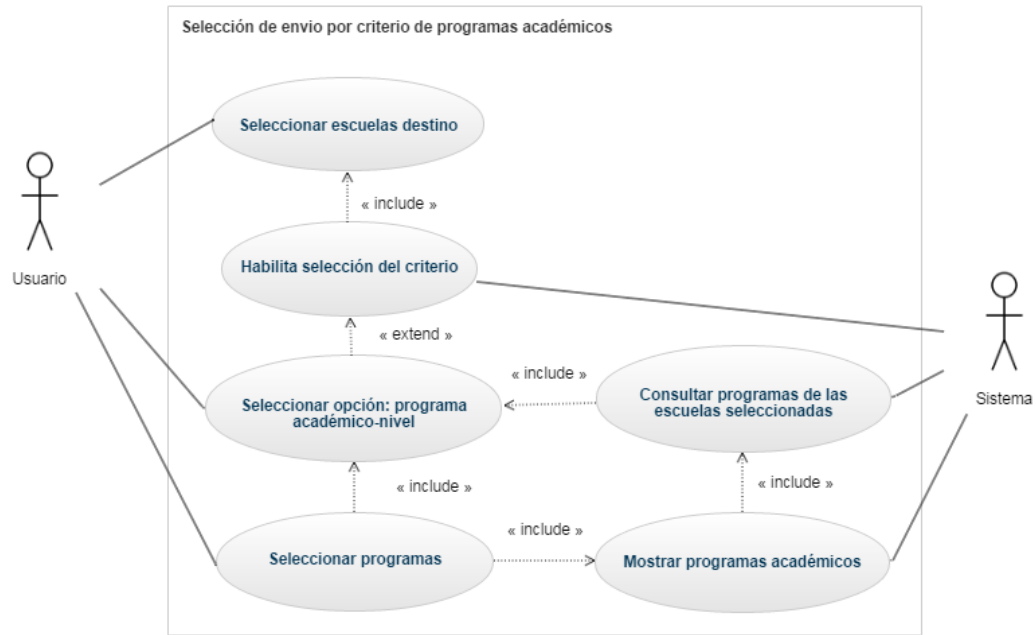
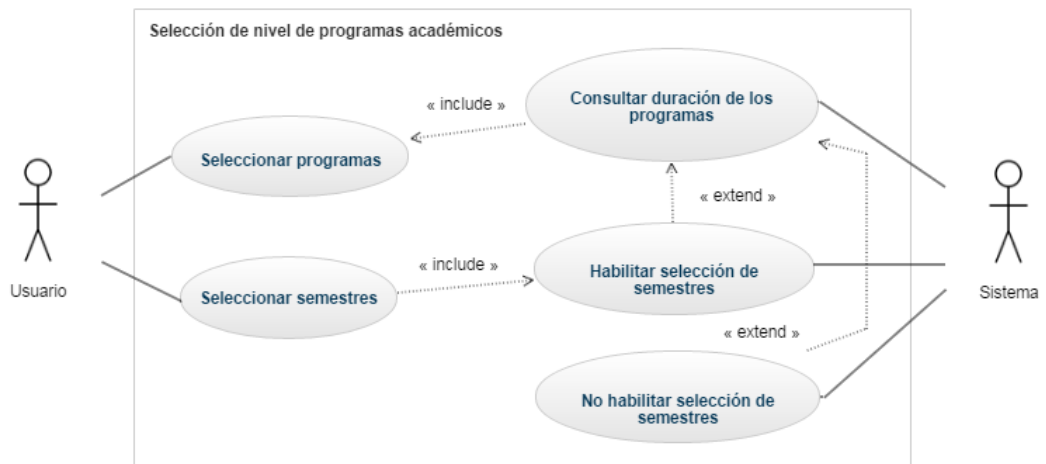


Figura 9. Diagrama de Casos de Uso: Selección de niveles para los programas académicos.



- **Correos UIS**

Figura 10. Diagrama de Casos de Uso: Ver correos.

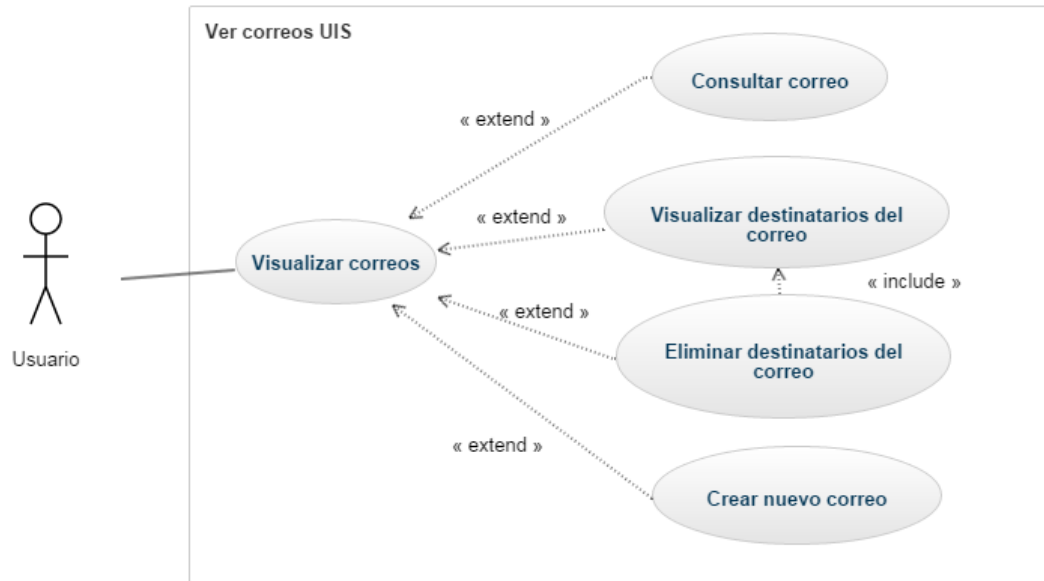


Figura 11. Diagrama de Casos de Uso: Crear correos UIS.

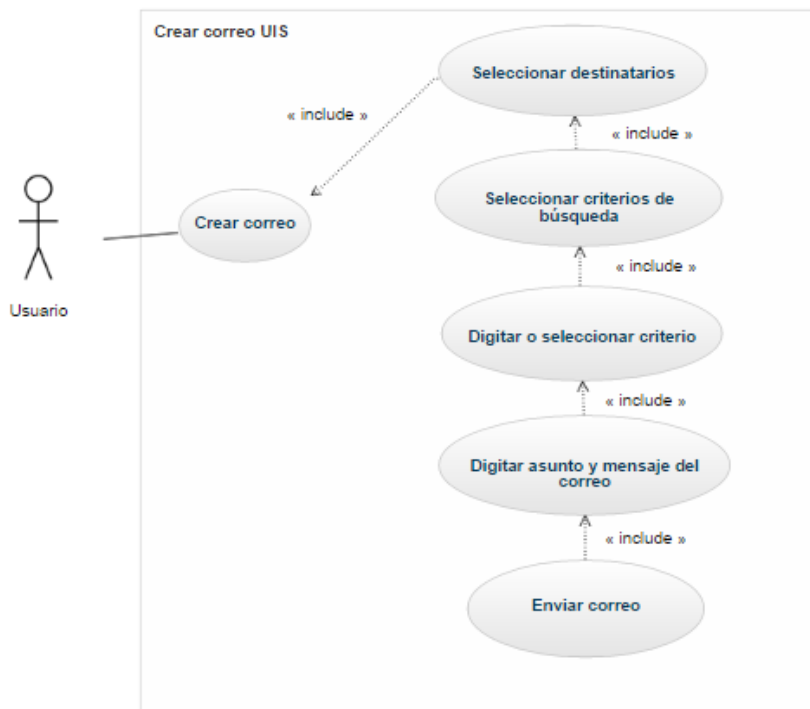


Figura 12. Diagrama de Casos de Uso: Visualizar y eliminar destinatarios del correo UIS.

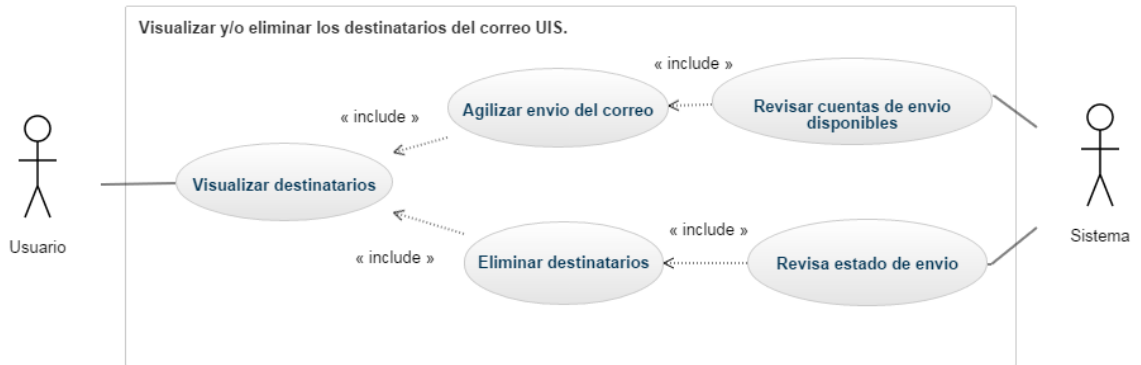
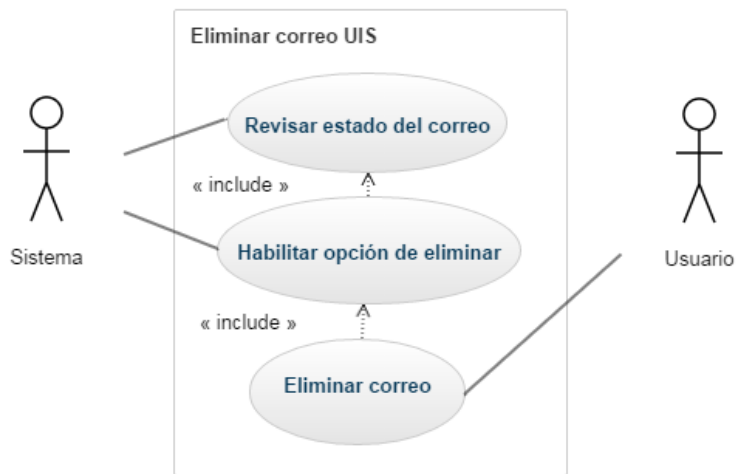


Figura 13. Diagrama de Casos de Uso: Eliminar correo UIS.



- **Menú del portal del profesor**

Figura 14. Diagrama de Casos de Uso: Visualizar menú actual del portal del profesor.

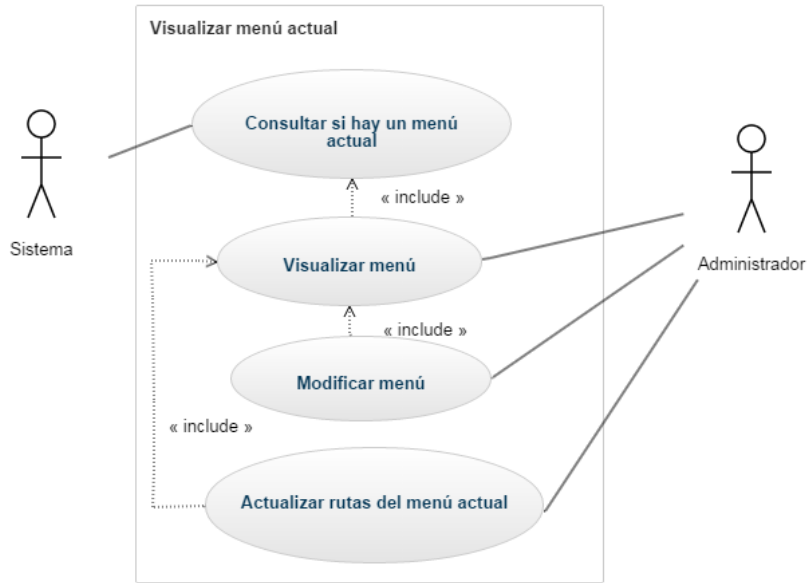


Figura 15. Diagrama de Casos de Uso: Modificar menú actual.

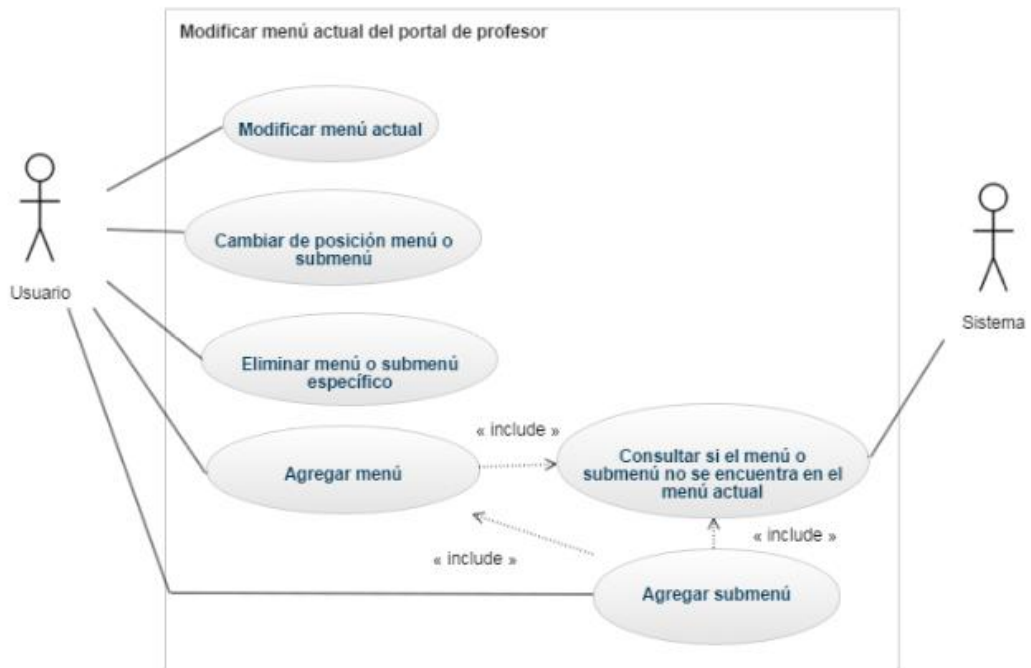
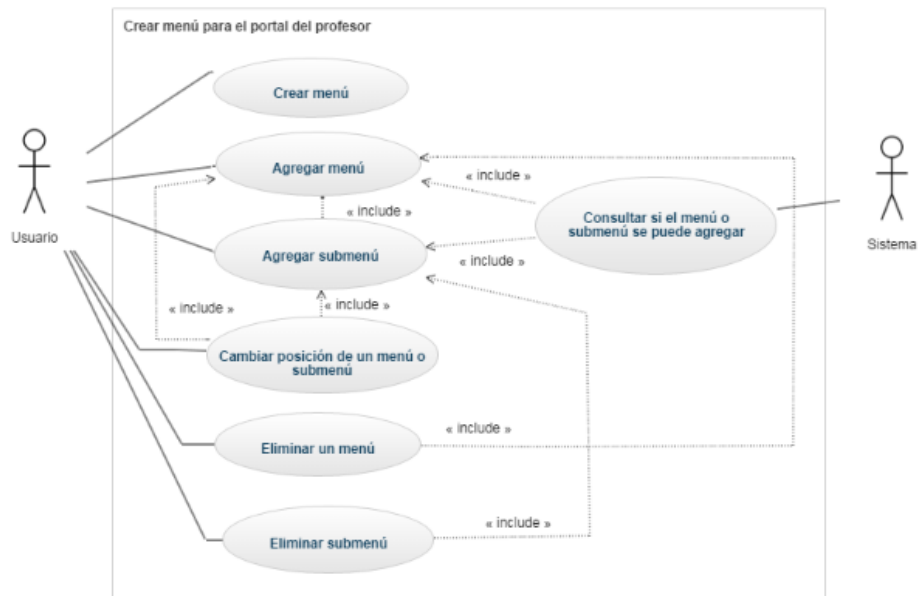


Figura 16. Diagrama de Casos de Uso: Crear menú para el portal del profesor.



- **Producción Intelectual**

Figura 17. Diagrama de Casos de Uso: Visualizar productos intelectuales.

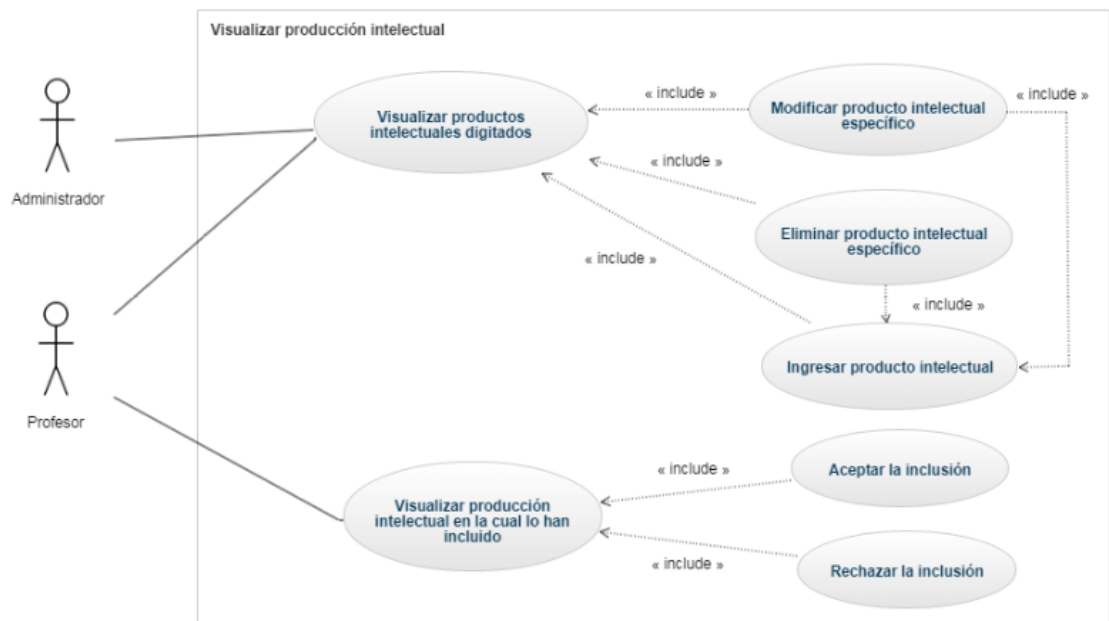


Figura 18. Diagrama de Casos de Uso: Ingresar producto intelectual.

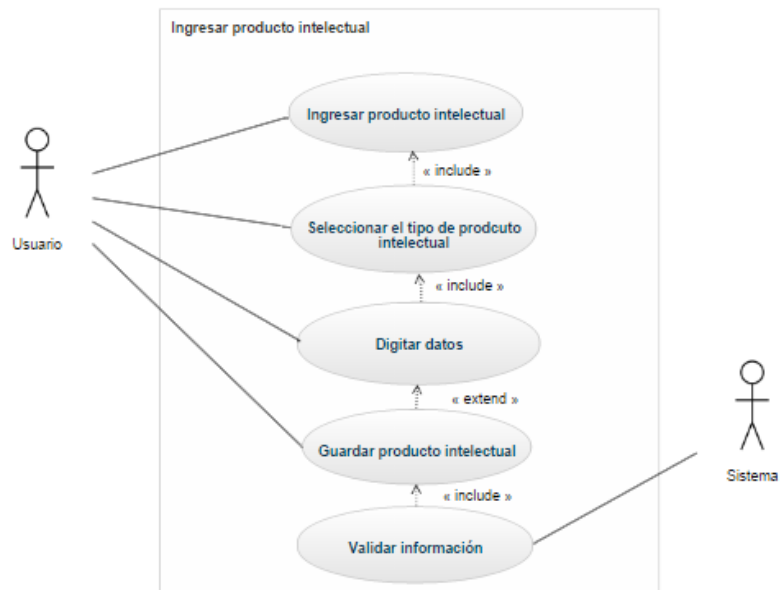


Figura 19. Diagrama de Casos de Uso: Modificar producto intelectual.

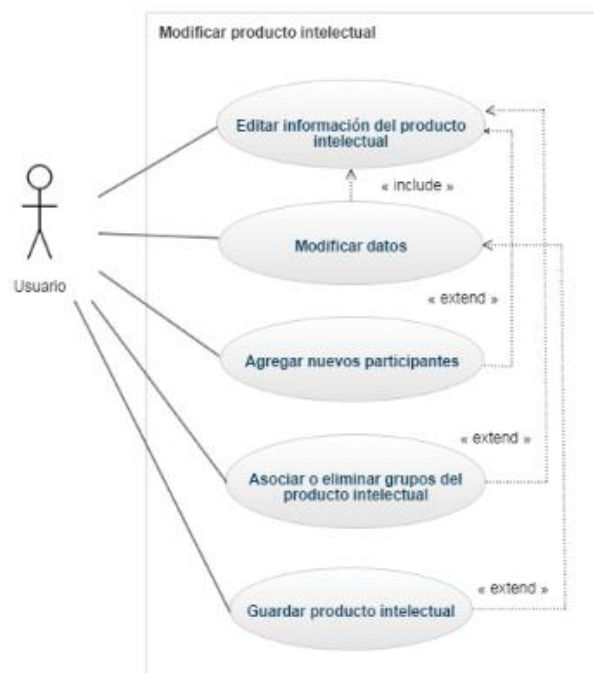


Figura 20. Diagrama de Casos de Uso: Consultar producción intelectual públicamente.

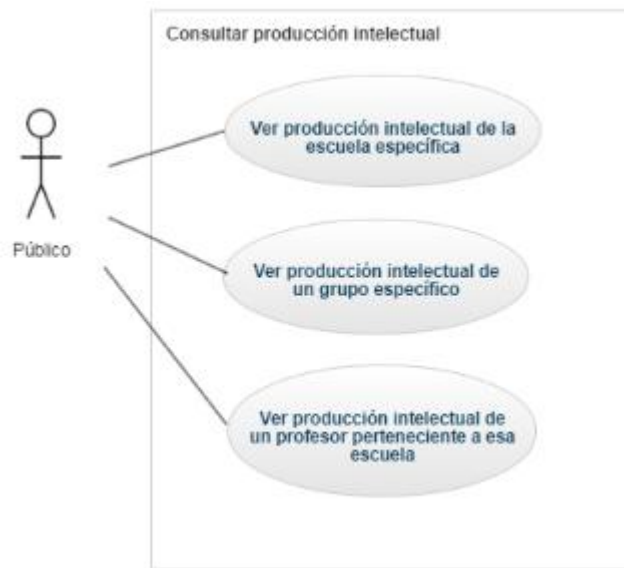


Figura 21. Diagrama de Casos de Uso: Consultar producción intelectual de un grupo específico.

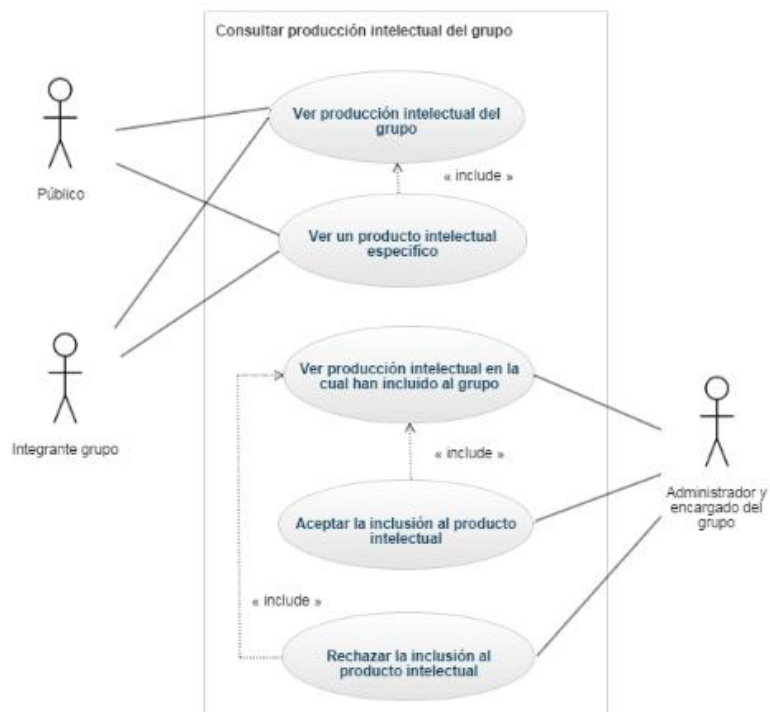
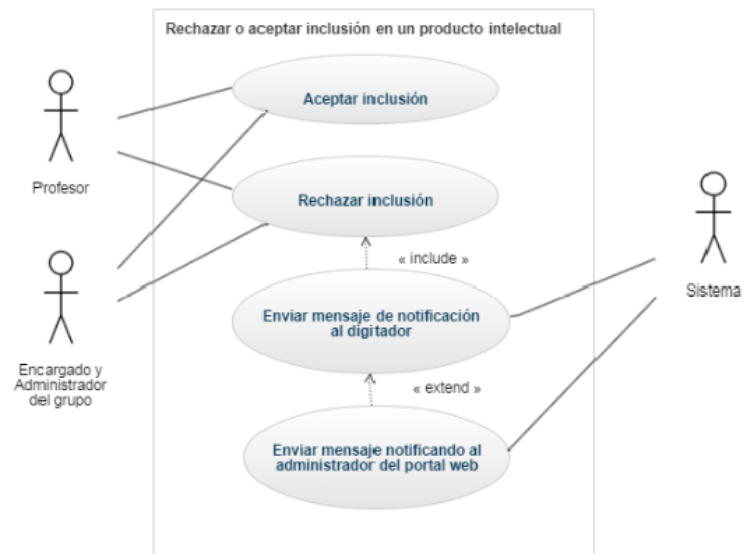
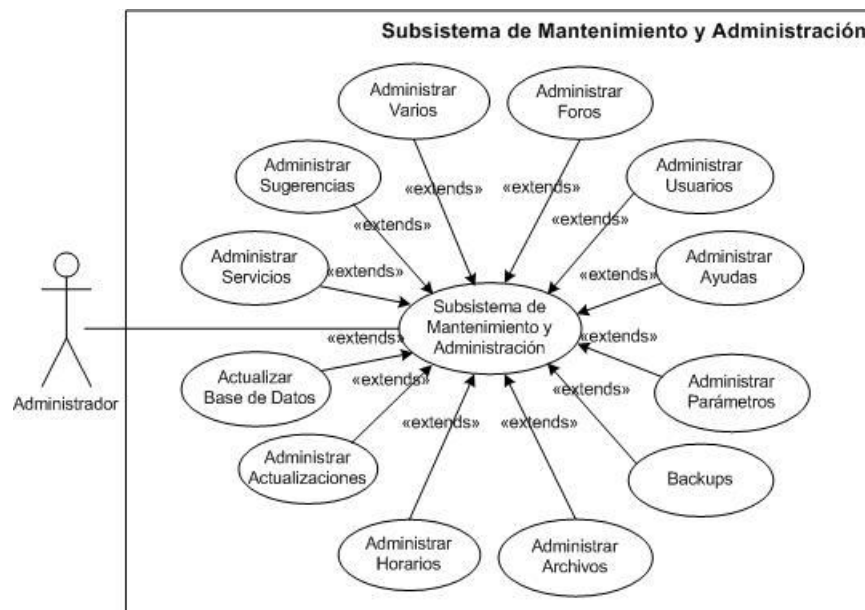


Figura 22. Diagrama de Casos de Uso: Rechazar o aceptar inclusión en un producto intelectual



- **Mantenimiento y Administración de los portales EIMTWEB y FIFMEWEB**

Figura 23. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.



7.1.2 Documentación de casos de uso del sistema: Una vez identificados los casos de uso, procedemos a documentar sus pasos, describiendo cada una de las acciones que se llevan a cabo en los casos de uso anteriormente mencionados. Los detalles completos se documentan en los anexos de este libro. (Ver Anexo E)

7.1.3 Diseño y Análisis

7.1.3.1 Diagrama Entidad/Relación de los servicios desarrollados

- **Eventos comunidad escuela**

Figura 24. Diagrama E/R: Módulo eventos comunidad escuela.

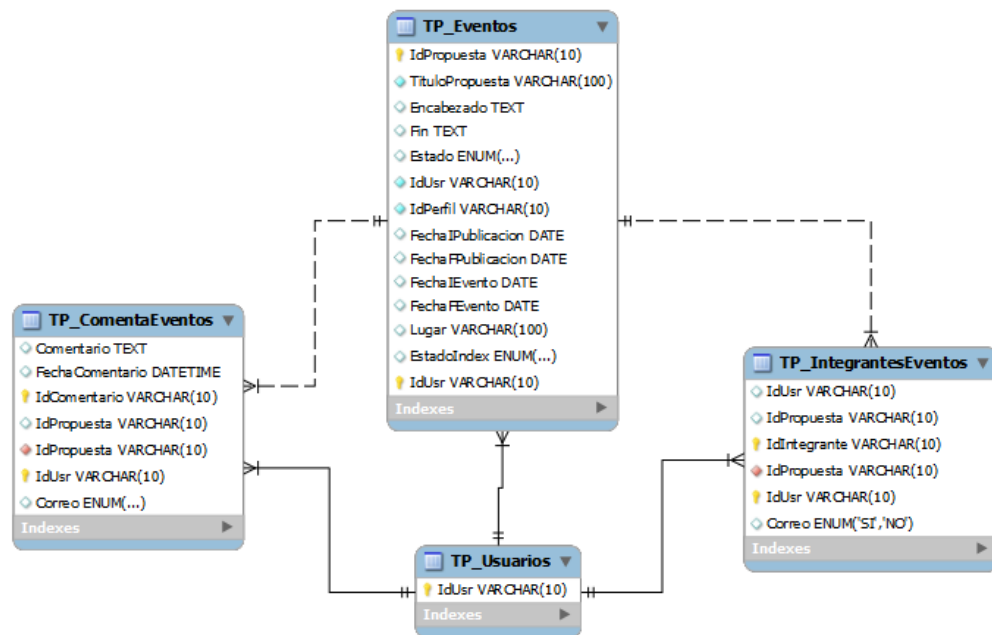


Figura 25. Diagrama E/R: Módulo eventos comunidad UIS

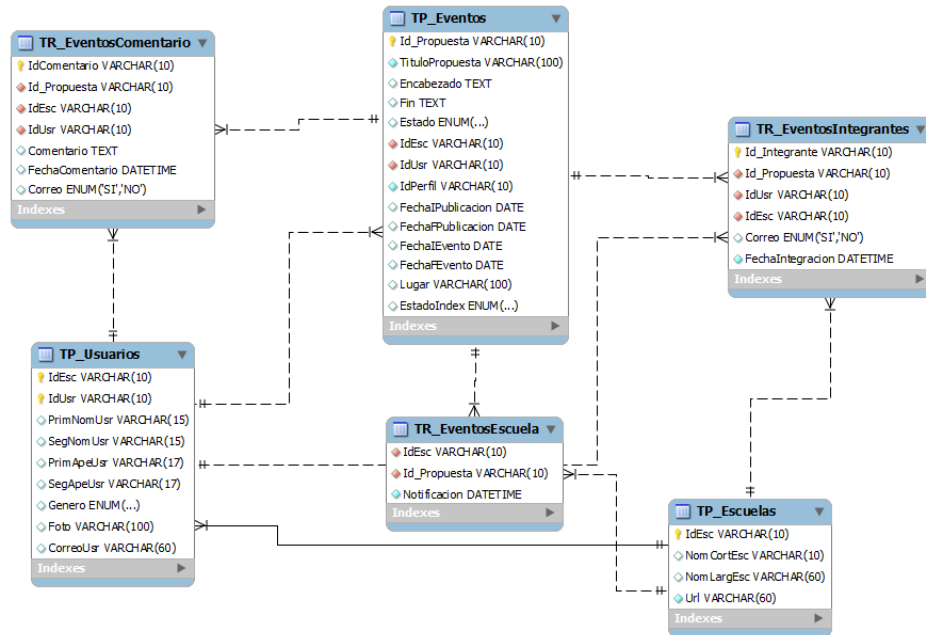


Figura 26. Diagrama E/R: Módulo foros comunidad escuela

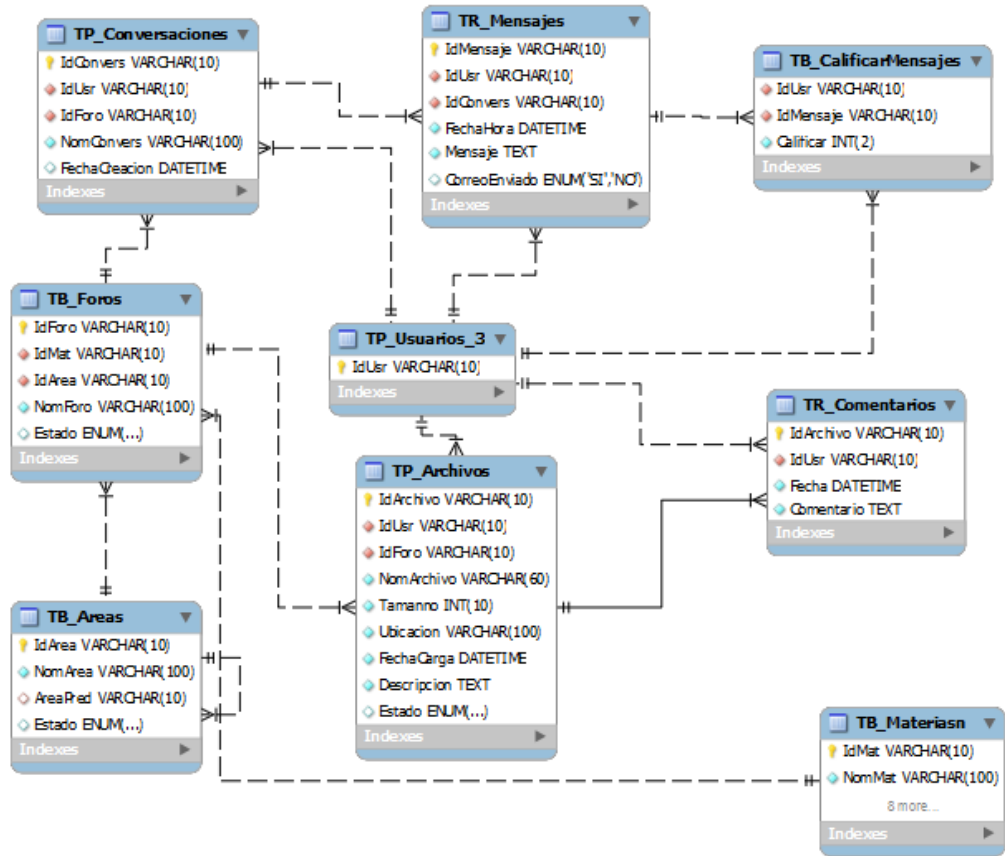
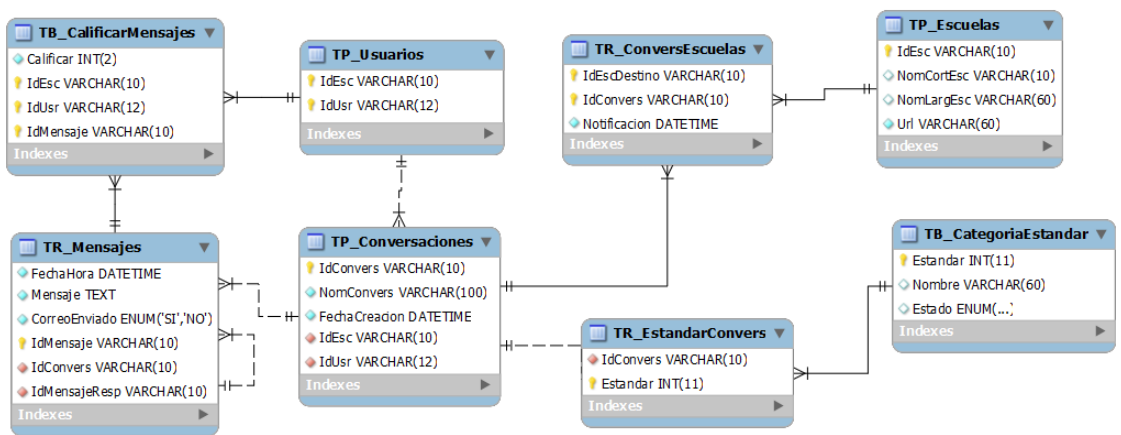


Figura 27. Diagrama E/R: Módulo foros comunidad UIS



- Correo por nivel de programa académico

Figura 28. Diagrama E/R: Correos por nivel en Base de datos Diamante

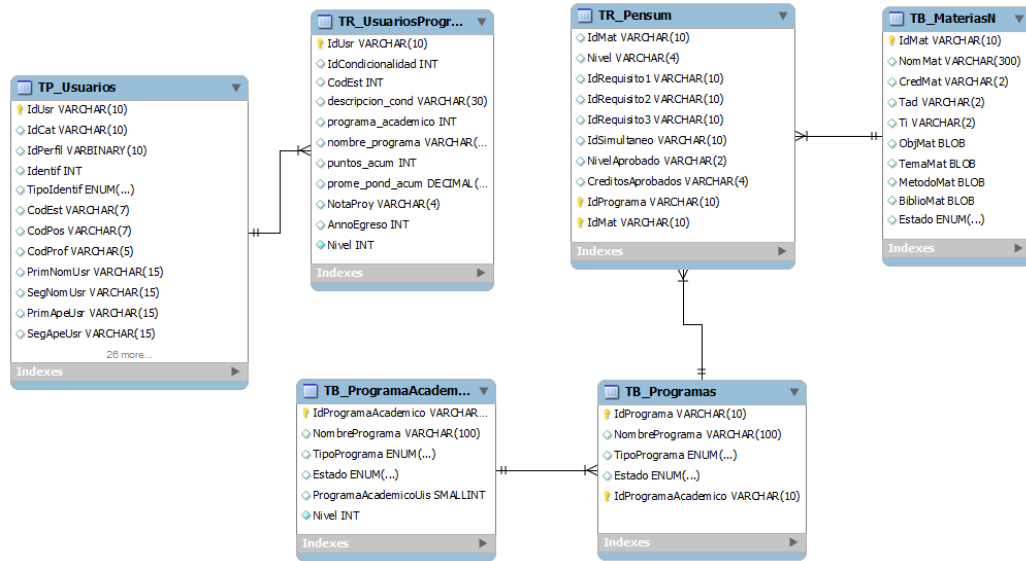
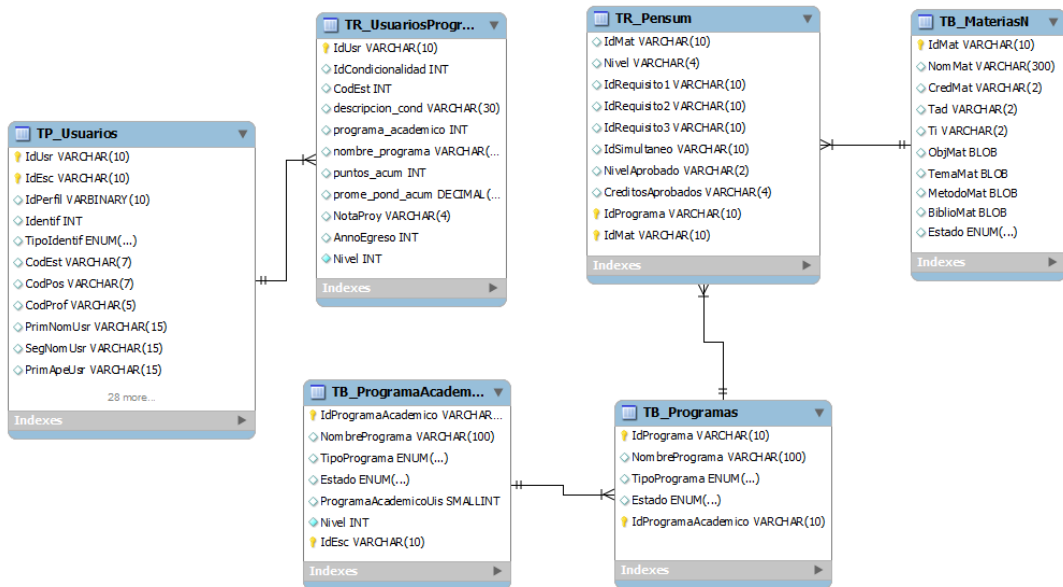
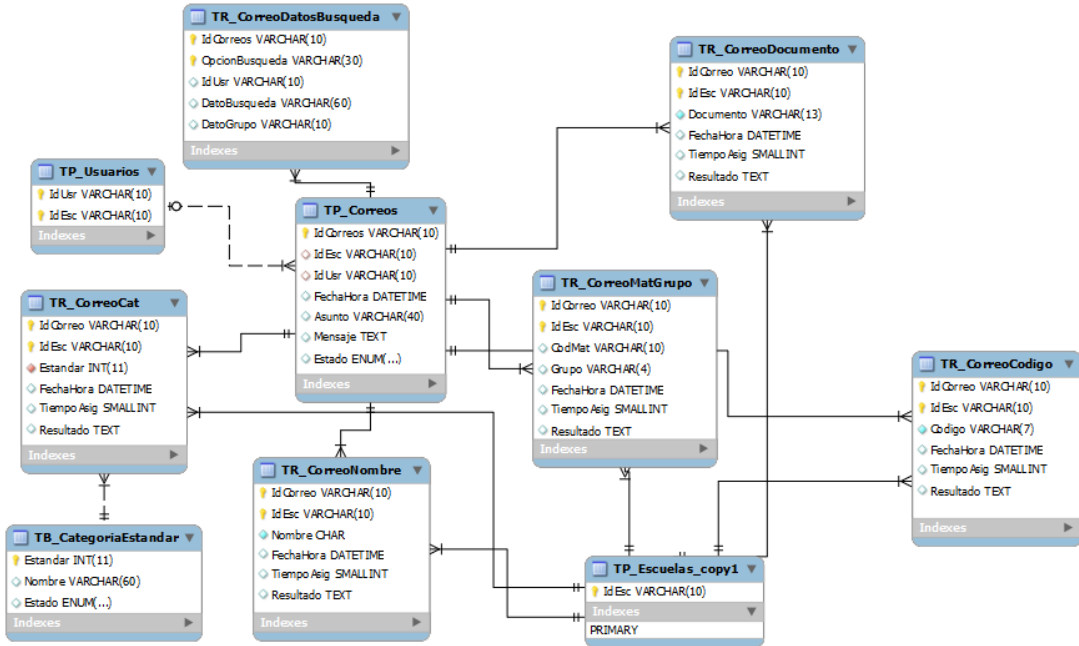


Figura 29. Diagrama E/R: Correos por nivel en Base de datos Poseidón



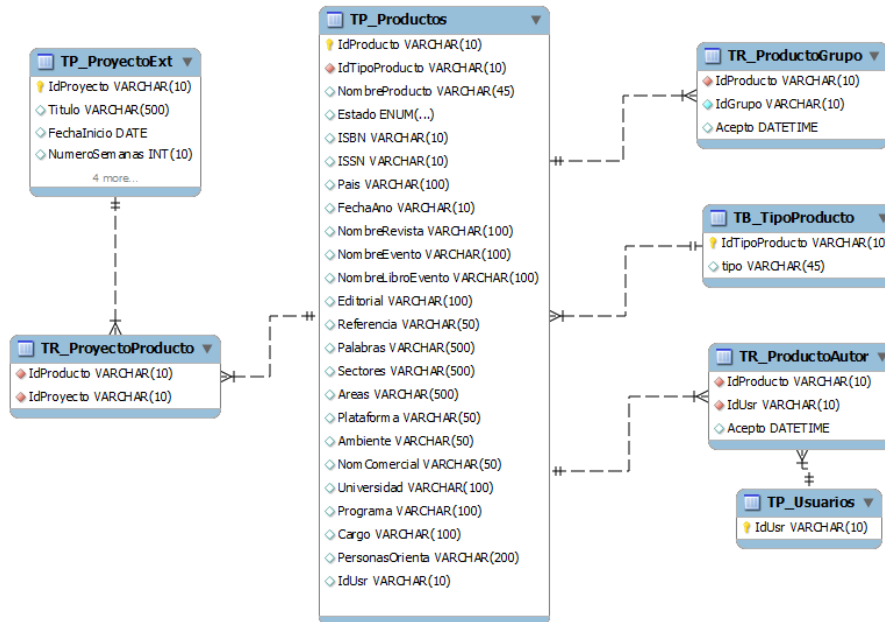
- Correo UIS

Figura 30. Diagrama E/R: Correo UIS



- **Producción intelectual**

Figura 31. Diagrama E/R: Módulo de producción intelectual



7.1.3.2 Descripción de las Entidades

Definición de las entidades utilizadas 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 EIMTWEB y E3TWeb.

Tabla 1. Descripción de las Entidades

| ENTIDAD | DESCRIPCIÓN |
|-------------|---|
| TP_Usuarios | Contiene la información de todos los usuarios registrados en el portal. |
| TP_Eventos | Contiene la información de los eventos que se crean en la respectiva escuela. |

| ENTIDAD | DESCRIPCIÓN |
|-----------------------|---|
| TP_ComentaEventos | Contiene información de la participación que realizan los usuarios a un respectivo evento a nivel escuela. |
| TP_IntegrantesEventos | Contiene los usuarios de la escuela que se han unido a un evento determinado, almacenando la relación entre el usuario y el evento. |
| TR_EventoComentario | Contiene información de los usuarios que han comentado en un evento a nivel comunidad UIS. |
| TP_Escuelas | Contiene las escuelas pertenecientes al portal. |
| TR_EventosIntegrantes | Relaciona los usuarios que se han unido a un evento a nivel comunidad UIS. |
| TR_EventosEscuela | Relaciona las escuelas a las que va dirigido el evento, y la fecha en la cual ha sido notificado ese evento para esa escuela. |
| TB_Foros | Contiene información del área a la cual es relacionada los foros comunidad escuela creados. |
| TP_Conversaciones | Contiene la información del asunto del foro. |
| TR_Mensajes | Contiene información de las participaciones de los usuarios a un respectivo foro. |
| TP_Conversaciones | Contiene información de los foros creados a nivel comunidad UIS. |

| ENTIDAD | DESCRIPCIÓN |
|------------------------|---|
| TR_ConversEscuelas | Relaciona las escuelas a las que va dirigido el respectivo foro, y la fecha en la cual han enviado correos de notificación del mismo. |
| TR_Mensajes | Contiene información de las participaciones de los usuarios al foro a nivel comunidad UIS. |
| TP_Correos | Contiene información de los correos UIS creados. |
| TR_CorreoDatosBusqueda | Contiene información de los datos de búsqueda para enviar un correo. |
| TR_CorreosCat | Contiene información de los destinatarios del correo UIS. |
| TB_Menu | Contiene la información del menú que se encuentra en el portal. |
| TB_MenuProfesor | Contiene la información de los diferentes menús que contiene el portal de profesor. |
| TP_Producto | Contiene información de los productos intelectuales ingresados. |
| TR_ProductoGrupo | Relaciona los grupos con los productos intelectuales en los cuales participaron. |
| TB_TipoProducto | Contiene las diferentes categorías a la que puede pertenecer un producto intelectual. |
| TR_ProductoAutor | Relaciona los Autores con los productos intelectuales en los cuales participaron. |
| TR_ProyectoProducto | Relaciona un proyecto de extensión con un producto intelectual. |

7.1.4 Modelo de Procesos del Sistema: Para una mejor interpretación de los modelos de los 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 F).

7.1.5 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.

7.2 MANTENIMIENTO Y ADMINISTRACIÓN

7.2.1 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.

- **Detección de sesión y recarga de página en Portal Grupo y Profesor.**

Funcionalidad:

Los portales de la Escuela, Profesor y Grupo, deben detectar que la sesión actual este vigente al activar el foco en la pestaña o ventana. De no estar vigente la sesión se debe realizar una recarga de la página.

Problemática:

En portal del grupo y profesor agregar o actualizar la función que detecta cambio de sesión y recarga página.

Desarrollo del problema:

Solucionando la incidencia presentada en los portales, se realizó reingeniería a la función que se encarga de manejar la sesión del usuario.

Descripción:

Se modificaron varios jsps que son responsables de mostrar las interfaces en los diferentes portales en los cuales navega el usuario y en las cuales es necesario utilizar la detección de sesión. Para ello se realizó reingeniería en la función que se encarga de detectar la sesión del usuario.

- **En trabajos de grado usar el campo impacto para los objetivos específicos.**

Funcionalidad:

El sistema de trabajos de grado permite a los estudiantes ingresar los datos al formulario de tema requeridos por las escuelas.

Problemática:

El campo Impacto que no se está usando en ninguna escuela debe modificarse para que ahora reciba los objetivos específicos, de tal forma que aquellas escuelas que lo consideren pertinente lo puedan usar con este fin en el tema de trabajo de grado.

Desarrollo del Problema:

Solucionando la problemática, se implementó el cambio en la organización del tema de proyecto, modificando la organización y un requerimiento que se

solicitaba, el cual era Impacto se eliminó y se colocó uno nuevo, el cual corresponde a Objetivos Específicos.

Descripción:

Se modificó varios jsp, entre los cuales están FormularTema.jsp, ModificarTodoTema.jsp, ModificarParteTema.jsp, entre otros, reemplazando el campo de Impacto que no era utilizado por Objetivos Específicos, también se cambió el orden de la presentación, colocando de manera organizada los Objetivos Específicos, después del Objetivo General; en la visualización de los temas de proyecto para la vista de la interfaz de los profesores, el comité, y los estudiantes.

- **En servicio de archivos no se permite el acceso en grupos.**

Funcionalidad:

El servicio archivos comunidad escuela permite a los grupos, entre otros, compartir todo tipo de documentos. El acceso a estos archivos (de los grupos) es restringido solo a sus miembros.

Problemática:

El acceso a los archivos de un grupo de la escuela solo es permitido a sus miembros, no obstante se presenta un error, y ellos solo acceden a los archivos de la primera página. Cuando se va a las siguientes páginas sale una emergente con el aviso de servicio no permitido.

Desarrollo del Problema:

Solucionando la incidencia presentada en el servicio de los archivos, se arregló la funcionalidad de poder cambiar a la siguiente página para poder visualizar los otros archivos que se encuentran en el grupo.

Descripción:

Se modificó el jsp archivosVerS, el cual es el responsable de mostrar la interfaz donde se selecciona el área y la subárea, en dicho jsp se agregó el identificador de servicio y la url anterior que hace referencia a la actual en el momento en que el usuario realiza la selección; luego el usuario escoge una determinada opción, se le reenvía a la url indicada con los parámetros establecidos concatenados con el identificador del servicio y la url anterior para permitir el acceso a dicha consulta. Esta agregación se realizó en las líneas de código afectadas. Luego de este procedimiento se analiza la primera página de archivos cargada de un subárea en general, la cual hace referencia academicoAccionesS.jsp, agregando dos variables, una de ellas almacena la url anterior a la página en la cual se encontraba el usuario, y la otra guarda el identificador del servicio; dichas variables se concatenaron y fueron utilizadas para ser enviadas como parámetros por medio de la url en el momento en que el usuario selecciona una subárea, las cuales otorgan los permisos para visualizar la siguiente o anterior página solicitada. Con esto se concluye la solución al problema de " UPS! Servicio no permitido".

- **Cambiar el nombre de los botones en el Aula Virtual y agregar textos explicativos alusivos.**

Funcionalidad:

El Aula Virtual permite la calificación de las actividades a través de botones que dan acceso a las diferentes funciones e interfaces a los profesores de las

asignaturas. A los estudiantes los botones permiten acceder a las notas y a los textos de realimentación del profesor.

Problemática:

Los textos sobre los botones no son claros, se hace necesario cambiarlos y agregar textos emergentes que aclaren las funcionalidades tanto a los profesores, como a los estudiantes. Se debe cambiar Ver Notas por Listado de Notas, Calificar Abiertas por Preguntas Abiertas, Asignar Notas por Publicar Notas. Agregar texto emergente de ayuda por medio de la herramienta tool tips.

Desarrollo del Problema:

Se implementó como solución, que al situarse en un botón aparezca un texto emergente explicativo que aclara la funcionalidad de dicho botón a los usuarios.

Descripción:

Se modificaron los jsp que estaban relacionados con el Aula Virtual, que era donde residía el problema, se implementó la herramienta de ayuda visual Tooltip, en los diferentes botones que fueron modificados. Así mismo en Aula virtual y otras dependencias de ella, se modificaron ciertos campos, como lo son: Ver Notas por Listado de Notas, Calificar Abiertas por Preguntas Abiertas, Asignar Notas por Publicar Notas.

- **Creación de ayudas para el servicio de trabajos de grado Inscripción de temas.**

Funcionalidad:

El sistema de trabajos de grado permite el ingreso de datos a los estudiantes a los campos del formulario definido por las escuelas.

Problemática:

Se requiere la creación de las Ayudas a los usuarios que permita ingresar el tema de proyecto de manera sencilla.

Desarrollo del Problema:

Se implementó una ayuda para Inscribir el Tema de proyecto de grado, para facilitar el uso de las tecnologías a los estudiantes, de forma clara y precisa.

Descripción:

Se realizó la ayuda para inscribir un Tema de Proyecto de Grado.

- **Reingeniería al diseño tooltip de las escuelas, grupos, profesores en el index de cada portal escuela.**

Funcionalidad:

El index del portal escuela muestra información acerca de los grupos, profesores pertenecientes a dicha escuela, e información de las escuelas pertenecientes a la Universidad, en el lateral izquierdo del index encontramos dicha información, la cual al ubicar el cursor, nos muestra un tooltip con la información relacionada.

Problemática:

Se hace necesario realizar una reingeniería al tooltip que es mostrado en el index, cada vez que el cursor se sitúa sobre algún grupo, profesor o escuela.

Desarrollo del Problema:

Se implementó un tooltip con un nuevo diseño, que muestra información acerca de la producción intelectual, sitio web del grupo y/o profesores.

Descripción:

Se modificó el jsp Laterallzquierdo, y se creó un nuevo jsp tooltipEscuelasIndex, con los cuales se implementó el nuevo diseño del tooltip. A su vez se automatizó la forma en que se listaban las diferentes escuelas, clasificándolas por la facultad a la que pertenecen.

7.2.2 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.

7.2.3 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.

8. PRUEBAS DEL SISTEMA

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

8.1 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.

8.1.1 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 G).

8.2 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.

8.3 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.

9. CONCLUSIONES

- El proyecto realizado ha contribuido de manera favorable, dando soporte a las escuelas gracias al mejoramiento de los módulos a través a la implementación de los nuevos servicios que han dado respuesta a las necesidades de los usuarios.
- La implementación del sistema de notificaciones en los foros y eventos, ha contribuido a que los usuarios se integren y participen más, creando comunidad.
- La implementación del nuevo módulo de Correos UIS permite a los usuarios llevar un seguimiento de sus correos, a su vez, modificar los destinatarios, y agilizar los envíos, con lo cual se conseguirá mayor eficiencia en la entrega de los mismos.
- El servicio armar menú profesor permite a los administradores de los diferentes portales optimizar tiempo y llevar un seguimiento visual del menú por medio de los portales web.
- El portal producción intelectual permite conocer los diferentes proyectos que son realizados en la comunidad UIS.
- El módulo de producción intelectual facilita al administrador o profesor ingresar un producto intelectual, modificar o eliminar uno ya existente. Ayudando a facilitar la gestión de la información.
- La implementación de la navegabilidad entre los portales del profesor, del grupo, de las escuelas, y del portal de producción intelectual; permite la interacción entre el usuario y los diferentes sitios web que provee el grupo Calumet. Con esto se logrará que el usuario pueda navegar de manera fácil y agradable a su vista.

10. 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 recomienda capacitar a los usuarios para que utilicen de una manera óptima los nuevos servicios, y con ellos ahorren tiempo y agilicen sus tareas relacionadas a los mismos.
- Se recomienda dar continuidad y promocionar el portal de producción intelectual, el cual es muy útil para mostrar a las personas interesadas en las diferentes producciones generadas en la universidad, y poder visualizarlas dependiendo de cada escuela, profesor o grupo de investigación. Debido a que esto generará un reconocimiento al compromiso y entrega de los implicados en el desarrollo de dichas producciones.

BIBLIOGRAFÍA

GISSET. Información acerca de la Arquitectura Cliente Servidor. [en línea]. <http://g701giadar.wikispaces.com/page/diff/Arquitctura+Cliente+Servidor/229968658> [citado el 01 Junio de 2014]

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.

ORACLE. Api de Java [en línea]. <http://docs.oracle.com/javase/7/docs/api/> [citado el 01 Junio de 2014]

ORACLE CORPORATION. MySQL Documentation: MySQL Reference Manuals.[en línea]. <http://dev.mysql.com/doc/> [citado el 06 Febrero de 2014]

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.

PONCE DE LEÓN, Diego. Tutoriales HTML. [en línea]. <http://www.htmlquick.com/es/tutorials.html> [citado el 01 Mayo de 2014]

RICH, Clark, et al. Html element index [online]. Available from Internet: <http://html5doctor.com/element-index/> [cited el 03 Abril de 2014]

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.

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.

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.

The jQuery Foundation. Api de la librería jQuery. [en línea]. <http://api.jquery.com/> [citado el 18 Mayo de 2014]

UNIVERSIDAD INDUSTRIAL DE SANTANDER. Grupo Desarrollador Calumet. Algunas herramientas de recursividad y utilidad para cuando se codifica JavaScript. [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 07 Mar. 2014]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#js>

----- . ¿Cómo extender y profundizar en herramientas para crear aplicaciones web front-end avanzadas? [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 12 Feb. 2014]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#liberias>

----- . Componentes de interfaz de usuario dinámica e interactiva para crear aplicaciones. [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 25 Feb. 2014]. Disponible en Internet:

<http://cormoran.uis.edu.co/eisi/Estandar/components.jsp>

-----. Estilos y diseños básicos y complementarios para crear componentes de la Librería Elise. [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 20 Feb. 2014]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/design.jsp>

-----. Guía de documentación de selectores, eventos, manipulación y procedimientos de jQuery. [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 10 Nov. 2014]. Disponible en Internet: <http://www.oscarotero.com/jquery/>

-----. Herramientas adicionales en el proceso de testear y codificar servicios y aplicaciones. [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 10 Nov. 2014]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#otros>

-----. Herramientas JavaScript y JQuery para la creación de procedimientos para aplicaciones [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 16 Mar. 2014]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/codes.jsp>

-----. Herramientas para facilitar y agilizar la creación de diseños para objetos HTML con CSS. [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 12 Feb. 2014]. Disponible en Internet: <http://cormoran.uis.edu.co/eisi/Estandar/tools.jsp#librerias>

-----. Plantillas estándar para crear servicios, aplicaciones o cualquier proyecto para Calumet. [en línea] [Bucaramanga, Colombia]: Elise Library V2. Noviembre 2014. [Citado 05 Feb. 2014]. Disponible en Internet:

<http://cormoran.uis.edu.co/eisi/Estandar/templates.jsp>

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.

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.

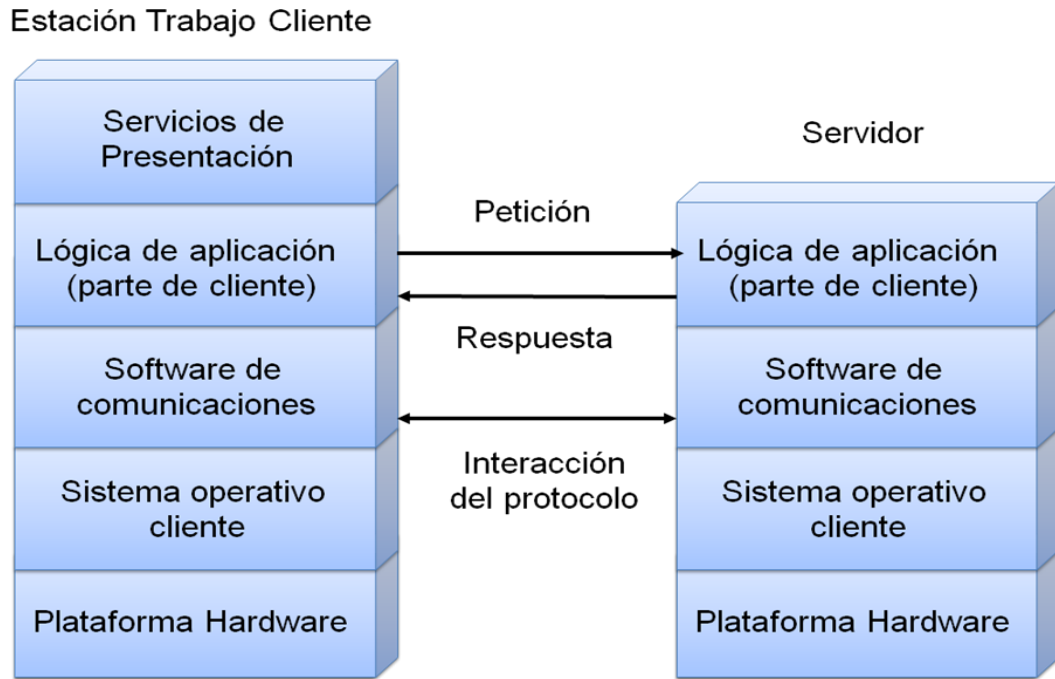
A continuación se muestra la clasificación de los sistemas Cliente / Servidor:

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 32. Arquitectura Cliente/Servidor de Dos Capas



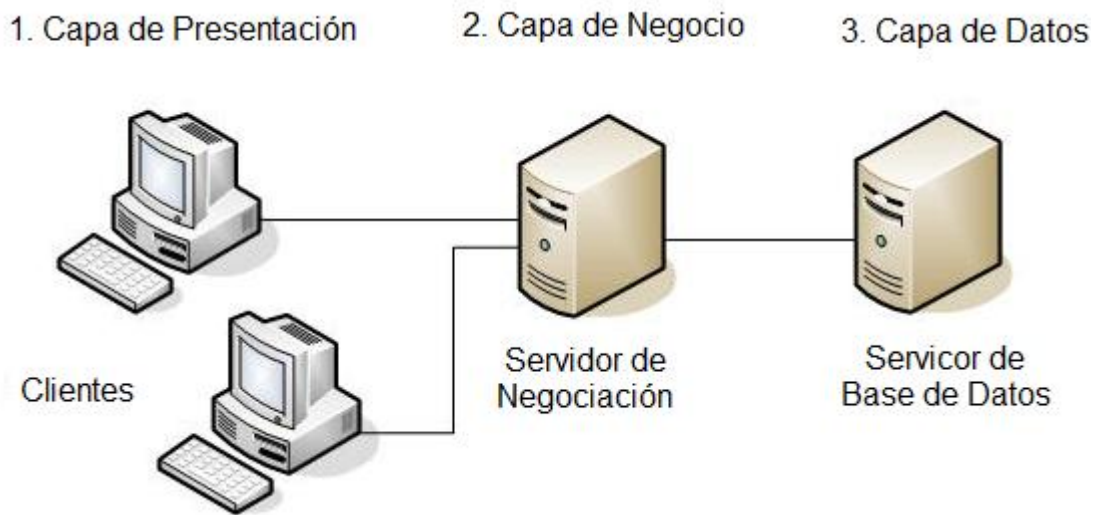
1.2. ARQUITECTURA CLIENTE/SERVIDOR DE TRES CAPAS

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 33. Arquitectura Cliente/Servidor de Tres Capas



Fuente: Arquitectura Cliente/Servidor de tres capas. [en línea]. http://www.ecured.cu/index.php/Archivo:Tres_capas.JPG [citado el 10 Mayo de 2014]

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

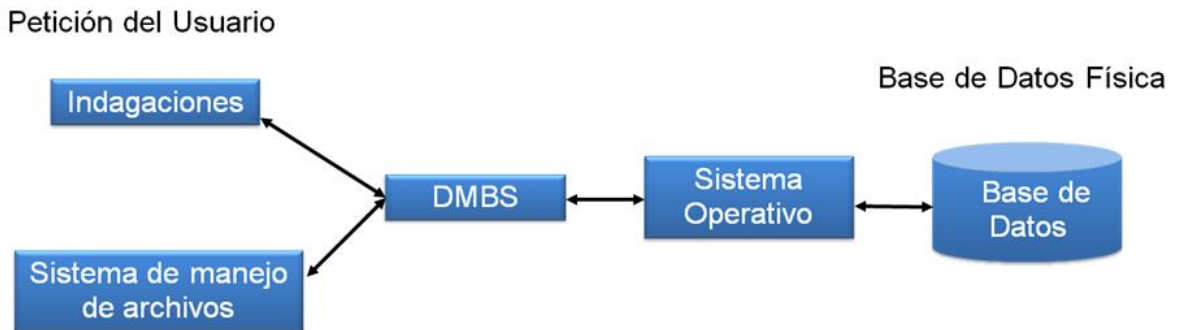
El sistema manejador de bases de datos es la porción más importante del software de un sistema de base de datos. Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica.

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 34. Función del DBMS.

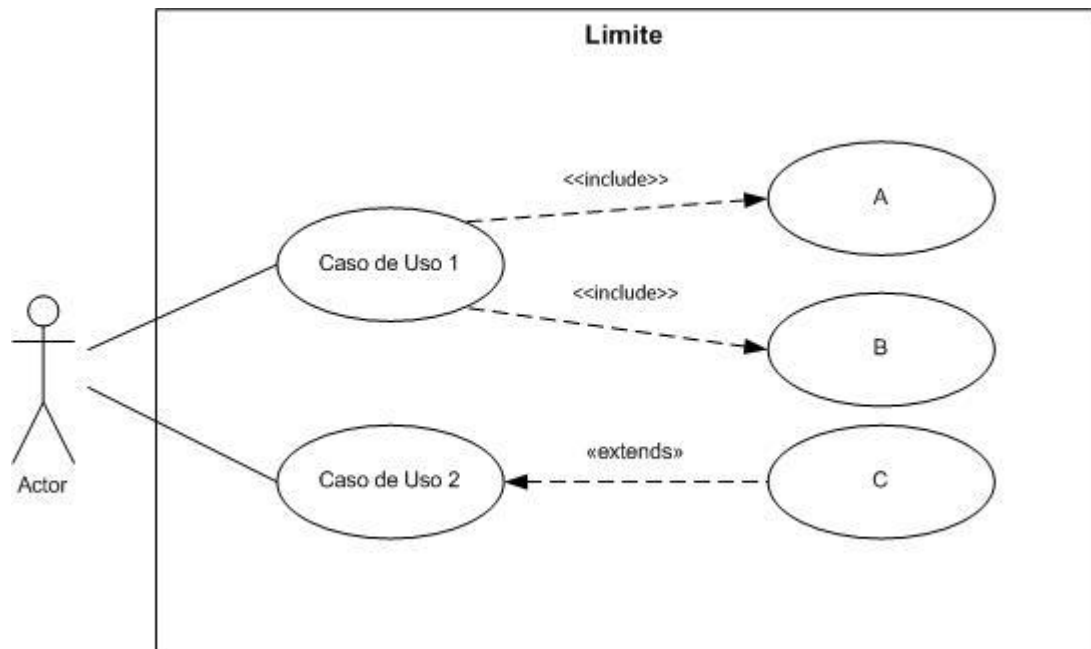


La figura muestra el DBMS como interface entre la base de datos física y las peticiones del usuario. El DBMS interpreta las peticiones de entrada/salida del usuario y las manda al sistema operativo para la transferencia de datos entre la unidad de memoria secundaria y la memoria principal. Es decir el 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 35. Diagramas de Casos de Uso



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 las siguientes 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 EIMTWEB, FIFMEWEB: Es el tipo de usuario común de los portales EIMTWEB y FIFMEWEB y a quien van dirigidos los servicios. Este usuario solo tiene control sobre sus privilegios.
- Profesor EIMTWEB, FIFMEWEB: Son usuarios que cuentan con ciertos privilegios, por lo cual tienen acceso a determinados servicios, y a realizar ciertas acciones.
- Sistema: Es el sistema encargado de realizar determinadas funciones, que le solicita el usuario.
- Público: Personas que ingresan a los portales, sin necesidad de registrarse solo tienen permisos públicos.
- Encargado del grupo: Usuario que pertenece a un grupo de investigación, desarrollo u otro determinado grupo, pero es el encargado de administrar el portal del grupo.
- Integrante del grupo: Usuario que pertenece a un grupo de investigación, desarrollo u otro determinado grupo.

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

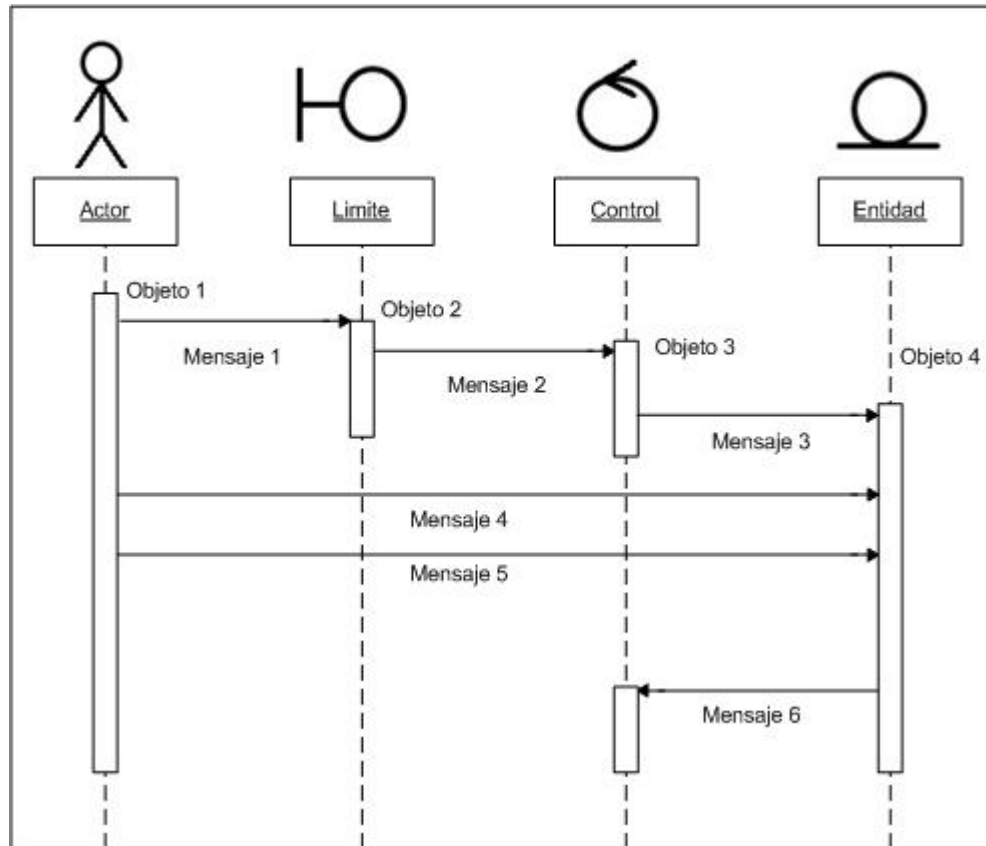
Objetos: Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de cada objeto será una línea discontinua conocida como la *línea de vida* de un objeto. Con la línea de vida se encuentra un pequeño rectángulo conocido 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 Inglés) La interfaz gráfica de usuario; es la interfaz de interacción del usuario y en la que más interactividades se presentan con otros objetos.

Figura 36. Diagrama de Secuencias



Anexo E. DOCUMENTACIÓN DE CASOS DE USO DEL SISTEMA

1. Sistema de notificaciones.

Tabla 2. Casos de uso: Sistema de notificaciones.

| TÍTULO | FUNCIONES PRIMARIAS |
|---------------------------------------|---|
| Enviar correo de notificación Eventos | Crea un correo de notificación diario a partir de participaciones o vinculaciones de los usuarios en los eventos a nivel comunidad UIS o escuela. El cual es enviado a los usuarios que han participado y al creador. |
| Enviar correo de notificación Foro | Crea un correo de notificación diario a partir de participaciones de los usuarios en los Foros a nivel comunidad UIS o escuela. El cual es enviado a los usuarios que han participado y al creador. |

2. Correos por nivel del semestre y programa académico

Tabla 3. Caso de Uso: Enviar correos por nivel del semestre.

| TÍTULO | FUNCIONES PRIMARIAS |
|-------------------------|---|
| Enviar correo por nivel | Permite enviar correos a los estudiantes de determinado semestre. |

Tabla 4. Casos de uso: Correos por nivel de semestre.

| TÍTULO | FUNCIONES PRIMARIAS |
|---------------|----------------------------|
|---------------|----------------------------|

| | |
|------------------------|---|
| Selección de programas | Despliega los programas académicos relacionados con las escuelas que ha seleccionado el usuario. |
| Selección de nivel | Dependiendo del tiempo de duración de los programas académicos seleccionados por el usuario, el sistema habilita los niveles de semestre correspondiente a los programas. |

3. Correos UIS

Tabla 5. Casos de uso: Correos UIS.

| TÍTULO | FUNCIONES PRIMARIAS |
|--|---|
| Ver correos | Permite a quien creó el correo visualizar todos los correos que ha creado, consultarlos, eliminarlos, o crear uno nuevo. |
| Crear correo UIS | Permite crear un correo a diferentes destinatarios, dependiendo del criterio que tenga el usuario que desea crearlo. |
| Consultar correo | Permite visualizar el asunto y el mensaje de un correo específico. |
| Eliminar correo | Permite eliminar un correo si y solo si el estado del correo lo permite. |
| Visualizar y eliminar destinatarios del correo | Permite visualizar los destinatarios del correo seleccionado, a su vez ver el estado del correo dependiendo del destinatario. A su vez eliminar un destinatario, si y solo si el estado del envío lo permite. |

4. Menú del profesor

Tabla 6. Casos de uso: Menú del profesor.

| TÍTULO | FUNCIONES PRIMARIAS |
|------------------------|--|
| Visualizar menú actual | Permite al administrador visualizar el menú actual del profesor y actualizar las rutas del menú actual, para con ello evitar inconsistencias en los direccionamientos. |
| Modificar menú actual | Permite al administrador editar el menú, permitiéndole cambiar de posición, eliminar o agregar un nuevo submenú o menú. |
| Crear menú | Permite al administrador crear un menú del profesor, en el cual puede definir los niveles y subniveles. |

5. Producción intelectual.

Tabla 7. Caso de Uso: Producción intelectual.

| TÍTULO | FUNCIONES PRIMARIAS |
|------------------------------------|--|
| Visualizar productos intelectuales | Permite al usuario visualizar los diferentes productos intelectuales que ha ingresado dicho usuario. A si mismo da la posibilidad de eliminar un producto intelectual ya existente. También permite visualizar al usuario de categoría profesor, los productos intelectuales que ha ingresado, y en los cuales lo han incluido, y le da la opción de aceptar o rechazar dicha inclusión. |

| TÍTULO | FUNCIONES PRIMARIAS |
|--|--|
| Ingresar producto intelectual | Permite ingresar un nuevo producto intelectual. |
| Modificar un producto intelectual | Permite editar un producto intelectual ya existente. |
| Consultar producción intelectual públicamente | Permite al público consultar la producción intelectual que ha realizado la escuela, y consultar la producción realizada o en la que ha participado los diferentes profesores y grupos de la escuela. |
| Consultar producción intelectual de un grupo específico. | Permite visualizar al administrador y encargado del grupo, los productos en los cuales han incluido al grupo, y darle la posibilidad de aceptar o rechazar dicha inclusión. |
| Rechazar o aceptar inclusión en un producto intelectual | Permite a los usuarios de categoría profesor o Administrador y encargado del grupo, visualizar los productos intelectuales en los cuales han sido incluidos, y poder aceptar o rechazar dicha propuesta. |

6. Mantenimiento y Administración de los portales EIMTWEB y FIFMEWEB.

Tabla 8. Caso de Uso: Mantenimiento y Administración de los portales EIMTWEB y FIFMEWEB.

| TÍTULO | FUNCIONES PRIMARIAS |
|-----------------------|---|
| Administrar Servicios | Crear, modificar y eliminar servicios del portal. Establecer parámetros. |

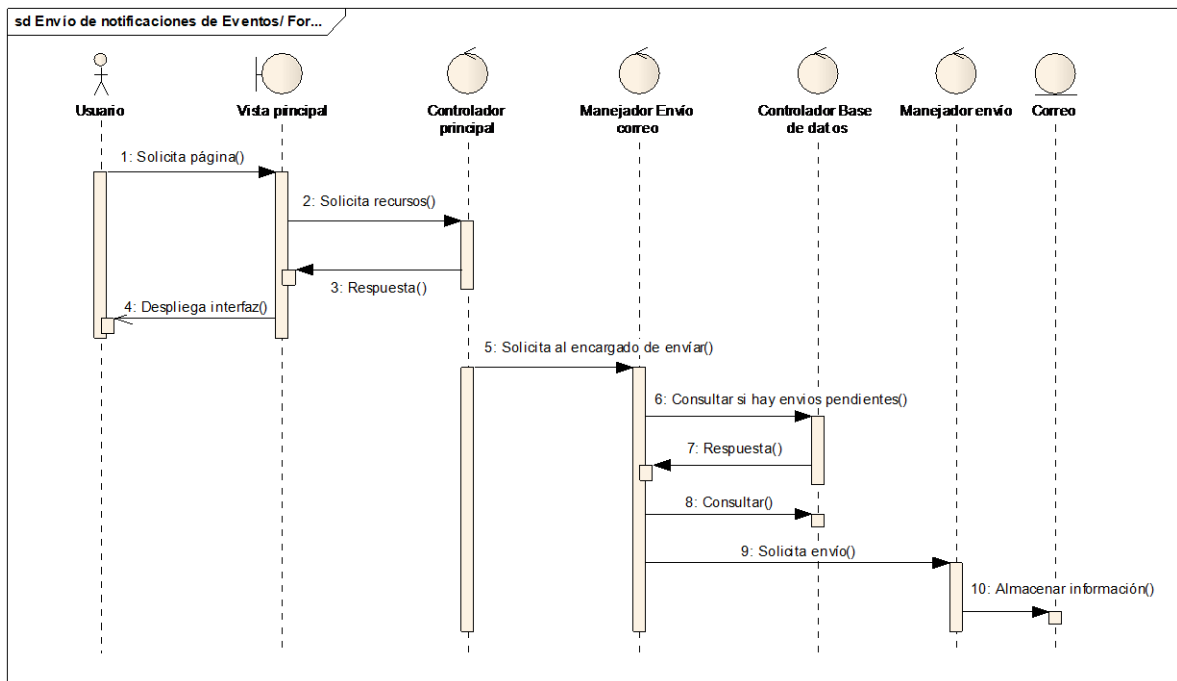
| TÍTULO | FUNCIONES PRIMARIAS |
|-----------------------------|--|
| Administrar Sugerencias | Revisar todas las sugerencias para atender o eliminar. |
| Administrar Varios | Ejecutar labores de mantenimiento que no clasifican en otras categorías. Llevar el contador de visitas a cero. Insertar, cambiar 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. |

| TÍTULO | FUNCIONES PRIMARIAS |
|--------------------------|--|
| Actualizar Base de Datos | Actualizar tablas de datos de División y Diamante. Ver contadores de registros. |

Anexo F. MODELO DE PROCESOS DEL SISTEMA

1. Sistema de notificaciones para Eventos y Foros a nivel comunidad UIS y escuela.

Figura 37. Diagrama de Secuencia: Envío de notificaciones de Eventos/foros.

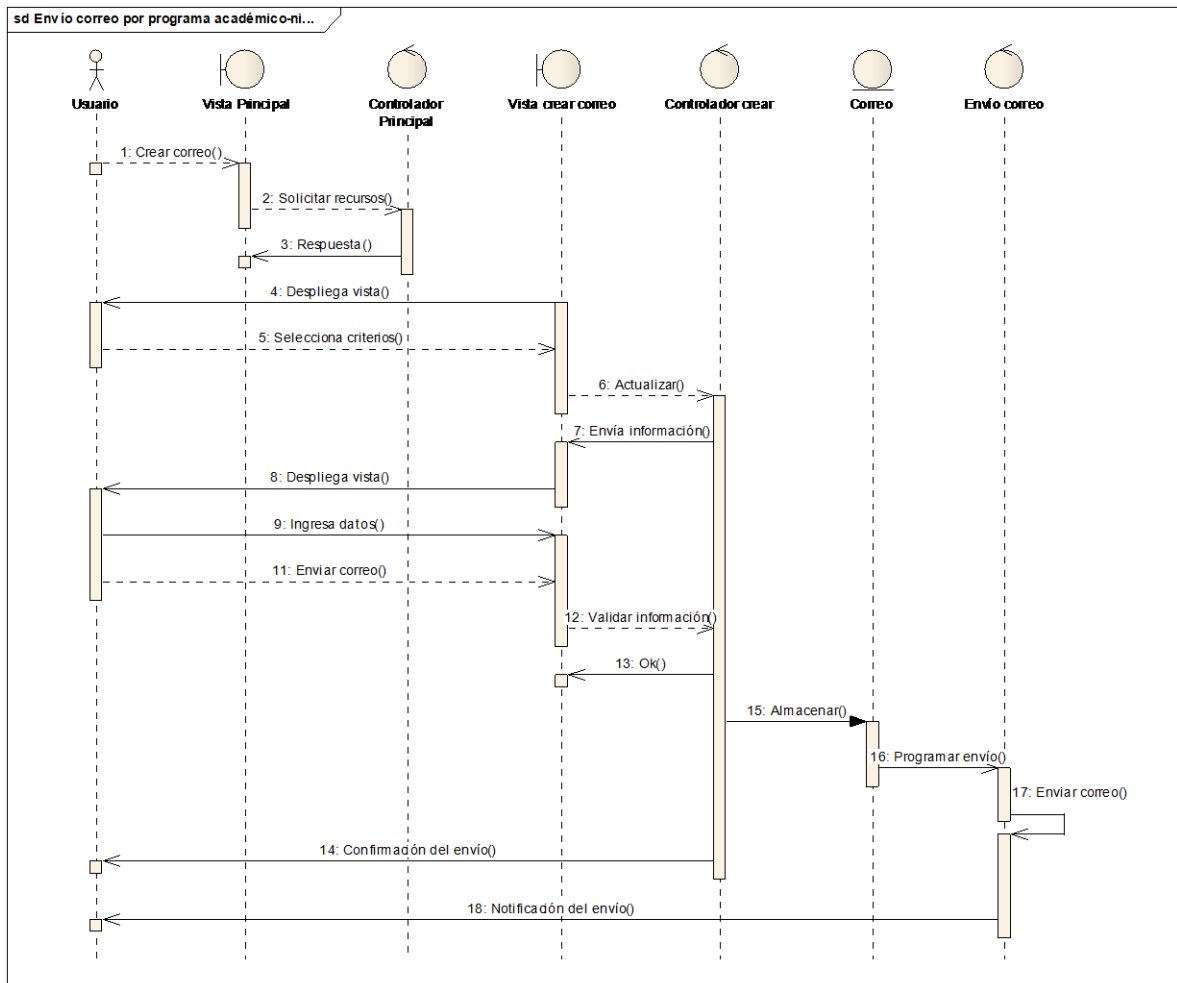


1. El usuario solicita la página de la respectiva escuela que necesita.
2. La vista principal solicita al controlador principal los recursos para realizar la visualización de la interfaz al usuario
3. El controlador principal envía los recursos solicitados a la vista principal
4. La vista principal muestra la interfaz al usuario
5. El controlador principal realiza un llamado al manejador de envío de correos
6. El manejador de envío de correos realiza una consulta al controlador de base de datos, la cual consiste en preguntar si necesita realizar envíos.
7. El controlador de base de datos responde al manejador de envío de correos
8. Si hay correos por enviar, el manejador de envío de correos realiza diferentes consultas al controlador de base de datos.
9. El controlador de base de datos responde con la información solicitada
10. El manejador de envío de correos recibe la información, la organiza y solicita el envío de correos al manejador de envío

11.El manejador de envío de correos solicita almacenar y actualizar la información relacionada con dicho correo

2. Correos por nivel del semestre y programa académico

Figura 38. Diagrama de Secuencia: Envío correo por programa académico-nivel.

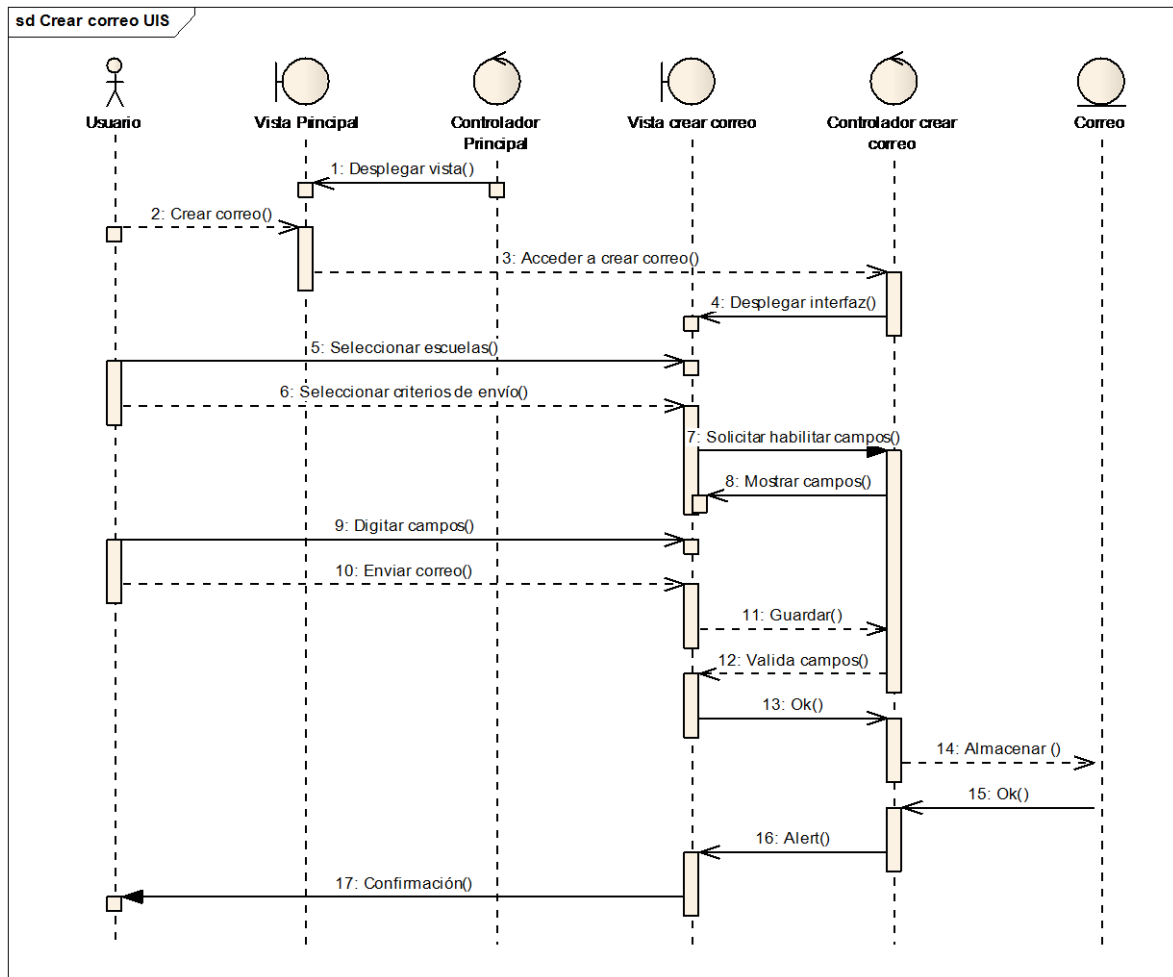


1. El usuario solicita a la vista principal crear un correo.
2. La vista principal solicita los recursos al controlador principal.
3. El controlador principal responde con los recursos solicitados.
4. Se despliega la interfaz de la vista crear correo al usuario
5. El usuario selecciona el criterio de búsqueda
6. La vista crear correo, actualiza la información, enviándola al controlador de crear

7. El controlador de crear envía la información correspondiente a la selección del usuario
8. La vista crear correo, despliega la interfaz con la información recibida
9. El usuario ingresa los datos, y los envía a la vista crear correo
10. El usuario hace click en enviar correo
11. La vista crear correo, valida la información con colaboración del controlador crear.
12. El controlador crear, envía un mensaje a la vista crear correo, con un ok al verificar la información.
13. La vista crear correo, envía un mensaje con la confirmación de que el envío se estará realizando y que puede consultar el estado del envío en el servicio dado.
14. El controlador crear, almacena la información en base de datos y programa el envío del correo
15. El manejador de envío de correo, obtiene la información almacenada en base de datos y envía el correo
16. Al ser enviado el correo, el manejador envío de correo notifica el envío del correo al usuario que creo el correo, y le envía un informe del mismo.
17. El controlador envía correo se llama así mismo para enviar el correo.
18. Cuando el controlador envía el correo, notifica al usuario por medio de un correo.

3. Correos UIS

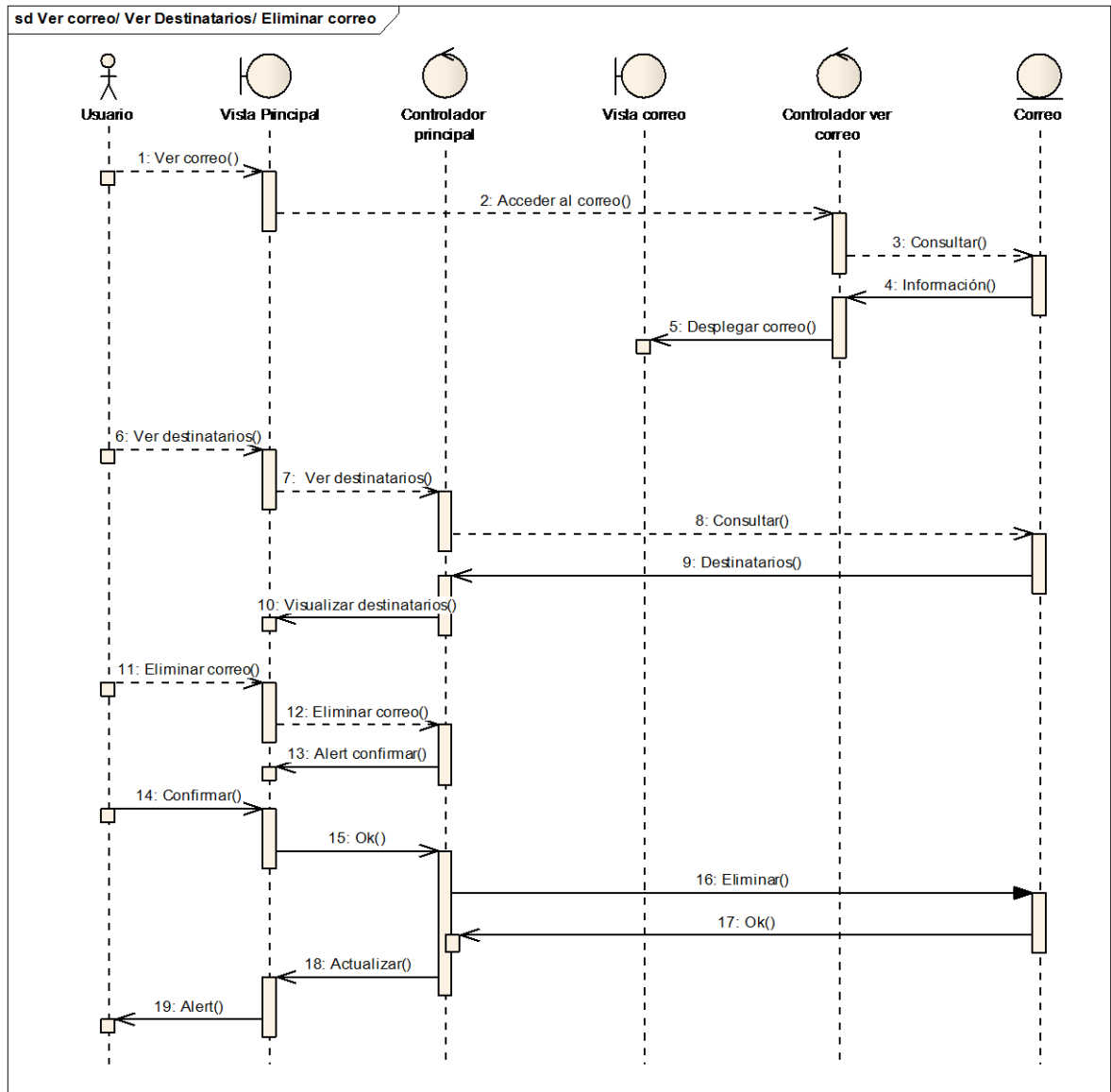
Figura 39. Diagrama de Secuencia: Crear correo.



1. El controlador principal despliega la vista de la interfaz al usuario.
2. El usuario selecciona Crear correo a la vista VistaPrincipal.
3. La vista principal solicita al controlador crear correo, los recursos para acceder a la interfaz crear correo.
4. El controlador crear correo, responde con los recursos solicitados para la visualización de la interfaz, y se hace visible la nueva interfaz.
5. El usuario selecciona las escuelas destino en la vista crear correo.
6. El usuario selecciona el criterio de búsqueda: Programa-Nivel en la vista crear correo.
7. La vista crear correo envía la información al controlador crear correo, para que de acuerdo a la selección habilite los campos a digitar.
8. El controlador crear correo, muestra los campos necesarios para la opción escogida.

9. El usuario digita los campos solicitados en la vista crear correo.
- 10.El usuario hace click en enviar el correo en la vista crear correo.
- 11.La vista crear correo, envía la información ingresada al controlador crear correo.
- 12.El controlador crear correo, valida los datos recibidos, y envía un mensaje a la vista crear correo sobre la validación.
- 13.La vista crear correo envía un recibido al controlador crear correo.
- 14.El controlador crear correo almacena la información del correo en base de datos.
- 15.El controlador crear correo, recibe un mensaje de confirmación de la inserción de los datos.
- 16.El controlador crear correo, envía un mensaje a la vista crear correo, informando que el correo se encuentra en proceso de envío, y que si lo desea puede visualizar el estado del mismo.
- 17.La vista crear correo envía un mensaje de confirmación al usuario, informándole que el correo ha sido guardado exitosamente.

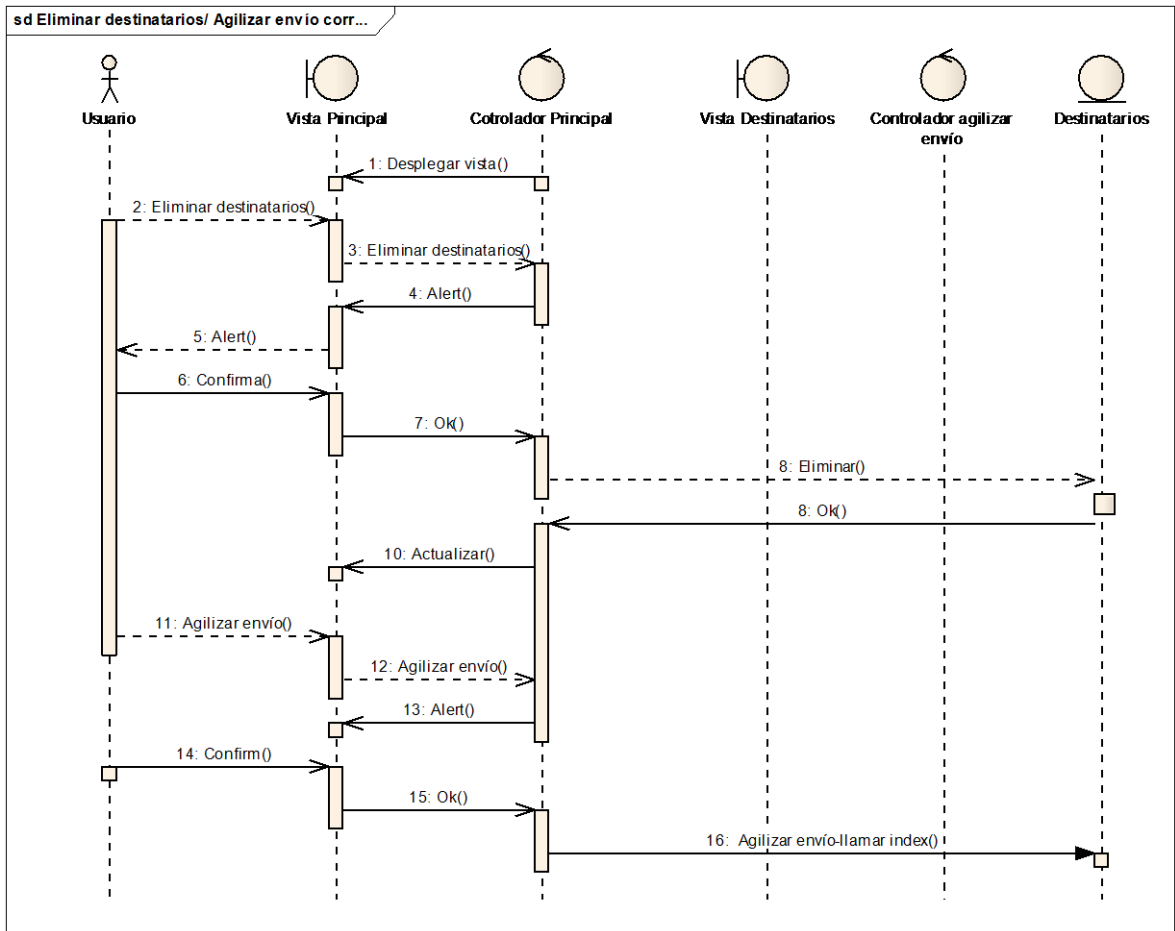
Figura 40. Diagrama de Secuencia: Ver correo/Ver destinatarios/ Eliminar todo el correo.



1. El usuario solicita ver correo en la vista VistaPrincipal.
2. La vista VistaPrincipal, envía la solicitud de ver correo al controlador ControladorCorreo.
3. El controlador ControladorCorreo consulta el correo en la base de datos Correo.
4. La base de datos Correo le envía los resultados al controlador ControladorCorreo.

5. El controlador ControladorCorreo despliega la interfaz de resultados en la vista VistaCorreo.
6. El usuario solicita ver destinatarios del correo en la vista VistaPrincipal.
7. La vista VistaPrincipal solicita ver destinatarios al controlador ControladorPrincipal.
8. El controlador ControladorPrincipal consulta en la base de datos Correo los destinatarios.
9. La base de datos Correo envía los destinatarios al controlador ControladorPrincipal.
10. El controlador ControladorPrincipal despliega la vista de los destinatarios en la vista VistaPrincipal.
11. El usuario solicita eliminar correo en la vista VistaPrincipal.
12. La vista VistaPrincipal envía la petición de eliminar correo al controlador ControladorPrincipal.
13. El controlador ControladorPrincipal recibe la petición y despliega un alert de confirmación en vista VistaPrincipal.
14. El usuario confirma la eliminación del correo.
15. La vista VistaPrincipal le envía la confirmación de eliminar correo al controlador ControladorPrincipal.
16. El controlador ControladorPrincipal elimina el correo de la base de datos Correo.
17. La base de datos Correo envía la confirmación del correo eliminado al controlador ControladorPrincipal.
18. El controlador ControladorPrincipal actualiza la vista VistaPrincipal.
19. La vista principal informa con un alert al usuario la eliminación del correo.

Figura 41. Diagrama de Secuencia: Eliminar destinatarios/ Agilizar envío del correo.

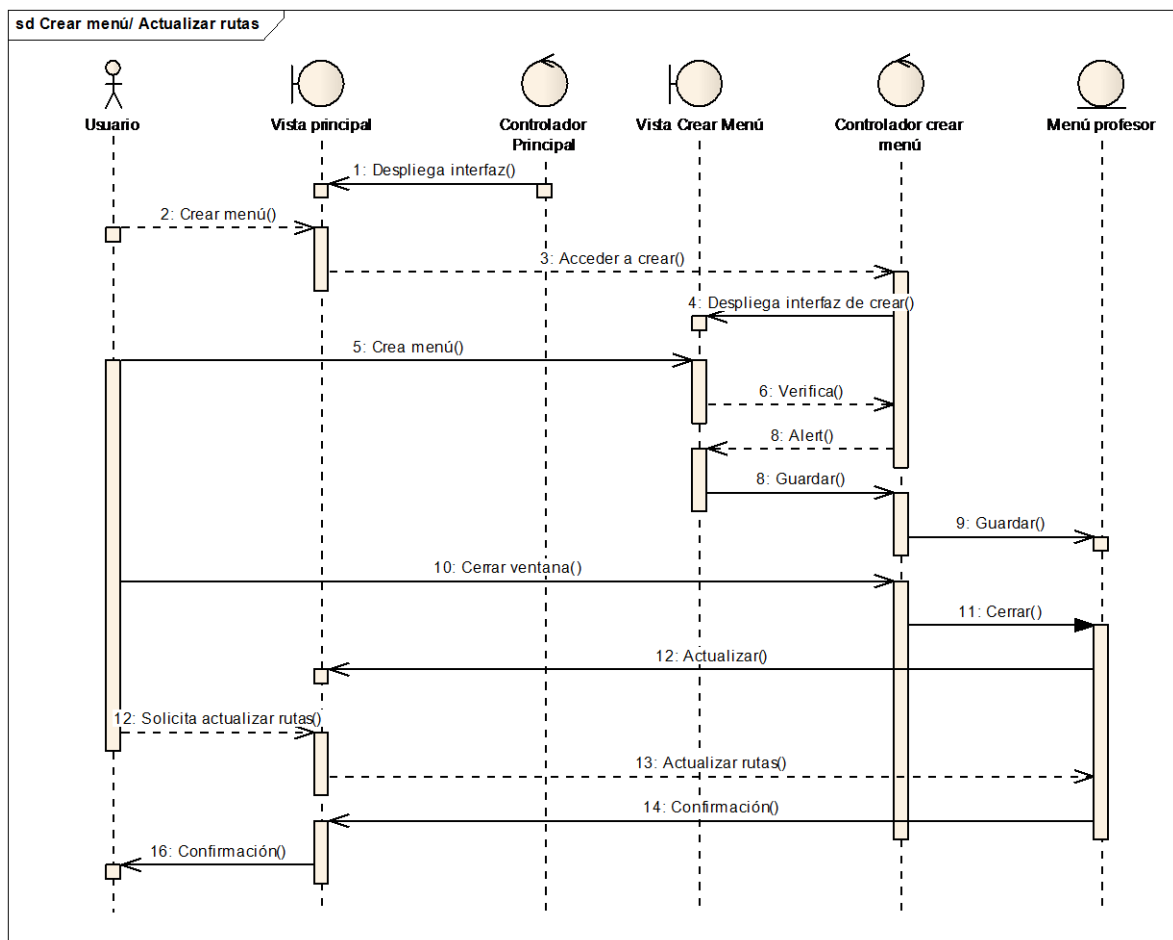


1. El controlador principal despliega la vista en la VistaPrincipal
2. El usuario selecciona eliminar destinatario en la vista principal
3. La vista principal, envía la solicitud de eliminar destinatario al controlador principal.
4. El controlador principal envía un mensaje de confirmación a la vista principal
5. La vista principal envía un mensaje al usuario para confirmar la eliminación.
6. El usuario confirma la eliminación del destinatario en la vista principal
7. La vista principal envía una confirmación al controlador principal
8. El controlador principal elimina el destinatario en la base de datos Destinatarios.
9. La base de datos Destinatarios envía la confirmación de la eliminación al controlador principal.

- 10.El controlador principal despliega la interfaz de resultados en la vista principal.
- 11.El usuario solicita agilizar el envío del correo a un destinatario en particular en la vista VistaPrincipal.
12. La vista principal solicita agilizar el envío al controlador principal
- 13.El controlador principal envía un mensaje de confirmación a la vista principal
- 14.El usuario confirma la agilización del envío
- 15.La vista principal envía la confirmación del envío al controlador agilizar envío.
- 16.La función del controlador agilizar envío es de llamar al encargado de enviar los correos.

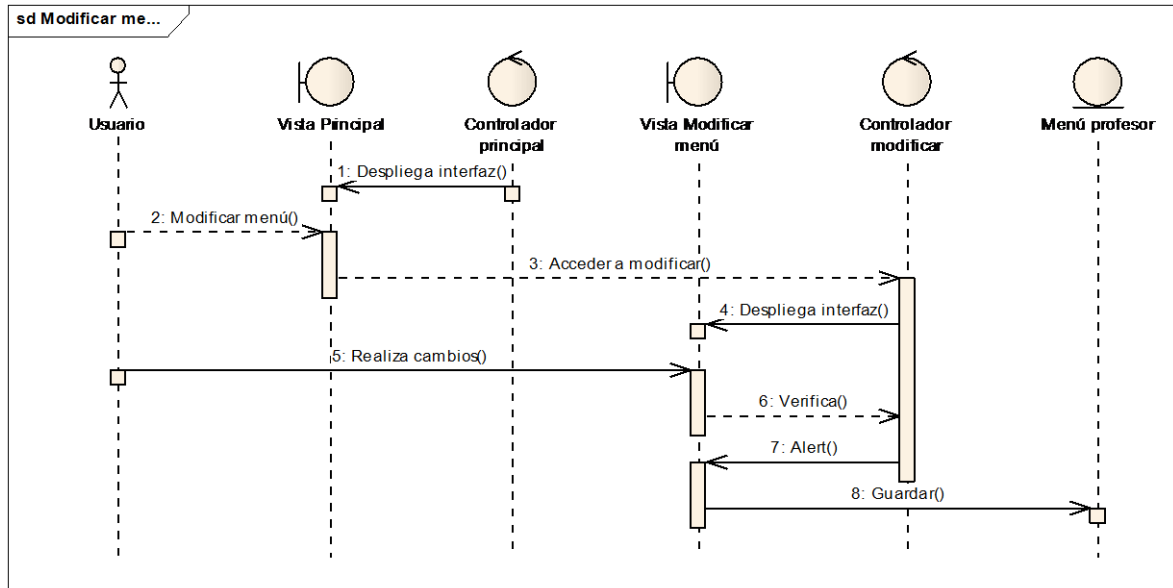
4. Menú del profesor

Figura 42. Diagrama de Secuencia: Crear Menú/ Actualizar rutas.



1. El controlador principal despliega la vista en la VistaPrincipal.
2. El usuario selecciona crear menú en la vista VistaPrincipal.
3. La vista principal solicita al controlador CrearMenú, los recursos para acceder a la interfaz CrearMenú.
4. El controlador CrearMenú, responde con los recursos solicitados para la visualización de la interfaz, y se hace visible la nueva interfaz.
5. El usuario crea el menú en la vista CrearMenú.
6. La vista CrearMenú solicita verificar al controlador Crear.
7. El controlador Crear le envía un alert, sí y solo sí, alguna elección en la creación es errónea o no permitida.
8. La vista crear menú envía una confirmación al controlador crear, para que solicite guardar la información.
9. La vista CrearMenú guarda en línea la información en la base de datos MenúProfesor.
10. El usuario cierra la ventana en la vista CrearMenú.
11. La vista CrearMenú informa al controlador Crear, que el usuario cerró la vista.
12. El controlador Crear actualiza en la vista VistaPrincipal el menú creado.
13. El usuario solicita actualizar rutas en la vista VistaPrincipal.
14. La vista VistaPrincipal actualiza las rutas en la base de datos MenúProfesor.
15. La base de datos MenúProfesor confirma la actualización.
16. La vista VistaPrincipal envía un alert de confirmación al usuario para informarle que la actualización fue exitosa.

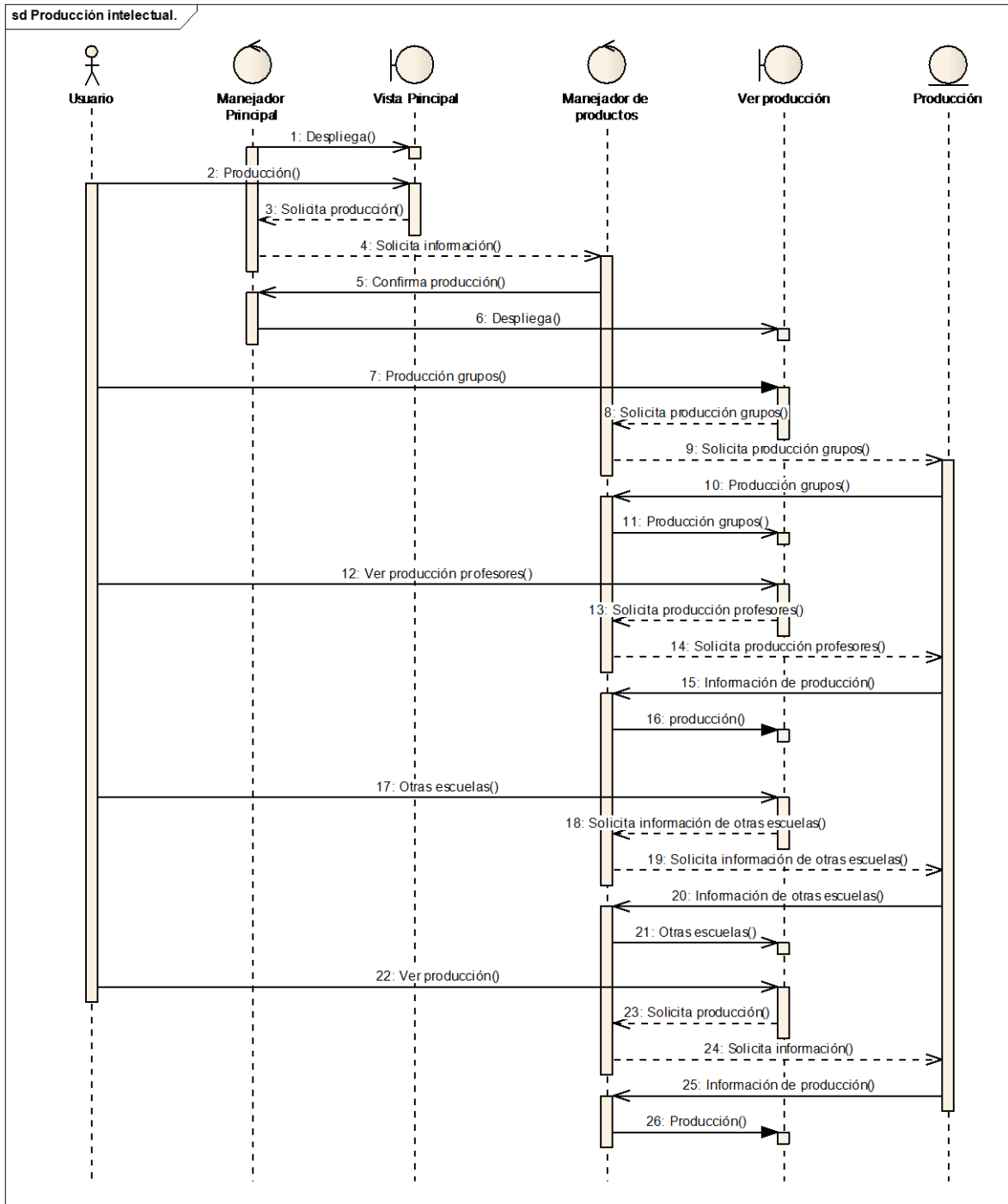
Figura 43. Diagrama de Secuencia: Modificar Menú.



1. El controlador principal despliega la vista en la VistaPrincipal.
2. El usuario selecciona modificar menú en la vista VistaPrincipal.
3. La vista principal solicita al controlador ModificarMenú, los recursos para acceder a la interfaz ModificarMenú.
4. El controlador ModificarMenú, responde con los recursos solicitados para la visualización de la interfaz, y se hace visible la nueva interfaz.
5. El usuario modifica el menú en la vista ModificarMenú.
6. La vista ModificarMenú solicita verificar al controlador ModificarMenú.
7. El controlador ModificarMenú le envía un alert, sí y solo sí, alguna elección en la creación es errónea o no permitida.
8. La vista ModificarMenú guarda en línea la información en la base de datos MenúProfesor.
9. El usuario cierra la ventana en la vista ModificarMenú.
10. La vista ModificarMenú informa al controlador ModificarMenú, que el usuario cerro la vista.
11. El controlador ModificarMenú actualiza en la vista VistaPrincipal el menú creado.

5. Producción intelectual

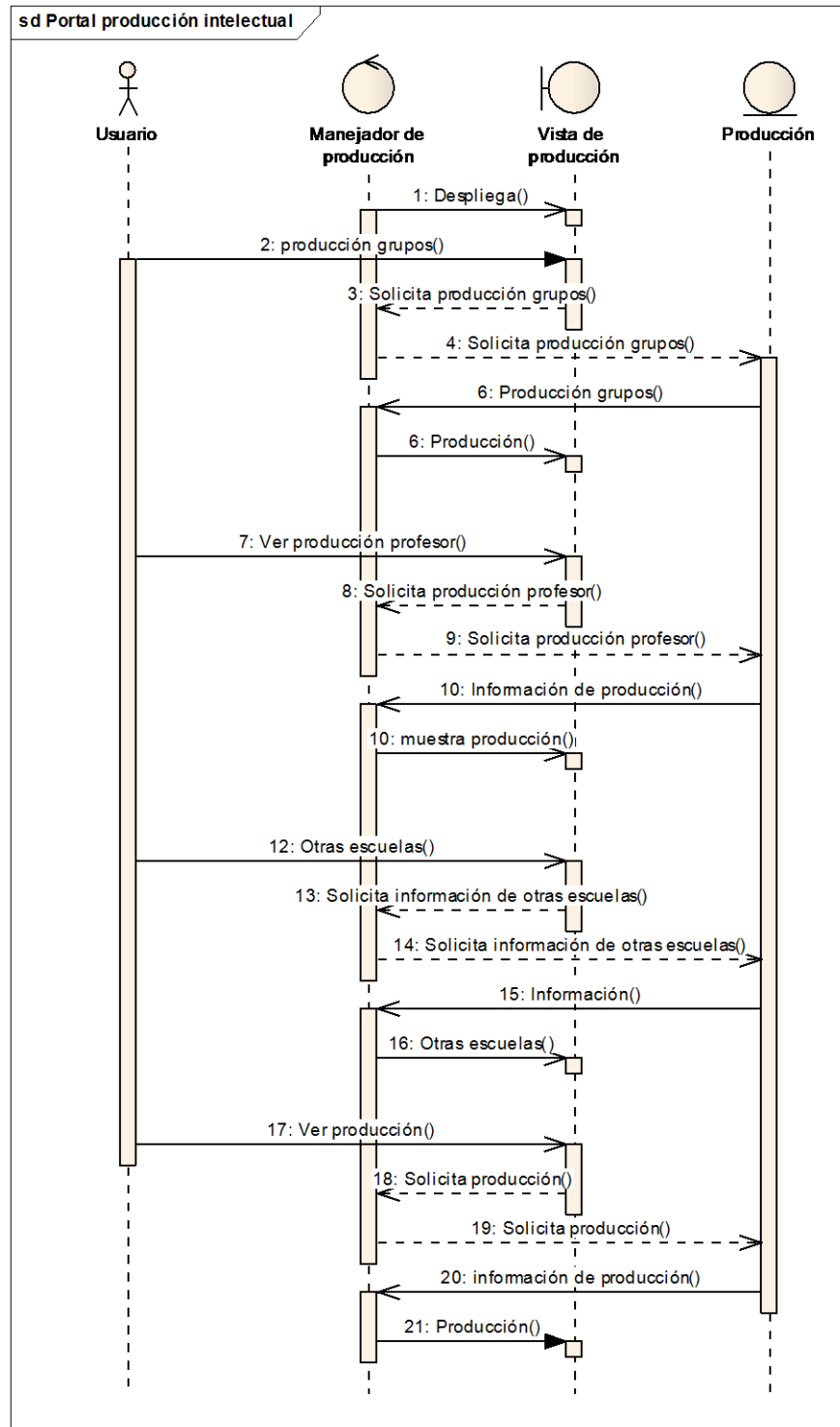
Figura 44. Diagrama de Secuencia: Producción intelectual.



El manejador principal despliega la vista principal. Ya sea el portal de la escuela, el portal de grupos o el portal de profesor.

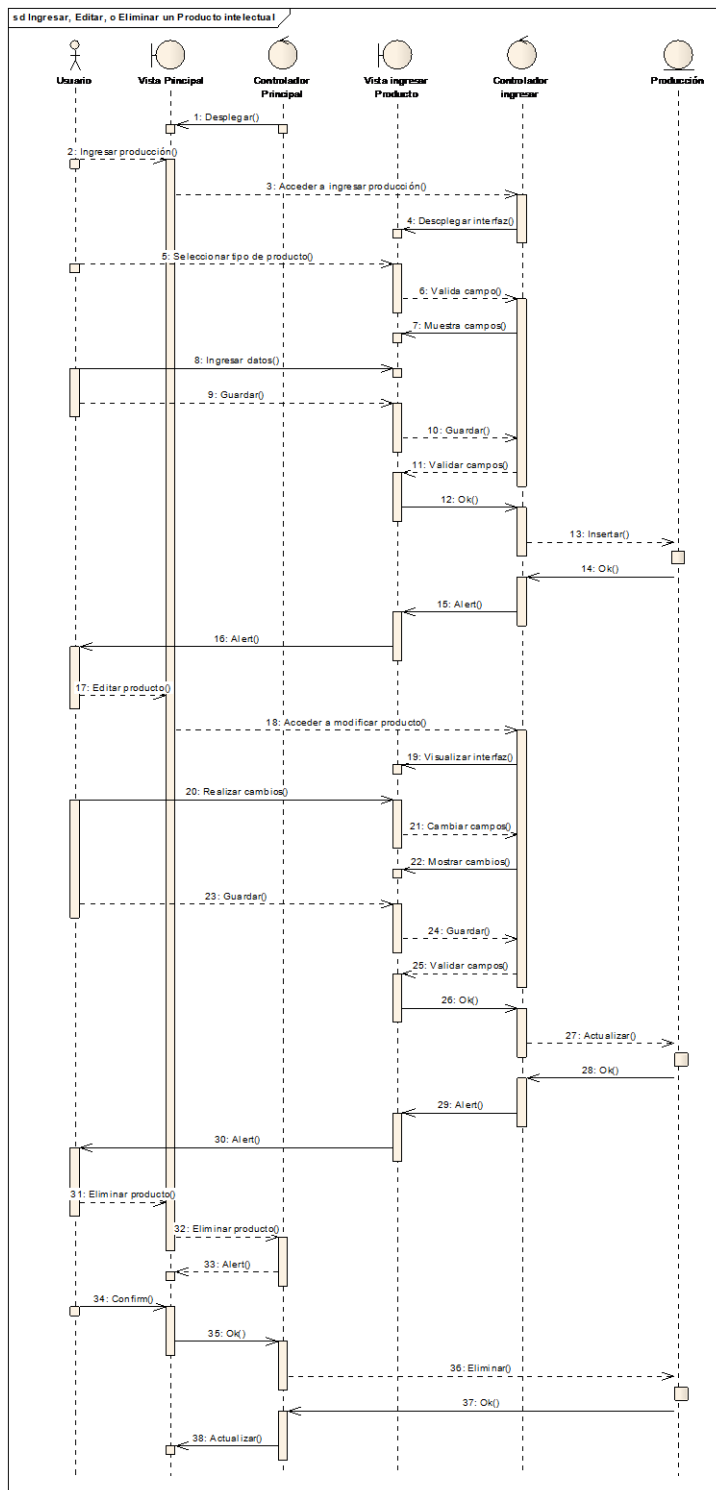
1. El usuario solicita el servicio de producción intelectual.
2. La vista principal solicita al manejador principal la producción.
3. El manejador principal solicita al manejador de producción la validación.
4. El manejador de producción confirma al manejador principal la validación del servicio.
5. El manejador principal despliega la vista de producción intelectual.
6. El usuario en la vista de producción intelectual accede a la producción de los grupos.
7. La vista principal solicita al manejador de producción la producción de los grupos.
8. El manejador de producción solicita la información a la base de datos.
9. La base de datos envía la información de la producción de grupos al manejador de producción.
10. El manejador de producción envía la información de los grupos a la vista de producción.
11. 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 45. Diagrama de Secuencia: Portal producción intelectual.



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.

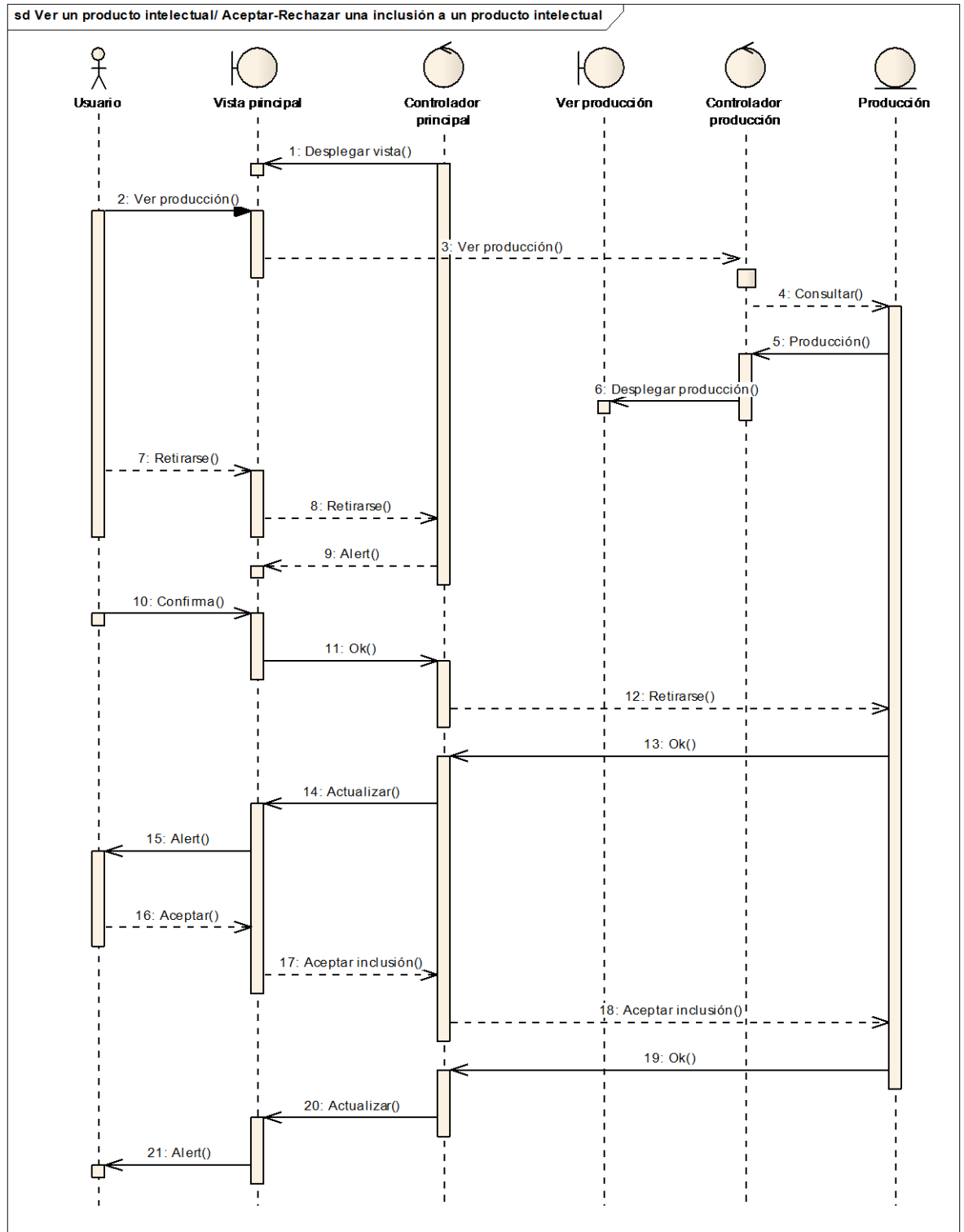
Figura 46. Diagrama de Secuencia: Ingresar un producto intelectual/ Modificar un producto intelectual/ Eliminar un producto intelectual.



1. El controlador ControladorPrincipal despliega la interfaz en la vista VistaPrincipal al usuario.
2. El usuario solicita ingresar un producto intelectual en la vista VistaPrincipal.
3. La vista VistaPrincipal le envía la solicitud de acceder a la interfaz de ingresar un producto intelectual al controlador ControladorIngresar.
4. El controlador ControladorIngresar recibe la petición y despliega la interfaz en la vista VistaIngresarProducto.
5. El usuario selecciona el tipo de producto en la vista VistaIngresarProducto.
6. La vista VistaIngresarProducto envía el tipo de producto seleccionado al controlador ingresar.
7. El controlador ingresar, muestra los campos que son relacionados a ese tipo de producto.
8. El usuario ingresa los datos en la vista VistaIngresarProducto.
9. El usuario solicita guarda en la vista VistaIngresarProducto.
10. La vista ingresar producto solicita guardar la información al controlador ingresar.
11. El controlador ingresar, valida los campos de la encuesta en la vista VistaIngresarProducto.
12. La vista VistaIngresarProducto le envía la confirmación de la validación de los campos al controlador ControladorIngresar.
13. El controlador ControladorIngresar inserta el producto ingresado en la base de datos Producción.
14. La base de datos Producción retorna un mensaje de confirmación al controlador ControladorIngresar.
15. El controlador ControladorIngresar despliega un alert de confirmación en la vista VistaIngresarProducto.
16. La vista VistaIngresarProducto envía un alert al usuario informándole sobre la creación de su producto.
17. El usuario solicita editar el producto intelectual en la vista VistaPrincipal.
18. La vista VistaPrincipal solicita acceder a editar producto intelectual al controlador ControladorIngresar.
19. El controlador ControladorIngresar recibe la petición y despliega la vista VistaIngresarProducto.
20. El usuario hace cambios al producto intelectual en la vista VistaIngresarProducto.
21. La vista VistaIngresarProducto solicita cambiar campos al controlador ControladorIngresar.
22. El controlador ControladorIngresar muestra los cambios en vista VistaIngresarProducto.
23. El usuario solicita guardar cambios en vista VistaIngresarProducto.
24. La vista VistaIngresarProducto envía la petición de guardar cambios al controlador ControladorIngresar.
25. El controlador ControladorIngresar recibe la petición y valida los campos en la vista VistaIngresarProducto.

26. La vista VistaIngresarProducto le envía una confirmación de validación de campos al controlador ControladorIngresar.
27. El controlador ControladorIngresar actualiza los campos en la base de datos Producción.
28. La base de datos Producción retorna un mensaje de confirmación al controlador ControladorIngresar.
29. El controlador ControladorIngresar despliega un alert de confirmación en vista VistaIngresarProducto.
30. la vista VistaIngresarProducto envía un alert con información acerca de guardar los cambios, al usuario.
31. El usuario solicita eliminar un producto intelectual en la vista VistaPrincipal.
32. La vista VistaPrincipal envía la petición de eliminar un producto intelectual, al controlador ControladorPrincipal.
33. El controlador ControladorPrincipal recibe la petición y despliega un alert de confirmación en vista VistaPrincipal.
34. El usuario confirma la eliminación del producto intelectual.
35. La vista VistaPrincipal le envía la confirmación de eliminar el producto intelectual al controlador ControladorPrincipal.
36. El controlador ControladorPrincipal elimina el producto intelectual de la base de datos Producción.
37. La base de datos Producción envía la confirmación de la eliminación al controlador ControladorPrincipal.
38. El controlador ControladorPrincipal actualiza la vista VistaPrincipal.

Figura 47. Diagrama de Secuencia: Gestionar Productos intelectuales



Ver un producto intelectual/ Aceptar-Rechazar una inclusión a un producto intelectual

1. El controlador ControladorPrincipal despliega la interfaz en la vista VistaPrincipal al usuario.
2. El usuario solicita ver un producto específico en la vista VistaPrincipal.
3. La vista VistaPrincipal, envía la solicitud de ver producto al controlador ControladorProducción.
4. El controlador ControladorProducción consulta el producto en la base de datos Producción.
5. La base de datos Producción le envía la información al controlador ControladorProducción.
6. El controlador ControladorProducción despliega la interfaz de resultados en la vista VistaProducción.
7. El usuario solicita retirarse de un producto específico en la vista VistaPrincipal.
8. La vista VistaPrincipal, envía la solicitud de retirarse de un producto al controlador ControladorProducción.
9. El controlador ControladorPrincipal recibe la petición y despliega un alert de confirmación en vista VistaPrincipal.
10. El usuario confirma que se quiere retirar del producto intelectual.
11. La vista VistaPrincipal le envía la confirmación de retirarse del producto intelectual al controlador ControladorPrincipal.
12. El controlador ControladorPrincipal retira el autor del producto intelectual de la base de datos Producción.
13. La base de datos Producción envía la confirmación de la eliminación de dicho autor en ese producto, al controlador ControladorPrincipal.
14. El controlador ControladorPrincipal actualiza la vista VistaPrincipal.
15. La vista VistaPrincipal envía un alert al usuario, informándole que ha sido retirado del producto.
16. El usuario selecciona aceptar la inclusión al producto intelectual.
17. La vista VistaPrincipal le envía la confirmación de aceptar la inclusión en el producto intelectual al controlador ControladorPrincipal.
18. El controlador ControladorPrincipal actualiza que el usuario aceptó la inclusión al producto intelectual en la base de datos Producción.
19. La base de datos Producción envía la confirmación de la actualización de dicho autor en ese producto, al controlador ControladorPrincipal.
20. El controlador ControladorPrincipal actualiza la vista VistaPrincipal.
21. La vista VistaPrincipal envía un alert de confirmación al usuario.

Anexo G. PRUEBAS DEL SISTEMA

1 Sistema de notificaciones para Eventos y Foros

Tabla 9. Pruebas Realizadas: Enviar correo de notificación de foros a nivel de comunidad UIS.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|---|------------------|
| Participar en foro | Se comprobó creando foros a nivel de comunidad UIS, y realizando diferentes participaciones de usuarios de otras escuelas en dicho foro. Se comprobó que se almacenará correctamente. | ✓ |
| Enviar correo de notificación | Verifica que en los foros que tienen nuevas participaciones, halla usuarios de la escuela, para notificarlos. Se comprobó depurando. Los resultados obtenidos fueron favorables debido a que envía el correo exitosamente a los usuarios implicados. | ✓ |
| Enviar correo de notificación de foros comunidad UIS. | Al acceder al index de cada portal de las escuelas, la notificación se activa, y de acuerdo a ciertas restricciones se realiza una notificación diaria. Por consiguiente la prueba realizada fue: crear un foro UIS, y realizar participaciones de otros usuarios pertenecientes a otras escuelas y de la | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--------------------|---|------------------|
| | misma escuela. Luego invocar al portal web para activar la notificación. Se recibió el correo de notificación satisfactoriamente. | |

Tabla 10. Pruebas Realizadas: Enviar correo de notificación de foros comunidad Escuela.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--|--|------------------|
| Participar en foro Escuela | Se comprobó creando foros a nivel de comunidad escuela, realizando diferentes participaciones de usuarios. Se comprobó que se almacena correctamente en base de datos. | ✓ |
| Enviar correo de notificación | Se verificó que validará el envío y se entregara el correo de notificación de manera exitosa. | ✓ |
| Enviar correo de notificación de foros comunidad Escuela | Al acceder al index de cada portal de las escuelas, la notificación se activa, y de acuerdo a ciertas restricciones se realiza una notificación diaria. Por consiguiente la prueba realizada fue: crear un foro escuela, y realizar comentarios de usuarios pertenecientes la escuela. Luego invocar al portal web para activar la | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--------------------|---|------------------|
| | notificación y el envío del correo. Como resultado se recibió el correo de notificación satisfactoriamente. | |

Tabla 11. Pruebas Realizadas: Enviar correo de notificación de eventos comunidad UIS.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|-------------------------------|--|------------------|
| Participar en evento UIS | Se comprobó creando un evento a nivel de comunidad UIS, realizando diferentes comentarios de usuarios. Se comprobó que se almacena correctamente en base de datos. | ✓ |
| Unirse al evento | Se unieron varios usuarios a un evento propuesto, con ello se comprobó que la información de los usuarios y la relacionada al evento se almacenara correctamente en base de datos. | ✓ |
| Enviar correo de notificación | Se verificó que validará el envío y se entregara el correo de notificación de manera exitosa, incorporando en el correo una lista de los nombres y la escuela de los usuarios que han comentado, y los que se han unido. | ✓ |
| Enviar correo de notificación | Al acceder al index de cada portal de las escuelas, la notificación se activa, | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---------------------------|---|------------------|
| de eventos comunidad UIS. | y de acuerdo a ciertas restricciones se realiza una notificación diaria. Por consiguiente la prueba realizada fue, crear un evento UIS, luego realizar participaciones de otros usuarios pertenecientes a otras escuelas y de la misma escuela, a su vez que algunos de estos usuarios se unieran a la propuesta. Luego invocar al portal web para activar la notificación. Como resultado se recibió el correo de notificación satisfactoriamente. | |

Tabla 12. Pruebas Realizadas: Enviar correo de notificación de eventos comunidad Escuela.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|------------------------------|--|------------------|
| Participar en evento escuela | Se comprobó creando foros a nivel de comunidad escuela, realizando diferentes participaciones de usuarios. Se comprobó que se almacena correctamente en base de datos. | ✓ |
| Unirse al evento | Se unieron varios usuarios al evento, con base a ello, se comprobó que la información se almacena correctamente. | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--|---|------------------|
| <p>Enviar correo de notificación</p> | <p>Se verificó que validará el envío y se entregara el correo de notificación de manera exitosa, creando una lista de los usuarios que se han unido a la propuesta y los que han participado.</p> | <p>✓</p> |
| <p>Enviar correo de notificación de eventos comunidad Escuela.</p> | <p>Al acceder al index de cada portal de las escuelas, la notificación se activa, y de acuerdo a ciertas restricciones se realiza una notificación diaria. Por consiguiente la prueba realizada fue, crear un evento escuela, luego realizar participaciones de otros usuarios pertenecientes a la escuela, a su vez que algunos de estos usuarios se unieran a la propuesta. Luego invocar al portal web para activar la notificación. Como resultado se recibió el correo de notificación satisfactoriamente.</p> | <p>✓</p> |

2 Correos por nivel del semestre y programa académico

Tabla 13. Pruebas Realizadas: Correos por nivel, selección programas académicos

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--------------------------------|--|------------------|
| Seleccionar escuelas destino | Se comprobó que al seleccionar las escuelas destino, consultara y mostrara los programas académicos relacionados. | ✓ |
| Seleccionar programa académico | Se verifico que al seleccionar programas académicos, habilitara el envío del correo por niveles o a todos los niveles. | ✓ |

Figura 48. Pruebas Realizadas: Correos por nivel, selección de nivel

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--------------------|---|------------------|
| Habilitar niveles | El sistema consulta si el usuario ha escogido programas académicos que tienen la misma duración en semestres, si es así, habilita los niveles de dichos programas, para que el usuario pueda escoger. De lo contrario, si los programas académicos son de diferente duración, el correo se envía a todos los semestres. | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|-----------------------|--|------------------|
| | Se comprobó que esta función se realizara correctamente. | |
| Seleccionar nivel(es) | Se verificó que al seleccionar un nivel o varios, habilitara para digitar el asunto y el mensaje del correo. | ✓ |

Figura 49. Pruebas Realizadas: Envío de correo por programa-nivel.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--|--|------------------|
| Correo por nivel: Guardar información | Se comprobó que se guardara la información en base de datos. | ✓ |
| Correo por nivel: Programación del Envío | Se verificó que la programación del envío del mensaje funcionara correctamente, es decir, que el mensaje se colocara en la lista de los que se encuentra por enviar y así mismo fuera enviado luego. | ✓ |
| Correo por nivel: Envío al destinatario | Se comprobó que el correo se enviara correctamente a los usuarios destino. | ✓ |

3 Correos UIS

Tabla 14. Pruebas Realizadas: Visualizar correos creados.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|-------------------------------------|---|------------------|
| Visualizar correos | Se verificó que se listaran todos los correos que se encuentran en base de datos y que han sido creados por el usuario en sesión. | ✓ |
| Consultar correo | Se comprobó que al consultar el correo, mostrara la información correspondiente al correo seleccionado. | ✓ |
| Visualizar destinatarios del correo | Se comprobó que muestre los destinatarios del correo seleccionado. | ✓ |
| Eliminar destinatarios del correo | Se verificó la eliminación de un correo con todos sus destinatarios, validando las restricciones que se establecieron. | ✓ |
| Crear nuevo correo | Se verificó que redirija correctamente a la nueva interfaz en la cual se pueda crear el correo. | ✓ |

Tabla 15. Pruebas Realizadas: Crear correos UIS.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--------------------|--|------------------|
| Crear correo | Al crear un correo, valida que se diligencien todos los campos | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|-------------------------------------|--|------------------|
| | obligatorios y lo almacena correctamente. | |
| Seleccionar destinatarios | Se comprobó que se puedan buscar las escuelas, y al colocar el cursor en el buscador que muestre la lista de las escuelas. | ✓ |
| Seleccionar criterio de búsqueda | Se verificó que listara los criterios de búsqueda. | ✓ |
| Digitar o seleccionar criterio | Se comprobó que al ingresar un dato busque el destinatario, y si se selecciona un criterio, muestre los criterios disponibles a escoger. | ✓ |
| Digitar asunto y mensaje del correo | Se validó que se pueda escribir el mensaje y el asunto. A su vez se valida que se diligencien todos los campos. | ✓ |
| Enviar correo | Valida que todos los campos obligatorios sean digitados. Cuando todos los campos son digitados, se coloca el correo en proceso de envío. Se comprobó que lo realiza correctamente. | ✓ |

Tabla 16. Pruebas Realizadas: Visualizar y eliminar destinatarios del correo UIS.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---------------------------|---|------------------|
| Visualizar destinatarios. | Se verificó que se listaran todos los destinatarios que se encuentran en base de datos, a su vez si el correo ha sido enviado que muestre los resultados del envío y la fecha de envío del mismo. Sí no se ha enviado que permita agilizar el envío si y solo sí el envío es posible en ese momento. Todo esto se comprobó satisfactoriamente, creando correos con diferentes criterios de búsqueda y varios destinatarios. | ✓ |
| Agilizar envío del correo | Se validó que agilizar el envío, mediante depuración; a su vez analizando las actualizaciones del estado del correo en base de datos y en la interfaz. | ✓ |
| Eliminar destinatarios | Elimina correctamente el correo destinado al tipo de categoría seleccionado. También valida el estado del correo para dar permisos al usuario de eliminarlo. | ✓ |

Tabla 17. Pruebas Realizadas: Eliminar correo UIS.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|------------------------------|---|------------------|
| Revisar estado del correo | Se comprobó que el sistema muestre correctamente el estado del correo en tiempo real. | ✓ |
| Habilitar opción de eliminar | Verifica correctamente el estado del correo; así mismo se comprobó que habilita la opción de eliminar al usuario. | ✓ |
| Eliminar correo | Elimina correctamente un correo UIS y los registros que dependan de éste en la base de datos. A su vez se comprobó que se validaran las restricciones establecidas. | ✓ |

4 Menú del portal del profesor

Tabla 18. Pruebas Realizadas: Crear menú para el portal del profesor.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--------------------|---|------------------|
| Crear menú | Se comprobó que redirija correctamente a la interfaz que permite crear el menú. | ✓ |
| Agregar menú | Se comprobó que agregara exitosamente menús al menú del profesor. | ✓ |
| | Se validó que agregara un submenú, | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|---|------------------|
| Agregar submenú | validando las restricciones establecidas en los requerimientos. | |
| Cambiar posición menú o submenú | Se comprobó que funciona correctamente el cambio de posición de los niveles y subniveles del menú. | ✓ |
| Eliminar un menú | Elimina correctamente el menú. | ✓ |
| Eliminar submenú | Elimina correctamente el submenú. | ✓ |
| Consultar si el menú o submenú se puede agregar | Valida correctamente si puede agregar o no el menú o submenú dependiendo de las restricciones establecidas. Se realizaron pruebas y se comprobó que el comportamiento que presenta el sistema es el esperado. | ✓ |

Tabla 19. Pruebas Realizadas: Visualizar menú actual/ Actualizar rutas de acceso.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---------------------------------|---|------------------|
| Consultar si hay un menú actual | Se verificó que el sistema consulte correctamente la información de la base de datos. | ✓ |
| | Muestra correctamente el menú del | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|--|------------------|
| Visualizar menú | portal de profesor que se encuentra en bases de datos. | |
| Modificar menú | Se comprobó que permita modificar el menú actual, redirigiendo correctamente a la interfaz que permite realizar dicha acción. | ✓ |
| Actualizar rutas del menú actual | Se comprobó que actualizará las rutas de acceso a cada servicio que se encuentre en el menú creado. Se verificó en base de datos la actualización. | ✓ |
| Crear menú para el portal del profesor. | Se verificó creando un menú para el portal del profesor, valida agregar subniveles a un nivel ya determinado, y que ese nivel o subnivel no se encuentre más de una vez. A su vez se comprobó que almacenara la información correctamente en base de datos y actualizara la información en la página correspondiente a la visualización. | ✓ |

Tabla 20. Pruebas Realizadas: Modificar menú actual.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|--|------------------|
| Modificar menú actual | Al ingresar a modificar menú, permite visualizar el menú actual y los menús que puede agregar. Se comprobó que funciona correctamente, y almacena la información en la base de datos y recarga la página con la nueva información. | ✓ |
| Cambiar de posición un menú o submenú | Se comprobó que permita cambiar la posición de un menú o submenú. | ✓ |
| Eliminar un menú o submenú | Elimina correctamente un menú o submenú seleccionado. | ✓ |
| Agregar un menú | Agrega un menú correctamente. | ✓ |
| Consultar si el menú o submenú se encuentra en el menú actual | El sistema verifica si el usuario puede agregar o no determinado menú o submenú, dependiendo del menú actual. | ✓ |
| Agregar submenú | Valida correctamente agregar un submenú si y solo si existe un menú | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--------------------|-----------------------------|------------------|
| | en el que se pueda agregar. | |

5 Producción Intelectual

Tabla 21. Pruebas Realizadas: Visualizar productos intelectuales.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--|---|------------------|
| Visualizar productos intelectuales digitados | Se comprobó que listara los productos digitados por el usuario. | ✓ |
| Modificar producto intelectual | Al visualizar los productos intelectuales digitados, permite modificarlo. Se comprobó que dirija a la interfaz relacionada con modificar el producto intelectual. | ✓ |
| Eliminar producto intelectual | Se verificó que elimina correctamente el producto intelectual de la base de datos y de la interfaz. | ✓ |
| Ingresar producto intelectual | Se comprobó que redirige correctamente a la interfaz que permite ingresar un producto intelectual. | ✓ |
| Ver producción intelectual en la | Se verificó que liste de manera correcta la producción intelectual en | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|------------------------|--|------------------|
| cual lo han etiquetado | la cual lo han etiquetado. | |
| Aceptar la inclusión | Se verificó que permita aceptar la inclusión al producto intelectual. | ✓ |
| Rechazar la inclusión | Se comprobó que permita rechazar la inclusión al producto intelectual. | ✓ |

Tabla 22. Pruebas Realizadas: Ingresar producto intelectual.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|--|------------------|
| Ingresar producto intelectual | Al ingresar un producto intelectual, valida que se diligencien todos los campos obligatorios y se almacenen correctamente en la base de datos. | ✓ |
| Seleccionar el tipo de producto intelectual | Verifica que se seleccione un tipo de producto intelectual. | ✓ |
| Digitar datos | Se validan los datos de acuerdo a los parámetros establecidos en los campos de texto. | ✓ |
| Guardar producto intelectual | Se valida los campos obligatorios a digitar para insertar la información del producto intelectual en la base de datos. Se comprobó que la información se almacena correctamente. | ✓ |
| Validar información | Valida la información digitada de acuerdo a las restricciones establecidas. | ✓ |

Tabla 23. Pruebas Realizadas: Modificar producto intelectual.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|--|--|------------------|
| Editar información del producto intelectual | Se verificó que permita editar un producto intelectual creado, y que mostrara la información correctamente. | ✓ |
| Modificar datos | Se comprobó que permite modificar los datos y actualiza la información en base de datos. | ✓ |
| Agregar nuevos participantes | Se comprobó que permita agregar nuevos participantes. | ✓ |
| Asociar o eliminar grupos del producto intelectual | Se verificó que eliminara correctamente uno o varios grupos asociados al producto intelectual, desde la interfaz y en base de datos. Así mismo se comprobó que permitiera agregar nuevos grupos al producto intelectual y que se almacenaran correctamente en base de datos. | ✓ |
| Guardar producto intelectual | Verifica los campos obligatorios, y reemplaza correctamente la nueva información en la base de datos. | ✓ |

Tabla 24. Pruebas Realizadas: Consultar producción intelectual públicamente.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|----------------------------------|---|------------------|
| Ver producción intelectual de la | Se comprobó que muestre la producción intelectual de la escuela | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|---|------------------|
| escuela | que se encuentra almacenada en la base de datos. | |
| Ver producción intelectual del grupo | Muestra la información de la producción intelectual del grupo que se encuentra almacenada en base de datos. | ✓ |
| Ver producción intelectual de un profesor | Se comprobó que muestra la producción intelectual del profesor seleccionado. | ✓ |

Tabla 25. Pruebas Realizadas: Consultar producción intelectual de un grupo específico.

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|--|------------------|
| Ver producción intelectual del grupo | Se comprobó que lista la producción intelectual del grupo que se encuentra almacenada en base de datos. | ✓ |
| Ver un producto intelectual | Se verificó que al hacer click en ver el producto intelectual, redirija al portal de producción intelectual para ver el detalle de la información del producto intelectual seleccionado. | ✓ |
| Ver producción intelectual en la cual han incluido al grupo | Se comprobó que muestra la producción intelectual en la cual han incluido al grupo, y que se encuentra almacenada en base de datos. | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|--|------------------|
| Aceptar la inclusión al producto intelectual | Se comprobó que a los usuarios que tienen el permiso de aceptar la inclusión, les muestre la interface con la producción intelectual. Se verificó que al aceptar la inclusión, se almacena la información en base de datos, y se actualice la información en la interface. | ✓ |
| Rechazar la inclusión al producto intelectual | Se comprobó que los usuarios que tengan | ✓ |

Tabla 26. Pruebas Realizadas: Rechazar o aceptar inclusión en un producto intelectual

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|-----------------------|---|------------------|
| Visualizar producción | Se comprobó que consultara los productos intelectuales en los cuales el usuario ha sido incluido. | ✓ |
| Aceptar la inclusión | Se verificó que al aceptar la inclusión en el producto intelectual, se actualizara la información en base de datos. Así mismo se actualizara la información en la página en la cual es mostrada dicha | ✓ |

| CASO DE USO | PRUEBA REALIZADA | RESULTADO |
|---|---|------------------|
| | información. | |
| Rechazar la inclusión | Se comprobó que al rechazar una inclusión en un producto, eliminara en base de datos la relación del usuario con ese producto. También se verificó que informara al usuario que digito el producto intelectual. | ✓ |
| Enviar mensaje de notificación al digitador | Se comprobó que si algún usuario rechaza la participación en algún producto en el cual ha sido incluido; y este usuario es el único autor en dicho producto, se notificara al digitador del producto | ✓ |
| Enviar mensaje de notificación al administrador | Se verificó que si el usuario que digito el producto intelectual no se encontraba activo, enviaba la notificación al administrador del portal de la respectiva escuela a la cuál pertenecía el digitador. | ✓ |