

**ADMINISTRACIÓN, ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN
DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE
GEOLOGÍA, INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES**

**YADIANA LAITÓN ZÁRATE
CARLOS FERNANDO RUIZ NIETO**

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

2013

**ADMINISTRACIÓN, ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN
DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE
GEOLOGÍA, INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES**

**YADIANA LAITÓN ZÁRATE
CARLOS FERNANDO RUIZ NIETO**

Trabajo de grado para optar al título de Ingeniero de Sistemas

**Director
Msc. LUIS IGNACIO GONZÁLEZ RAMÍREZ
Magíster en Informática**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECAÑICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA
2013**

Agradecimientos

Principalmente a Dios por darme la oportunidad de haber ingresado a esta gran universidad, a mi madre por apoyarme siempre, a mi padre que desde el cielo me cuida llevándome por el camino correcto, a mis hermanos que siempre fueron incondicionales, a Calumet porque con ellos aprendí y apliqué mis conocimientos y por supuesto a Carlos, quien es el principal autor de mi amor por esta carrera, quien me enseñó a ser la mejor y compartió conmigo su inteligencia, siempre fue mi apoyo incondicional.

Yadiana Laitón Zárte

A mi profesor de algoritmos, quien me dio las bases, a mis compañeros de la UNAB con quienes di mis primeros pasos, a mis compañeros de Calumet con quienes aprendí muchas cosas, a mi familia por siempre estar ahí para apoyarme y confiar en mí, y por supuesto a Yadiana, con quién compartí más que una relación, una amistad, a ella le doy mil gracias por haber estado conmigo, en las buenas y en las malas.

A todos los hago partícipes de este pequeño triunfo, el primero de muchos.

Carlos Fernando Ruiz Nieto

CONTENIDO

INTRODUCCIÓN	17
1 PRESENTACIÓN DEL PROYECTO.....	18
1.1 ORIENTACIÓN SOBRE EL CONTENIDO DEL INFORME.....	18
1.2 ANTECEDENTES	19
1.3 DEFINICIÓN DEL PROBLEMA.....	19
1.4 OBJETIVOS	21
1.4.1 Objetivo General	21
1.4.2 Objetivos Específicos	21
1.5 JUSTIFICACIÓN	25
1.6 ALCANCES Y LIMITACIONES.....	26
2 MARCO TEÓRICO.....	27
2.1 ARQUITECTURA CLIENTE/SERVIDOR	27
2.1.1 Características de la arquitectura Cliente/Servidor	28
2.1.2 Clasificación de las arquitecturas Cliente/Servidor.....	29
2.1.3 Arquitectura Cliente/Servidor aplicada	32
2.1.4 Ventajas del esquema Cliente/Servidor.....	32
2.1.5 Desventajas del esquema Cliente/Servidor.....	33
2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS	33
2.2.1 Código del Lado del Cliente (Client Side Scripts).....	34
2.2.2 Código del Lado del Servidor (Server Side Scripts)	34
2.2.3 Tecnología aplicada	35
2.3 BASES DE DATOS	37
2.3.1 Modelos de Bases De Datos	38
2.3.2 Acceso a Base de Datos	39
2.3.3 Conectores más utilizados	39
2.3.4 Manejadores o Gestores de Bases de Datos	41
2.3.5 MySQL	42

2.3.6	Ventajas de MySQL	43
2.4	NETBEANS	43
2.5	SISTEMA DE CONTROL DE VERSIONES	44
2.5.1	Subversión	45
2.6	PROGRAMACIÓN UTILIZADA	45
2.6.1	Clases	46
2.6.2	Objetos	46
2.6.3	Atributos	46
2.6.4	Métodos.....	47
2.6.5	Herencia.....	47
2.6.6	Beneficios de la Programación Orientada a Objetos.....	47
2.6.7	Java y JDK (Java Development Kit)	47
2.7	SERVIDORES WEB.....	48
2.7.1	Servidor Jakarta Tomcat	48
3	MARCO METODOLÓGICO	50
3.1	PROTOTIPO EVOLUTIVO.....	50
3.2	LENGUAJE DE MODELADO UNIFICADO	52
3.2.1	Diagramas de UML	53
3.2.2	Diagramas de casos de uso	53
3.2.3	Diagramas de secuencias	55
3.3	ESTÁNDARES DE PROGRAMACIÓN	57
3.3.1	Modelo de datos.....	57
3.3.2	Nombres de las tablas.....	58
3.3.3	Clases	59
3.3.4	Páginas JSP	59
3.3.5	Organización de Directorios	59
4	DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO.....	60
4.1	PROTOTIPO ESPERADO	60
4.1.1	Diagramas de Casos de Uso.....	70
4.1.2	Documentación de Casos de Uso del Sistema	76

4.1.3	Diseño y Análisis	85
4.1.4	Modelo de Procesos del Sistema	90
4.1.5	Implementación, Implantación y Pruebas Generales	123
4.2	MANTENIMIENTO Y ADMINISTRACIÓN	124
4.2.1	Actividades de Mantenimiento.....	124
4.2.2	Actividades de Soporte a Usuarios	132
4.2.3	Actividades de Administración.....	132
5	PRUEBAS DEL SISTEMA.....	134
5.1	PRUEBAS DE VERIFICACIÓN.....	134
5.1.1	Pruebas por componente	134
5.2	PRUEBAS DE INTEGRACIÓN.....	143
5.3	PRUEBAS DE VALIDACIÓN.....	143
6	CONCLUSIONES.....	144
7	RECOMENDACIONES	146
	BIBLIOGRAFÍA	147
	ANEXOS.....	149

LISTA DE TABLAS

Tabla 1. Casos de uso: Gestión ayudas	76
Tabla 2. Casos de uso: Gestión Preguntas Frecuentes.....	76
Tabla 3. Casos de uso: Sugerencias Ayudas	77
Tabla 4. Casos de Uso: Portal Ayudas	77
Tabla 5. Caso de Uso: Cronogramas Académicos	79
Tabla 6. Casos de uso: Inscripciones	80
Tabla 7. Casos de uso: Ver Inscripciones.....	82
Tabla 8. Casos de uso: Mantenimiento y Administración de los portales GEOWEB y E3TWeb	83
Tabla 9. Descripción de las Entidades.....	87
Tabla 10. Pruebas Realizadas: Gestión Ayudas.....	135
Tabla 11. Pruebas realizadas: Gestión Preguntas Frecuentes.....	136
Tabla 12. Pruebas realizadas: Sugerencias Ayudas.....	137
Tabla 13. Pruebas realizadas: Portal Ayudas	137
Tabla 14. Pruebas realizadas: Cronogramas Académicos	138
Tabla 15. Pruebas realizadas: Inscripciones.	140
Tabla 16. Pruebas realizadas: Ver Inscripciones	142

LISTA DE FIGURAS

Figura 1. Modelo Cliente/Servidor.....	28
Figura 2. Arquitectura Cliente/Servidor de Dos Capas	30
Figura 3. Arquitectura Cliente/Servidor de Tres Capas.....	31
Figura 4. Modelo de acceso a JSP	37
Figura 5. Función del DBMS.....	42
Figura 6. Prototipo Evolutivo.....	50
Figura 7. Diagramas de Casos de Uso	54
Figura 8. Diagrama de Secuencias.....	57
Figura 9. Diagrama de Casos de Uso: Gestión Ayudas.....	70
Figura 10. Diagrama de Casos de Uso: Gestión Preguntas Frecuentes.	71
Figura 11. Diagrama de Casos de Uso: Sugerencias Ayudas.	71
Figura 12. Diagrama de Casos de Uso: Portal Ayudas.....	72
Figura 13. Diagrama de Casos de Uso: Cronogramas Académicos.....	73
Figura 14. Diagrama de Casos de Uso: Inscripciones.....	74
Figura 15. Diagrama de Casos de Uso: Ver Inscripciones.	74
Figura 16. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.	75
Figura 17. Diagrama E/R: Ayudas	85
Figura 18. Diagrama E/R: Cronogramas Académicos	86
Figura 19. Diagrama E/R: Inscripciones	87
Figura 20. Diagrama de secuencia: Gestionar Ayudas - Crear Ayuda.....	90
Figura 21. Diagrama de secuencia: Gestionar Ayudas - Editar Ayuda	91
Figura 22. Diagrama de secuencia: Gestionar Ayudas - Eliminar Ayuda.....	92
Figura 23. Diagrama de secuencia: Gestionar Ayudas - Crear Menú Ayuda.....	94
Figura 24. Diagrama de secuencia: Gestionar Ayudas - Editar Menú Ayuda	95
Figura 25. Diagrama de secuencia: Gestionar Ayudas - Eliminar Menú Ayuda.....	96
Figura 26. Diagrama de secuencia: Gestionar Ayudas – Sugerencias Ayudas	97

Figura 27. Diagrama de secuencia: Gestionar Preguntas Frecuentes - Crear Pregunta Frecuente	99
Figura 28. Diagrama de secuencia: Gestionar Preguntas Frecuentes - Editar Pregunta Frecuente	100
Figura 29. Diagrama de secuencia: Gestionar Preguntas Frecuentes - Eliminar Pregunta Frecuente	102
Figura 30. Diagrama de secuencia: Portal Ayudas - Ver Temas Ayudas	103
Figura 31. Diagrama de secuencia: Portal Ayudas - Ver Preguntas Frecuentes	104
Figura 32. Diagrama de secuencia: Portal Ayudas - Enviar Sugerencia.....	105
Figura 33. Diagrama de secuencia: Gestionar Cronogramas Académicos - Crear Cronograma Académico	106
Figura 34. Diagrama de secuencia: Gestionar Cronogramas Académicos - Editar Cronograma Académico	108
Figura 35. Diagrama de secuencia: Gestionar Cronogramas Académicos - Eliminar Cronograma Académico	109
Figura 36. Diagrama de secuencia: Gestionar Cronogramas Académicos - Programar Cronograma Académico	110
Figura 37. Diagrama de secuencia: Gestionar Cronogramas Académicos – Ver Calendario	111
Figura 38. Diagrama de secuencia: Gestionar Inscripciones - Crear Inscripción	113
Figura 39. Diagrama de secuencia: Gestionar Inscripciones - Editar Inscripción	114
Figura 40. Diagrama de secuencia: Gestionar Inscripciones - Eliminar Inscripción	116
Figura 41. Diagrama de secuencia: Gestionar Inscripciones - Duplicar Inscripción	117
Figura 42. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Estados	118
Figura 43. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Enviar Correo.....	119

Figura 44. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Eliminar Interesado	120
Figura 45. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Información Interesado.....	121
Figura 46. Diagrama de secuencia: Ver Inscripciones - Diligenciar Formulario ...	122
Figura 47. Portal de Ayudas	149

LISTA DE ANEXOS

Anexo A. MANUAL DE USUARIO	149
----------------------------------	-----

RESUMEN

TÍTULO: ADMINISTRACIÓN, ANÁLISIS DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE GEOLOGÍA E INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONESⁱ.

AUTORESⁱⁱ: Yadiana Laitón Zárate
Carlos Fernando Ruiz Nieto

PALABRAS CLAVE: Portal Web, Ayuda, Calendario Académico, Inscripción, ESGEO (Escuela de Geología), E3T (Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones).

DESCRIPCIÓN

En la actualidad las Escuelas de Geología e Ingeniería Eléctrica, Electrónica y de Telecomunicaciones cuentan con un sistema de información orientado a la Web que proporciona servicios y canales de comunicación a docentes, estudiantes y administrativos de las mismas. Estos servicios ofrecen soluciones a los problemas y necesidades de los usuarios, por esta razón las labores de mantenimiento y actualización son indispensables.

Actualmente las escuelas cuentan con un servicio de sugerencias, a través del cual se da soporte a los usuarios que tienen inquietudes o desconocen el funcionamiento del sitio. Con el fin de mejorar el soporte, se creó una aplicación que provee ayudas prácticas, que dan solución a preguntas frecuentes y da capacitación en el funcionamiento de los servicios que se ofrecen. A través de este servicio de ayudas se logra una mejor atención a los usuarios y se amplía el conocimiento que tiene la comunidad acerca del portal de sus respectivas escuelas.

Las escuelas cuentan con programas de especialización, maestrías y doctorado. Cada escuela debe elaborar un calendario académico con fechas, ciudad, lugar y docente por cada cohorte de cada programa. Para esto se implementó un servicio que asiste a la persona que organiza el calendario, manteniéndolo constantemente informado de cruces de horarios, cruce de docentes, cruces de aulas, y otras limitantes al momento de armar el calendario académico de cada cohorte de cada programa.

Además parte del manejo de los posgrados y otras actividades organizadas por las escuelas, incluye la inscripción de posibles interesados a ingresar a ellos. Se creó un servicio que permite a las escuelas, realizar la inscripción de los interesados en ingresar a sus programas o actividades. A través de este servicio el personal de las escuelas podrá gestionar mejor la información que reciben, manteniendo una comunicación sencilla con los interesados.

ⁱ Trabajo de grado Modalidad Practica Empresarial.

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

ABSTRACT

TITLE: ADMINISTRATION, ANALYSIS, DESIGN, DEVELOPMENT AND IMPLEMENTATION OF THE ACADEMIC MANAGEMENT SYSTEM FOR THE GEOLOGY AND ELECTRICAL, ELECTRONICS, AND COMMUNICATIONS ENGINEERING PORTALSⁱ.

AUTHORSⁱⁱ: Yadiana Laitón Zárate
Carlos Fernando Ruiz Nieto

KEYWORDS: Web Portal, Help, Academic Schedule, Inscription, ESGEO (Geology School), E3T (Electrical, Electronic, and Communications Engineering School).

DESCRIPTION

Currently Geology and Electrical, Electronic, and Communications Engineering School have a Web information system that provides services and communication channels to teachers, students and administration staff. These services offer solutions to the problems and needs of the users, therefore the maintenance and updating are essential.

Currently schools have a suggestions service, which gives support to users who have concerns or do not know how the site works. In order to improve this support, an application that provides practical help was created, giving solutions to common questions and providing capacitation on the services offered.

The schools have specialization, masters and doctoral programs. Each school must make an academic schedule with dates, city, teacher and place for each cohort of each program. A service that assists the person organizing the schedule has been implemented, keeping him constantly informed of crossings in the schedules of teachers and classrooms, and other constraints when assembling the academic schedule of each cohort of each program.

Also part of the management of graduate programs and other activities organized by schools, includes the registration of interested parties to join them. A service that allows schools to register those interested in joining their programs or activities was created. Through this service the school staff can improve the management of putty the information they receive, maintaining easy communication with the interested ones.

ⁱ Degree Work: Business Practice.

ⁱⁱ Industrial and Business Studies School, School of Systems Engineering and Computing Science.
Director: M.Sc. Luis Ignacio González Ramírez

INTRODUCCIÓN

Los portales GEOWEB y E3TWeb pertenecientes a las Escuelas de Geología e Ingeniería Eléctrica, Electrónica y de Telecomunicaciones respectivamente, se han convertido en uno de los principales canales de comunicación e integración entre los miembros de la comunidad de las escuelas. Gracias a la aceptación y utilización por parte de los usuarios surge el objetivo fundamental de fortalecer y mejorar los servicios que se ofrecen, además de la creación de nuevos servicios para suplir las necesidades crecientes de los miembros de la escuela.

El grupo de desarrollo de software Calumet se encarga de desarrollar, administrar y mantener los portales GEOWeb y E3TWeb 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 continuo soporte ofrecido a los portales GEOWeb y E3TWeb por el grupo Calumet se hace de manera integral permitiendo el mejoramiento continuo de los servicios ofrecidos así como la creación de nuevos servicios, tales como un módulo de ayudas para brindar un mejor soporte a los usuarios, un gestor para la creación de cronogramas académicos de programas de posgrado y un gestor para la inscripción de interesados a programas y/o actividades.

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 PRESENTACIÓN DEL PROYECTO

1.1 ORIENTACIÓN SOBRE EL CONTENIDO DEL INFORME

Este documento presenta en detalle las etapas llevadas en el desarrollo del proyecto: ADMINISTRACIÓN, ANÁLISIS, DISEÑO, DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE GESTIÓN ACADÉMICA DE LOS PORTALES DE GEOLOGÍA, INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES; la información está distribuida de la siguiente forma:

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

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

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

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

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

CAPITULO 6. Conclusiones del trabajo realizado.

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

1.2 ANTECEDENTES

Desde su creación en 1969, el programa de Ingeniería de Sistemas de la Universidad Industrial de Santander ha crecido y evolucionado considerablemente. Cambios tecnológicos han impulsado la constante mejora del plan de estudios, así como la implementación de canales de comunicación que faciliten la integración de la comunidad, la cual está conformada por sus estudiantes, profesores y personal administrativo. Es así como surge la idea de crear una herramienta que en primer lugar, facilite la interacción entre los miembros de la escuela, desarrollo llevado a cabo en el año 2004 por el grupo de desarrollo de software Calumet el cual conformado y liderado por estudiantes de pregrado de Ing. de Sistemas da vida a la primera versión del portal web de la Escuela de Ingeniería de Sistemas EISIWeb.

Dado el crecimiento 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.

1.3 DEFINICIÓN DEL PROBLEMA

Los portales de las Escuelas de Geología e Ingeniería Eléctrica, Electrónica y de Telecomunicaciones actualmente cuentan con un sistema de información orientado a la Web el cual se encarga de la administración y control de las diferentes actividades que se realizan dentro de estas, así como el control de usuarios y servicios que se les proporciona a la comunidad.

Actualmente el portal de las escuelas cuenta con un servicio de sugerencias, a través del cual se da soporte a los usuarios que tienen inquietudes o desconocen el funcionamiento del sitio. Muchas de estas dudas son complejas y no se logra una completa solución de las mismas, razón por la cual en varias ocasiones es necesario dar soporte personalizado a los usuarios.

Las escuelas de la Universidad Industrial de Santander cuentan con programas de especialización, maestrías y doctorado, que son dirigidos y organizados por cada escuela. Por lo anterior, es que cada escuela debe elaborar un calendario académico con fechas, ciudad, salón y docente por cada cohorte de cada programa. Cada uno de estos calendarios debe ser aprobado por el Consejo Académico de la Universidad. Los programas a su vez son abiertos en diferentes ciudades, con lo cual es posible que existan cohortes del mismo programa en el mismo nivel en diferentes ciudades. La realización de estos calendarios es una labor bastante dispendiosa y complicada para los coordinadores ya que deben tener en cuenta que no se presenten cruces de horarios, profesores, salones, ciudades y otros. Este cronograma se elabora de manera manual a través de múltiples hojas de cálculo, proceso durante el cual ocurren cruces que no son fácilmente identificables, y que deben ser modificados durante la marcha, causando problemas tanto para los docentes que deben trasladarse a diferentes locaciones, como a los estudiantes que no pueden llevar un correcto desarrollo de su posgrado.

Parte del manejo de los posgrados y otras actividades organizadas por las escuelas, incluye la inscripción de posibles interesados a ingresar a ellos, actualmente no existe un servicio que los asista en esta labor, por esta razón los interesados deben diligenciar formularios físicos donde proveen sus datos más relevantes para así poder ser contactados más adelante.

1.4 OBJETIVOS

1.4.1 Objetivo General

Realizar las funciones de administración, mantenimiento, análisis, diseño y desarrollo de nuevos servicios para el portal web existente en las escuelas de Geología e Ingeniería Eléctrica, Electrónica y de Telecomunicaciones, para hacer óptimos, eficientes y sostenibles los trámites dentro de las escuelas y el acceso a la información de la misma.

1.4.2 Objetivos Específicos

1.4.2.1 Realizar labores de administración del portal web de la Escuela de Geología y de la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones teniendo en cuenta principalmente:

- Generar copias de seguridad diariamente de las bases de datos.
- Salvar la información de los portales web una vez por semana por medio de copias de respaldo y mantener un histórico de los mismos en caso de alguna falla.
- Hacer seguimiento del uso que hacen los usuarios a los servicios de Foros, Eventos Propuestos, Cartelera, Archivos y Mi perfil dentro de los portales GEOWEB, E3TWEB, para detectar usos indebidos o incorrectos por parte de éstos.
- Actualizar periódicamente las bases de datos con el objeto de mantener al día la información referente a matrículas, horarios, estados y categorías de los usuarios.

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

1.4.2.2 Llevar a cabo labores de mantenimiento a los portales GEOWEB, E3TWEB, en las que podemos encontrar:

- Implementar los nuevos servicios o mejoras realizadas por otros desarrolladores del grupo Calumet dentro de los portales GEOWEB, E3TWEB, ajustando el nuevo código, modificando las bases de datos y finalmente realizando las pruebas necesarias para verificar que el funcionamiento sea correcto.
- 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 del portal web.

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

- Crear el servicio de ayudas del portal web que permita, de manera sencilla a los usuarios hacer uso de los servicios, para de esta forma lograr una mayor y mejor interacción entre los miembros de la comunidad. Los usuarios podrán acceder a las ayudas bien sea siguiendo la estructura de menús existente, o bien, por el módulo de preguntas frecuentes. A los administradores, para crear las ayudas de cada servicio se les ofrecerá una interface cómoda para subir textos, PDFs, imágenes, enlaces a videos y otros.
- Crear un planeador de soporte para la generación de calendarios académicos. Este planeador permitirá realizar el calendario académico de las diferentes cohortes de los diferentes programas de posgrado teniendo en cuenta: el cruce de horarios, de programas, profesores, asignaturas, salones, número de horas, ciudad y fecha.
- Realizar una aplicación que soporte la pre-inscripción de los interesados en los diferentes programas, u otras actividades, que realicen las escuelas. De esta forma se podrán capturar automáticamente datos, y procesarlos de acuerdo a las necesidades de los programas.

1.4.2.4 Desempeñar labores de soporte a los usuarios de los portales GEOWEB, E3TWEB, 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 de los Portales GEOWEB, E3TWEB promoviendo así su utilización.
- Atender usuarios por olvido de la contraseña, creación de grupos, solicitudes de propuestas de eventos o de cartelera y creación de agendas con eventos y foros con sus respectivas conversaciones.
- Crear usuarios de forma manual, para personas no pertenecientes a la escuela y que por algún motivo necesitan registrarse en el sitio.
- Modificar los estados de los usuarios de acuerdo a la relación con la Escuela (Activo, Inactivo, Suspendido).

1.4.2.5 Capacitar a los estudiantes que recibirán las funciones de administración, mantenimiento, creación, y mejora de nuevos servicios dentro de los portales GEOWEB, E3TWEB 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 y base de datos.

- Llevar a cabo la familiarización con el entorno del portal web.

1.5 JUSTIFICACIÓN

Los servicios del portal de las escuelas deben mejorar constantemente y proporcionalmente a los cambios que se presenten en su entorno, a su vez deben dar correcta solución a los problemas y necesidades que surjan por parte de los usuarios, razón por la cual las labores de mantenimiento y actualización son indispensables.

Con el fin de mejorar el soporte que se ofrece a los usuarios se implementó un servicio que provee ayudas prácticas, da solución a preguntas frecuentes, y da capacitación en el funcionamiento de los servicios que se ofrecen. A través de este servicio de ayudas se ofrece una mejor atención a los usuarios y se mejora el conocimiento que tienen los estudiantes, profesores, y en general, la comunidad acerca del portal de sus respectivas escuelas.

El sistema de ayudas cubre una falencia presente en los sitios de las escuelas, ya que muchos de los servicios no son usados debido al desconocimiento de los usuarios. El usuario, ubicado en el sitio, tiene acceso constante al servicio de ayudas para que lo asista en el momento que lo requiera.

Además se diseñó y desarrolló un servicio que asiste la organización de cronogramas académicos de programas de posgrado, de tal forma, que constantemente exista información de cruces de horarios, cruces de aulas, y otras limitantes al momento de programar el calendario académico de cada cohorte de cada programa.

Finalmente se creó un nuevo servicio que permite a las escuelas, a través de su portal web, realizar la inscripción de los interesados en ingresar a sus programas y actividades. A través de este servicio el personal de las escuelas puede gestionar mejor la información que reciben y agilizar procesos internos, manteniendo una comunicación sencilla con los interesados. Además, se puede contar con información valiosa para ofrecer posibles nuevos programas a potenciales interesados.

1.6 ALCANCES Y LIMITACIONES

El servicio de ayudas brinda orientación a estudiantes, profesores y personal administrativo sobre el respectivo uso de los servicios del portal, además el módulo de preguntas frecuentes contiene ayudas orientadas a tipos de usuario y el módulo de sugerencias da la posibilidad a los usuarios de expresar al grupo ayudas que consideran necesarias.

La implementación del servicio de cronogramas académicos brinda asistencia a la hora de gestionar múltiples cronogramas, ya que previene cruces en los horarios de docentes en varias cohortes de varios programas académicos. Les permite a los estudiantes de programas de posgrado y docentes de los mismos tener un conocimiento pleno de las materias en las cuales están involucrados.

Además el servicio de inscripciones se presenta como una herramienta útil para la gestión de preinscripciones e inscripciones a programas académicos y actividades. Se facilita la agrupación de participantes, y se consolida como un canal de comunicación inmediato entre quien crea la inscripción y todos aquellos que se inscriben.

2 MARCO TEÓRICO

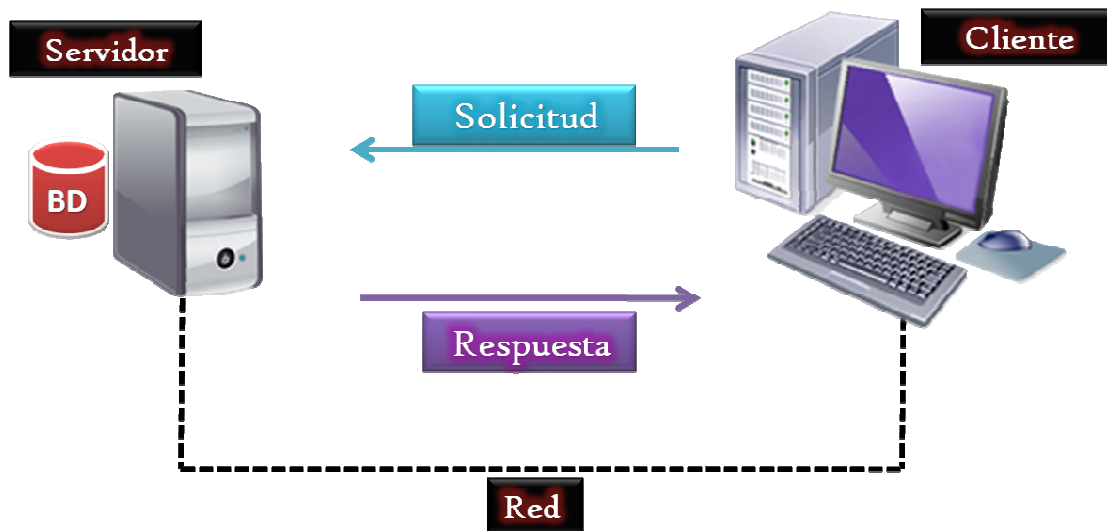
2.1 ARQUITECTURA CLIENTE/SERVIDOR

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

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

La idea es tratar un computador como un instrumento, que por sí solo pueda realizar muchas tareas, pero con la consideración de realizar aquellas que son más adecuadas a sus características. Si esto se aplica tanto a clientes como servidores, se entiende que la forma estándar de aplicación y uso de sistemas C/S, es mediante la explotación de los computadores a través de interfaces gráficas de usuario; mientras que la administración de datos y seguridad e integridad que deja a cargo de computadores centrales tipo mainframe. Usualmente la mayoría del trabajo pesado se hace en el proceso llamado servidor y los procesos cliente sólo se ocupan de la interacción con el usuario (aunque esto puede variar). La arquitectura C/S es una extensión de programación modular en la que la base fundamental es separar una gran pieza de software en módulos con el fin de hacer más fácil el desarrollo y mejorar su mantenimiento.

Figura 1. Modelo Cliente/Servidor



Fuente: Internet. <http://ulyla003.wordpress.com/page/4/>

2.1.1 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.

2.1.2 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:

2.1.2.1 Arquitectura Cliente/Servidor de dos capas

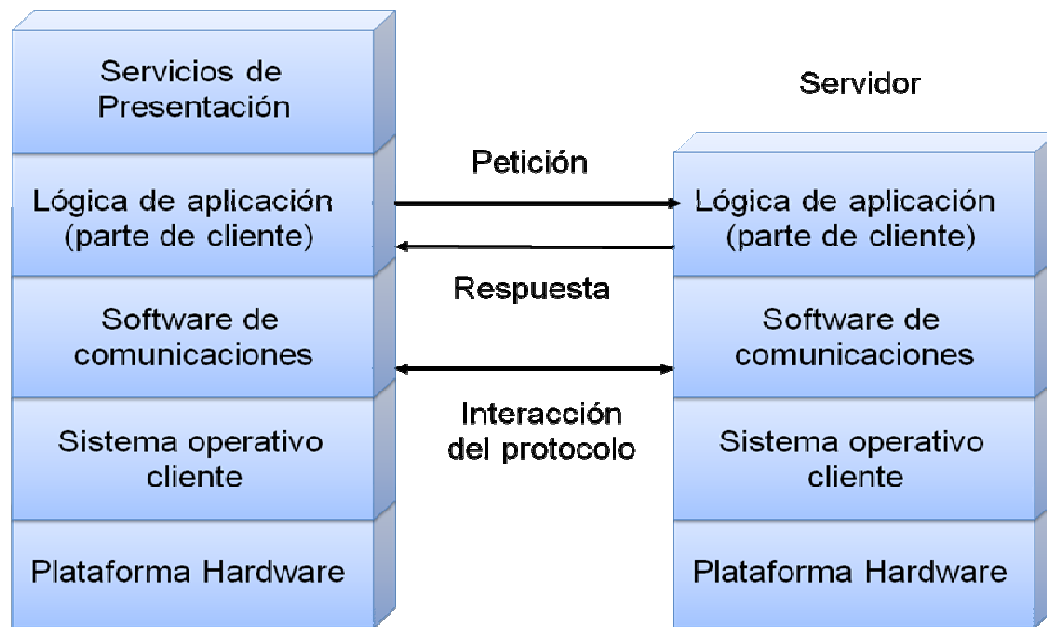
Consiste en una capa de presentación y lógica de la aplicación; y otra de la base de datos, cuando el cliente solicita recursos entonces el servidor responde directamente a la solicitud con sus propios recursos.

- 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 2. Arquitectura Cliente/Servidor de Dos Capas

Estación Trabajo Cliente



Fuente: Autores.

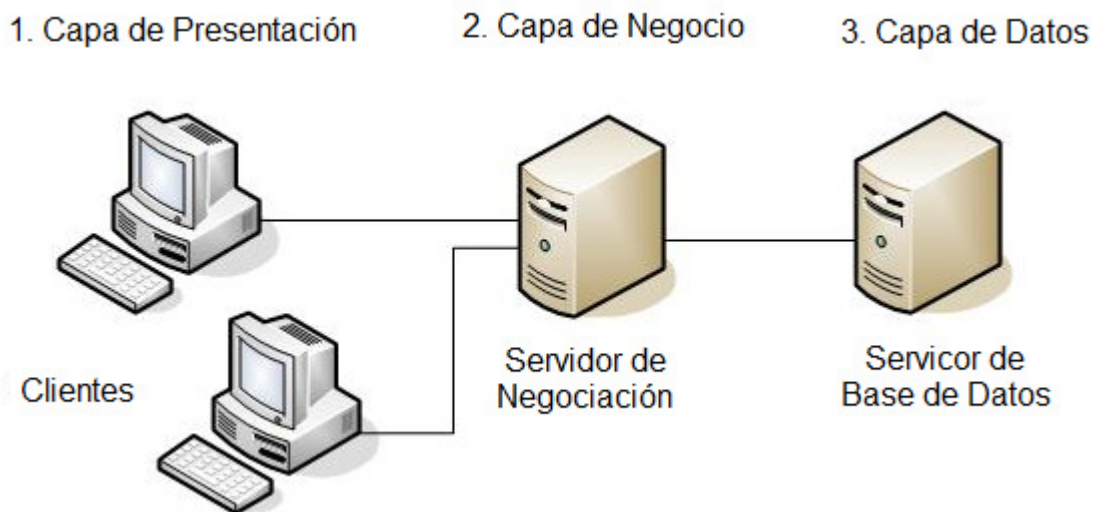
2.1.2.2 Arquitectura Cliente/Servidor de tres capas

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 3. Arquitectura Cliente/Servidor de Tres Capas



Fuente: Internet. http://www.ecured.cu/index.php/Archivo:Tres_capas.JPG

2.1.3 Arquitectura Cliente/Servidor aplicada

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

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

2.1.4 Ventajas del esquema Cliente/Servidor

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

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

2.1.5 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.

2.2 TECNOLOGÍAS DE DESARROLLO DE PÁGINAS WEB DINÁMICAS

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

Existen diferentes tecnologías para el desarrollo de páginas dinámicas entre ellas están:

2.2.1 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.

2.2.2 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.

2.2.3 Tecnología aplicada

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

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

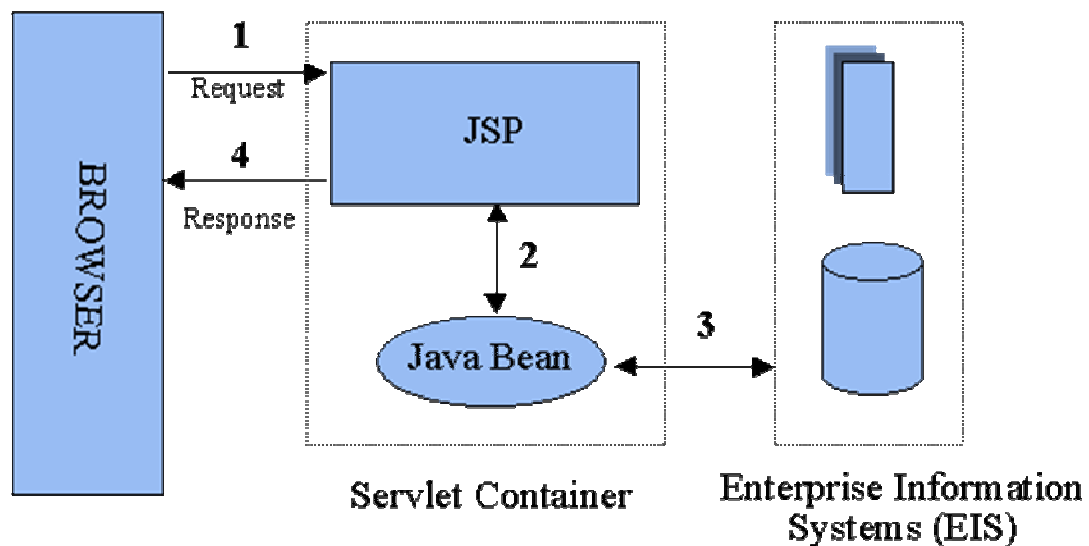
Una página JSP se compila a una aplicación Java la primera vez que se invoca, y de esta aplicación Java se crea una clase que empieza a ejecutarse en el servidor como un servlet. Un JSP es una página web con etiquetas especiales y código

Java incrustado, mientras que un servlet es un programa que recibe peticiones y genera a partir de ellas una página web.

2.2.3.1 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 a su vez 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 4. Modelo de acceso a JSP



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

La primera vez que un archivo JSP es invocado, este es compilado en un objeto, la respuesta del objeto es HTML estándar, el cual es interpretado por el navegador para ser presentado al usuario. Después de la compilación, el objeto de la página es almacenado en la memoria del servidor. En las peticiones posteriores a esta página, el servidor revisa si el archivo JSP ha cambiado. Si no ha cambiado, el servidor utiliza el objeto de la página compilada guardado en memoria para generar la respuesta al cliente, en caso contrario el servidor automáticamente compila el archivo de la página y reemplaza el objeto en la memoria.

2.3 BASES DE DATOS

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso, con una redundancia controlada y una estructura que refleja las interrelaciones y restricciones

existentes en el mundo real. En la base de datos se almacena información considerada necesaria para una determinada organización o negocio.

2.3.1 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:

2.3.1.1 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.

2.3.1.2 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.

2.3.1.3 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.

2.3.2 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.

2.3.3 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.

2.3.4 Manejadores o Gestores de Bases de Datos

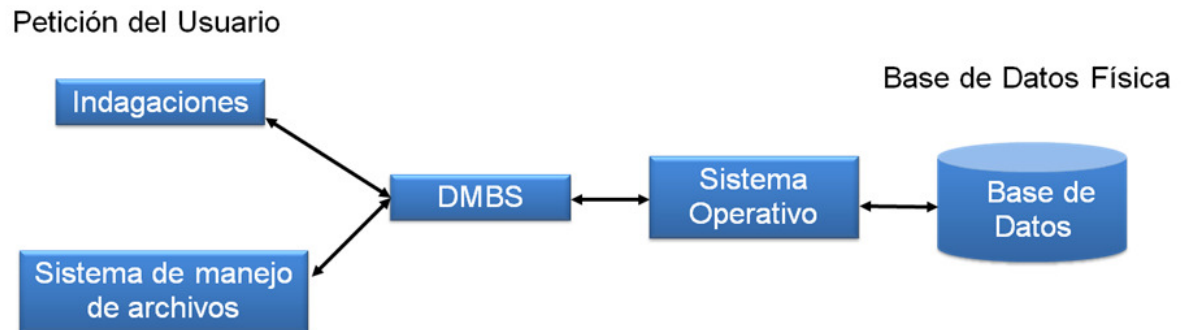
El sistema manejador de bases de datos es la porción más importante del software de un sistema de base de datos. Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica.

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 5. Función del DBMS.



Fuente: Autores.

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

2.3.5 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.

2.3.6 Ventajas de MySQL

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

2.4 NETBEANS

Es un entorno de desarrollo integrado (IDE), siendo una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas escritos en JAVA, pero puede servir para cualquier otro tipo lenguaje de programación. Netbeans es un producto libre y gratuito sin restricciones de uso.

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

2.5 SISTEMA DE CONTROL DE VERSIONES

- Un sistema de control de versiones es un software que administra el acceso a un conjunto de ficheros, y mantiene un historial de cambios realizados. El control de versiones es útil para guardar cualquier documento que cambie con frecuencia, o el código fuente de un programa.
- Normalmente consiste en una copia maestra en un repositorio central, y un programa cliente con el que cada usuario sincroniza su copia local. Además, el repositorio guarda registro de los cambios realizados por cada usuario, y permite volver a un estado anterior en caso de necesidad.

- Existen multitud de sistemas de control de versiones, pero sin duda, el más popular es CVS (Concurrent Versions System). CVS tuvo el mérito de ser el primer sistema usado por el movimiento de código abierto para que los programadores colaboran remotamente mediante el envío de parches. Es de uso gratuito, código abierto, y emplea fusión de cambios.
- Subversión se creó para igualar y mejorar la funcionalidad de CVS, preservando su filosofía de desarrollo.

2.5.1 Subversión

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

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

2.6 PROGRAMACIÓN UTILIZADA

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

características y funciones; abstrae algunas características de sistemas naturales complejos como son:

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

2.6.1 Clases

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

2.6.2 Objetos

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

2.6.3 Atributos

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

2.6.4 Métodos

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

2.6.5 Herencia

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

2.6.6 Beneficios de la Programación Orientada a Objetos

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

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

2.6.7 Java y JDK (Java Development Kit)

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

Permite escribir aplicaciones que puedan ejecutarse en casi cualquier plataforma. El lenguaje toma parte de la sintaxis de C y C++, pero tiene un modelo de objetos

más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria. Además, cuenta con una característica denominada “recolección de basura”, que examina la memoria y libera cualquier variable u objeto que no esté siendo usado. El JDK es un software que provee herramientas de desarrollo para la creación de programas en java.

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

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

2.7 SERVIDORES WEB

Es un tipo de software que se encuentra a la espera de una petición hecha por una aplicación cliente y da respuesta a dicha petición a través de una página web. Para cada transacción el servidor debe realizar dos acciones básicas: integrar todos los componentes de la página (texto, imágenes, vídeo, scripts, etc.) y enviarla rápidamente al usuario. A continuación se describe el servidor Web que se ajusta a la tecnología escogida para el proyecto.

2.7.1 Servidor Jakarta Tomcat

Servidor de aplicaciones Java basado en los estándares definidos por Sun Microsystems. Tomcat es desarrollado como parte del proyecto de código abierto Jakarta de la fundación de software Apache y es uno de los servidores de

aplicaciones Java más utilizados, en especial porque es liviano, cumple con todos los estándares, sencillo de instalar, tiene muy buena documentación y es gratuito, además por ser escrito en Java funciona en cualquier sistema operativo que disponga de la Máquina Virtual de Java (JVM).

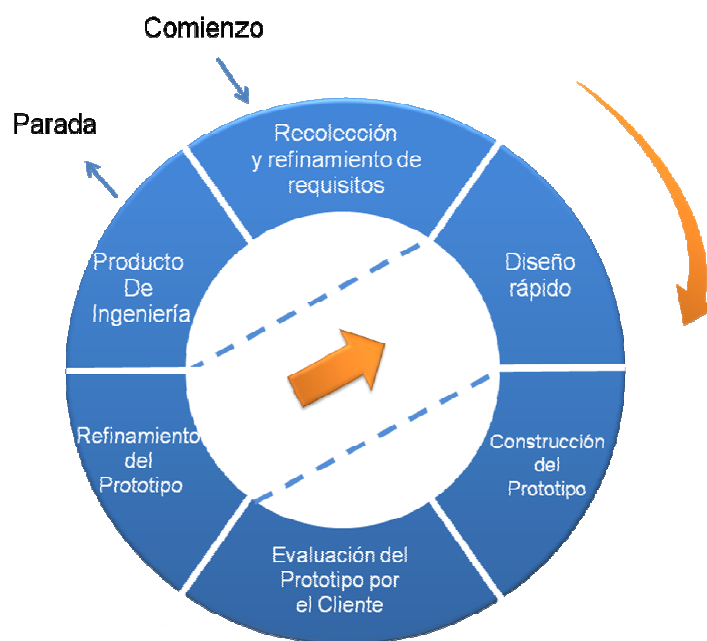
Es posible ejecutarlo desde la línea de comandos (consola o terminal), después de configurar algunas variables de entorno, sin embargo configurar cada variable de entorno y seguir los parámetros de la líneas de comando usados por Tomcat es tedioso y expuesto a errores, en su lugar se proporciona código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

3 MARCO METODOLÓGICO

3.1 PROTOTIPO EVOLUTIVO

Para realizar los nuevos servicios para el Portal de la Escuela de Geología y de la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones de la Universidad Industrial de Santander se propone como metodología de desarrollo el prototipo evolutivo.

Figura 6. Prototipo Evolutivo.



Fuente: Autores.

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

- Los portales GEOWeb y E3TWeb presentan constantemente necesidades por parte de los usuarios, por lo tanto estos portales requieren 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 GEOWeb y E3TWeb durante el desarrollo del proyecto.
- Durante la primera etapa de los nuevos servicios es necesario tener una clara interpretación de las especificaciones dadas por los usuarios y escuelas, pero muchas veces los usuarios no tienen una idea clara de lo que necesitan. Por esto la construcción de prototipos brinda la posibilidad de efectuar refinamientos de los requerimientos en forma sucesiva a fin de acercarse al producto deseado.
- Otra ventaja es tener la posibilidad de realizar cambios en etapas tempranas y crear varios prototipos evaluables durante el desarrollo, obteniéndose de este modo una metodología integral para el proceso de evaluación del programa.
- Esta metodología favorece la autocrítica al sistema, lo que conlleva a que se produzca muchas pruebas antes de dar un nuevo prototipo, así como mejoras rápidas a problemas que puedan surgir durante su uso.

Procedimiento a seguir para la metodología planteada:

- Se toma nota de los requisitos para la construcción de los prototipos.
- Se especifican en detalle los objetivos globales del software a realizar, mediante una reunión entre el desarrollador y el usuario, en la cual se identifican los requisitos básicos y se concluyen las áreas donde se necesita mayor definición.
- Posteriormente se presenta al usuario el diseño de un prototipo que se enfoca en los aspectos visuales del software, métodos de entrada y formatos de salida, para proceder a la construcción.
- El prototipo es evaluado por el usuario y se utiliza para filtrar los requisitos del software a desarrollar.
- Se produce un proceso interactivo en el que el prototipo es depurado para satisfacer necesidades del usuario, de igual forma el desarrollador obtiene una mejor comprensión de lo que hay que hacer para la entrega del producto final de ingeniería requerido por el usuario.

3.2 LENGUAJE DE MODELADO UNIFICADO

El Lenguaje de Modelado Unificado o Unified Modeling Language (UML), es el más utilizado en la actualidad. Es un lenguaje grafico 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.

UML no es un método de desarrollo porque no indica los pasos a seguir para llegar al código, es decir, no especifica como pasar del análisis al diseño y de este al código. Al no ser un método de desarrollo resulta ser independiente del ciclo de desarrollo que se siga, puede encajar en un ciclo en cascada, evolutivo, espiral o en métodos ágiles de desarrollo.

3.2.1 Diagramas de UML

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

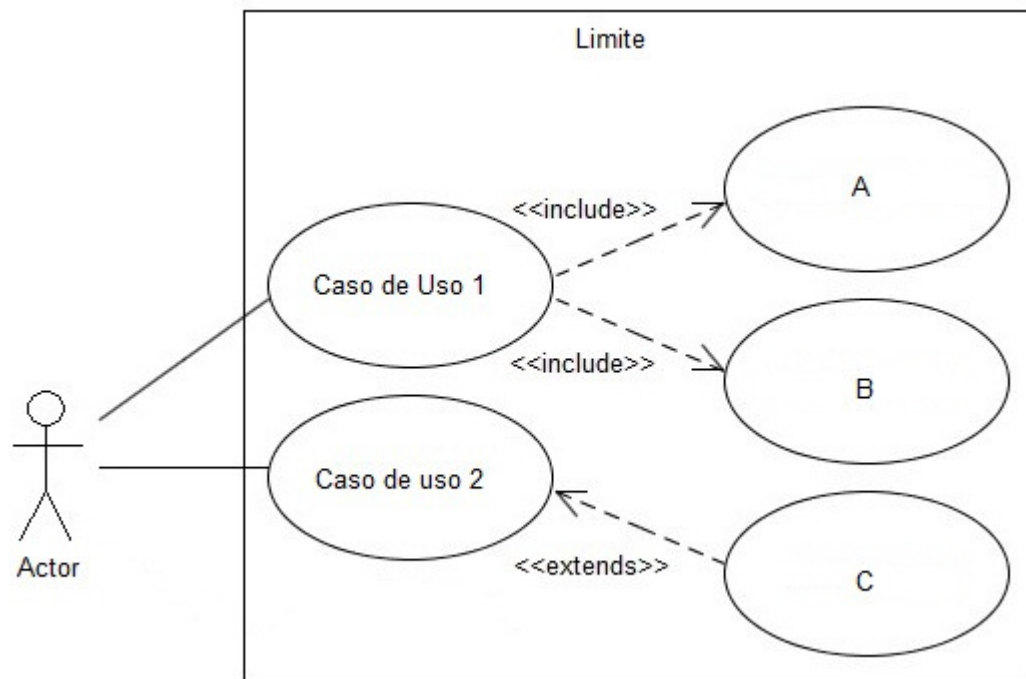
- UML tiene una notación gráfica muy expresiva que permite representar todas las fases de un proyecto informático: desde el análisis con casos de uso, el diseño con diagramas de clases, objetos, etc.
- UML facilita el entendimiento de la información, la función y el comportamiento de un sistema, haciendo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

3.2.2 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.

Figura 7. Diagramas de Casos de Uso



Fuente: Autores.

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 tres 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.

- Súper Administrador: Es el tipo de usuario que puede administrar, controlar y modificar los portales GEOWeb y E3TWeb, sus parámetros y sus usuarios.
- Usuario GEOWeb, E3TWeb: Es el tipo de usuario común de los portales GEOWeb y E3TWeb y a quien van dirigidos los servicios. Este usuario solo tiene control sobre sus privilegios.

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

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

3.2.3 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.

Objetos: Se ubican en la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen al diagrama. La línea que está debajo de

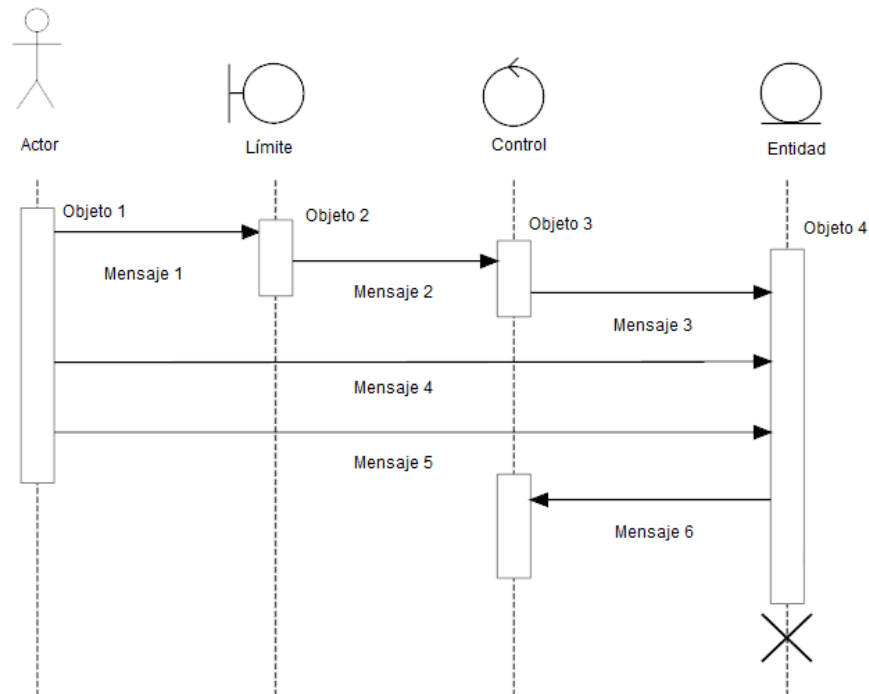
cada objeto será una línea discontinua conocida como la *línea de vida* de un objeto. Con la línea de vida se encuentra un pequeño rectángulo conocido como *activación*, el cual representa la ejecución de una operación que realiza el objeto.

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

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

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

Figura 8. Diagrama de Secuencias



Fuente: Autores.

3.3 ESTÁNDARES DE PROGRAMACIÓN

3.3.1 Modelo de datos

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

3.3.2 Nombres de las tablas

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

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

- *Tabla básica:* Aquella cuyos registros son necesarios para el correcto funcionamiento de la base de datos. Estas tablas no experimentan muchos cambios en los datos. El prefijo a anteponer a los nombres de estas tablas es “TB_”, es decir la tabla que almacena las ciudades donde hay programas académicos de posgrado se llamada “TB_Ciudades”.
- *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 “AyudasFrecuentes” es conocida como “TR_AyudasFrecuentes”.
- *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 EISIWeb. El prefijo a anteponer a los nombres de estas tablas es “TP_”, es decir la tabla “Usuarios”, es conocida como “TP_Usuarios”.

3.3.3 Clases

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

3.3.4 Páginas JSP

Los nombres de las paginas JSP que componen el portal GEOWeb y E3TWeb 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:

CrearAyuda.jsp, VerInscripciones.jsp.

3.3.5 Organización de Directorios

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

- El sitio cuenta con un directorio llamado “images”; en éste se encuentran almacenados todos los archivos .jpg, .gif, .png.
- Para el desarrollo de este proyecto se crearon los directorios “Ayudas”, “CronogramaAcademico” e “Inscripciones” los cuales almacenan los archivos jsp correspondientes a el funcionamiento de dichos servicios.

4 DESARROLLO DE LA HERRAMIENTA, ADMINISTRACIÓN Y MANTENIMIENTO

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

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

4.1 PROTOTIPO ESPERADO

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

Análisis de Requisitos.

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 el servicio de ayudas del portal web que permita, de manera sencilla a los usuarios hacer uso de los servicios, para de esta forma lograr una mayor y mejor interacción entre los miembros de la comunidad. Los usuarios podrán acceder a las ayudas bien sea siguiendo la estructura de menús existente, o bien, por el módulo de preguntas frecuentes. A los administradores, para crear las ayudas de cada servicio se les ofrecerá una interface cómoda para subir textos, PDFs, imágenes, enlaces a videos y otros.**

Objetivo Inicial:

- Implementar un servicio que provea ayudas prácticas, de solución a preguntas frecuentes, y ofrezca capacitación en el funcionamiento de los servicios que se ofrecen, haciendo uso de textos, imágenes y videos explicativos.

Requisitos finales del objetivo:

Los siguientes son los requisitos para el módulo de administrador de ayudas:

- Sólo tienen acceso a este servicio los usuarios con perfil de administradores.
- Permite crear ayudas de los servicios de los portales, en las cuales se debe poder utilizar contenido multimedia como imágenes, videos, animaciones, etc.
- Al crear una ayuda, se puede escoger un título, una descripción, temas de primer y segundo nivel, y seleccionar los tipos de usuarios a los cuales está enfocada la ayuda.
- Permite crear preguntas frecuentes con título y descripción, las cuales se pueden relacionar con un conjunto de ayudas.

- Permite crear menús personalizados de los temas de primer y segundo nivel para el portal de ayudas.
- Permite consultar sugerencias realizadas por usuarios en el portal de ayudas.

Los siguientes son los requisitos para el portal de ayudas público:

- Permite buscar ayudas, utilizando temas de primer y segundo nivel, que están relacionados con los servicios en los portales.
- Permite buscar preguntas frecuentes, las cuales se despliegan dependiendo del tipo de usuario, y en las cuales se contiene un grupo de ayudas que solucionan la pregunta frecuente.
- Permite enviar sugerencias relacionadas con las ayudas, como una manera de involucrar a los usuarios de los portales en la creación y mejora de las ayudas.

2. Crear un planeador de soporte para la generación de calendarios académicos. Este planeador permitirá realizar el calendario académico de las diferentes cohortes de los diferentes programas de posgrado teniendo en cuenta: el cruce de horarios, de programas, profesores, asignaturas, salones, número de horas, ciudad y fecha.

Objetivo inicial:

- Crear un servicio que asista la organización de cronogramas académicos de programas académicos, de tal forma, que constantemente exista información de cruces de horarios, cruces de aulas, y otras limitantes al momento de programar el calendario académico de cada cohorte de cada programa.

Requisitos finales del objetivo:

- Este servicio sólo puede ser utilizado para aquellos usuarios que tengan permiso.
- Permite la creación de cronogramas académicos, especificando el programa académico, la ciudad y la cohorte, además de señalar el rango de fechas en el cual será válido éste cronograma académico.
- Permite programar materias, en las cuales se señala el docente que dictará la materia y la fecha, horas y salón donde tomará lugar.
- El sistema señala todas las fechas en las cuales ocurren cruces, se distingue entre cruces de docentes y cruces de salones.
- Permite visualizar el cronograma académico en forma de calendario, señalando con diferente color las fechas en las cuales se dictan materias.
- Permite visualizar un consolidado de todos los cronogramas académicos vigentes en forma de calendario.

3. Realizar una aplicación que soporte la pre-inscripción de los interesados en los diferentes programas, u otras actividades, que realicen las escuelas. De esta forma se podrán capturar automáticamente datos, y procesarlos de acuerdo a las necesidades de los programas.

Objetivo inicial:

- Implementar un servicio que soporte la creación de pre-inscripciones en línea a programas académicos o actividades, facilitando la gestión de la información de los interesados.

Requisitos finales del objetivo:

Los siguientes son los requisitos del módulo administrador de inscripciones:

- Permite crear inscripciones, en las cuales se señala un título y descripción, el rango de fechas de validez de la inscripción, y se agregan los campos personalizados que deben diligenciar los interesados, además se puede señalar si la inscripción es pública (usuarios externos) o privada (usuarios del portal).
- Permite agregar campos personalizados, ofreciendo diferentes tipos de campos tales como texto, párrafo, selección múltiple y salto de sección.
- Permite modificar el orden de los campos, así como editarlos y eliminarlos.
- Permite editar, duplicar y eliminar inscripciones.
- Identifica y agrupa a los usuarios dependiendo de su estado, el cual puede ser “Interesado”, “Pre-inscrito” o “Inscrito”.
- Permite el envío de correos.
- Permite ver la información diligenciada por los usuarios, así como actualizarla en caso de ser necesario.
- Permite cambiar el estado de los interesados.

Los siguientes son los requisitos del formulario de inscripción:

- Este servicio puede ser accedido por usuarios externos si se especifica la inscripción como pública, y sólo por usuarios del portal si se especifica la inscripción como privada.
- Se carga la información diligenciada por el usuario si ya se ha inscrito previamente, dándole la posibilidad de editar su inscripción.
- Si la persona sólo diligencia sus datos personales ingresa a la inscripción como “interesado” y si diligencia todo el formulario ingresa a la inscripción como “pre-inscrito”.

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

Administración.

Objetivos iniciales:

- Efectuar labores de administración de los portales web de la Escuela de Estudios Industriales y Empresariales, de la Escuela de Ingeniería de Petróleos y la Escuela de Diseño Industrial; teniendo en cuenta entre otras:
 - Generar Backups (copias de respaldo) diariamente de la Base de Datos.
 - Salvar la información de los sitios web semanalmente, por medio de copias de respaldo y mantener un histórico de los mismos en caso de alguna falla.
 - 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 EEIEWeb. EIPWeb y EDIWeb 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 EEIEWeb, EIPWeb y EDIWeb, en los que podemos encontrar:
 - Implementar los nuevos servicios o mejoras realizadas por otros desarrolladores del grupo CALUMET, dentro de los portales EEIEWeb,

EIPWeb y EDIWeb, ajustando el nuevo código, modificando la Base de Datos y realizando las pruebas necesarias.

- 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.
- Realizar el mantenimiento de un archivo clasificado de las solicitudes de cambios y correcciones que hagan los usuarios.

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 la base de datos Diamante y de los directorios de los sitios EEIWeb, EIPWeb y EDIWeb.
- 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 EEIWeb, EIPWeb y EDIWeb, brindando así solución a sus diferentes necesidades,

conflictos o situaciones que se puedan presentar. Entre estas labores se destacan:

- Capacitar usuarios y estudiantes del primer nivel en el uso de servicios dentro del portal EEIEWeb, EIPWeb y EDIWeb, 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 EEIEWeb, EIPWeb o EDIWeb 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 EEIEWeb, EIPWeb y EDIWeb, 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 EEIEWeb, EIPWeb y EDIWeb en cuanto a:
 - Implantación del sitio local para la creación de nuevos servicios y realización de pruebas.
 - Realizar inducción en cuanto al manejo y utilización de los JSP, JavaBeans y Base de Datos.
 - Llevar a cabo la familiarización con el entorno de los portales web.

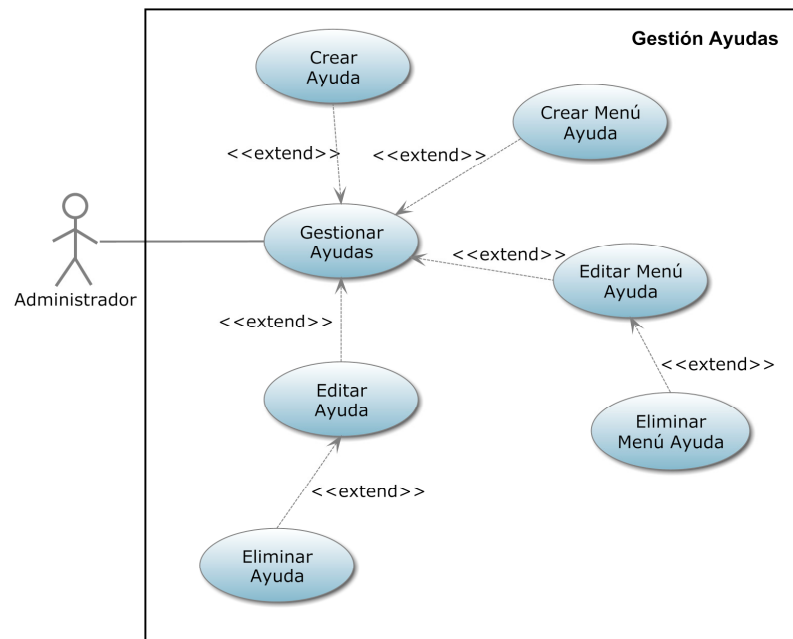
Requisitos finales del objetivo:

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

4.1.1 Diagramas de Casos de Uso

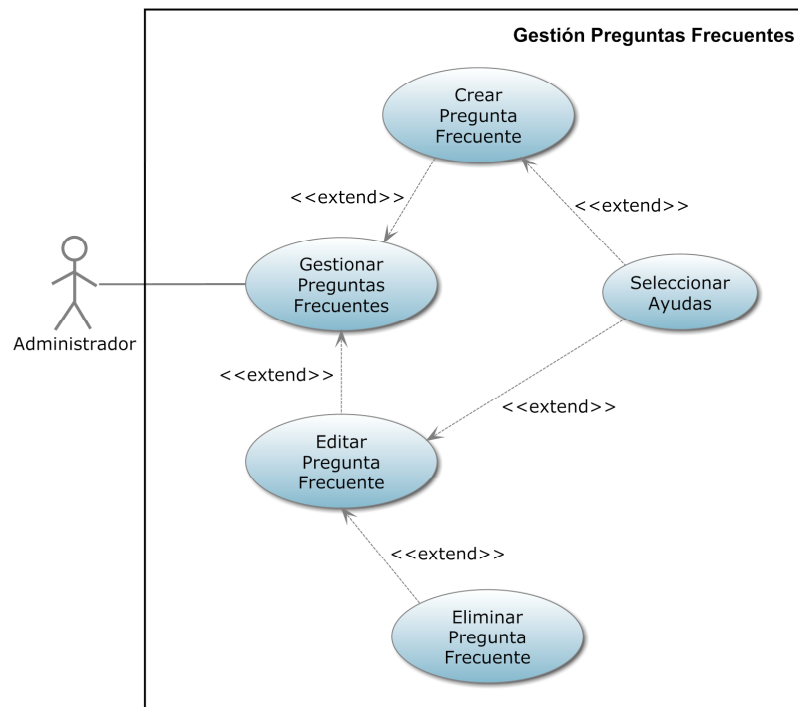
4.1.1.1 Servicio de Ayudas

Figura 9. Diagrama de Casos de Uso: Gestión Ayudas.



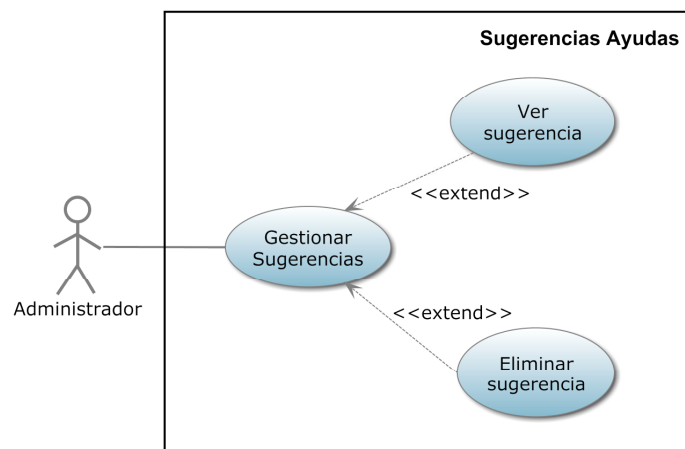
Fuente: Autores.

Figura 10. Diagrama de Casos de Uso: Gestión Preguntas Frecuentes.



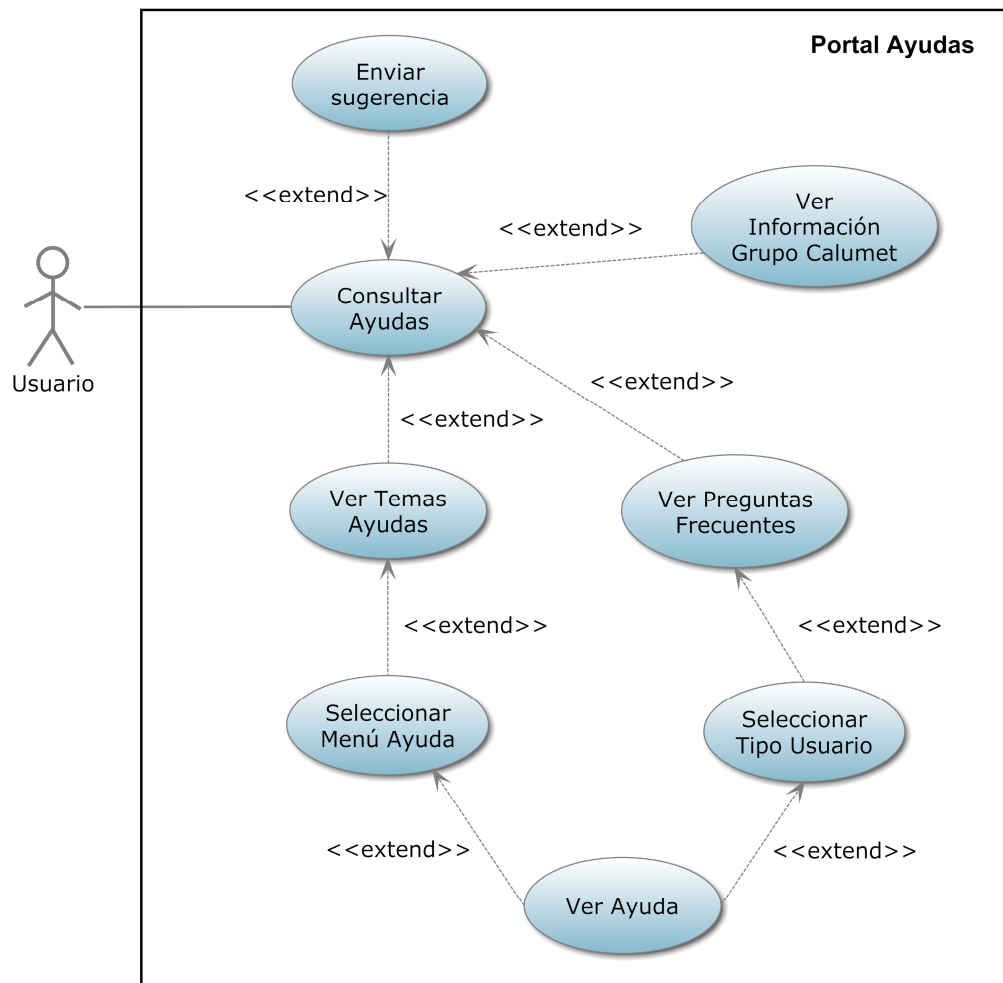
Fuente: Autores.

Figura 11. Diagrama de Casos de Uso: Sugerencias Ayudas.



Fuente: Autores.

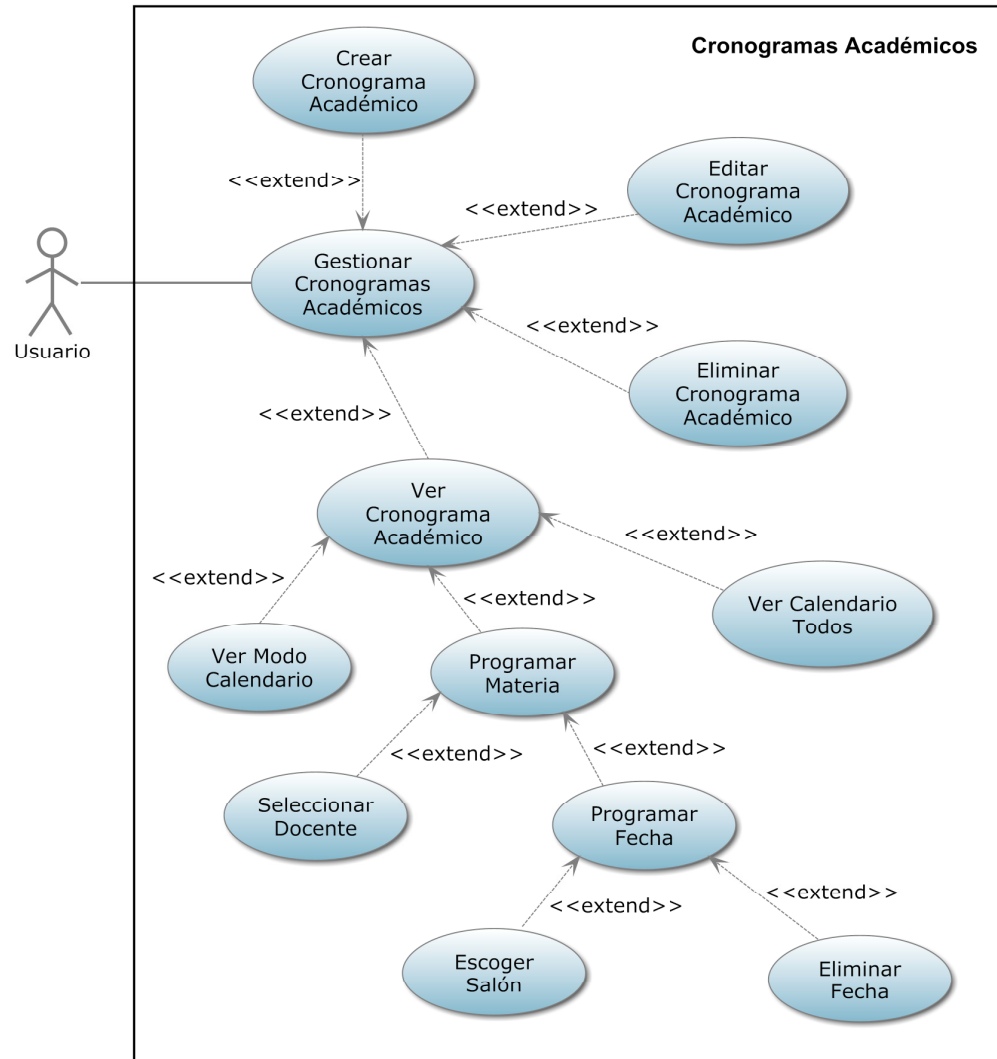
Figura 12. Diagrama de Casos de Uso: Portal Ayudas.



Fuente: Autores.

4.1.1.2 Servicio de Cronogramas Académicos

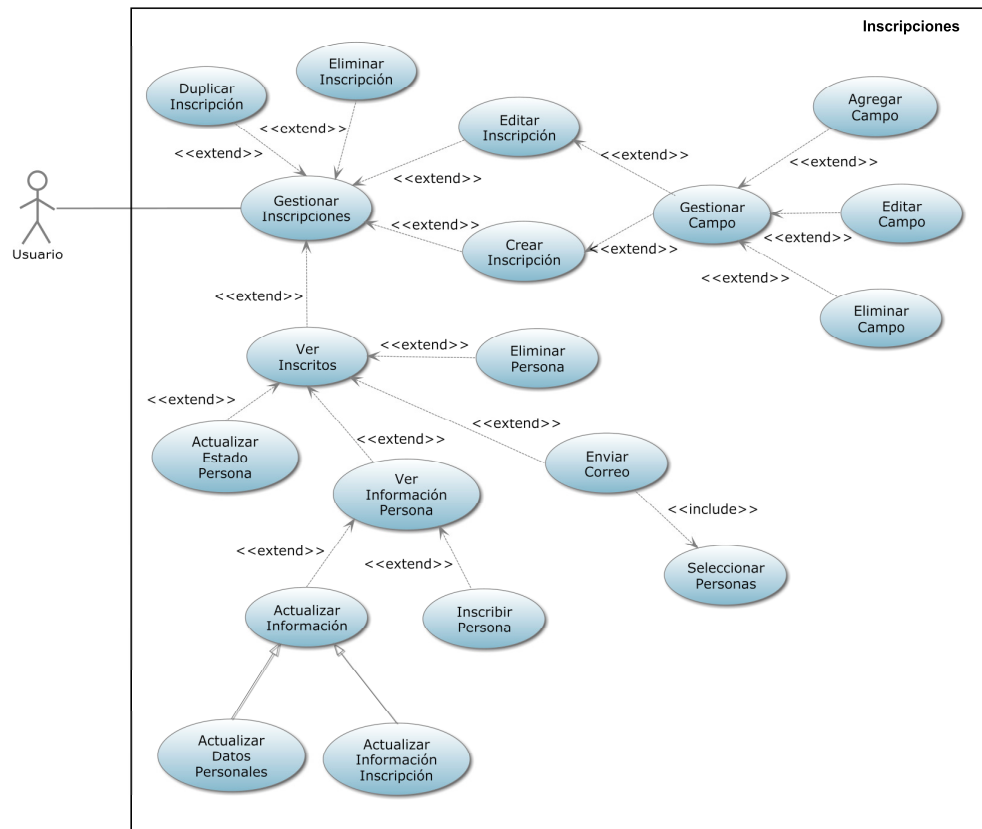
Figura 13. Diagrama de Casos de Uso: Cronogramas Académicos.



Fuente: Autores.

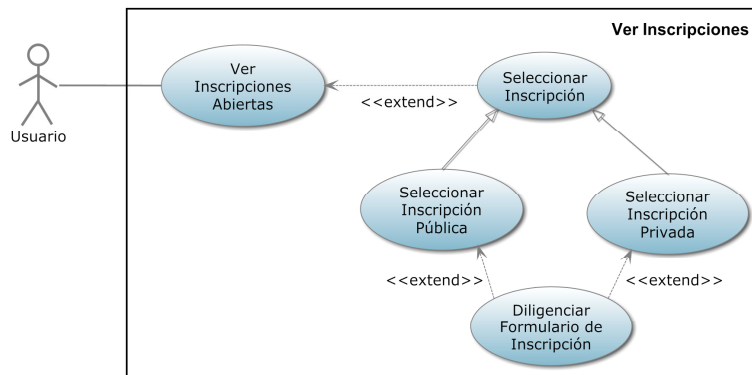
4.1.1.3 Servicio de Inscripciones

Figura 14. Diagrama de Casos de Uso: Inscripciones.



Fuente: Autores.

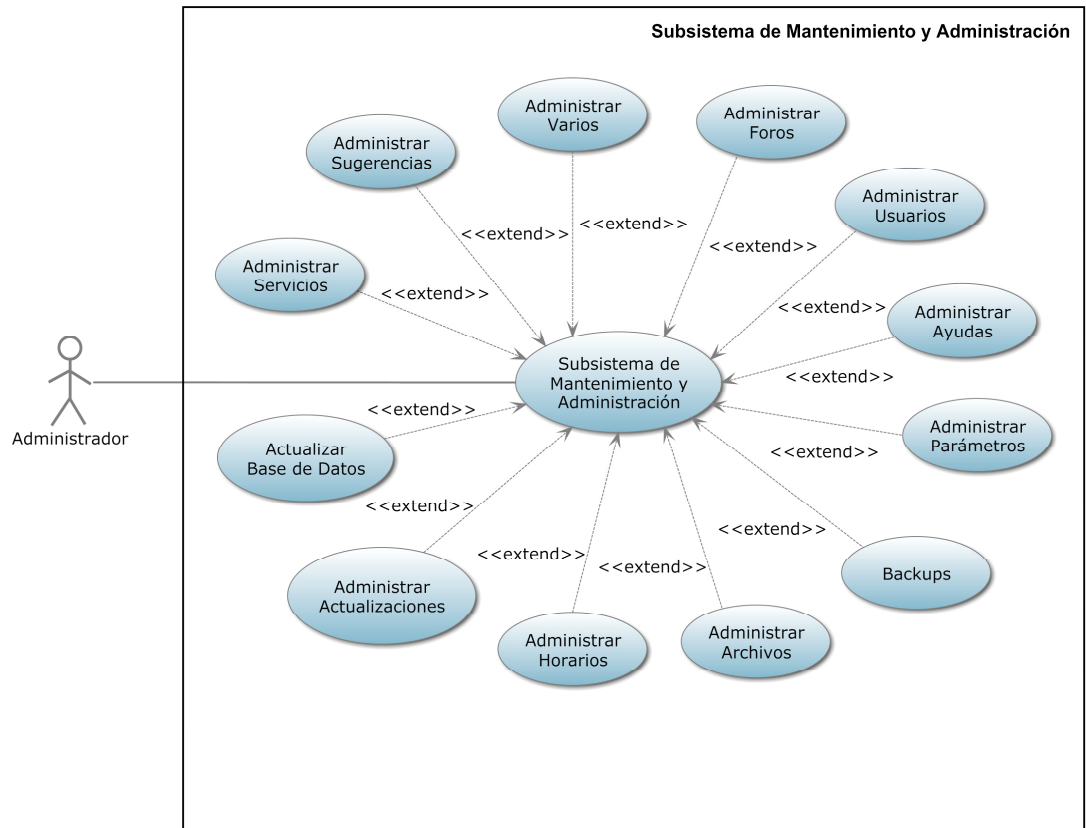
Figura 15. Diagrama de Casos de Uso: Ver Inscripciones.



Fuente: Autores.

4.1.1.4 Mantenimiento y Administración de los portales GEOWEB y E3TWEB

Figura 16. Diagrama de Casos de Uso: Subsistema de Mantenimiento y Administración.



Fuente: Autores.

4.1.2 Documentación de Casos de Uso del Sistema

4.1.2.1 Servicio de Ayudas

Tabla 1. Casos de uso: Gestión ayudas

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Ayudas	Acceder al servicio que lista los menús de ayudas y las ayudas creadas en el portal de ayudas de las escuelas.
Crear Ayuda	Crear una nueva ayuda para el portal de ayudas.
Editar Ayuda	Editar una ayuda creada anteriormente.
Eliminar Ayuda	Eliminar una ayuda que ha sido creada anteriormente.
Crear Menú Ayuda	Crear un menú de ayuda de primer o segundo nivel para el portal de ayudas.
Editar Menú Ayuda	Editar un menú de ayuda de primer o segundo nivel.
Eliminar Menú Ayuda	Eliminar un menú de ayuda que ha sido creado anteriormente.

Fuente: Autores.

Tabla 2. Casos de uso: Gestión Preguntas Frecuentes

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Preguntas Frecuentes	Acceder al servicio que lista las preguntas frecuentes.

Crear Pregunta Frecuente	Crear una nueva pregunta frecuente para el portal de ayudas.
Editar Pregunta Frecuente	Editar una pregunta frecuente que ha sido creada anteriormente.
Eliminar Pregunta Frecuente	Eliminar una pregunta frecuente que ha sido creada anteriormente.
Seleccionar Ayudas	Seleccionar las ayudas que están relacionadas con la pregunta frecuente que se está creando o editando.

Fuente: Autores.

Tabla 3. Casos de uso: Sugerencias Ayudas

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Sugerencias	Acceder al servicio que lista las sugerencias recibidas por usuarios en el portal de ayudas.
Ver Sugerencia	Ver sugerencia de usuario recibida en el portal de ayudas.
Eliminar Sugerencia	Eliminar sugerencia de usuario recibida en el portal de ayudas.

Fuente: Autores.

Tabla 4. Casos de Uso: Portal Ayudas

TÍTULO	FUNCIONES PRIMARIAS
Consultar Ayudas	Acceder al portal de ayudas de las escuelas.

Ver Temas Ayudas	Listar las ayudas organizadas por temas de servicios del portal de primer y segundo nivel.
Seleccionar Menú Ayuda	Seleccionar menú de ayuda de primer o segundo nivel.
Ver Preguntas Frecuentes	Listar las ayudas organizadas por preguntas frecuentes y por tipos de usuario.
Seleccionar Tipo Usuario	Seleccionar tipo de usuario para ver las preguntas frecuentes y las ayudas correspondientes a esa pregunta.
Ver Ayuda	Desplegar ayuda con su respectivo contenido.
Enviar Sugerencia	Enviar sugerencia relacionada con el portal de ayudas.
Ver Información Grupo Calumet	Desplegar información relacionada con el grupo de desarrollo Calumet.

Fuente: Autores.

4.1.2.2 Servicio de Cronogramas Académicos

Tabla 5. Caso de Uso: Cronogramas Académicos

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Cronogramas Académicos	Acceder al servicio que lista los cronogramas académicos que se hayan creado.
Crear Cronograma Académico	Crear nuevo cronograma académico para un determinado programa académico.
Editar Cronograma Académico	Editar la información básica de un cronograma académico que haya sido creado anteriormente.
Eliminar Cronograma Académico	Eliminar un cronograma académico que haya sido creado anteriormente.
Ver Cronograma Académico	Ver un cronograma académico de un programa académico con las materias para ser programadas.
Programar Materia	Acceder al servicio que permite programar una materia de un cronograma académico con docente, fechas y salones.
Seleccionar Docente	Seleccionar el docente que dictará una materia de un cronograma académico.

Programar Fecha	Seleccionar las fechas en que se dicta una materia.
Escoger Salón	Seleccionar el salón para cada fecha en que se dicta una materia.
Eliminar Fecha	Eliminar una fecha de la programación de una materia.
Ver Modo Calendario	Ver un cronograma académico en forma de calendario.
Ver Calendario Todos	Ver un consolidado de todos los cronogramas académicos vigentes en forma de calendario.

Fuente: Autores.

4.1.2.3 Servicio de Inscripciones

Tabla 6. Casos de uso: Inscripciones

TÍTULO	FUNCIONES PRIMARIAS
Gestionar Inscripciones	Acceder al servicio que lista las inscripciones que se hayan creado.
Crear Inscripción	Crear una nueva inscripción de un programa académico o de una actividad.

Editar Inscripción	Editar una inscripción de programa o actividad que haya sido creada anteriormente.
Eliminar Inscripción	Eliminar una inscripción de programa o actividad creada anteriormente.
Gestionar Campo	Opciones que tiene el usuario respecto a los campos que haya creado o que desea crear.
Agregar Campo	Agregar un nuevo campo en el formulario, el cual puede ser texto, párrafo o selección múltiple.
Editar Campo	Editar la información de un campo que haya sido agregado en el formulario
Eliminar Campo	Eliminar un campo que haya sido agregado en el formulario.
Ver Inscritos	Ver las personas inscritas en la inscripción, aparecen agrupadas en estados “Interesado”, “Pre-Inscrito” e “Inscrito”.
Actualizar Estado	Actualizar el estado de un interesado en la inscripción.
Eliminar Persona	Eliminar interesado del proceso de inscripción.
Ver Información Persona	Ver la información diligenciada por el interesado en el proceso de inscripción.

Actualizar Información	Opciones que tiene el creador de la inscripción para actualizar la información de los interesados.
Actualizar Datos Personales	Actualizar únicamente datos personales.
Actualizar Información Inscripción	Actualizar información diligenciada por interesados en el proceso de inscripción
Inscribir Persona	Cambiar estado de cualquier interesado directamente a “Inscrito”.

Fuente: Autores.

Tabla 7. Casos de uso: Ver Inscripciones

TÍTULO	FUNCIONES PRIMARIAS
Ver Inscripciones Abiertas	Acceder al servicio que lista las inscripciones que se encuentren vigentes, las cuales pueden ser privadas (usuarios del portal) o públicas (usuarios externos).
Seleccionar Inscripción	Opciones que tiene un usuario que desea inscribirse.
Seleccionar Inscripción Pública	La inscripción puede ser pública, por lo cual no requerirá que el usuario esté registrado en el portal en el cual se lleva a cabo la inscripción.
Seleccionar Inscripción Privada	La inscripción puede ser privada, por lo cual se requiere que el usuario se encuentre registrado y logueado en el portal en el cual

	se lleva a cabo la inscripción.
Diligenciar Formulario de Inscripción	El interesado diligencia el formulario de inscripción.

Fuente: Autores.

4.1.2.4 Mantenimiento y Administración de los portales GEOWeb y E3TWEB

Tabla 8. Casos de uso: Mantenimiento y Administración de los portales GEOWEB y E3TWeb

TÍTULO	FUNCIONES PRIMARIAS
Administrar Servicios	Crear, modificar y eliminar servicios del portal. Establecer parámetros.
Administrar Sugerencias	Revisar todas las sugerencias para atender o eliminar.
Administrar Varios	Ejecutar labores de mantenimiento que no clasifican en otras categorías. Llevar el contador de visitas a cero. Insertar, cambiar y eliminar banner.
Administrar Foros	Ver información y eliminar conversaciones de los usuarios en foros.

Administrar Usuarios	<p>Crear usuarios manualmente.</p> <p>Ver información de usuarios.</p> <p>Cambiar parámetros.</p> <p>Cambiar clave.</p> <p>Autorizar servicios a categorías y perfiles.</p> <p>Crear, modificar, eliminar categorías y perfiles.</p> <p>Enviar Correos.</p>
Administrar Ayudas	<p>Crear, modificar y eliminar las ayudas en el portal de ayudas de las escuelas.</p>
Administrar Parámetros	<p>Crear, modificar y eliminar los diferentes parámetros del portal.</p>
Backups	<p>Generar copias de la base de datos División y Diamante.</p>
Administrar Archivos	<p>Ver información, descargar y eliminar archivos cargados por los usuarios.</p>
Administrar Horarios	<p>Cambiar, restaurar, ver incoherencias y cambios de horarios. Crear grupo de materia.</p>
Administrar Actualizaciones	<p>Ver historial y estadísticas de cambios del portal web.</p>
Actualizar Base de Datos	<p>Actualizar tablas de datos de División y Diamante.</p> <p>Ver contadores de registros.</p>

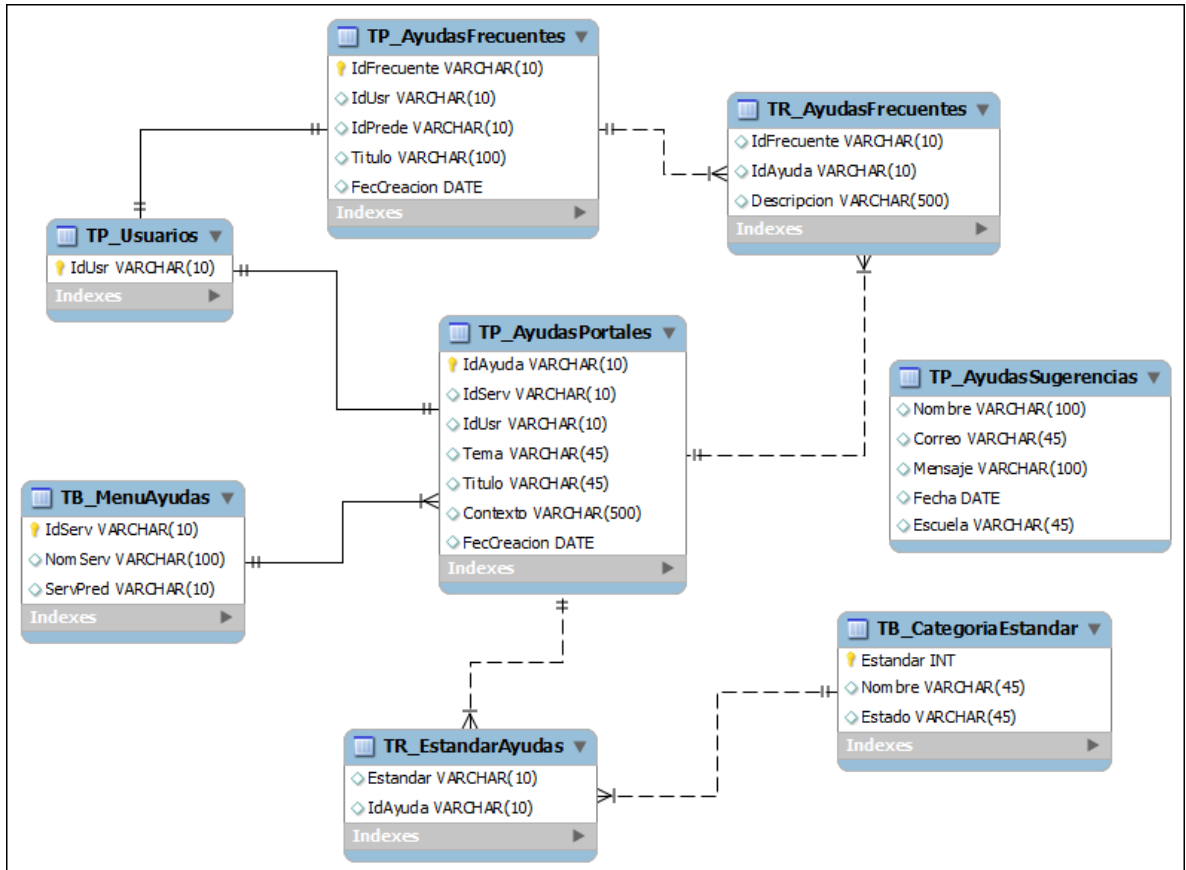
Fuente: Autores.

4.1.3 Diseño y Análisis

4.1.3.1 Diagrama Entidad/Relación de los servicios desarrollados

- Servicio de Ayudas:

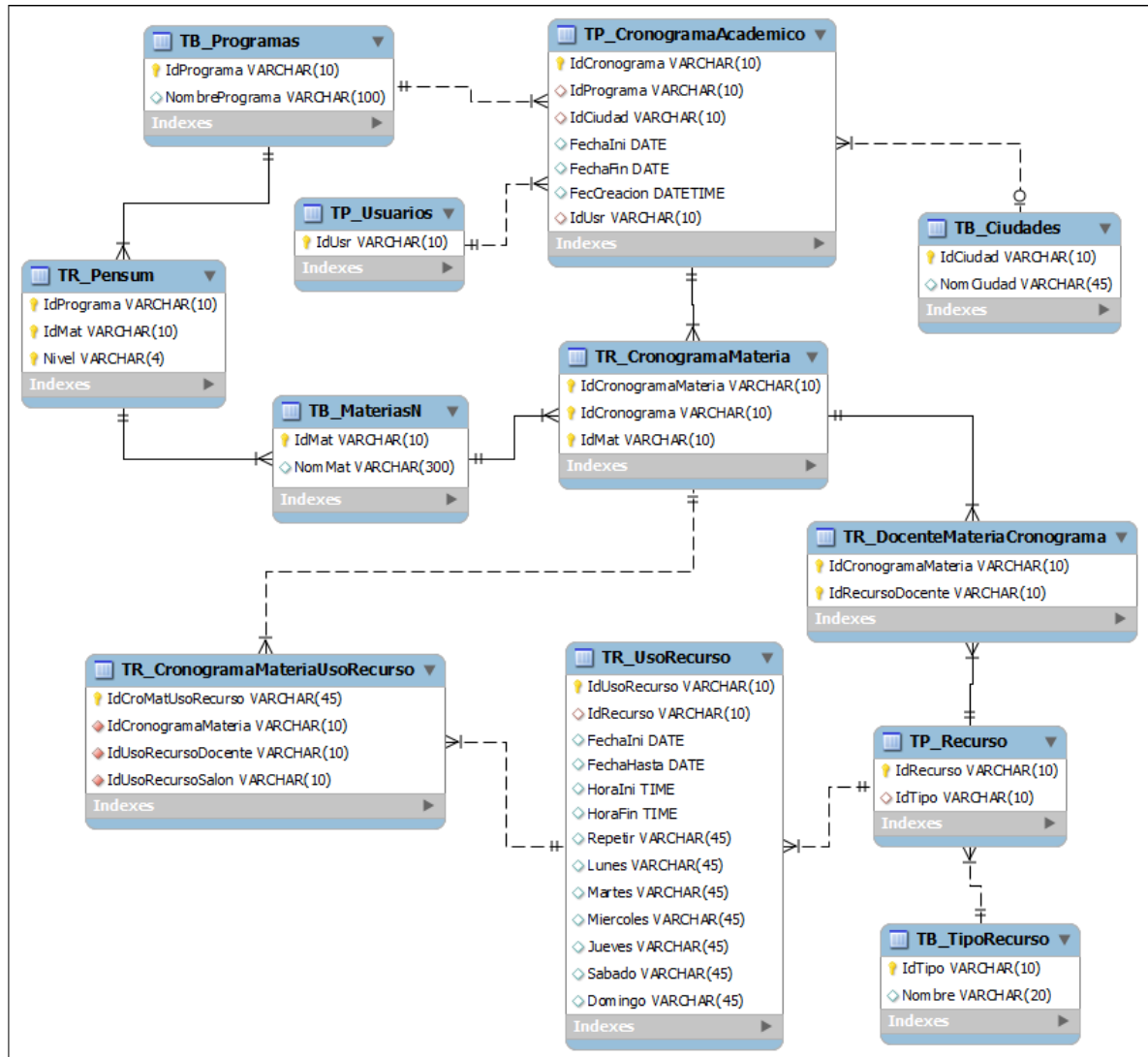
Figura 17. Diagrama E/R: Ayudas



Fuente: Autores.

- Servicio de Cronogramas Académicos:

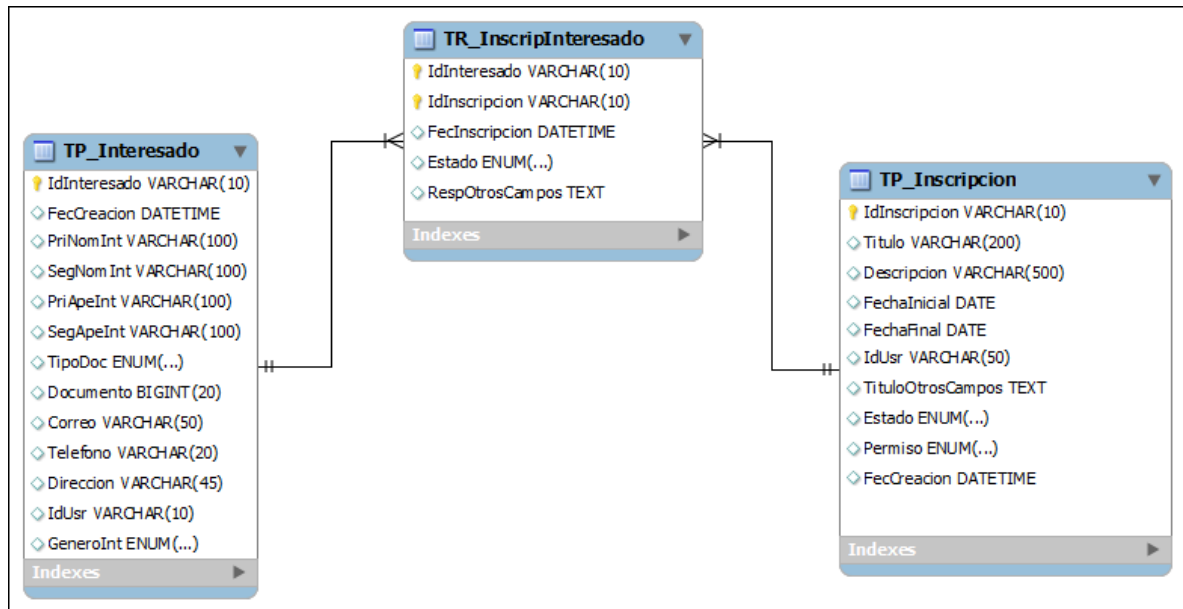
Figura 18. Diagrama E/R: Cronogramas Académicos



Fuente: Autores.

- Servicio de Inscripciones.

Figura 19. Diagrama E/R: Inscripciones



Fuente: Autores.

4.1.3.2 Descripción de las Entidades

Definición de las entidades utilizadas y creadas en la base de datos Poseidón ubicada en el servidor de la escuela de Ingeniería Civil y de la base de datos Diamante ubicada en el servidor de los portales de las escuelas GEOWEB y E3TWeb.

Tabla 9. Descripción de las Entidades

ENTIDAD	DESCRIPCIÓN
TP_Usuarios	Contiene la información de todos los usuarios registrados en el portal.

TP_AyudasPortales	Contiene las ayudas creadas para el portal de ayudas.
TP_AyudasFrecuentes	Contiene las preguntas frecuentes para el portal de ayudas.
TR_AyudasFrecuentes	Relaciona una pregunta frecuente con una o más ayudas para el portal de ayudas.
TR_EstandarAyudas	Relaciona una ayuda con una o más categorías estándar.
TB_CategoriaEstandar	Contiene las categorías estándar de los tipos de usuarios en el portal.
TP_AyudasSugerencias	Contiene las sugerencias recibidas en el portal de ayudas.
TB_MenuAyudas	Contiene los menús de primer y segundo nivel del portal de ayudas.
TP_CronogramaAcademico	Contiene la información básica de los cronogramas académicos de programas académicos y su respectiva cohorte.
TR_CronogramaMateria	Relaciona un cronograma académico con cada una de las materias del programa académico del mismo.
TR_DocenteMateriaCronograma	Contiene los docentes que dictan cada materia de un cronograma académico.
TR_CronogramaMateriaUsoRecurso	Relaciona la materia de un cronograma académico con el tiempo que está ocupado un docente o un salón.

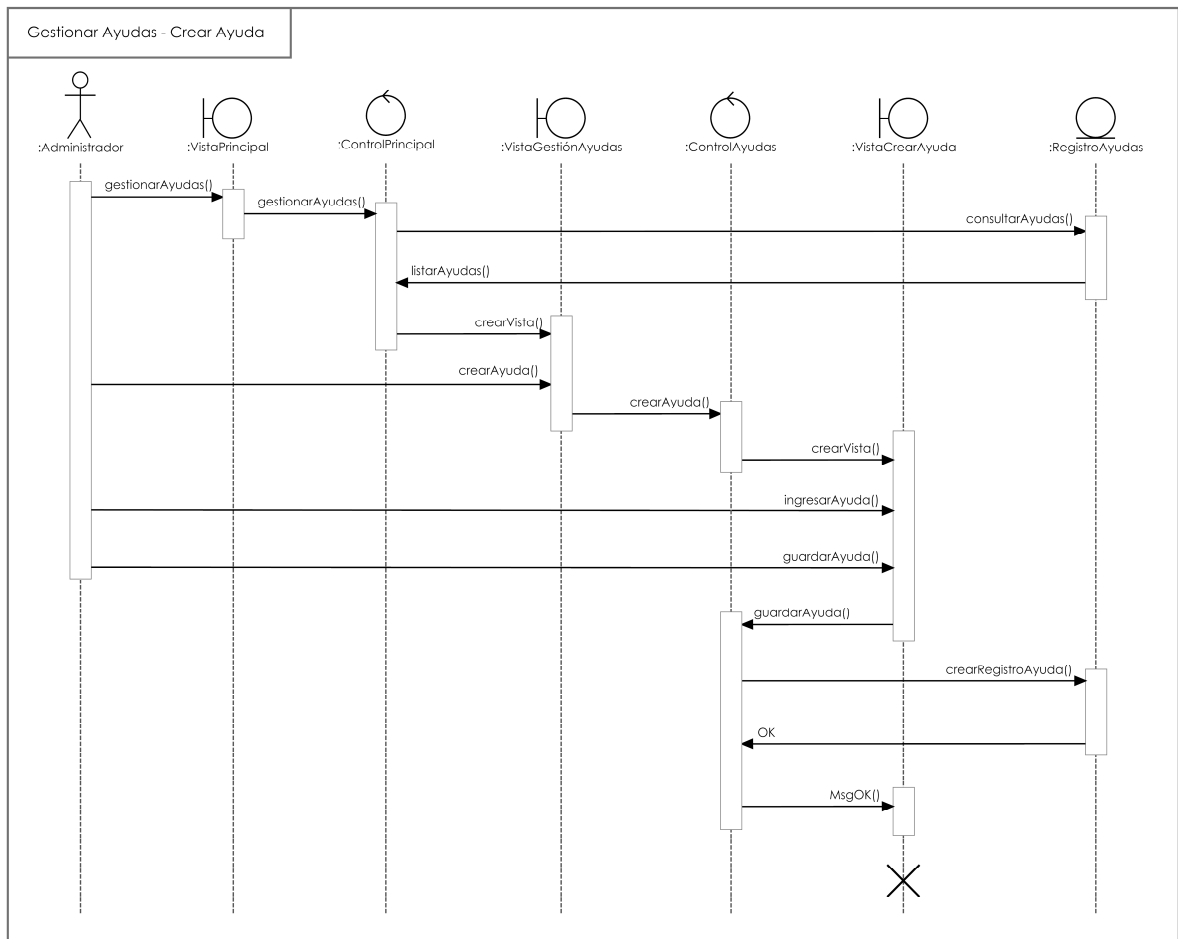
TB_Programas	Contiene los programas académicos de la escuela a la cual pertenece el portal.
TR_Pensum	Relaciona las materias de todos los programas académicos.
TB_MateriasN	Contiene todas las materias de todos los programas académicos en la escuela.
TB_Ciudades	Contiene ciudades en las cuales hay sedes de la universidad.
TP_Recurso	Contiene la información básica de los recursos en el portal.
TR_UsoRecurso	Relaciona recursos con el tiempo que están ocupados.
TB_TipoRecurso	Contiene los tipos de recurso.
TP_Interesado	Contiene la información personal de todos aquellos que apliquen para una inscripción.
TP_Inscripcion	Contiene las inscripciones que se hayan creado en el portal.
TR_InscripInteresado	Relaciona una inscripción con un interesado y almacena la información que éste haya diligenciado.

Fuente: Autores.

4.1.4 Modelo de Procesos del Sistema

4.1.4.1 Servicio de Ayudas

Figura 20. Diagrama de secuencia: Gestionar Ayudas - Crear Ayuda

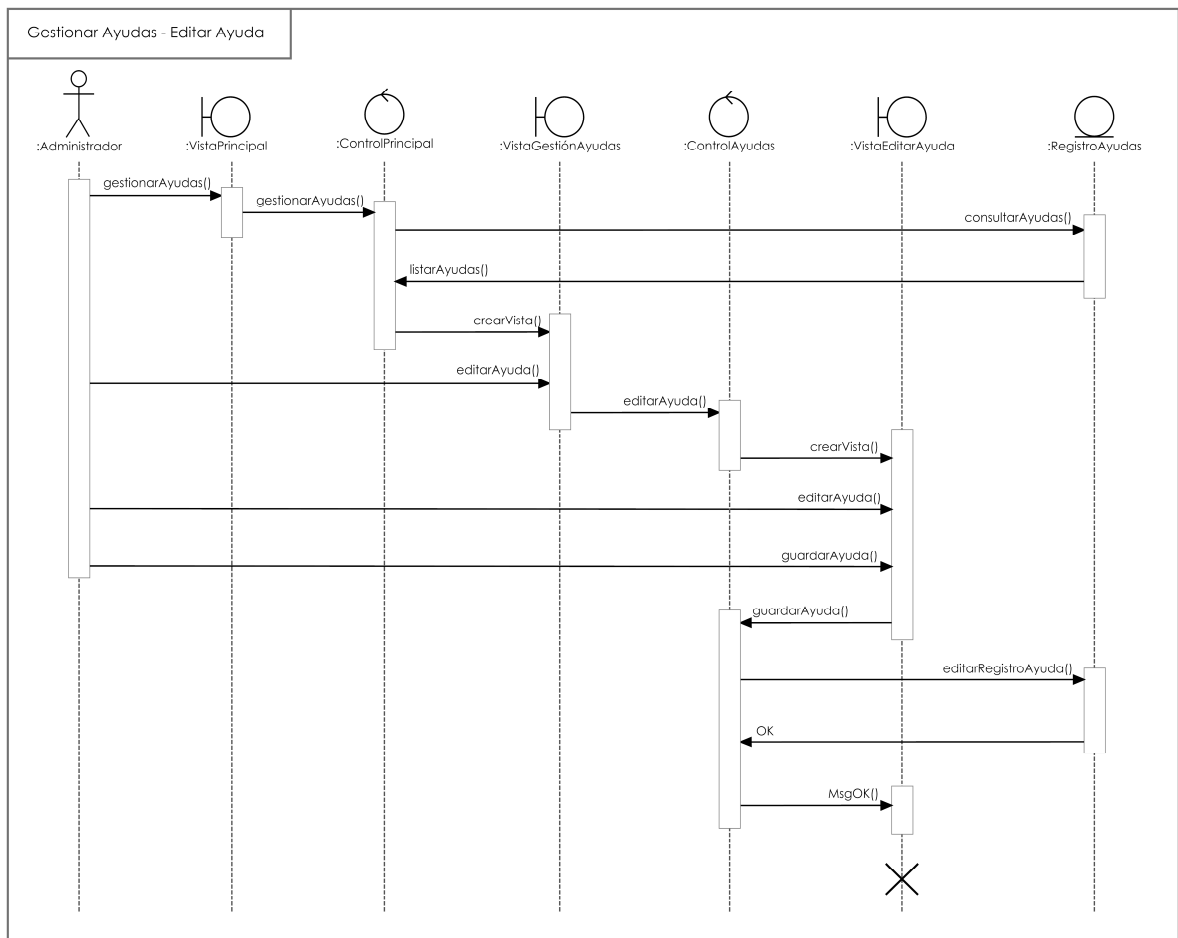


Fuente: autores.

1. El Administrador solicita gestionar ayudas en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las ayudas en base de datos y despliega la vista VistaGestiónAyudas.
3. El Administrador solicita crear ayuda en la vista VistaGestiónAyudas.

4. El controlador ControlAyudas recibe la petición y despliega la vista VistaCrearAyuda.
5. El Administrador ingresa la información de la ayuda y la guarda.
6. El controlador ControlAyudas recibe la petición de guardar la ayuda, crea el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaCrearAyuda.
7. Se destruye la vista VistaCrearAyuda.

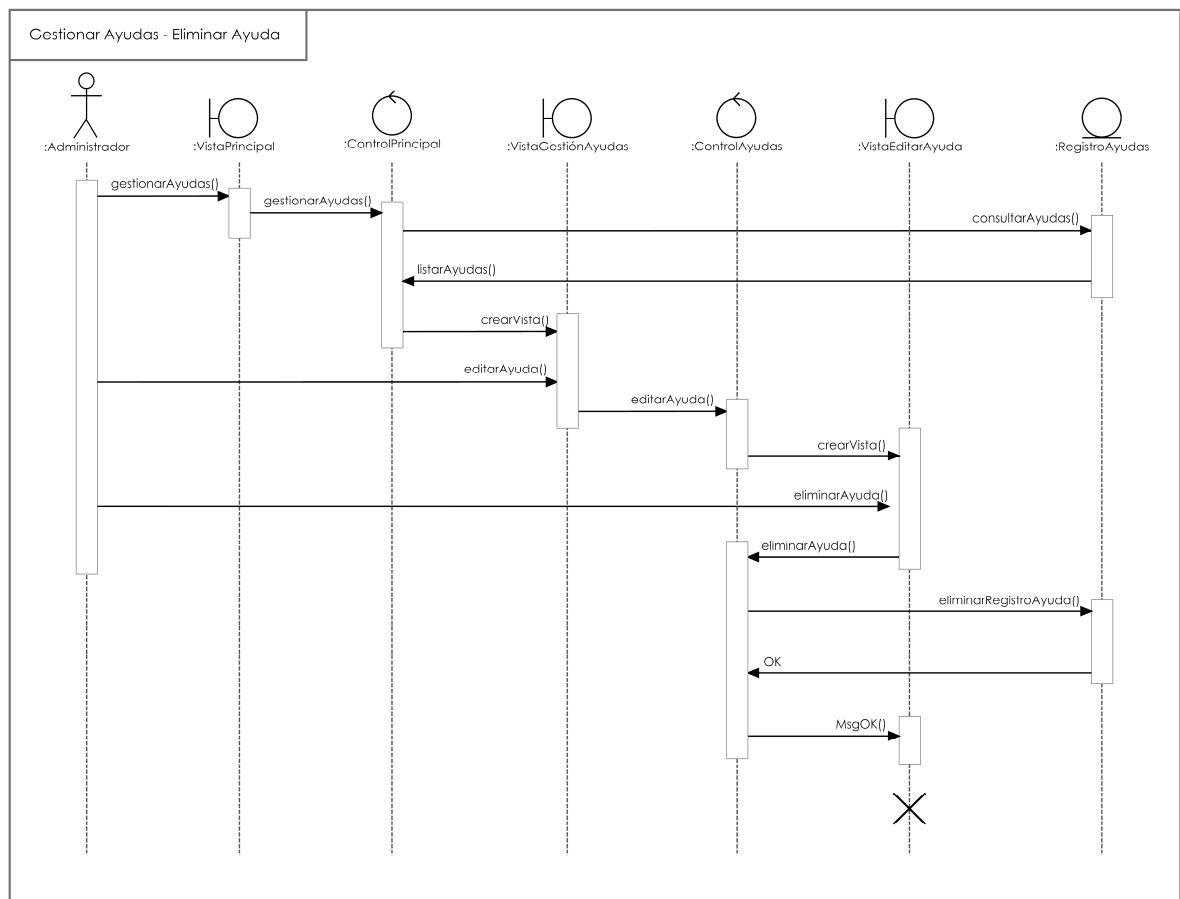
Figura 21. Diagrama de secuencia: Gestionar Ayudas - Editar Ayuda



Fuente: autores.

1. El Administrador solicita gestionar ayudas en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las ayudas en base de datos y despliega la vista VistaGestiónAyudas.
3. El Administrador solicita editar una ayuda en la vista VistaGestiónAyudas.
4. El controlador ControlAyudas recibe la petición y despliega la vista VistaEditarAyuda.
5. El Administrador actualiza la información de la ayuda y la guarda.
6. El controlador ControlAyudas recibe la petición de editar la ayuda, actualiza el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaEditarAyuda.
7. Se destruye la vista VistaEditarAyuda.

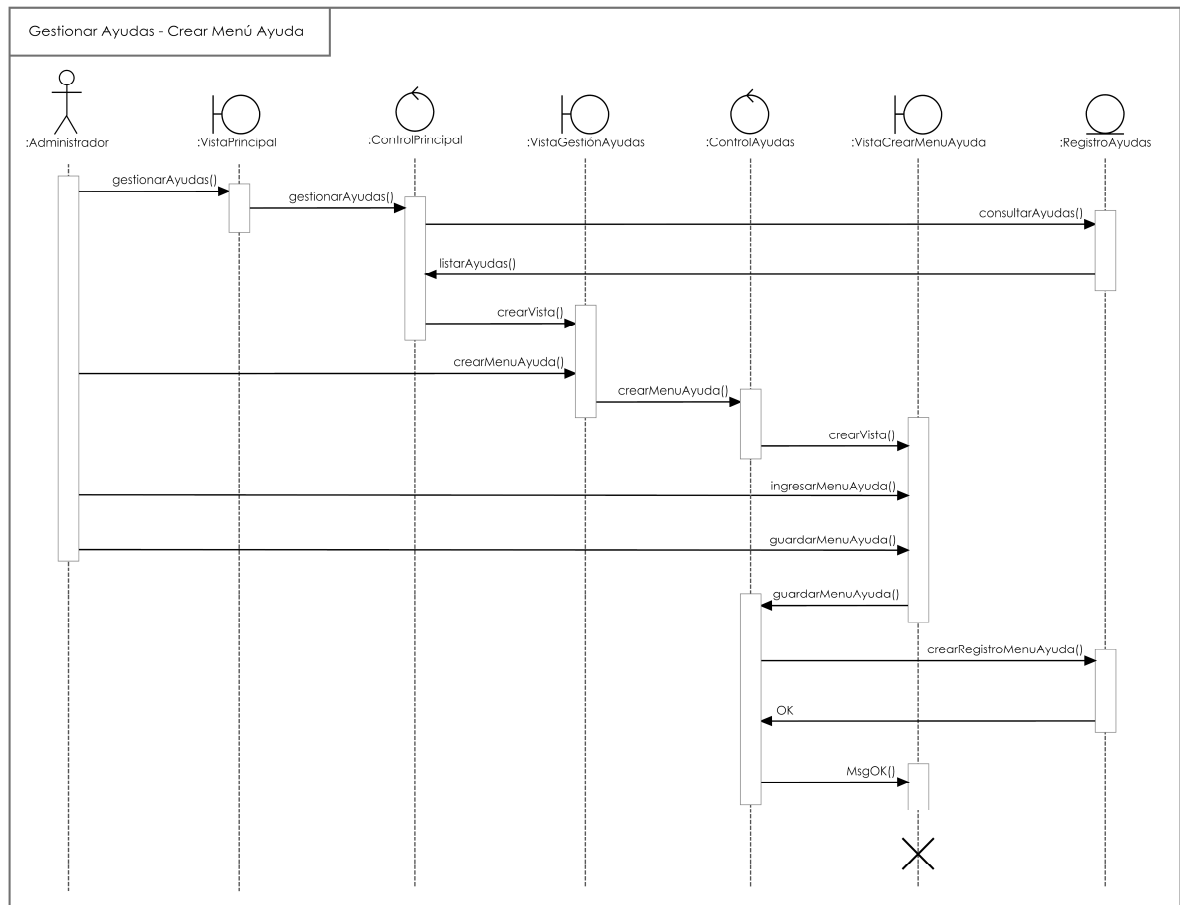
Figura 22. Diagrama de secuencia: Gestionar Ayudas - Eliminar Ayuda



Fuente: Autores

1. El Administrador solicita gestionar ayudas en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las ayudas en base de datos y despliega la vista VistaGestiónAyudas.
3. El Administrador solicita editar una ayuda en la vista VistaGestiónAyudas.
4. El controlador ControlAyudas recibe la petición y despliega la vista VistaEditarAyuda.
5. El Administrador solicita eliminar la ayuda.
6. El controlador ControlAyudas recibe la petición de eliminar la ayuda, elimina el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaEditarAyuda.
7. Se destruye la vista VistaEditarAyuda.

Figura 23. Diagrama de secuencia: Gestionar Ayudas - Crear Menú Ayuda

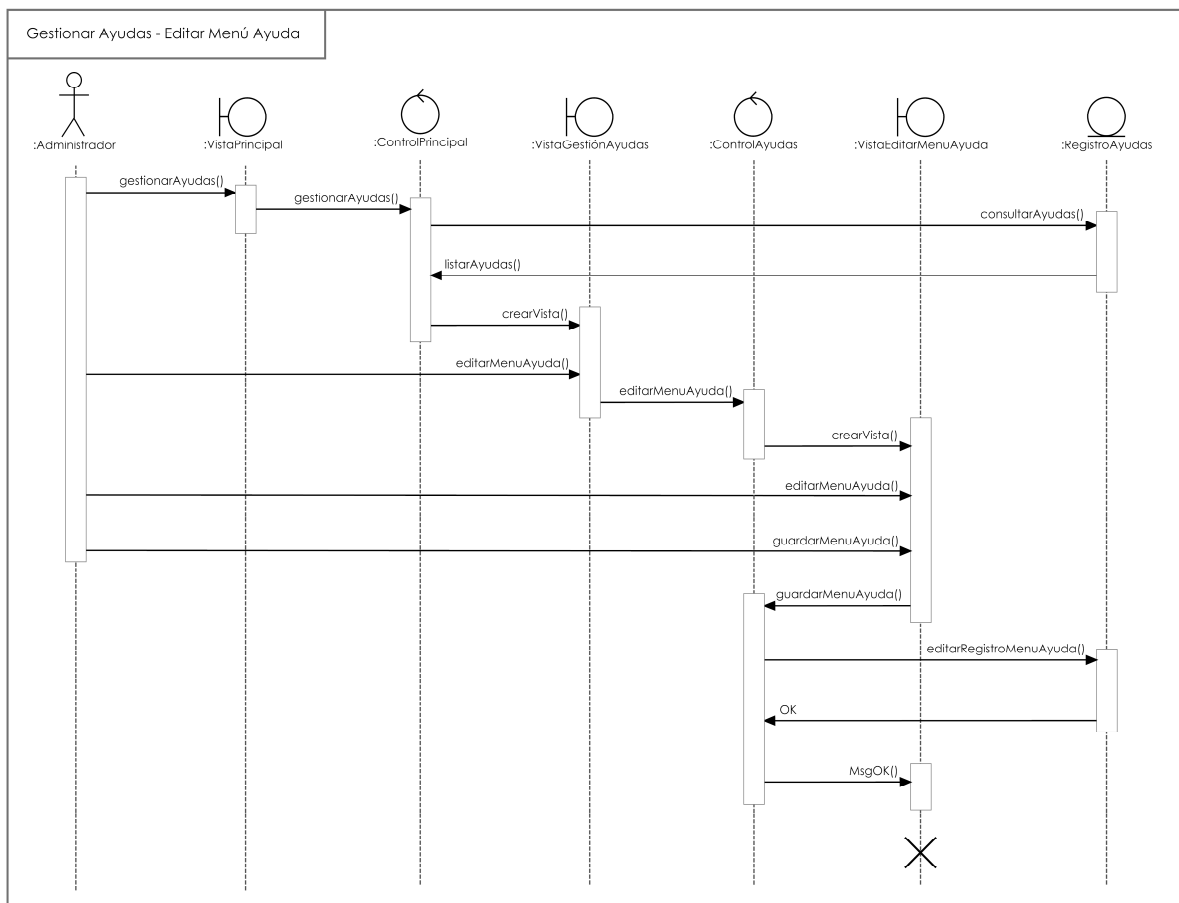


Fuente: Autores.

1. El Administrador solicita gestionar ayudas en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta los menús de ayuda en base de datos y despliega la vista VistaGestiónAyudas.
3. El Administrador solicita crear un menú de ayuda en la vista VistaGestiónAyudas.
4. El controlador ControlAyudas recibe la petición y despliega la vista VistaCrearMenuAyuda.
5. El Administrador ingresa la información del menú de ayuda y lo guarda.

6. El controlador ControlAyudas recibe la petición de guardar el menú de ayuda, crea el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaCrearMenuAyuda.
7. Se destruye la vista VistaCrearMenuAyuda.

Figura 24. Diagrama de secuencia: Gestionar Ayudas - Editar Menú Ayuda

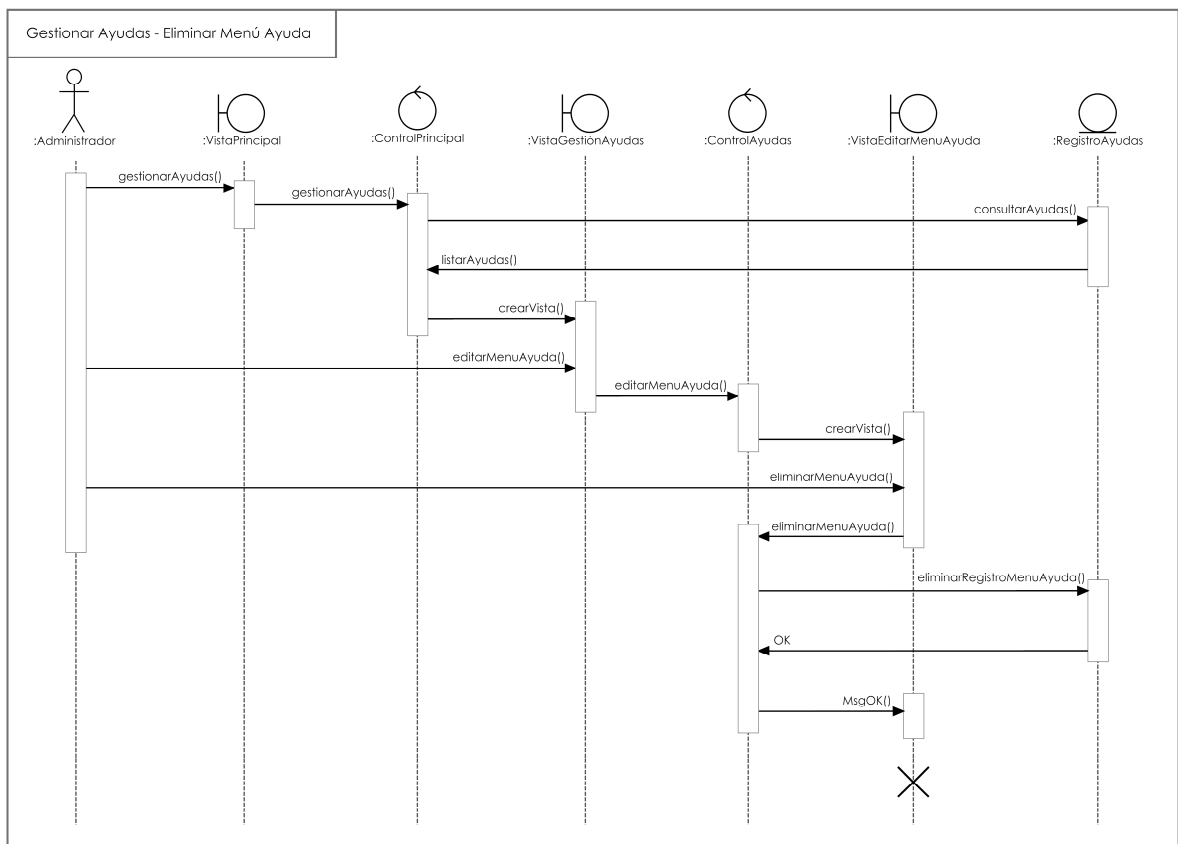


Fuente: Autores.

1. El Administrador solicita gestionar ayudas en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta los menús de ayudas en base de datos y despliega la vista VistaGestiónAyudas.

3. El Administrador solicita editar un menú de ayuda en la vista VistaGestiónAyudas.
4. El controlador ControlAyudas recibe la petición y despliega la vista VistaEditarMenuAyuda.
5. El Administrador solicita eliminar el menú de ayuda.
6. El controlador ControlAyudas recibe la petición de eliminar el menú de ayuda, elimina el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaEditarMenuAyuda.
7. Se destruye la vista VistaEditarMenuAyuda.

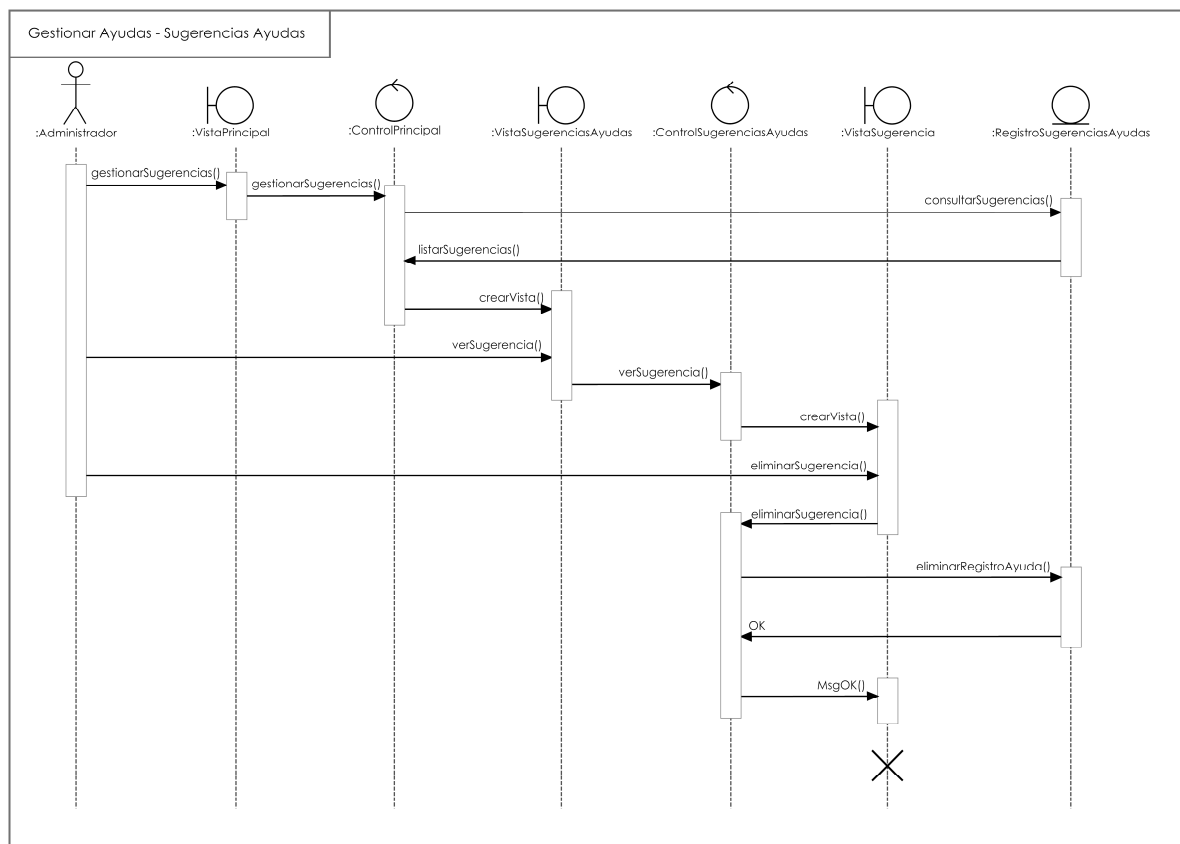
Figura 25. Diagrama de secuencia: Gestionar Ayudas - Eliminar Menú Ayuda



Fuente: Autores.

1. El Administrador solicita gestionar ayudas en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta los menús de ayudas en base de datos y despliega la vista VistaGestiónAyudas.
3. El Administrador solicita editar un menú de ayuda en la vista VistaGestiónAyudas.
4. El controlador ControlAyudas recibe la petición y despliega la vista VistaEditarMenuAyuda.
5. El Administrador solicita eliminar el menú de ayuda.
6. El controlador ControlAyudas recibe la petición de eliminar el menú de ayuda, elimina el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaEditarMenuAyuda.
7. Se destruye la vista VistaEditarMenuAyuda.

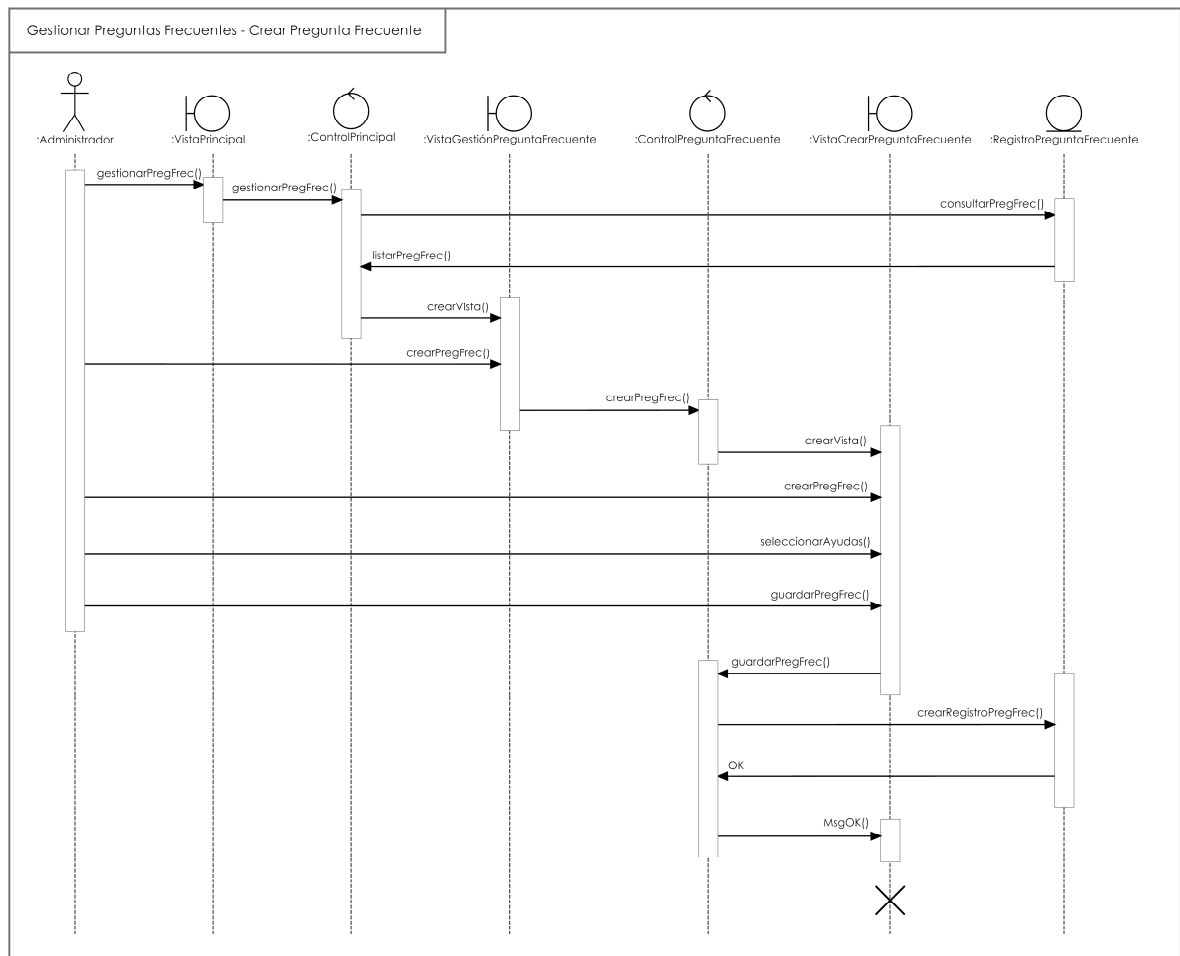
Figura 26. Diagrama de secuencia: Gestionar Ayudas – Sugerencias Ayudas



Fuente: Autores.

1. El Administrador solicita gestionar las sugerencias de ayudas en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las sugerencias en base de datos y despliega la vista VistaSugerenciasAyudas.
3. El Administrador solicita ver una sugerencia en la vista VistaSugerenciasAyudas.
4. El controlador ControlSugerenciasAyudas recibe la petición y despliega la vista VistaSugerencia.
5. El Administrador solicita eliminar la sugerencia.
6. El controlador ControlAyudas recibe la petición de eliminar la sugerencia, elimina el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaSugerencia.
7. Se destruye la vista VistaSugerencia.

Figura 27. Diagrama de secuencia: Gestionar Preguntas Frecuentes - Crear Pregunta Frecuente

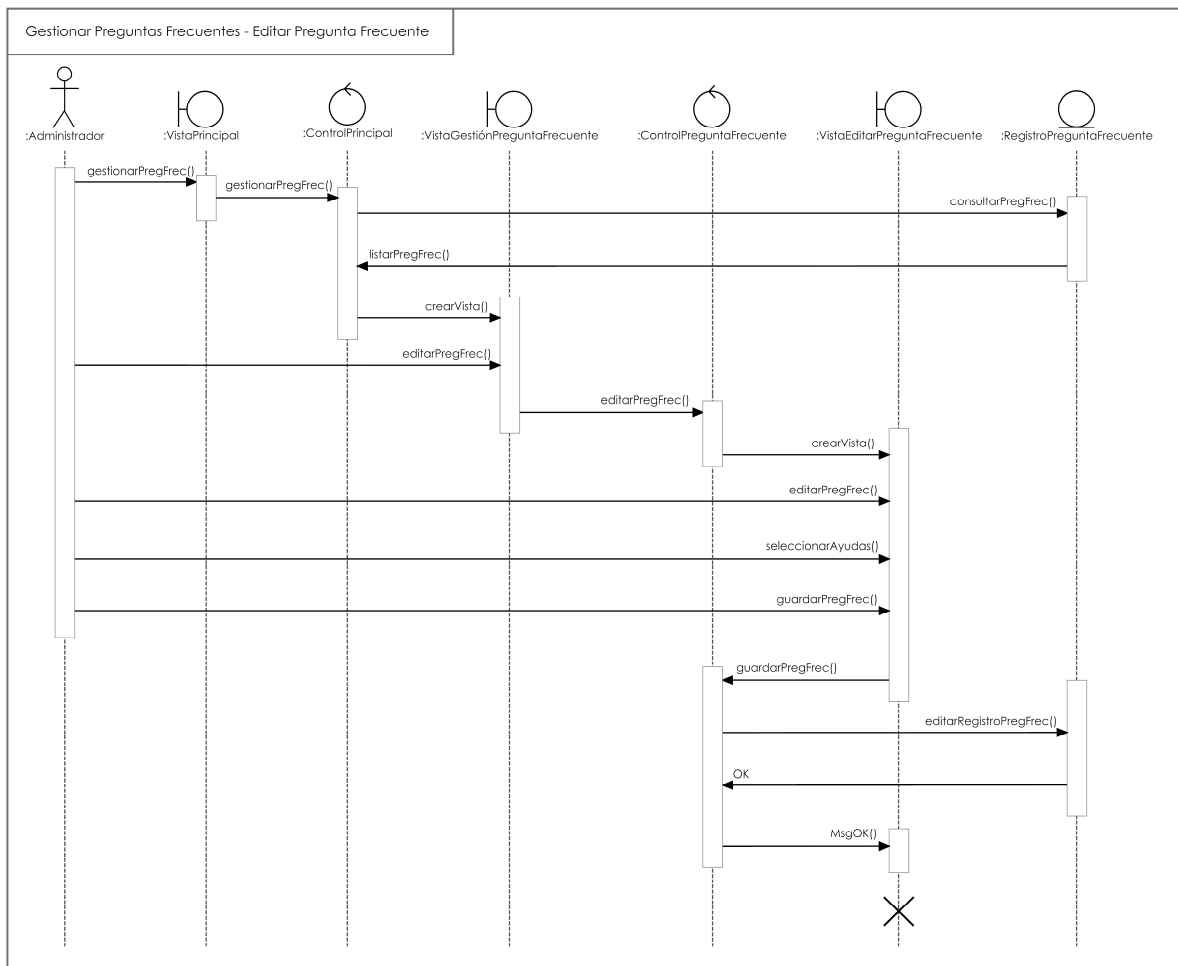


Fuente: Autores.

1. El Administrador solicita gestionar preguntas frecuentes en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las preguntas frecuentes en base de datos y despliega la vista VistaGestiónPreguntaFrecuente.
3. El Administrador solicita crear pregunta frecuente en la vista VistaGestiónPreguntaFrecuente.

4. El controlador ControlPreguntaFrecuente recibe la petición y despliega la vista VistaCrearPreguntaFrecuente.
5. El Administrador ingresa la información de la pregunta frecuente y la guarda.
6. El controlador ControlPreguntaFrecuente recibe la petición de guardar la pregunta frecuente, crea el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaCrearPreguntaFrecuente.

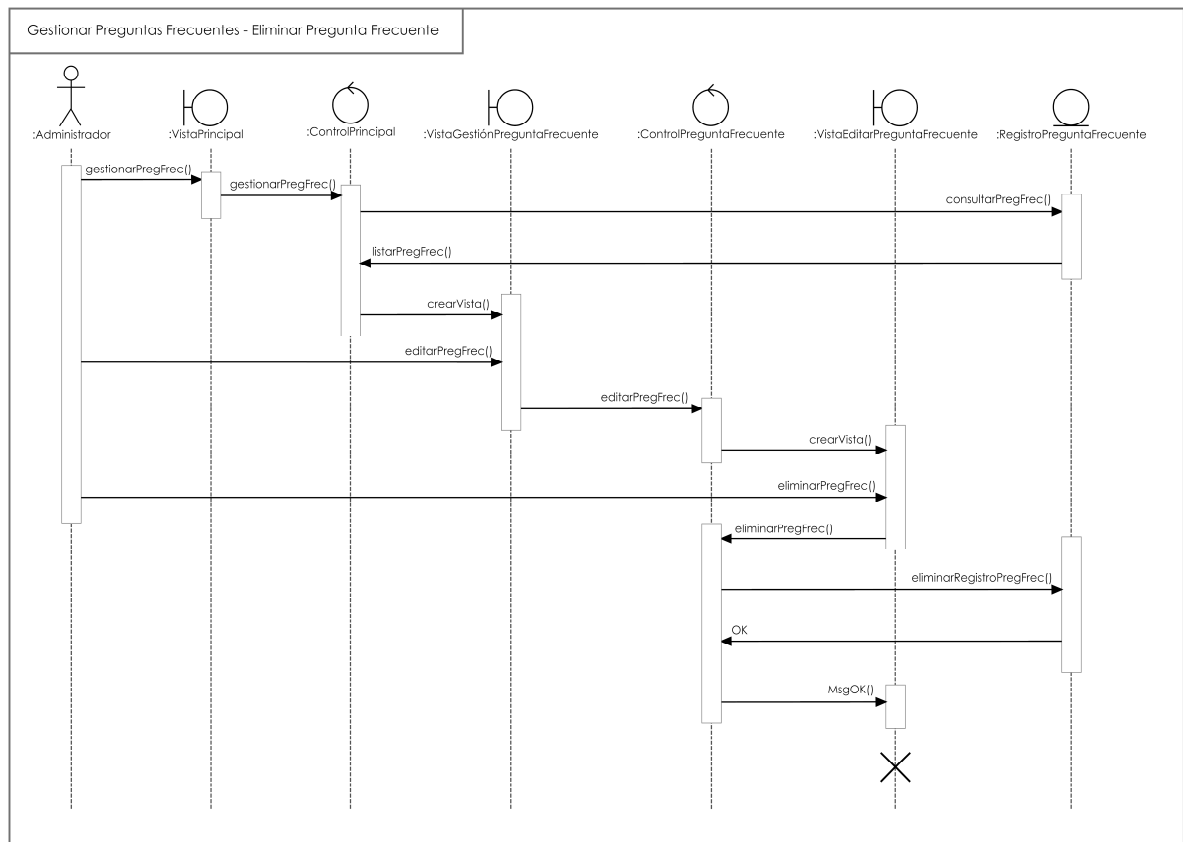
Figura 28. Diagrama de secuencia: Gestionar Preguntas Frecuentes - Editar Pregunta Frecuente



Fuente: Autores.

1. El Administrador solicita gestionar preguntas frecuentes en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las preguntas frecuentes en base de datos y despliega la vista VistaGestiónPreguntaFrecuente.
3. El Administrador solicita editar una pregunta frecuente en la vista VistaGestiónPreguntaFrecuente.
4. El controlador ControlPreguntaFrecuente recibe la petición y despliega la vista VistaEditarPreguntaFrecuente.
5. El Administrador actualiza la información de la pregunta frecuente y la guarda.
6. El controlador ControlPreguntaFrecuente recibe la petición de actualizar la pregunta frecuente, edita el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaEditarPreguntaFrecuente.
7. Se destruye la vista VistaEditarPreguntaFrecuente.

Figura 29. Diagrama de secuencia: Gestionar Preguntas Frecuentes - Eliminar Pregunta Frecuente

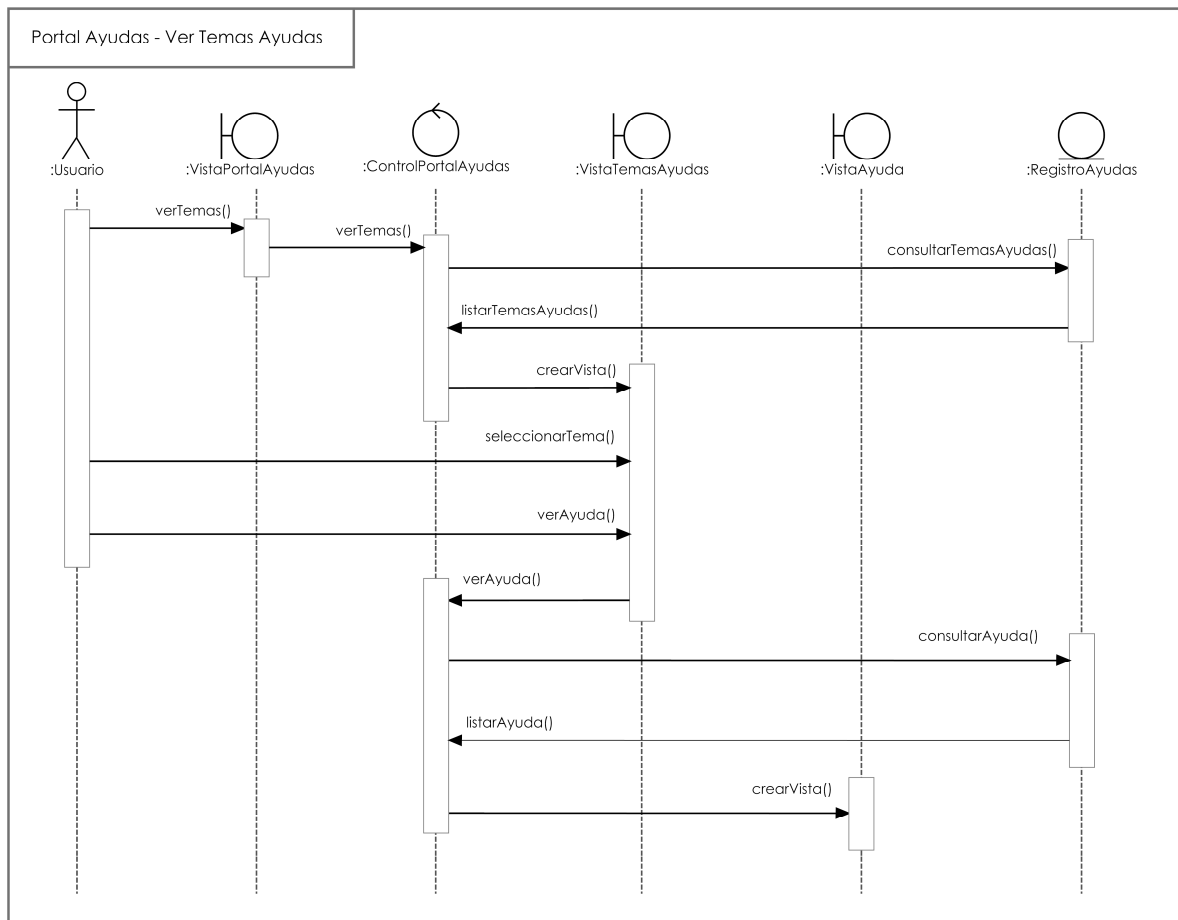


Fuente: Autores.

1. El Administrador solicita gestionar preguntas frecuentes en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las preguntas frecuentes en base de datos y despliega la vista VistaGestiónPreguntaFrecuente.
3. El Administrador solicita editar una pregunta frecuente en la vista VistaGestiónPreguntaFrecuente.
4. El controlador ControlPreguntaFrecuente recibe la petición y despliega la vista VistaEditarPreguntaFrecuente.
5. El Administrador solicita eliminar la pregunta frecuente.

6. El controlador `ControlPreguntaFrecuente` recibe la petición de eliminar la pregunta frecuente, elimina el registro en base de datos y retorna un mensaje de Ok al usuario en la vista `VistaEditarPreguntaFrecuente`.
7. Se destruye la vista `VistaEditarPreguntaFrecuente`.

Figura 30. Diagrama de secuencia: Portal Ayudas - Ver Temas Ayudas

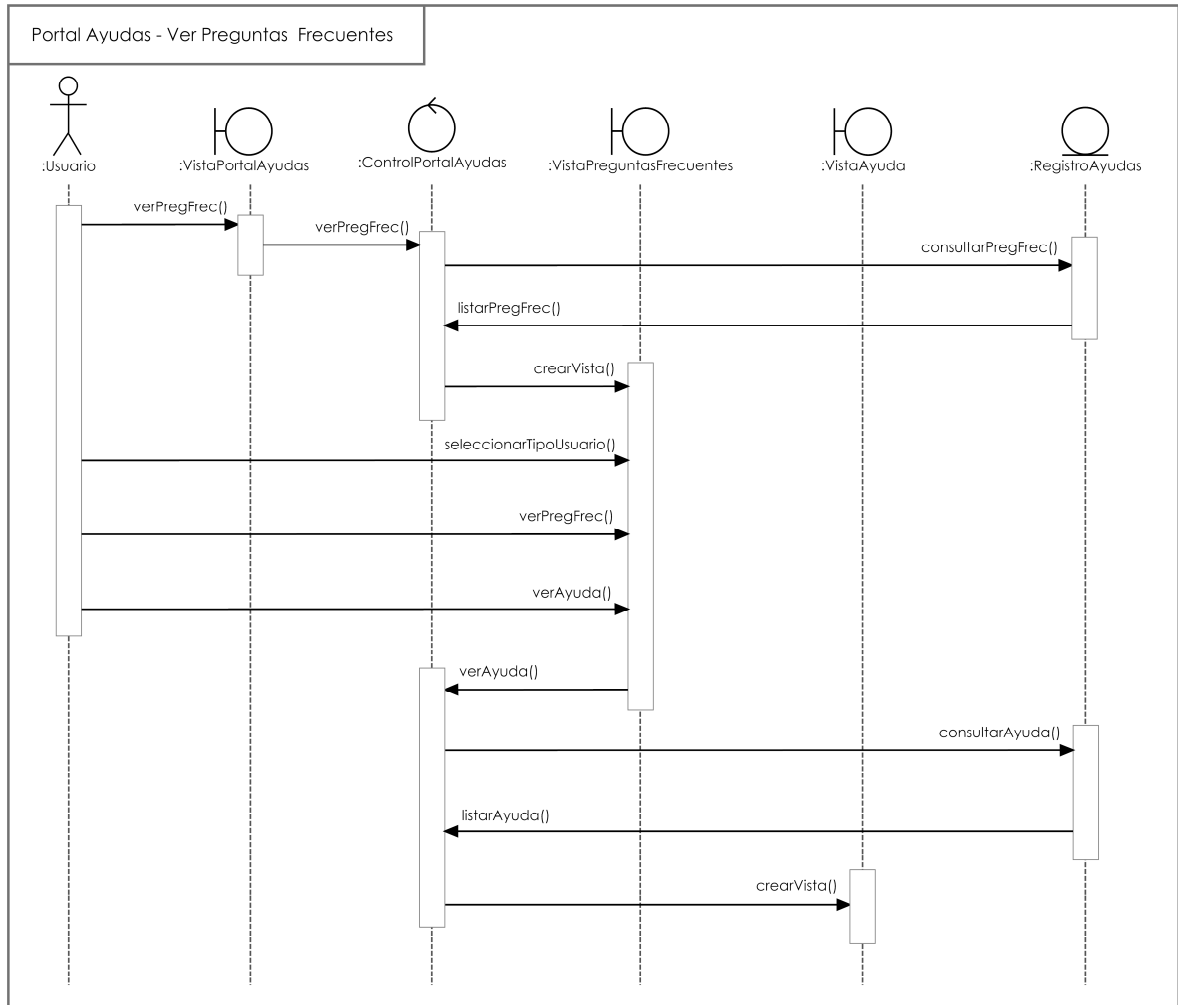


Fuente: Autores.

1. El usuario solicita ver los temas de ayuda en la vista `VistaPortalAyudas`.
2. El controlador `ControlPortalAyudas` recibe la petición, consulta los temas de ayuda en base de datos y despliega la vista `VistaTemasAyudas`.
3. El usuario selecciona el tema de ayuda que le interese.

4. El usuario selecciona la ayuda que desea ver.
5. El controlador ControlPortalAyudas recibe la petición de ver ayuda, consulta la ayuda en base de datos y despliega la vista VistaAyuda.

Figura 31. Diagrama de secuencia: Portal Ayudas - Ver Preguntas Frecuentes

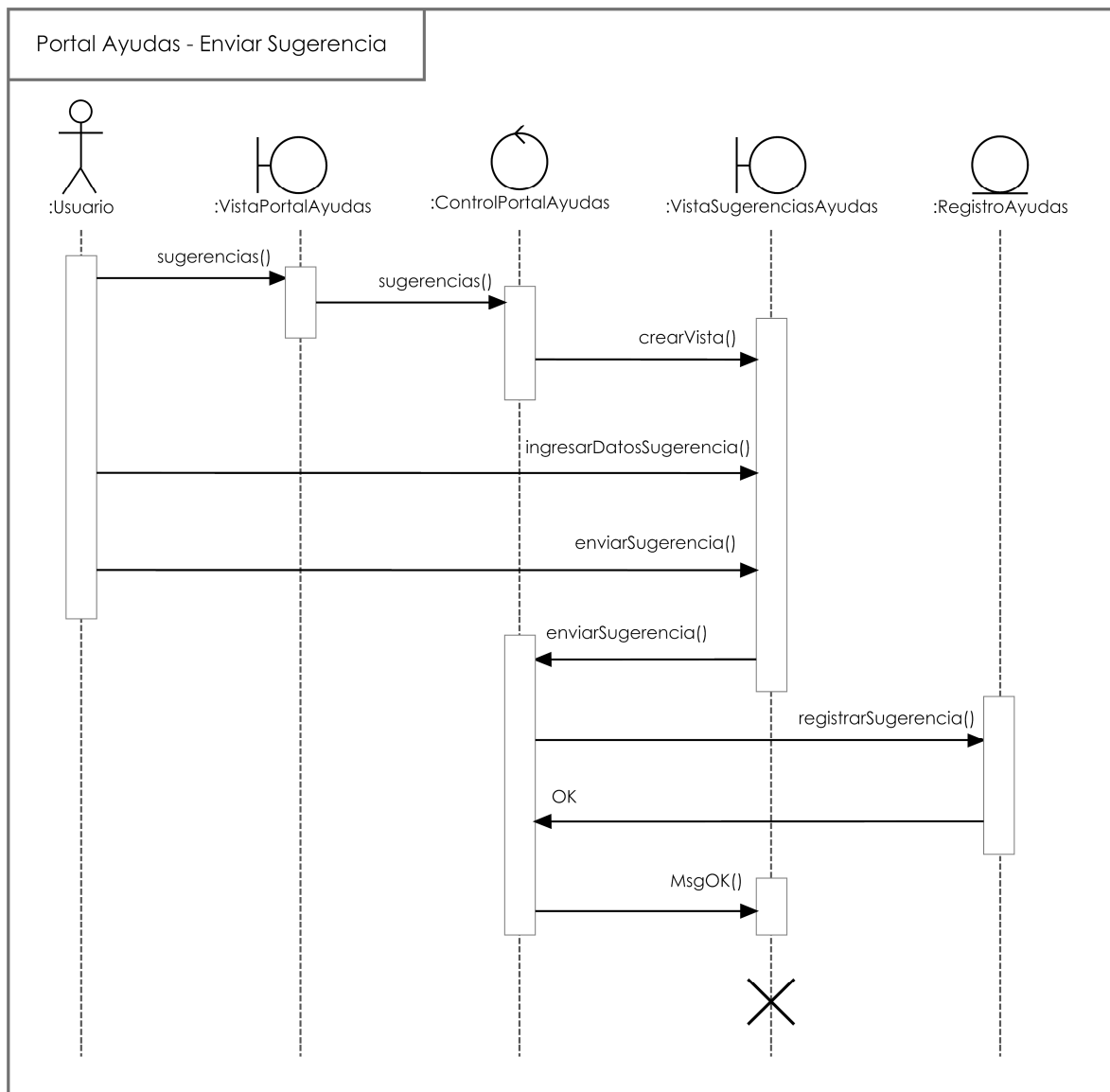


Fuente: Autores.

1. El usuario solicita ver las preguntas frecuentes en la vista VistaPortalAyudas.
2. El controlador ControlPortalAyudas recibe la petición, consulta las preguntas frecuentes en base de datos y despliega la vista VistaPreguntasFrecuentes.

3. El usuario selecciona su tipo de usuario.
4. El usuario selecciona la pregunta frecuente que le interese.
5. El usuario selecciona la ayuda que desea ver.
6. El controlador ControlPortalAyudas recibe la petición de ver ayuda, consulta la ayuda en base de datos y despliega la vista VistaAyuda.

Figura 32. Diagrama de secuencia: Portal Ayudas - Enviar Sugerencia

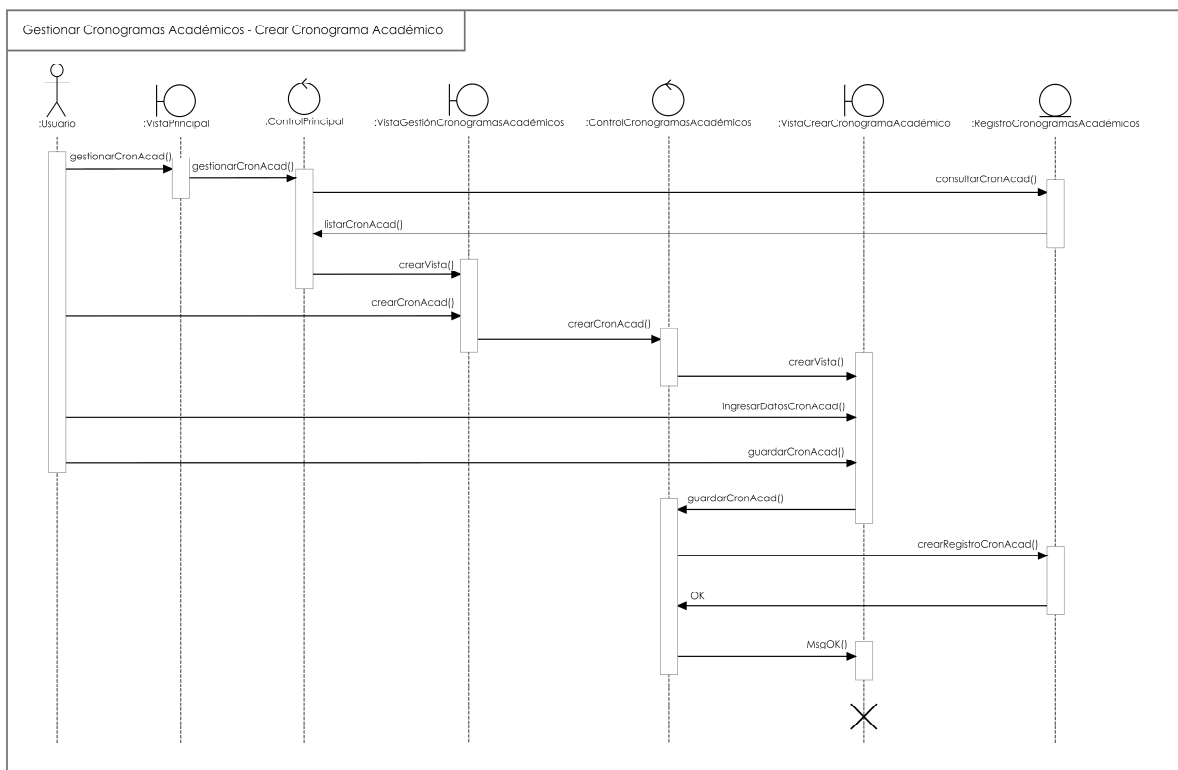


Fuente: Autores.

1. El usuario solicita enviar una sugerencia en la vista VistaPortalAyudas.
2. El controlador ControlPortalAyudas recibe la petición y crea la vista VistaSugerenciasAyudas.
3. El usuario ingresa los datos de la sugerencia y la envía.
4. El controlador ControlPortalAyudas recibe la solicitud de enviar sugerencia, crea el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaSugerenciasAyudas.
5. Se destruye la vista VistaSugerenciasAyudas.

4.1.4.2 Servicio de Cronogramas Académicos

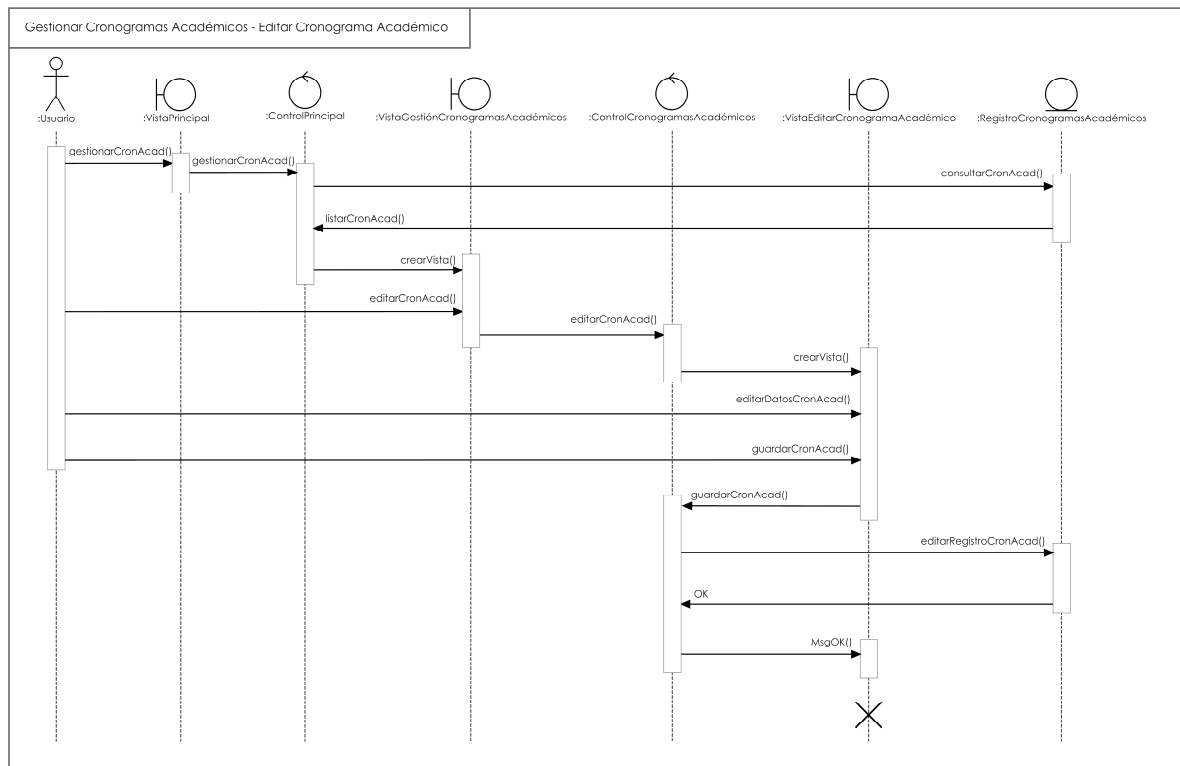
Figura 33. Diagrama de secuencia: Gestionar Cronogramas Académicos - Crear Cronograma Académico



Fuente: Autores.

1. El Usuario solicita gestionar cronogramas académicos en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta los cronogramas académicos en base de datos y despliega la vista VistaGestiónCronogramasAcadémicos.
3. El Usuario solicita crear cronograma académico en la vista VistaGestiónCronogramasAcadémicos.
4. El controlador ControlCronogramasAcadémicos recibe la petición y despliega la vista VistaCrearCronogramaAcadémico.
5. El Usuario ingresa la información del cronograma académico y lo guarda.
6. El controlador ControlCronogramasAcadémicos recibe la petición de guardar el cronograma académico, crea el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaCrearCronogramaAcadémico.
7. Se destruye la vista VistaCrearCronogramaAcadémico.

Figura 34. Diagrama de secuencia: Gestionar Cronogramas Académicos - Editar Cronograma Académico

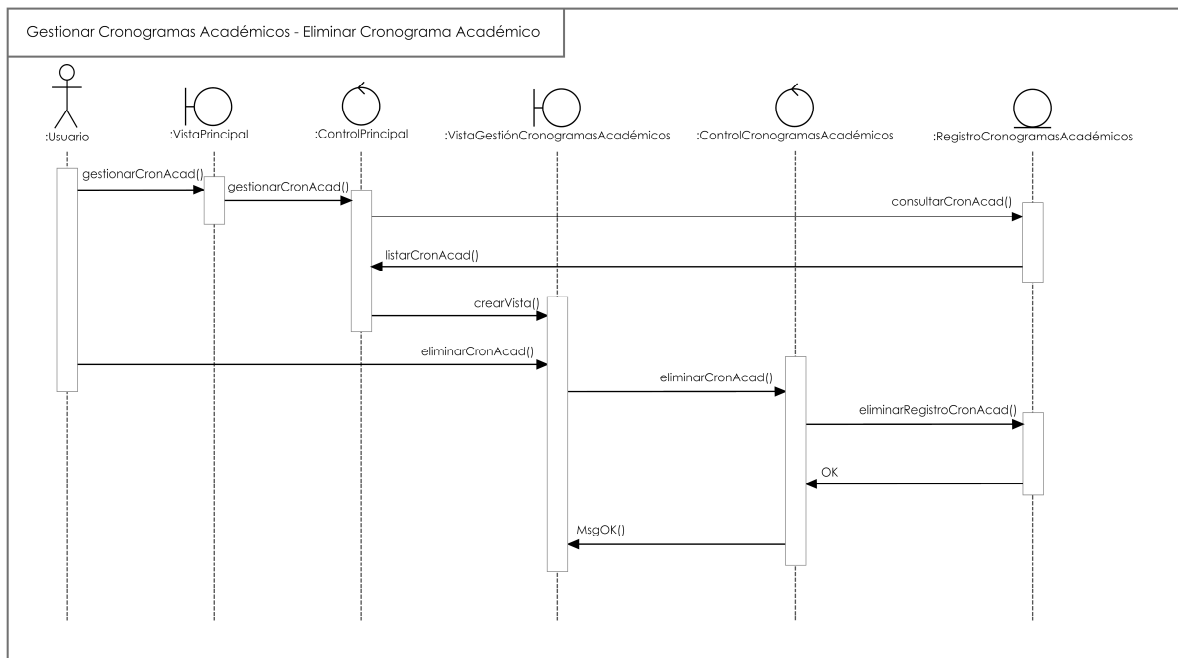


Fuente: Autores.

1. El Usuario solicita gestionar cronogramas académicos en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta los cronogramas académicos en base de datos y despliega la vista VistaGestiónCronogramasAcadémicos.
3. El Usuario solicita editar un cronograma académico en la vista VistaGestiónCronogramasAcadémicos.
4. El controlador ControlCronogramasAcadémicos recibe la petición y despliega la vista VistaEditarCronogramaAcadémico.
5. El Usuario actualiza la información del cronograma académico y lo guarda.

6. El controlador `ControlCronogramasAcadémicos` recibe la petición de editar el cronograma académico, actualiza el registro en base de datos y retorna un mensaje de Ok al usuario en la vista `VistaEditarCronogramaAcadémico`.
7. Se destruye la vista `VistaEditarCronogramaAcadémico`.

Figura 35. Diagrama de secuencia: Gestionar Cronogramas Académicos - Eliminar Cronograma Académico

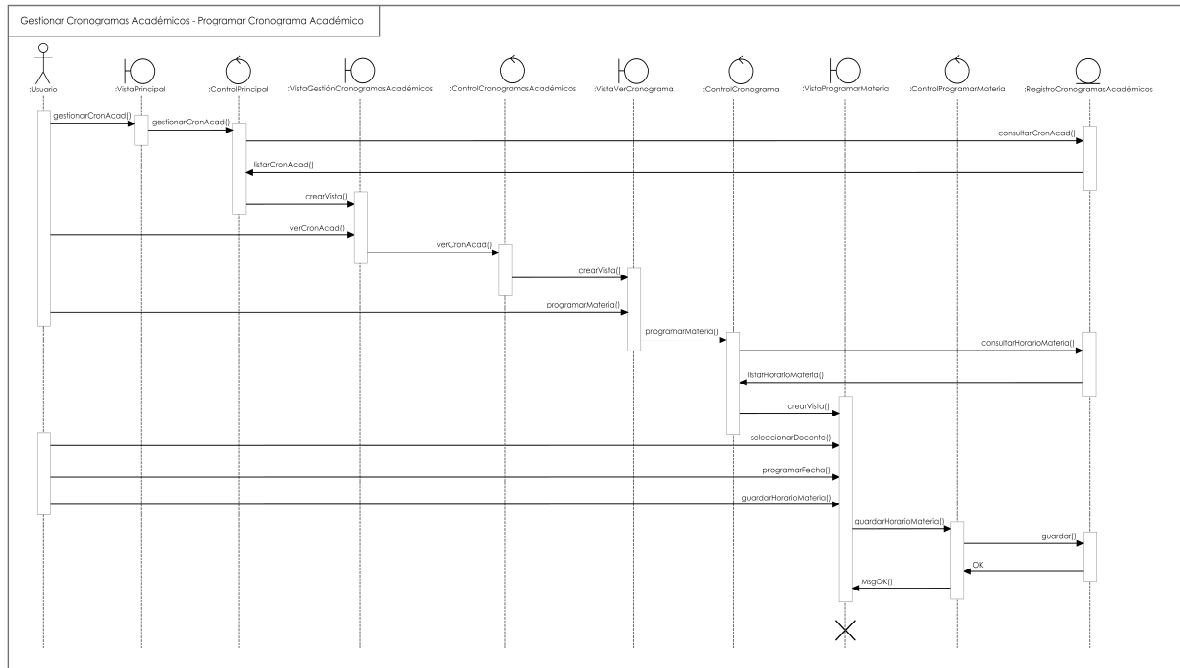


Fuente: Autores.

1. El Usuario solicita gestionar cronogramas académicos en la vista `VistaPrincipal`.
2. El controlador `ControlPrincipal` recibe la petición, consulta los cronogramas académicos en base de datos y despliega la vista `VistaGestiónCronogramasAcadémicos`.
3. El Usuario solicita eliminar un cronograma académico en la vista `VistaGestiónCronogramasAcadémicos`.

4. El controlador `ControlCronogramasAcadémicos` recibe la petición de eliminar el cronograma académico, elimina el registro en base de datos y retorna un mensaje de Ok al usuario en la vista `VistaGestiónCronogramasAcadémicos`.

Figura 36. Diagrama de secuencia: Gestionar Cronogramas Académicos - Programar Cronograma Académico

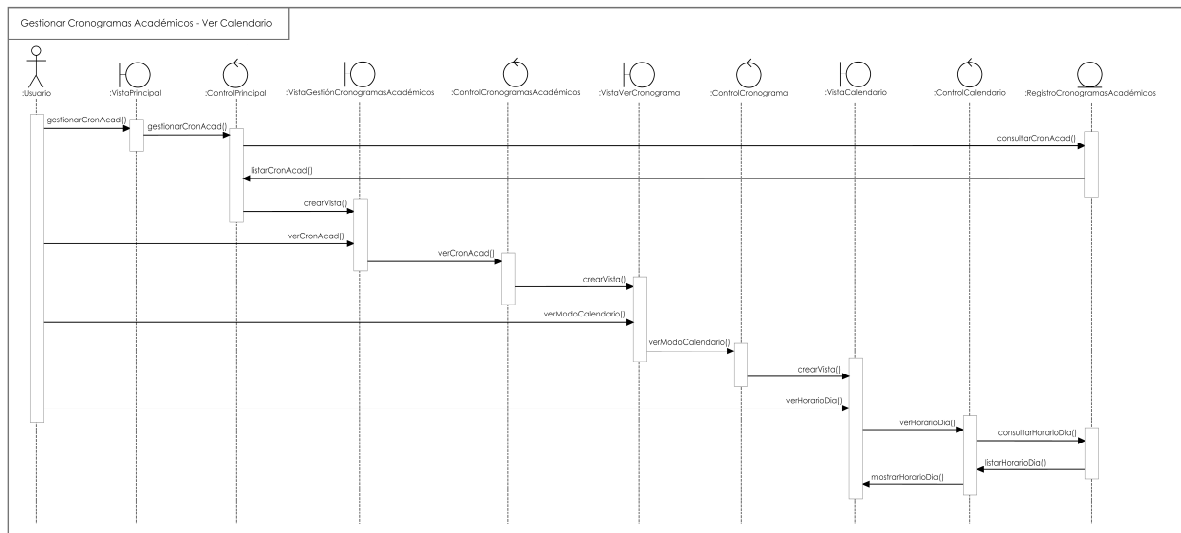


Fuente: Autores.

1. El Usuario solicita gestionar cronogramas académicos en la vista `VistaPrincipal`.
2. El controlador `ControlPrincipal` recibe la petición, consulta los cronogramas académicos en base de datos y despliega la vista `VistaGestiónCronogramasAcadémicos`.
3. El Usuario solicita ver un cronograma académico en la vista `VistaGestiónCronogramasAcadémicos`.
4. El controlador `ControlCronogramasAcadémicos` recibe la petición y despliega la vista `VistaVerCronograma`.

5. El Usuario solicita programar una materia en la vista VistaVerCronograma.
6. El controlador ControlCronograma recibe la petición, consulta el horario de la materia en base de datos y despliega la vista VistaProgramarMateria.
7. El Usuario selecciona el docente que dictará la materia.
8. El Usuario programa una o varias fechas, en las cuales selecciona rango de horas y salón.
9. El Usuario guarda el horario que programó.
10. El controlador ControlProgramarMateria recibe la petición de guardar el horario de la materia, crea los registros en base de datos y retorna un mensaje de Ok al usuario en la vista VistaProgramarMateria.
11. Se destruye la vista VistaProgramarMateria.

Figura 37. Diagrama de secuencia: Gestionar Cronogramas Académicos – Ver Calendario



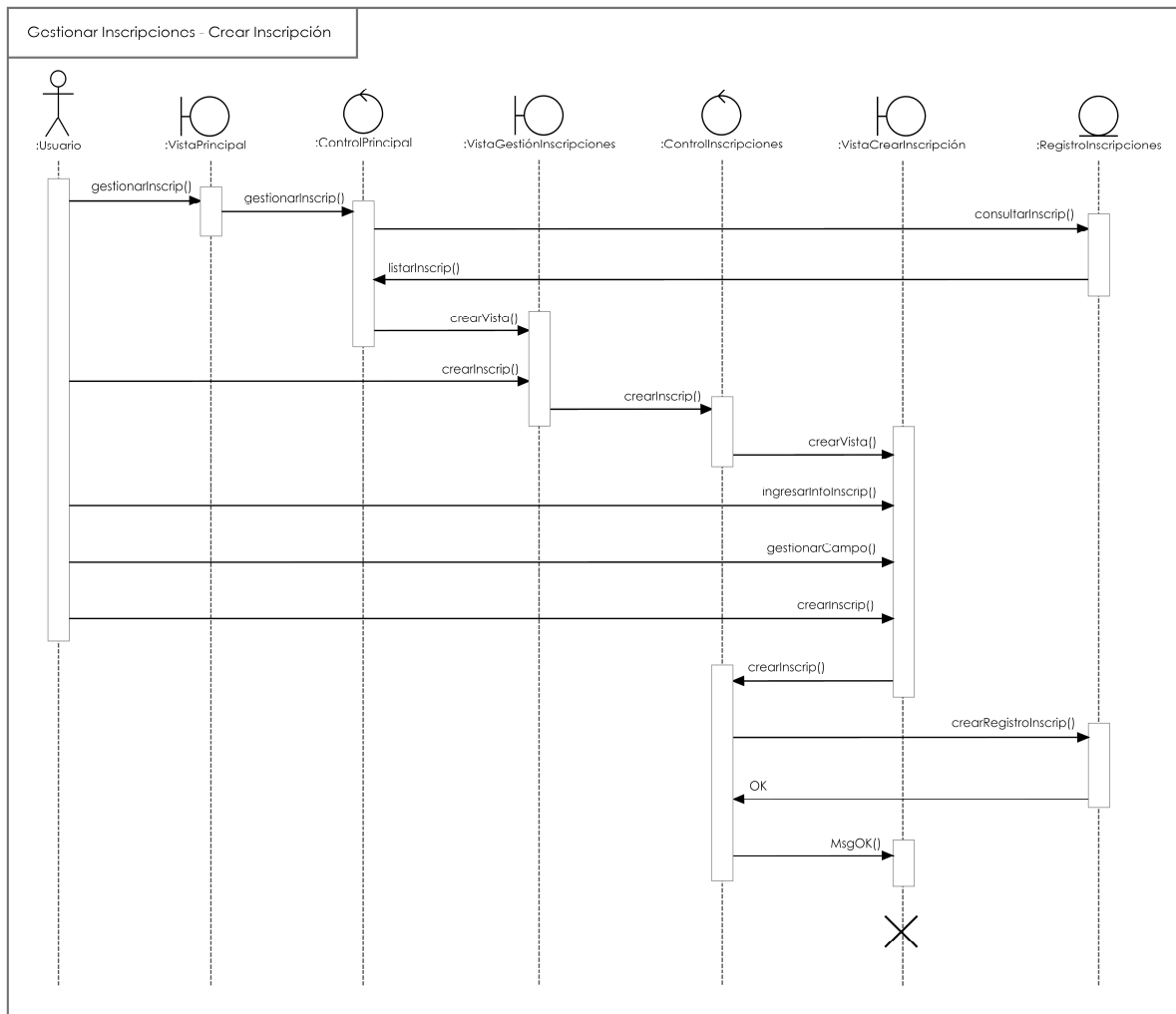
Fuente: Autores.

1. El Usuario solicita gestionar cronogramas académicos en la vista VistaPrincipal.

2. El controlador ControlPrincipal recibe la petición, consulta los cronogramas académicos en base de datos y despliega la vista VistaGestiónCronogramasAcadémicos.
3. El Usuario solicita ver un cronograma académico en la vista VistaGestiónCronogramasAcadémicos.
4. El controlador ControlCronogramasAcadémicos recibe la petición y despliega la vista VistaVerCronograma.
5. El Usuario solicita ver el cronograma en forma de calendario.
6. El controlador ControlCronograma recibe la petición y despliega la vista VistaCalendario.
7. El usuario solicita ver el horario completo de un día del año.
8. El controlador ControlCalendario recibe la petición y consulta en base de datos el horario para este día, lo despliega en la vista VistaCalendario.

4.1.4.3 Servicio de Inscripciones

Figura 38. Diagrama de secuencia: Gestionar Inscripciones - Crear Inscripción

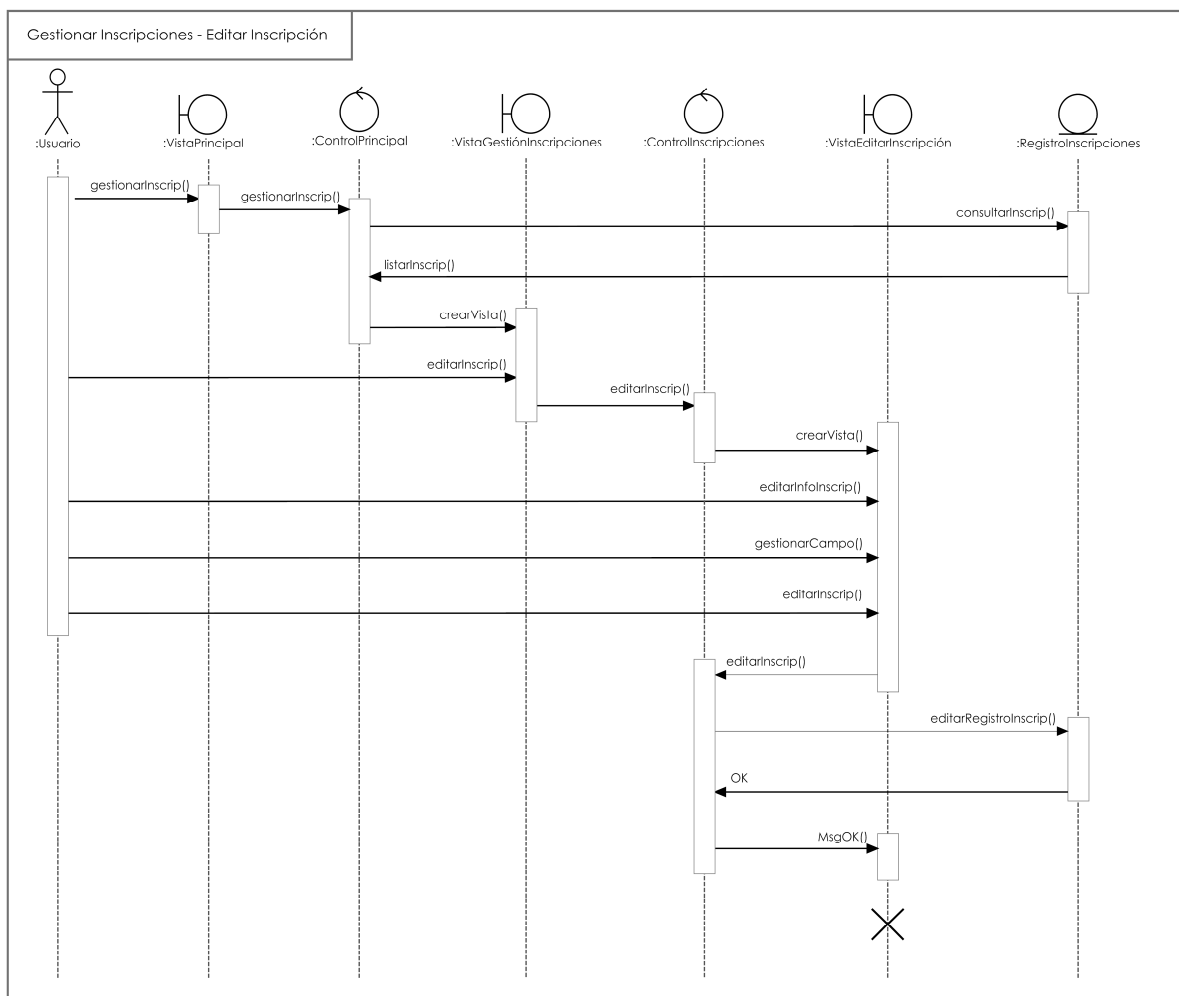


Fuente: Autores.

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestionarInscripciones.
3. El Usuario solicita crear inscripción en la vista VistaGestiónInscripciones.

4. El controlador `ControlInscripciones` recibe la petición y despliega la vista `VistaCrearInscripción`.
5. El Usuario ingresa la información de la inscripción, agrega los campos que necesite y la guarda.
6. El controlador `ControlInscripciones` recibe la petición de guardar la inscripción, crea el registro en base de datos y retorna un mensaje de Ok al usuario en la vista `VistaCrearInscripción`.
7. Se destruye la vista `VistaCrearInscripción`.

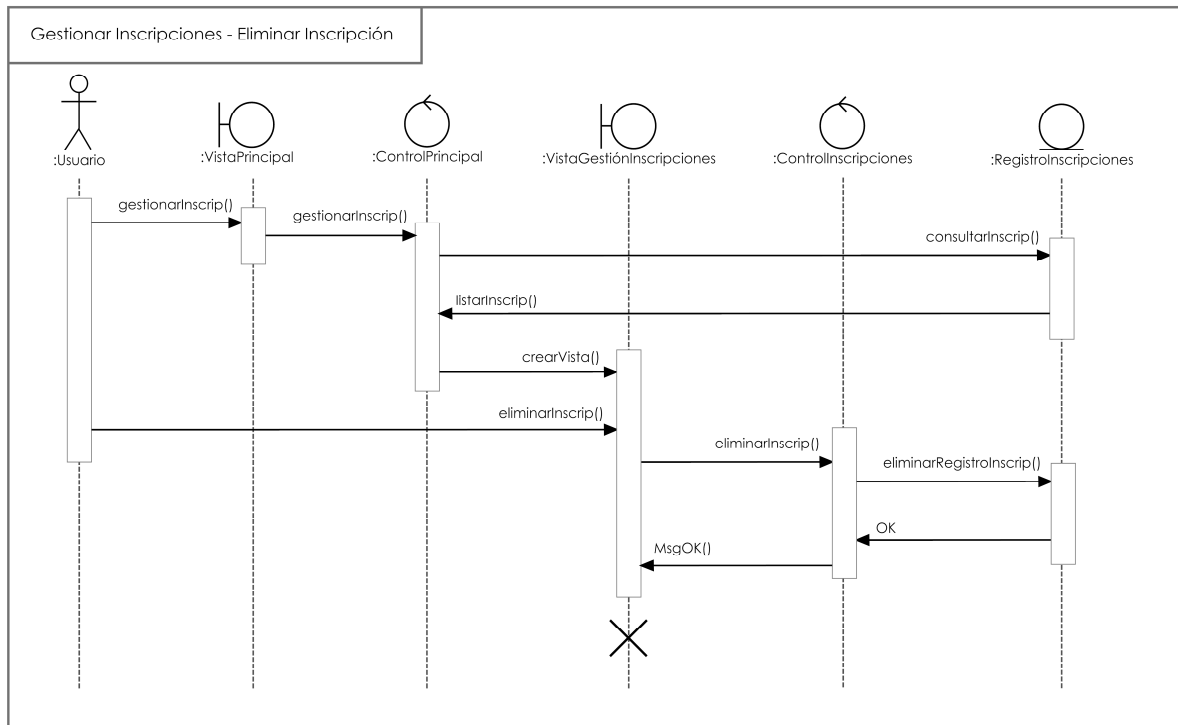
Figura 39. Diagrama de secuencia: Gestionar Inscripciones - Editar Inscripción



Fuente: Autores.

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita editar una inscripción en la vista VistaGestiónCronogramasAcadémicos.
4. El controlador ControllInscripciones recibe la petición y despliega la vista VistaEditarInscripcion.
5. El Usuario actualiza la información de la inscripción, agrega o elimina campos y la guarda.
6. El controlador ControllInscripciones recibe la petición de editar la inscripción, actualiza el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaEditarInscripción.
7. Se destruye la vista VistaEditarInscripción.

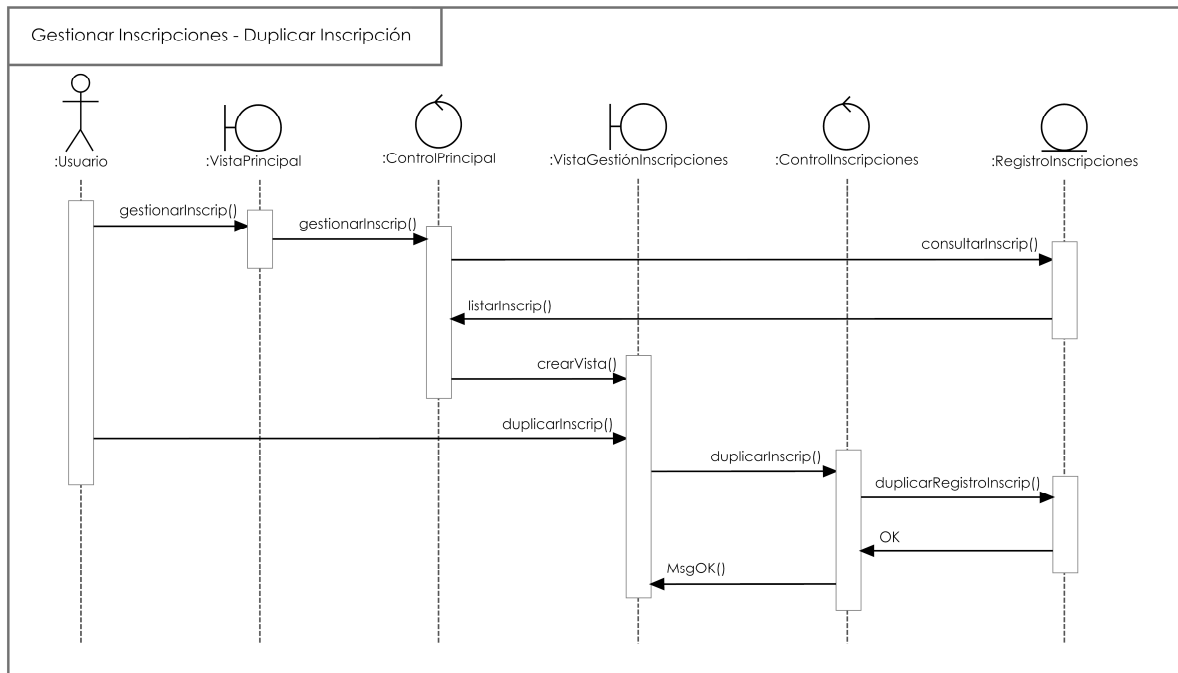
Figura 40. Diagrama de secuencia: Gestionar Inscripciones - Eliminar Inscripción



Fuente: Autores.

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita eliminar una inscripción en la vista VistaGestiónInscripciones.
4. El controlador ControlInscripciones recibe la petición de eliminar la inscripción, elimina el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaGestiónInscripciones.

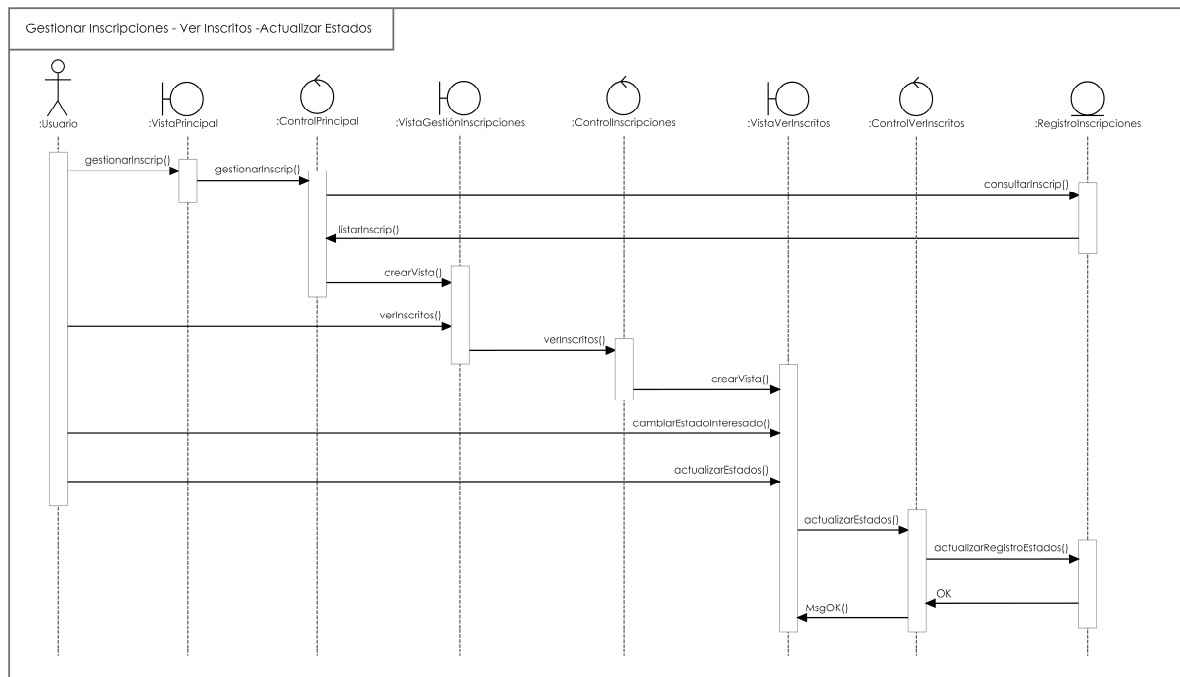
Figura 41. Diagrama de secuencia: Gestionar Inscripciones - Duplicar Inscripción



Fuente: Autores.

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita duplicar una inscripción en la vista VistaGestiónInscripciones.
4. El controlador ControlInscripciones recibe la petición de duplicar la inscripción, duplica el registro en base de datos y retorna un mensaje de Ok al usuario en la vista VistaGestiónInscripciones.

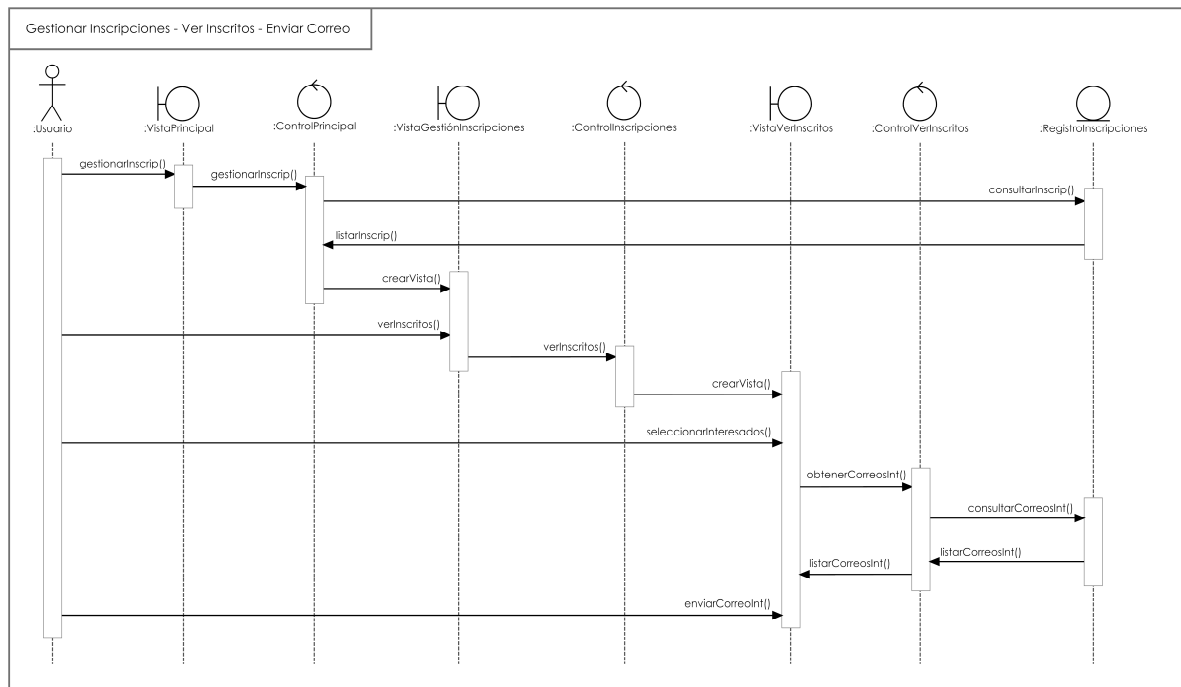
Figura 42. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Estados



Fuente: Autores.

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita ver los inscritos de una inscripción en la vista VistaGestiónInscripciones.
4. El controlador ControllInscripciones recibe la petición de ver los inscritos de la inscripción y despliega la vista VistaVerInscritos.
5. El Usuario cambia el estado de los interesados que va a inscribir y envía los datos.
6. El controlador ControllInscripciones recibe la petición, actualiza los registros de los interesados en la base de datos y retorna un mensaje de Ok al usuario en la vista VistaVerInscritos.

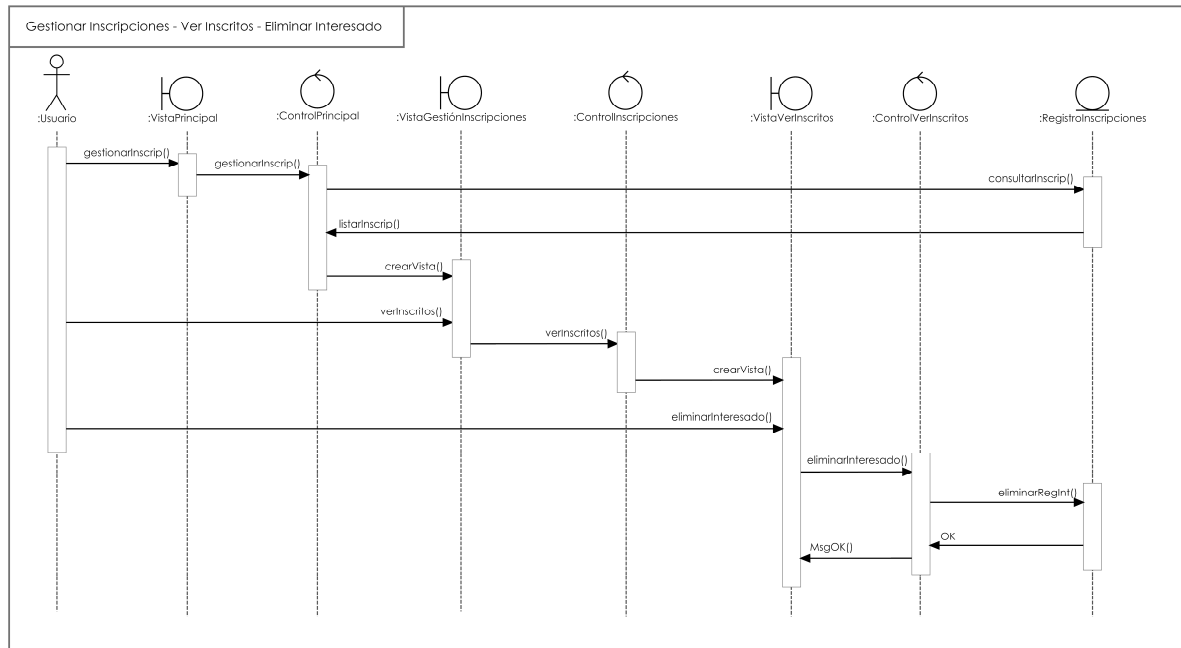
Figura 43. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Enviar Correo



Fuente: Autores.

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita ver los inscritos de una inscripción en la vista VistaGestiónInscripciones.
4. El controlador ControlInscripciones recibe la petición de ver los inscritos de la inscripción y despliega la vista VistaVerInscritos.
5. El Usuario selecciona los interesados a los cuales desea enviarles un correo.
6. El controlador ControlInscripciones recibe la petición, consulta los correos de los interesados en la base de datos y retorna la consulta a la vista VistaVerInscritos.
7. El Usuario envía el correo.

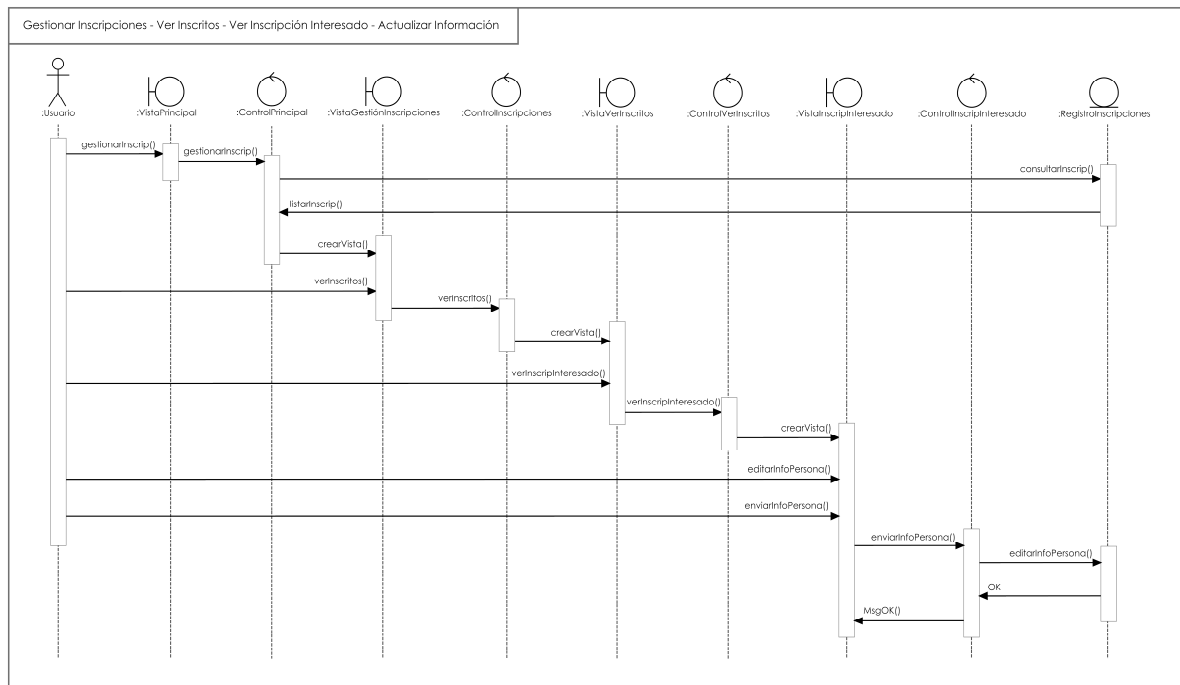
Figura 44. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Eliminar Interesado



Fuente: Autores.

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita ver los inscritos de una inscripción en la vista VistaGestiónInscripciones.
4. El controlador ControllInscripciones recibe la petición de ver los inscritos de la inscripción y despliega la vista VistaVerInscritos.
5. El Usuario solicita eliminar un interesado.
6. El controlador ControlVerInscritos recibe la petición, elimina el registro del interesado en la base de datos y retorna un mensaje de Ok al usuario en la vista VistaVerInscritos.

Figura 45. Diagrama de secuencia: Gestionar Inscripciones - Ver Inscritos - Actualizar Información Interesado

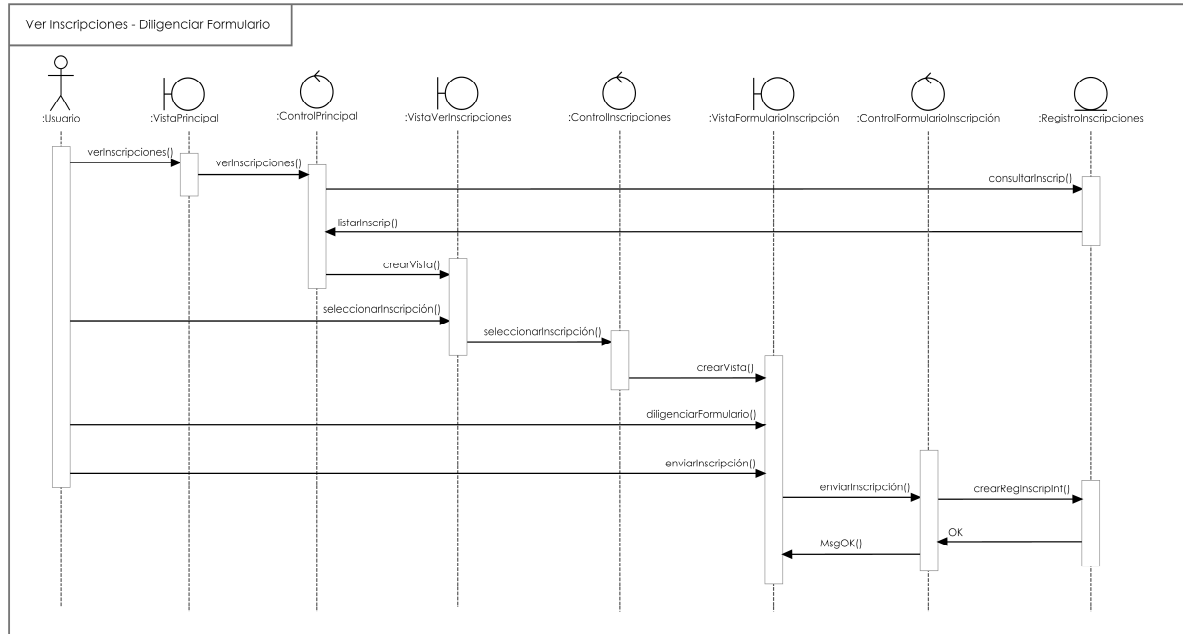


Fuente: Autores.

1. El Usuario solicita gestionar inscripciones en la vista VistaPrincipal.
2. El controlador ControlPrincipal recibe la petición, consulta las inscripciones en base de datos y despliega la vista VistaGestiónInscripciones.
3. El Usuario solicita ver los inscritos de una inscripción en la vista VistaGestiónInscripciones.
4. El controlador ControlInscripciones recibe la petición y despliega la vista VistaVerInscritos.
5. El Usuario solicita ver la inscripción de un interesado en la vista VistaVerInscritos.
6. El controlador ControlVerInscritos recibe la petición y despliega la vista VistaInscripInteresado.
7. El Usuario edita la información personal o los campos de la inscripción del interesado y envía los datos.

8. El controlador `ControllInscriInteresado` recibe la petición, actualiza el registro del interesado en la base de datos y retorna un mensaje de Ok al usuario en la vista `VistaInscriInteresado`.

Figura 46. Diagrama de secuencia: Ver Inscripciones - Diligenciar Formulario



Fuente: Autores.

1. El Usuario solicita ver las inscripciones abiertas en la vista `VistaPrincipal`.
2. El controlador `ControlPrincipal` recibe la petición, consulta las inscripciones en base de datos y despliega la vista `VistaVerInscripciones`.
3. El usuario selecciona la inscripción a la cual desea aplicar en la vista `VistaVerInscripciones`.
4. El controlador `ControllInscripciones` recibe la petición y despliega la vista `VistaFormularioInscripción`.
5. El usuario diligencia los datos de la inscripción y la envía.

6. El controlador ControlFormularioInscripción recibe la petición, crea el registro de la inscripción del interesado en la base de datos y retorna un mensaje de Ok al usuario en la vista VistaFormularioInscripción.

4.1.5 Implementación, Implantación y Pruebas Generales

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

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

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

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

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

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

4.2 MANTENIMIENTO Y ADMINISTRACIÓN

4.2.1 Actividades de Mantenimiento

Dentro de las funciones que se realizan en la administración y mantenimiento de los servicios de los portales web se encuentra la tarea de corregir errores, las cuales se denominan incidencias y que se corrigen durante la primera fase como integrantes del Grupo Calumet. Las siguientes son las incidencias que se realizaron paralelamente al desarrollo de este proyecto.

- **Asociar varios foros a un mismo código de asignatura.**

Funcionalidad:

Los foros en el Portal se pueden crear en cada asignatura de los planes de estudio. Al crear conversaciones, o participar, se envían correos a los matriculados en las correspondientes asignaturas.

Desarrollo del problema:

Inicialmente los foros de materias sólo podían crearse una sola vez, sin importar el área (programa) con el cual estuvieran relacionados. Se modificó esto permitiendo que un foro relacionado con una materia pueda existir en más de un área a la vez, pero manteniendo la restricción de que sólo puede existir una sola vez dentro de un área en particular. Además se cambió la base de datos “División” por “Diamante” como origen desde donde se listan las materias.

Problemática:

Permitir que más de un foro de asignatura se relacione con un mismo código de materia en diferentes áreas (programas), además consultar el código de las asignaturas no de BD división, sino de TB_MateriasN.

Descripción:

Se cambió la consulta que lista las materias en:

/eisi/Administrador/AdminListarMaterias.jsp

Se cambió la forma como se validan los foros en:

/eisi/Administrador/adminParametForosA.jsp

- **Cambiar textos de correos de alertas a la creación de conversaciones y participaciones en los foros.**

Funcionalidad:

El portal envía correos a los interesados cuando se crea una nueva conversación y cuando se hacen nuevas participaciones, de esta forma mantiene al tanto a los miembros de la comunidad en aquellos debates en los que han participado.

Desarrollo del problema:

Dentro de la funcionalidad de foros, se envía un correo a los usuarios que están relacionados con el mismo, avisándoles de las nuevas participaciones, pero el envío de este correo se realizaba al día siguiente. Se cambió esta funcionalidad y ahora se envía un correo en el momento en que se realiza una nueva participación, manteniendo al tanto de la conversación a quienes hagan parte de la misma. Además como los foros son abiertos y cualquier usuario puede participar, los usuarios que han hecho participaciones pero no están directamente relacionados con el foro reciben correo también.

Problemática:

Revisar en el correo de invitación, que llega al crear conversación y con los participantes del día anterior en los foros, a ingresar a foros, que debido a la reingeniería reciente ya no da el path de ingreso correcto.

Descripción:

Se modificó la funcionalidad del envío de correos al hacer participaciones en conversaciones, de forma que se envíe un correo al momento de hacer una participación a todos los involucrados en la misma y a todos los que hayan participado.

- **Arreglo solicitud de cambio de tutor.**

Funcionalidad:

El Portal de las Escuelas hace soporte a la gestión de trabajos de grado de las escuelas. Las solicitudes permiten cambiar autores, título, director... de los proyectos de grado.

Desarrollo del problema:

Dentro de las solicitudes que se pueden hacer en la plataforma de trabajos de grado, existe la posibilidad de realizar solicitud de cambio de tutor. Esta solicitud es recibida por el comité de trabajos de grado, que la puede aprobar, al momento de aprobarse, el sistema hace el cambio inmediato del tutor actual por el tutor seleccionado. Se arregló la consulta que consulta el IdUsuario del nuevo tutor, ya que no se tenía en cuenta el campo correcto.

Problemática:

Arreglar solicitud cambio tutor que está funcionando mal para el comité.

Descripción:

Se corrigió la consulta que trae la información del nuevo codirector o tutor, ya que no se tenía en cuenta el campo correcto que almacenaba su IdUsuario. Esto se hizo en:

/eisi/TrabajoGrado/Comite/EvaluarSolicitudes2.jsp

/eisi/TrabajoGrado/Comite/IEvaluarSolicitudes2.jsp

- **Perfil de los usuarios que publican en cartelera.**

Funcionalidad:

Dentro de las políticas del portal de las escuelas se encuentra la de hacer público el autor y su relación con la escuela de toda información que se suba a la comunidad o al público.

Desarrollo del problema:

Los usuarios tienen dentro de los servicios que les ofrece el portal, la posibilidad de realizar publicaciones en cartelera, la cual se puede ver cuando se ingresa al sitio, por lo cual es necesario que sea claro el nombre del usuario que realiza la publicación y su relación con la escuela. Por esta razón fue necesario relacionar a los usuarios que no tienen un perfil, con un perfil que ya existe cuyo nombre es "UsuarioPortal".

Problemática:

En cartelera los estudiantes y egresados salen con perfil null, pero esto se creó un perfil llamado "Usuario Portal".

Descripción:

Se actualizó la funcionalidad al listar las publicaciones en cartelera, para que los usuarios sin perfil en el portal, tengan junto a su nombre en la publicación su relación con la escuela, en este caso se utilizó un perfil que ya existe que es "PE19" y que corresponde a "UsuarioPortal". Se modificó el código en:

`/eisi/web/Eventos/EventosNoticiasP.jsp`

- **Actualización de HV de en Alianza.**

Funcionalidad:

Alianza es la bolsa de oportunidades laborales y de prácticas empresariales del Portal. Para poder aplicar a estas oportunidades y prácticas los usuarios deben subir su HV en un formato predeterminado por la escuela y en PDF.

Desarrollo del problema:

Cuando una persona aplica a una oportunidad laboral en Alianza, sube su hoja de vida en formato PDF y el sistema renombra esta con el IdUsuario de la persona que la sube (Ej: HOJA_DE_VIDA_U1524.pdf). De esta forma se asegura que los nombres sean únicos. Ocurría que cuando una persona actualizaba su hoja de vida, para el administrador de la oportunidad laboral no era evidente el cambio, ya que si ya había consultado la hoja de vida de esta persona poco tiempo antes, la misma permanecía en caché, por lo cual debía actualizar constantemente el navegador. Se agregó la fecha y la hora en el nombre del archivo, de esta manera con cada actualización del usuario, se crea una hoja de vida con un nombre completamente diferente, y no ocurren problemas de caché para quien administra las ofertas laborales (Ej: HOJA_DE_VIDA_U1524_2013-04-22-17-55-38.pdf).

Problemática:

Al actualizar la HV en las propuestas de Alianza, no sube el archivo siempre.

Descripción:

Se agregó el Bean FechaHora y se utilizaron sus métodos para capturar la fecha y hora actual del sistema para agregarse al nombre de los archivos de hojas de vida en: /eisi/Alianza/listaImágenes.jsp

- **Al subir archivo de respuesta al Actividad del Aula, esta colapsa.**

Funcionalidad:

El Aula Virtual programa una serie de actividades evaluables a los que pueden responder los subgrupos de clase subiendo un archivo, que será revisado por el docente y valorado.

Desarrollo del problema:

Dentro del Aula Virtual, un docente puede programar actividades, las cuales los estudiantes deben responder subiendo un archivo, estas actividades pueden ser evaluadas individualmente o por grupo, y el docente puede responderles a los estudiantes con un comentario, el cual es visible para estos en la interfaz de Aula Virtual. Al momento de listar las actividades, es necesario capturar los comentarios de las mismas realizados por el profesor, y para esto es necesario distinguir que tipo de nota recibe, ya que los comentarios se almacenan en tablas distintas.

Problemática:

El aula colapsa al subir archivos como respuesta de una actividad por parte de los estudiantes.

Descripción:

Se modificaron las consultas que traen los comentarios de las actividades, diferenciando de las actividades que reciben nota individual de las que reciben nota grupal, ya que sólo se capturaban los comentarios de las actividades que

reciben nota individual, lo cual interfiere con el funcionamiento del sitio. Se modificó el código en:

/eisi/Aula/Clase/TodosActivoListarActOVA2.jsp

- **Correo de invitación egresado: texto debe corregirse.**

Funcionalidad:

El Portal debe de promover el registro de los egresados de años anteriores a su implantación en cada escuela. Para esto se puede desde la página de inicio enviar un texto al correo de los egresados motivándolos a realizar su registro, y a participar en el Portal.

Desarrollo del problema:

Al ingresar a alguno de los portales, los usuarios encuentran el botón “Invite a un Egresado”, a través del cual ingresan a un formulario donde pueden enviarle un mensaje a algún egresado de la escuela y que desean se registre en el portal. Se modificó el texto informativo.

Problemática:

Arreglar correo de invitación a egresados. Ahora el acceso a registrarse es por otro lado.

Descripción:

Se modificó el texto que se enviaba en el correo al invitar a un egresado en:

/eisi/inicioinvegre.jsp

4.2.2 Actividades de Soporte a Usuarios

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

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

4.2.3 Actividades de Administración

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

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

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

5 PRUEBAS DEL SISTEMA

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

5.1 PRUEBAS DE VERIFICACIÓN

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

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

Las siguientes tablas describen las pruebas de cada caso de uso de los servicios que fueron desarrollados.

5.1.1 Pruebas por componente

Esta prueba se realizó para cada caso de uso de cada servicio desarrollado y descrito anteriormente:

5.1.1.1 Servicio de Ayudas

Tabla 10. Pruebas Realizadas: Gestión Ayudas.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Ayudas	Al acceder con perfil de administrador al servicio de gestionar ayudas, lista los menús de ayudas y las ayudas creadas en el portal de ayudas de las escuelas.	✓
Crear Ayuda	Al crear una ayuda, valida que se diligencien todos los campos obligatorios y almacena correctamente.	✓
Editar Ayuda	Cuando se edita una ayuda, valida los campos obligatorios y almacena correctamente.	✓
Eliminar Ayuda	Elimina correctamente una ayuda y los registros que dependan de esta en la base de datos	✓
Crear Menú Ayuda	Cuando se crea un nuevo menú ayuda, valida que se diligencien todos los campos obligatorios.	✓
Editar Menú Ayuda	Cuando se edita un menú ayuda, valida los campos obligatorios.	✓

Eliminar Menú Ayuda	Elimina el menú ayuda y las ayudas que dependan de este en la base de datos	✓
---------------------	---	---

Fuente: Autores.

Tabla 11. Pruebas realizadas: Gestión Preguntas Frecuentes

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Preguntas Frecuentes	Al acceder con perfil de administrador al servicio, lista las preguntas frecuentes y las ayudas relacionadas con esta pregunta.	✓
Crear Pregunta Frecuente	Al crear una pregunta frecuente, valida que se diligencien todos los campos obligatorios y almacena correctamente.	✓
Editar Pregunta Frecuente	Cuando se edita una pregunta frecuente, valida los campos obligatorios y almacena correctamente.	✓
Eliminar Pregunta Frecuente	Elimina correctamente una pregunta frecuente de la base de datos.	✓
Seleccionar Ayudas	Al seleccionar las ayudas, las relaciona correctamente con la pregunta frecuente en la base de datos.	✓

Fuente: Autores.

Tabla 12. Pruebas realizadas: Sugerencias Ayudas

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Sugerencias	Al acceder con perfil de administrador al servicio de gestionar sugerencias, lista las sugerencias hechas por usuarios en el portal de ayudas.	✓
Ver Sugerencia	Al seleccionar una sugerencia, muestra correctamente la información diligenciada por el usuario en el portal de ayudas.	✓
Eliminar Sugerencia	Elimina correctamente la sugerencia de la base de datos.	✓

Fuente: Autores.

Tabla 13. Pruebas realizadas: Portal Ayudas

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Consultar Ayudas	Al acceder al portal de ayudas de las escuelas lista el correctamente los menús.	✓
Ver Temas Ayudas	Lista las ayudas organizadas por temas de servicios del portal de primer y segundo nivel.	✓
Seleccionar Menú Ayuda	Al seleccionar menú de ayuda de primer o segundo nivel, lista las ayudas relacionadas con ese menú.	✓

Ver Preguntas Frecuentes	Cuando selecciona Ver Preguntas Frecuentes, lista las ayudas organizadas por preguntas frecuentes y por tipos de usuario.	✓
Seleccionar Tipo Usuario	Al seleccionar el tipo de usuario lista correctamente las preguntas frecuentes y las ayudas correspondientes a esa pregunta.	✓
Ver Ayuda	Despliega la ayuda con su respectivo contenido, la cual corresponde al servicio escogido o a la pregunta frecuente.	✓
Enviar Sugerencia	Se almacenan correctamente las sugerencias en el portal de ayudas.	✓
Ver Información Grupo Calumet	Despliega información relacionada con el grupo de desarrollo Calumet.	✓

Fuente: Autores.

5.1.1.2 Servicio de Cronogramas Académicos

Tabla 14. Pruebas realizadas: Cronogramas Académicos

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Cronogramas Académicos	Al acceder al servicio de Cronogramas Académicos, lista correctamente los cronogramas creados por el usuario.	✓

Crear Cronograma Académico	Al crear un cronograma académico, valida que se diligencien todos los campos obligatorios y almacena correctamente.	✓
Editar Cronograma Académico	Cuando se edita un cronograma académico, valida los campos obligatorios y actualiza correctamente.	✓
Eliminar Cronograma Académico	Elimina un cronograma académico y los registros que dependan de este en la base de datos	✓
Ver Cronograma Académico	Al acceder a un cronograma académico lista correctamente las materias y los horarios que han sido programados para esta	✓
Programar Materia	Al programar una materia, lista correctamente las fechas que ya han sido programadas, almacena correctamente la información ingresada.	✓
Seleccionar Docente	Lista correctamente los docentes y actualiza el horario del docente al momento de programar una fecha, así mismo valida correctamente cruces de horario de este docente con otras materias.	✓
Programar Fecha	Al seleccionar una fecha, valida correctamente cruces.	✓
Escoger Salón	Al seleccionar un salón, valida correctamente cruces de este salón con	✓

	otras materias.	
Eliminar Fecha	Elimina correctamente una fecha del horario de una materia.	✓
Ver Modo Calendario	Lista correctamente las fechas programadas para un cronograma académico.	✓
Ver Calendario Todos	Lista correctamente las fechas programadas para todos los cronogramas académicos activos	✓

Fuente: Autores.

5.1.1.3 Servicio de Inscripciones

Tabla 15. Pruebas realizadas: Inscripciones.

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Gestionar Inscripciones	Al acceder al servicio de Inscripciones, lista correctamente las inscripciones creadas por el usuario.	✓
Crear Inscripción	Al crear una inscripción, valida que se diligencien todos los campos obligatorios y almacena correctamente.	✓
Editar Inscripción	Editar una inscripción de programa o actividad que haya sido creada anteriormente.	✓

Eliminar Inscripción	Elimina una inscripción y los registros de interesados inscritos a esta en la base de datos	✓
Gestionar Campo	Al seleccionar un campo agregado, muestra correctamente las opciones de subir campo, bajar campo, editar campo y eliminar campo.	✓
Agregar Campo	Agrega campos de tipo texto, párrafo y selección múltiple correctamente.	✓
Editar Campo	Edita correctamente la información de los campos agregados tales como título, subtítulo y opciones en el caso de selección múltiple.	✓
Eliminar Campo	Elimina correctamente campos agregados en la interfaz.	✓
Ver Inscritos	Lista correctamente las personas inscritas, discriminándolos según su estado.	✓
Actualizar Estado	Actualiza los cambios de estado de uno o varios interesados.	✓
Eliminar Persona	Elimina correctamente un interesado de la inscripción.	✓
Ver Información Persona	Lista correctamente la información diligenciada por un interesado.	✓
Actualizar	Muestra correctamente las opciones de	✓

Información	actualizar información.	
Actualizar Datos Personales	Actualiza correctamente los datos personales de un interesado.	✓
Actualizar Información Inscripción	Actualiza correctamente los campos con las respuestas diligenciadas por el interesado.	✓
Inscribir Persona	Cambia el estado del interesado a “Inscrito” correctamente.	✓

Fuente: Autores.

Tabla 16. Pruebas realizadas: Ver Inscripciones .

CASO DE USO	PRUEBA REALIZADA	RESULTADO
Ver Inscripciones Abiertas	Al acceder al servicio de Inscripciones, lista correctamente las inscripciones activas públicas y privadas.	✓
Seleccionar Inscripción	Muestra correctamente las opciones que tiene un usuario que desea inscribirse.	✓
Seleccionar Inscripción Pública	Valida correctamente que la inscripción esa pública y no necesita sesión por parte del usuario.	✓
Seleccionar Inscripción Privada	Valida correctamente que la inscripción es privada y por lo tanto el usuario se debe encontrar logueado.	✓

Diligenciar Formulario de Inscripción	Autocompleta correctamente los datos personales del interesado cuando ya se encuentra logueado o cuando ingresa la cédula. Además valida correctamente los campos obligatorios de la inscripción.	✓
---	---	---

Fuente: Autores.

5.2 PRUEBAS DE INTEGRACIÓN

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

5.3 PRUEBAS DE VALIDACIÓN

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

6 CONCLUSIONES

- La implementación del Portal de Ayudas se consolida como un soporte significativo para las escuelas y sus usuarios, solucionando las necesidades e inquietudes de los mismos a través de información clara y eficaz, la cual fomentará un mejor uso de los servicios del portal.
- Con el desarrollo del servicio de Cronogramas Académicos se optimizó la programación de los calendarios académicos de posgrado para las diferentes cohortes en distintas ciudades, previniendo de cruces en cuanto a la disponibilidad de los recursos como docentes y salones al momento de programar una materia.
- El servicio de Inscripciones ofrece a las escuelas la posibilidad de crear formularios personalizados de manera sencilla, sin la necesidad de acudir a servicios externos. A través de estas se podrán dar a conocer los programas académicos y eventos de las escuelas, y los administrativos contarán con una herramienta sencilla para contactar a todos aquellos que estén interesados en participar.
- Para que haya continuidad en el trabajo realizado en el grupo Calumet, es necesario orientar a los nuevos integrantes en la familiarización con las herramientas utilizadas para el mantenimiento de los portales, así como en la dinámica de trabajo que hay entre los miembros.
- La realización de copias de seguridad de base de datos y de los portales es una práctica indispensable. Dado que los servidores utilizados no cuentan con sistemas RAID y en caso de fallo sería imposible recuperar la información alojada en el disco duro.

- 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.
- El modelo de ciclo de vida, Prototipo Evolutivo, es ideal para aquellos proyectos cuyos requerimientos pueden cambiar durante el desarrollo del mismo.
- El uso de la librería de JavaScript jQuery, se presenta como una herramienta muy útil para el desarrollo web, ya que ofrece una serie de funcionalidades con las cuales se logran excelentes resultados en menos tiempo y espacio.

7 RECOMENDACIONES

- Se sugiere mejorar el servicio ya existente de encuestas, teniendo como base el servicio creado de inscripciones, para enfocarlo a la creación de formularios en línea que puedan ser accedidos desde cualquier sitio y utilizados por cualquier miembro de la comunidad.
- La extensión del servicio de inscripciones, que actualmente permite crear inscripciones por escuela, para que permita crear inscripciones para toda la comunidad.
- Ampliar las opciones de tipo de pregunta existentes al crear una inscripción, para permitir nuevos tipos de preguntas así como la posibilidad de preguntas que dependan de otras.
- Crear un canal de comunicación entre los usuarios inscritos a programas o eventos y el administrador de la inscripción.
- Continuar con el desarrollo del portal de ayudas, el cual está en su primera versión.

BIBLIOGRAFÍA

1. JOHNSON, James. Bases de datos: Modelos lenguajes y diseño. 1ª ed. Oxford, 2000. Presenta temas de teoría de bases de datos: modelos y métodos de acceso, administración, diseño de aplicaciones.
2. PIATTINI, Mario, CALVO-MANZANO, José A., CERVERA, Joaquín, FERNANDEZ, Luis. Análisis y diseño detallado de Aplicaciones Informáticas de Gestión. Alfaomega, 2000. Este libro contiene información sobre técnicas para el buen modelado de aplicaciones informáticas.
3. SCHMULLER, JOSEPH. Aprendiendo UML en 24 horas. 1ª ed. México: Alhambra Mexicana S.A, 2000. En este libro se encuentra una guía muy práctica que permite conocer y entender sobre UML.
4. STALLINGS, William. Sistemas Operativos: Aspectos Internos y Principios de Diseño. Madrid: Pearson Prentice Hall, 2005. Este libro se ocupa de los conceptos completos de las características de los sistemas operativos.
5. STUMPF, Robert, TEAGUE, Lavette. Object-Oriented Systems Analysis and Design with UML. Prentice Hall. 2004. Este libro introduce los conceptos y métodos del análisis y diseño de sistemas orientados a objetos.
6. WEITZENFELD, Alfredo. Ingeniería de Software Orientada a Objetos con UML, JAVA e Internet. Thomson International, 2003. En este libro se encuentra información sobre desarrollo de software orientado a objetos.
7. <http://g701giadar.wikispaces.com/Arquitctura+Cliente+Servidor>. Información acerca de la Arquitectura Cliente Servidor.

8. <http://manuales.dgsca.unam.mx/jsp>. Manuales básicos de JSP.
9. <http://www.aprender-informatica.com/>. (n.d). Tutoriales del lenguaje de programación JavaScript.
10. <http://www.htmlquick.com/es/tutorials.html>, tutoriales de cómo programar en html de gran ayuda en detalles básicos y sencillos.
11. <http://www.mysql.com>. Sitio oficial de MySQL, que permite participar en capacitaciones en línea y tener las ultimas noticias acerca del producto para su mayor aprovechamiento.
12. <http://www.programacion.com/java/tutorial>. En este sitio se encuentran una gran variedad de tutoriales relacionados con: Los APIS, las herramientas y los servicios que proporciona los Servlets y las Java Server Pages (JSP), los tópicos necesarios para la programación de acceso a bases de datos en Java.
13. http://www.programacionfacil.com/programacion:manual_java_jsp, manual de programación para Java Jsp (Java Server Pages).
14. <http://www.youblisher.com/p/155105-Aprendiendo-UML/>. Manual de conceptos básicos sobre UML.

ANEXOS

Anexo A. MANUAL DE USUARIO

El manual de usuario se presenta como una guía para que el usuario pueda acceder a los servicios creados, donde se explica su funcionamiento paso a paso. Se puede encontrar en el Portal de Ayudas en todos los portales que administra el grupo Calumet a través del siguiente enlace:

<http://cormoran.uis.edu.co/eisi/Ayudas/>

Figura 47. Portal de Ayudas



Fuente: Autores.