

**SISTEMA DE INFORMACIÓN INTRANET PARA EL APOYO DE LA
GESTION NORMATIVA EN INSTITUCIONES DE EDUCACIÓN
SUPERIOR**

**JUAN PABLO ROJAS CÁRDENAS
MARTHA LIGIA SUÁREZ ROJAS**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA**

2004

**SISTEMA DE INFORMACIÓN INTRANET PARA EL APOYO DE LA
GESTION NORMATIVA EN INSTITUCIONES DE EDUCACIÓN
SUPERIOR**

**JUAN PABLO ROJAS CÁRDENAS
MARTHA LIGIA SUÁREZ ROJAS**

**Proyecto de grado para optar al título de
Ingenieros de Sistemas**

Director

Ing. LUIS CARLOS GÓMEZ FLÓREZ

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS
BUCARAMANGA**

2004

DEDICATORIA

Este trabajo de grado está dedicado a nuestro hijo Juan Diego por ser fuente constante de alegrías y motivaciones y a nuestros padres por su confianza y apoyo incondicional

Juan Pablo y Martha Ligia

AGRADECIMIENTOS

Gracias a todos los que nos ayudaron a conseguir este triunfo.

A nuestras familias, en especial a nuestros padres, a Paula y a Yesid, ya que con su paciencia, entrega y apoyo incondicional hicieron posible la realización de este trabajo de grado.

Al Ing. Luis Carlos Gómez Flórez, por sus valiosos aportes, esmero y dedicación. Fue parte fundamental en el desarrollo del proyecto.

Al grupo de Investigación de Sistemas y Tecnologías de Información –STI , por su colaboración y apoyo hacia nuevos retos.

A nuestros compañeros de Universidad, por el conocimiento compartido, y amistad brindada en todo momento.

A Carlos “Abrego”, Daveiva Barrera y Fabian Contreras, por brindarnos su gran amistad.

CONTENIDO

PREFACIO	1
PARTE I – ASPECTOS GENERALES	5
CAPITULO 1	6
1.1 INTRODUCCIÓN	6
1.2 ANTECEDENTES Y DESCRIPCIÓN DEL PROBLEMA	6
1.2.1. Investigación del Manejo de la Información Normativa en algunas Instituciones de Educación Superior	7
1.2.2. Proceso actual para la Gestión Normativa en IES	10
1.2.3. Proceso propuesto para la Gestión Normativa en IES	12
1.3 OBJETIVOS	14
1.3.1 Objetivo General	14
1.3.2 Objetivos Específicos	14
1.4 IMPACTO	16
1.5 RESUMEN CAPITULO 1	18
CAPITULO 2: MARCO TEORICO	19
2.1 INTRODUCCIÓN	19
2.2 TECNOLOGIAS INTERNET	19
2.2.1. Internet	20
2.2.2. Extranet	20
2.2.3. Intranet	21
2.3 TECNOLOGÍAS DISPONIBLES PARA EL DESARROLLO	22
2.3.1. Programación del lado del cliente	24
2.3.2. Programación del lado del servidor	26

2.4	ARQUITECTURA DE SOFTWARE	30
2.4.1.	Cliente – Servidor	32
2.4.2.	Múltiples Capas	33
2.5	ACTOS ADMINISTRATIVOS	37
2.5.1.	El objeto o contenido de los actos administrativos	38
2.5.2.	Las Resoluciones	39
2.6	RESUMEN CAPITULO 2	40
	RESUMEN PARTE I	41
3.	NOTAS BIBLIOGRÁFICAS	42
	PARTE II – FASE DE INICIO	45
	CAPITULO 3	46
4.	RECOPIACIÓN DE REQUISITOS E IDENTIFICACIÓN DE CASOS DE USO	46
4.1.	INTRODUCCIÓN	46
4.2	LISTA DE CARACTERÍSTICAS	46
4.3.	MODELO DEL DOMINIO	48
4.4.	ACTORES	49
4.5.	MODELO DE CASOS DE USO	50
4.5.1.	Casos de Uso Comunidad Universitaria	50
4.5.2.	Casos de Uso Administrador	51
4.5.3.	Casos de Uso Secretaría del Ente	54
4.5.4.	Casos de Uso Autores de Proyecto	55
4.6.	LISTA INICIAL DE RIESGOS	56
4.6.1.	Riesgos del tamaño del producto	56
4.6.2.	Riesgos del Impacto del negocio	57
4.6.3.	Riesgos relacionados con el cliente	58
4.6.4.	Riesgos del proceso	58
4.6.5.	Riesgos del entorno de desarrollo	58
4.7.	RESUMEN CAPITULO 3	58

CAPITULO 4	60
5. ANÁLISIS	60
5.1 INTRODUCCION	60
5.2 MODELO DE ANÁLISIS	60
5.2.1. Diagrama de Colaboración para la realización del Caso de Uso Búsqueda	60
5.2.2. Diagrama de Colaboración para la realización del Caso de Uso Publicar Acto Administrativo	61
5.3 ARQUITECTURA CANDIDATA	62
5.3.1. Identificación paquetes de Análisis	62
5.3.2. Requisitos Especiales Comunes	63
5.4 RESUMEN CAPITULO 4	64
RESUMEN PARTE II	65
6. NOTAS BIBLIOGRÁFICAS	68
PARTE III – FASE DE ELABORACIÓN	69
CAPITULO 5	71
7. RECOPIACIÓN DE REQUISITOS FALTANTES, Y NUEVA VERSIÓN DE LOS MODELOS	71
7.1 INTRODUCCION	71
7.2 LISTADO DE CARACTERÍSTICAS RESTANTES	71
7.3 MODELO DEL DOMINIO	72
7.4 NUEVA VERSIÓN MODELOS DE CASOS DE USO	75
7.4.1. Modelo de Casos de Uso : Gestión de Configuración	75
7.4.2. Modelo de Casos de Uso: Gestión de Publicación	75
7.4.3. Modelo de Casos de Uso: Gestión de Proyectos	76
7.4.4. Modelo de Casos de Uso: Gestión de Consulta	76
7.5 RESUMEN CAPITULO 5	77

CAPITULO 6	78
8. ANÁLISIS	78
8.1 MODELO DE ANÁLISIS	78
8.1.1. análisis de la arquitectura	78
8.1.2. Análisis de Casos de Uso	81
8.2 RESUMEN CAPITULO 6	85
CAPITULO 7	86
9. DISEÑO	86
9.1 MODELO DE DISEÑO	86
9.1.1. Diseño de la arquitectura	86
9.1.2. Diseño de subsistemas	90
9.2 MODELO DE IMPLEMENTACIÓN	97
9.2.1. software del sistema	97
9.3 RESUMEN CAPITULO 7	98
RESUMEN PARTE III	99
10. NOTAS BIBLIOGRAFICAS	102
PARTE IV – FASE DE CONSTRUCCIÓN	103
CAPITULO 8	104
11. SUBSISTEMAS ESPECÍFICOS DE APLICACIÓN	104
11.1. GESTIÓN DE CONFIGURACIÓN	104
11.1.1. Implementación	105
11.1.2. Pruebas	106
11.2. GESTIÓN DE PUBLICACIÓN	107
11.2.1. Implementación	107
11.2.2. Pruebas	109
11.3. GESTIÓN DE PROYECTOS	112
11.3.1. Implementación	112

11.3.2. Pruebas	114
11.4. GESTIÓN DE CONSULTA	116
11.4.1. Implementación	116
11.2.2. Pruebas	118
11.5. RESUMEN CAPITULO 8	120
CAPITULO 9	121
12. SUBSISTEMAS GENERALES DE APLICACIÓN	121
12.1. GESTIÓN DE SESION	121
12.1.1. Implementación	122
12.1.2. Pruebas	123
12.2. GESTIÓN DE INTERFAZ	125
12.2.1. Implementación	126
12.2.2. Pruebas	128
12.3. PLANIFICACIÓN DE LA FASE DE TRANSICIÓN	129
12.4. RESUMEN CAPITULO 9	130
13. ELABORACIÓN DE MANUALES DEL SISTEMA	131
RESUMEN PARTE IV	131
14. NOTAS BIBLIOGRAFICAS	133
PARTE V – CONCLUYENDO EL TRABAJO DE GRADO	134
CAPITULO 10	134
15. REFLEXIONES FINALES	134
15.1. INTRODUCCIÓN	134
15.2. NARRANDO LA EXPERIENCIA	134
15.3. CONCLUSIONES	137
15.4. RECOMENDACIONES	138
BIBLIOGRAFÍA	140

LISTA DE TABLAS

PARTE I

Tabla 1.1. Estado actual del área normativa en algunas IES de Bucaramanga	8
Tabla 1.2. Proceso actual de gestión normativa en IES	11
Tabla 1.3. Proceso propuesto para el apoyo de la gestión normativa en IES	14
Tabla 1.4. Lenguajes del lado del cliente	25
Tabla 1.5. Lenguajes del lado del servidor	28
Tabla 1.6. Comparación de lenguajes del lado del servidor	28
Tabla 1.7. Lineamientos básicos para diseñar y desarrollar Interfaces	36

PARTE II

Tabla 2.1. Características del sistema	47
Tabla 2.2. Actores	49
Tabla 2.3. Casos de Uso Comunidad Universitaria	50
Tabla 2.4. Casos de Uso Administrador	52
Tabla 2.5.. Casos de Uso Secretaría de Ente	54
Tabla 2.6. Casos de Uso Autor Proyectos AA	55
Tabla 2.7. Riesgos del Tamaño del Producto	56
Tabla 2.8. Riesgos del Impacto en el Negocio.	57
Tabla 2.9. Riesgos Relacionados con el Cliente.	58
Tabla 2.10. Riesgos del Proceso.	58

Tabla 2.11	Riesgos del Entorno de Desarrollo.	58
Tabla 2.12	Casos de Uso identificados durante la fase de Inicio	65
Tabla 2.13.	Diagrama: Vista de la arquitectura contenidas en la Fase de Inicio.	67

PARTE III

Tabla 3.1.	Características restantes del sistema.	72
Tabla 3.2.	Entidades del modelo del dominio	72
Tabla 3.3.	Modelo de casos de uso: Gestión de Configuración	75
Tabla 3.4.	Modelo de Casos de Uso: Gestión de Publicación	75
Tabla 3.5.	Modelo de Casos de Uso: Gestión de Proyectos	76
Tabla 3.6.	Modelo de Casos de Uso: Gestión de Consulta	76
Tabla 3.7.	Paquete de Análisis: Gestión de Configuración	79
Tabla 3.8.	Paquete de Análisis: Gestión de Publicación	79
Tabla 3.9.	Paquete de Análisis: Gestión de Proyectos A.A.	80
Tabla 3.10.	Paquete de Análisis: Gestión Consulta.	80
Tabla 3.11.	Casos de uso significativos del paquete de análisis: Gestión de Configuración	82
Tabla 3.12.	Casos de uso significativos del paquete de análisis: Gestión de Publicación	83
Tabla 3.13.	Casos de uso significativos del paquete de análisis: Gestión de Proyectos	83
Tabla 3.14.	Casos de uso significativos del paquete de análisis: Gestión de Consulta	89
Tabla 3.15.	Descripción de Subsistemas de Diseño.	91
Tabla 3.16	Casos de Uso al final de la fase de Elaboración	99

Tabla 3.17. Diagramas de la vista de la arquitectura contenidas en esta parte	101
---	-----

PARTE IV

Tabla 4.1 . Código PHP para la creación de los Entes	105
Tabla 4.2. Prueba para la creación de los entes	107
Tabla 4.3. Código PHP que permite la publicación de un proyecto	108
Tabla 4.4. Prueba para la aprobación de un proyecto y publicación del mismo como acto administrativo	111
Tabla 4.5. Código PHP que permite modificar un proyecto	113
Tabla 4.6. . Prueba para modificar un proyecto y a la vez enviarlo a estudio	115
Tabla 4.7 . Código PHP que permite consultar un Proyecto de Acto Administrativo	117
Tabla 4.8. . Prueba para buscar un Proyecto de Acto Administrativo	119
Tabla 4.9 . Código PHP que permite validar el ingreso de un usuario al sistema con sus respectivos permisos	122
Tabla 4.10 . Prueba para validar el ingreso de un usuario al Sistema	124
Tabla 4.11 . Código PHP que permite validar el ingreso de un ente al sistema	128
Tabla 4.12. Prueba para validar el ingreso de un Ente en el sistema	129
Tabla 4.13. Resumen de Casos de Uso	131
Tabla 4.14. Equivalencia entre los subsistemas hallados en las fases de elaboración y construcción	132

LISTA DE FIGURAS

PARTE I

Figura 1.1. Figura enriquecida del proceso actual de gestión normativa en IES	10
Figura 1.2. Proceso actual de gestión normativa en IES	11
Figura 1.3. Figura enriquecida del proceso propuesto para apoyar la gestión normativa en IES	12
Figura 1.4. Proceso propuesto para apoyar la gestión normativa en IES	13
Figura 1.5. Alcance de Intranet, Extranet e Internet.	19
Figura 1.6. Programación en el lado del cliente	22
Figura 1.7. Programación en el lado del Servidor	23
Figura 1.8. Esquema mixto. Programación en el cliente y en el servidor.	23
Figura 1.8. Arquitectura Cliente – Servidor	32
Figura 1.9. Arquitectura de Tres capas. Caso mas general de arquitectura multicapas	33

PARTE II

Figura 2.1. Modelo del dominio	48
Figura 2.2. Casos de uso de la comunidad universitaria	50

Figura 2.3. Casos de uso del Administrador	52
Figura 2.4. Casos de Uso Secretaría del Ente	54
Figura 2.5. Casos de uso Autor Proyectos A.A.	55
Figura 2.6. Diagrama de Colaboración: Consultar Actos Administrativos.	61
Figura 2.7.. Diagrama de Colaboración: Publicación de Actos Administrativos.	61
Figura 2. 8. Paquete de Análisis: Gestión Crear Usuario	62
Figura 2.9. Paquete de Análisis: Configuración	62
Figura 2.10. Paquete de Análisis: Publicación	63
Figura 2.11. Paquete de Análisis: Gestión de Consulta.	63

PARTE III

Figura 3.1. Modelo del dominio	74
Figura 3.2. Modelo de Casos de Uso: Gestión de Configuración	75
Figura 3.3. Modelo de Casos de Uso: Gestión de Publicación	75
Figura 3.4. Modelo de Casos de Uso: Gestión de Proyectos	76
Figura 3.5. Modelo de Casos de Uso: Gestión de Consulta	76
Figura 3.6. Paquete: Gestión de Configuración	79
Figura 3.7. Paquete: Gestión de Publicación	79
Figura 3.8. Paquete: Gestión de Proyectos A.A.	80
Figura 3.9. Paquete: Gestión de consulta	80
Figura 3.10. Diagrama de Colaboración: CU Crear Ente	82

Figura 3.11. Diagrama de Colaboración: CU Crear tipo Acto Administrativo	82
Figura 3.12. Diagrama de Colaboración: CU Crear ID de la Intranet	82
Figura 3.13. Diagrama de Colaboración: CU Publicar Acto Administrativo	83
Figura 3.14. Diagrama de Colaboración: CU Elaborar Proyecto	83
Figura 3.15. Diagrama de Colaboración: CU Consultar Acto Administrativo	84
Figura 3.16. Diagrama de despliegue del sistema SIGNO	88
Figura 3.17. Subsistemas de diseño	91
Figura 3.18. Diagrama de Secuencia de las interacciones entre subsistemas.	94
Figura 3.19. Diseño General de la interfaz del sistema SIGNO.	95
Figura 3.20. Diseño General de la interfaz del Sistema de Información generado	96

PARTE IV

Figura 4.1. Interfaz que permite la inclusión de la información de cada Ente	106
Figura 4.2 . Interfaz que permite la publicación de un proyecto	109
Figura 4.3 . Interfaz que permite Modificar un proyecto	113
Figura 4.4. . Interfaz que permite consultar un Proyecto de Acto Administrativo	118
Figura 4.5 . Interfaz que permite ingresar al sistema	123
Figura 4.6 . Interfaz que permite Ingresar la información del ente	128

LISTA DE ANEXOS

Anexo A. El Proceso Unificado de Desarrollo de Software y UML	142
Anexo B. Pruebas del Sistema	186

RESUMEN

TITULO:

SISTEMA DE INFORMACIÓN INTRANET PARA EL APOYO DE LA GESTIÓN NORMATIVA EN INSTITUCIONES DE EDUCACIÓN SUPERIOR *

AUTORES:

SUÁREZ ROJAS, Martha Ligia.

ROJAS CÁRDENAS, Juan Pablo **

PALABRAS CLAVES:

Acto Administrativo, Intranet, Educación Superior, Gestión Normativa, Proceso Unificado.

DESCRIPCIÓN:

Este trabajo de grado expone una propuesta de investigación alrededor del proceso de gestión normativa en Instituciones de Educación Superior y su necesidad de cambio; propuesta que se plasmó con la herramienta software SIGNO 1.0, Sistema de Información Intranet que brindará a dichas instituciones y a su comunidad universitaria un medio que les permita estimular el espíritu crítico y la participación democrática en la elaboración de la documentación normativa. El desarrollo del sistema de información Intranet estuvo guiado por El Proceso Unificado de Desarrollo de Software, un marco de trabajo genérico caracterizado por tres aspectos fundamentales: Es iterativo e incremental, dirigido por casos de uso y centrado en la arquitectura. Por otra parte, para la documentación se utilizó el Lenguaje Unificado de Modelado (UML), un lenguaje estándar que permite visualizar, especificar, construir y documentar todos los elementos que componen un producto software. El sistema se realizó a lo largo de varias fases, desde la fase de inicio, donde se establecieron las características, siguiendo con la fase de elaboración que permitió definir la línea base de la arquitectura y finalizando con la fase de construcción, en la que se obtuvo la herramienta software en su versión operativa inicial.

Cabe destacar del presente trabajo de grado, su adaptabilidad a cualquier Institución de Educación Superior, debido a los aspectos comunes que en Actos Administrativos poseen dichas instituciones además de la aplicación de la ingeniería de sistemas en campos, en los cuales, no solo brinda apoyo para optimizar procesos, sino promueve cambios metodológicos que aporten al desarrollo Institucional y organizacional.

* Proyecto de grado en la modalidad investigación

** Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería de Sistemas e Informática, GÓMEZ FLÓREZ, Luis Carlos

ABSTRACT

TITLE:

SYSTEM OF INFORMATION INTRANET FOR THE SUPORT OF NORMATIVE MANAGEMENT IN INSTITUTIONS OF HIGHER EDUCATION *

AUTHORS:

SUÁREZ ROJAS, Martha Ligia.

ROJAS CÁRDENAS, Juan Pablo **

KEY WORDS:

Administrative acts, intranet, higher education, normative management, unified process.

DESCRIPTION

This project is proposing an investigation about the process of normative management in institutions of higher education and their need for change; this proposal formed with the software tool SIGNO 1.0, system of information intranet thah will offer to the above mentioned institutions and to their university community a way that allows them to stimulate the critical spirit and the democratic participation in the production of the normative documentation.

The development of the tool was guided by The Unified Software Development Process, a general-purpose framework featured by three main aspects: It is iterative and incremental , use-case directed, and architecture centered. On the other hand, for the documentation it was used the Unified Modeling Language (UML), an standardized language for specifying, visualizing, constructing and documenting all the artifacts composing a software product. The system was made up throughout several phases, starting with the initial phase, where all the features were established, passing by the elaboration phase, which allowed to define the architectural base line and finalizing with the construction phase, in which the first functional version of the software tool was obtained.

It's relevant to emphasize with this present proyect, its adaptability to any institution of higher education. Due to the common aspects in administrative acts these said. Institution possess, decides the aplicattion of engineering system, in which, they not only offer support to optimize processes, but promote metodological change that contribute to institutional development and organization

* Graduation project in the category of investigation

** Physical – Mechanical Engineering's Department, Systems and Informatics Engineering School, GÓMEZ FLÓREZ, Luis Carlos

PREFACIO

Es común, encontrar en la sociedad cierto desinterés hacia el estudio de la normatividad en la que ella está inmersa , esto conlleva a un desconocimiento de las reglas con las cuales la vida en sociedad como se concibe hoy en día no fuera posible.

A esta situación no son ajenas las Instituciones de Educación Superior, donde la normatividad que ellas expiden y que ellas deben acatar, pasan desapercibidas para la mayoría de la comunidad universitaria, creando así un inmenso vacío en la aplicación, control y transparencia de la gestión normativa en dichas instituciones, generando por tanto falencias en el fortalecimiento institucional.

En la actualidad, la normatividad vigente en las Instituciones de Educación Superior - IES¹ se basa en el apego al derecho y a la defensa de un espíritu democrático. Sin embargo, como resultado de la situación anteriormente mencionada, la misma practica refleja la ausencia de una verdadera cultura política, en la cual se den espacios de participación que aporten al desarrollo institucional y al compromiso personal para este. En este sentido es necesario hacer posible que los miembros participen del proceso de gestión normativa mediante tres acciones básicas: en primer lugar, conocer la normatividad vigente y su soporte legal, en segundo lugar, la participación en el ejercicio de esta

¹ De aquí en adelante, cada vez que en el documento se mencione IES, se hace referencia a Instituciones de Educación Superior.

normatividad y en el control de su cumplimiento, y en tercer lugar contribuir al desarrollo de la normatividad mediante sus cuestionamientos e iniciativas. De tal forma que se cree un escenario, de no solo contribución al desarrollo institucional, sino también, espacios de formación para la participación.

Dentro de este contexto surge el presente trabajo de grado, como una herramienta de apoyo a la gestión normativa en Instituciones de Educación superior pretendiendo de esta manera facilitar el logro de los objetivos que tiene la educación superior en Colombia.

El contenido de este trabajo se encuentra dividido en 5 partes:

- La primera parte comprende dos capítulos, en el primero se presentan los aspectos que se han tenido en cuenta para desarrollar este trabajo de grado, desde la problemática identificada, hasta la materialización del proyecto como una alternativa de solución a la situación dada. Adicionalmente se presentan los tópicos que se han tenido en cuenta para la explicación del desarrollo del proyecto, los cuales están relacionados con trabajos realizados al interior de EISI en el área de intranets, manejo de información normativa y por último se presentan los beneficios que el proyecto trae a las IES. El segundo capítulo comprende un marco conceptual que proporciona los aspectos básicos necesarios para la comprensión del proyecto.
- Con la segunda parte se empieza a mostrar el desarrollo gradual de la herramienta, siguiendo como metodología el Proceso Unificado de Desarrollo de Software. Esta parte comprende la Fase de Inicio dentro de la cual se plantean 2 capítulos: El primero – Recopilación de requisitos de Casos de Uso - , permite tener más claridad acerca del contexto global en el cual se enmarca el proyecto con base en tres aspectos vitales como son: el aporte del sistema a la solución de la problemática detectada en las IES, los tipos de usuarios que

interactuarán con el sistema y por último la identificación de los diferentes riesgos con el fin de controlarlos para que no afecten el desarrollo del proyecto. En segundo capítulo - Análisis - se comienza a estudiar, clasificar y organizar los requisitos hallados en el capítulo 1.

- La tercera parte comprende la Fase de Elaboración, en la cual se puede apreciar el avance del proyecto, ya que aquí se hallan la mayoría de requisitos y se complementan los modelos hallados en la fase de Inicio. Esta parte contiene 3 capítulos: Recopilación de Requisitos, Análisis y Diseño.
- En la cuarta parte se expone la Fase de Construcción, en ella, el énfasis está en la implementación y pruebas del software, adicionalmente, durante la fase de construcción se crearán diversos documentos que complementan la herramienta, tales como la guía del usuario final. Esta parte comprende 2 capítulos: en el primero se muestra algunas secciones de implementación y pruebas de los subsistemas Generales de la aplicación. En el segundo capítulo se presentan las mismas actividades para los subsistemas Específicos de la aplicación.
- La última parte, comprende dos secciones: primero, en el capítulo 10, se ofrece una serie de recomendaciones y conclusiones, y por último se reserva un espacio para las fuentes, incluyendo la bibliografía y los anexos.

Antes de seguir con la lectura de este documento, se recomienda considerar las siguientes sugerencias:

- ✓ La forma apropiada de comprender este trabajo de investigación es comenzar la lectura desde el primer capítulo, donde se ofrecen los diversos conceptos que sitúan al proyecto en su contexto de interés. Luego se recomienda ir

progresando ordenadamente a través del documento, el cual plasma el desarrollo del Sistema de Información Intranet.

- ✓ El desarrollo del Sistema de Información Intranet se llevó a cabo siguiendo la metodología del Proceso Unificado de Desarrollo de Software, por tanto para entender las partes dos, tres y cuatro, donde se exponen los avances graduales que llevaron a obtener la herramienta, sería de gran ayuda estar familiarizado con dicha metodología. Si ese no es el caso, se recomienda que antes de empezar a leer dichas partes, se revise el anexo A que ofrece información sobre el Proceso Unificado y el Lenguaje Unificado de Modelado, respectivamente.

- ✓ Si se tiene interés por conocer los diagramas UML que describen el Sistema de Información Intranet, puede ver la evolución gradual de ellos a lo largo de las partes dos y tres, aunque puede remitirse directamente a la parte 3 donde se encuentran los modelos finales.

PARTE I – ASPECTOS GENERALES

En esta parte se presentan los fundamentos básicos del presente trabajo de grado, con el fin de proporcionar una descripción de las características que identifican el mismo. Inicialmente se tratan los aspectos generales que describen el entorno en el cual se enmarca este proyecto. Es aquí donde se explica, tanto la problemática que se detectó en Instituciones de Educación Superior, específicamente en el área normativa, como la alternativa de solución planteada; además se muestra el impacto que dicha alternativa trae tanto a la Universidad Industrial de Santander como a las Instituciones que lo implanten. Posteriormente se presenta el marco teórico necesario para el entendimiento de los conceptos que se manejan a lo largo del libro y utilizados para el desarrollo del proyecto. Por último, se da a conocer el marco metodológico que se sigue para llevar a cabo el desarrollo del software involucrado.

CAPITULO 1

1. ASPECTOS GENERALES

1.1. INTRODUCCION

La normatividad vigente en las Instituciones de Educación Superior se caracteriza por el apego al derecho y a la defensa de un espíritu democrático. Sin embargo la misma practica refleja la ausencia de una verdadera cultura política, en donde se den espacios de participación que aporten al desarrollo institucional y al compromiso personal para este. En este sentido es necesario hacer posible que los miembros participen del proceso de desarrollo normativo mediante tres acciones básicas: en primer lugar, conocer la normatividad vigente y su soporte legal, en segundo lugar, la participación en el ejercicio de esta normatividad y en el control de su cumplimiento, y en tercer lugar contribuir al desarrollo de la normatividad mediante sus cuestionamientos e iniciativas, de tal forma que se cree un escenario, de no solo contribución al desarrollo institucional, sino también, espacios de formación para la participación.

1.2. ANTECEDENTES Y DESCRIPCIÓN DEL PROBLEMA.

La Educación Superior en Colombia es un proceso que está orientado primordialmente, al cultivo de la ciencia, la investigación y el desarrollo tecnológico, con un enfoque pedagógico y curricular que capacite al estudiante como persona y ciudadano, a la vez que ofrezca a la sociedad y al país la

formación del talento humano que requiere su progreso en todos los campos del conocimiento.

Para cumplir a cabalidad estos objetivos, las Instituciones de Educación Superior – IES, están en continua expansión, conllevando a que el cúmulo de información que manejan dichas instituciones se incremente considerablemente, haciendo cada vez más complejo y dispendioso el manejo de esta información.

Así mismo para optimizar su funcionamiento, las IES han adoptado una estructura administrativa jerárquica y descentralizada, es decir, cada una de las diferentes áreas (financiera, administrativa, normativa, académica, recursos humanos, etc.), que componen la estructura de las IES, poseen cierto grado de autonomía, que les permite el manejo de su información.

Los recursos destinados para el manejo de la información en las diferentes áreas, varían según el grado de interés que las IES tenga en ellas, creando así grandes diferencias en cuanto a medios utilizados para el manejo de la información entre las diferentes áreas que integran las IES, es el caso del área normativa, que aunque da la estructura legal bajo la cual están formadas y se rigen las IES, es una de las áreas más descuidadas en cuanto se refiere a herramientas para el manejo de la información, a esta afirmación se ha llegado luego de conocer la situación actual de dicha área, por medio de la investigación que se realizó en las IES más importantes de la ciudad de Bucaramanga, y que se presenta a continuación

1.2.1 Investigación del Manejo de la Información Normativa en algunas Instituciones de Educación Superior.

Esta investigación se llevo a cabo en diferentes IES de la	Organismo(s) que emiten actos	Proceso de publicación	Almacenamiento y de Actos	Recursos Hardware y Software que posee
--	-------------------------------	------------------------	---------------------------	--

ciudad de Bucaramanga, con el fin de analizar el estado actual del manejo de la información normativa, así mismo, la infraestructura tecnológica que permiten la comunicación en red (hardware y software)	Institución	administrativos	Administrativos	
Unidades Tecnológicas de Santander –UTS.		Directivas	Los actos administrativos después de elaborados, se archivan físicamente según el área a la que pertenezcan. Su publicación se hace mediante carteleras	Computadores, cableado, puntos de red y demás arquitectura (hardware) que se necesita para implantar una intranet
Universidad Autónoma de Bucaramanga – UNAB		Directivas y el manejo lo realiza Secretaría General	Los documentos se digitan en Word y se archivan físicamente. El departamento de sistemas realiza continuamente copias de seguridad de dichos documentos	Existe una arquitectura de red (hardware) y un sistema intranet de información, pero en este no se publica ni se maneja información normativa
Servicio Nacional de Aprendizaje –SENA		Directivas	existe un sistema de información monousuario en el que se almacenan, modifican y consultan los actos administrativos. Cada vez que hay un tope de información, se hace una copia de seguridad en CD.	Red.
Universidad Pontificia Bolivariana. –UPB		Directivas	Esta institución utiliza varios medios para almacenar y consultar los documentos generados por los actos administrativos, el más antiguo es el archivo físico y desde hace aprox. 2 años está en funcionamiento la Intranet-Extranet de la Institución; en dicho sistema se guarda la información respectiva de los actos administrativos, y allí todos los diferentes entes de la universidad tienen la posibilidad de	Red Sistema Intranet - Extranet

		consultar dicha información	
Universidad Industrial de Santander –UIS	Directivas	Sistema Intranet que entre otras funcionalidades, permite la publicación de los Actos Administrativos	Red Sistema Intranet - Extranet

Tabla 1.1. Estado actual del área normativa en algunas IES de Bucaramanga

Esta investigación realizada en algunas de las diferentes IES demuestra que aunque la información normativa desempeña un papel básico en el funcionamiento de estas instituciones, ha sido descuidada considerablemente originando problemas como inseguridad, desconfianza en cuanto a la veracidad de los documentos, entrega tardía de la información cuando ésta es requerida, sobrecosto en la nómina debido a la necesidad de incluir más personal para manejar los diferentes archivos, además de la desinformación de la comunidad universitaria en este campo.

Para una mayor comprensión de la problemática expuesta anteriormente, se presentan las figuras 1.1 y 1.2, en las cuales se muestra el proceso actual que las IES realizan para elaborar una norma. En estas figuras se puede apreciar que las IES poseen organismos (Entes) encargados de dar solución a las diferentes problemáticas que surgen al interior de la Institución por medio de normas, las cuales una vez emitidas son publicadas a la comunidad universitaria (generalmente por medio de carteleras) y aplicadas a dicha comunidad.

1.2.2 Proceso actual para la Gestión Normativa en IES

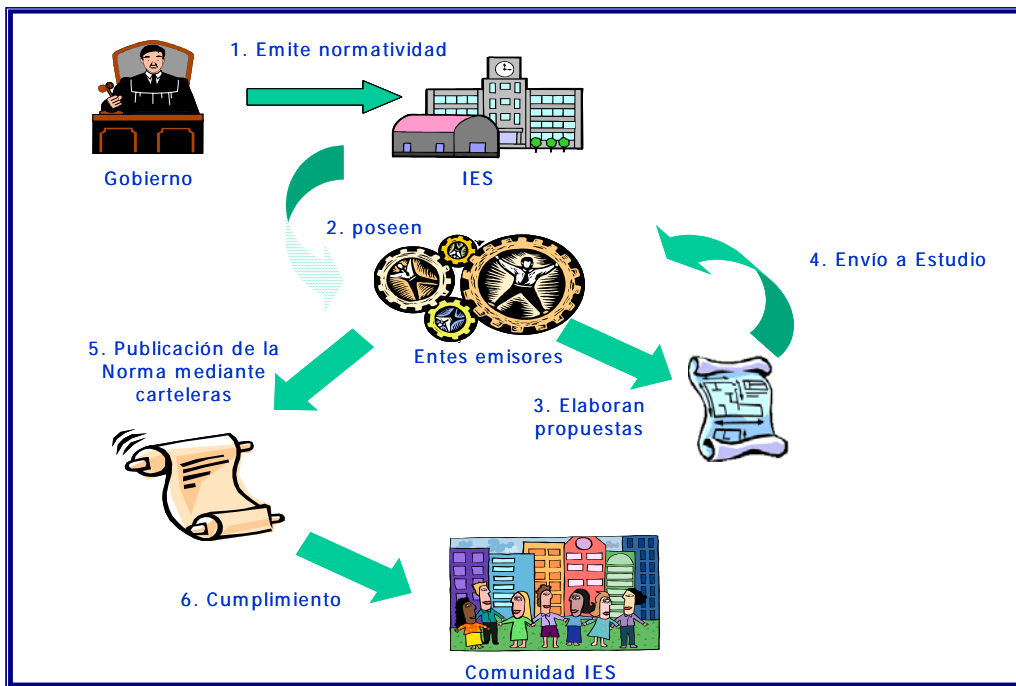


Figura 1.1. Figura enriquecida del proceso actual de gestión normativa en IES

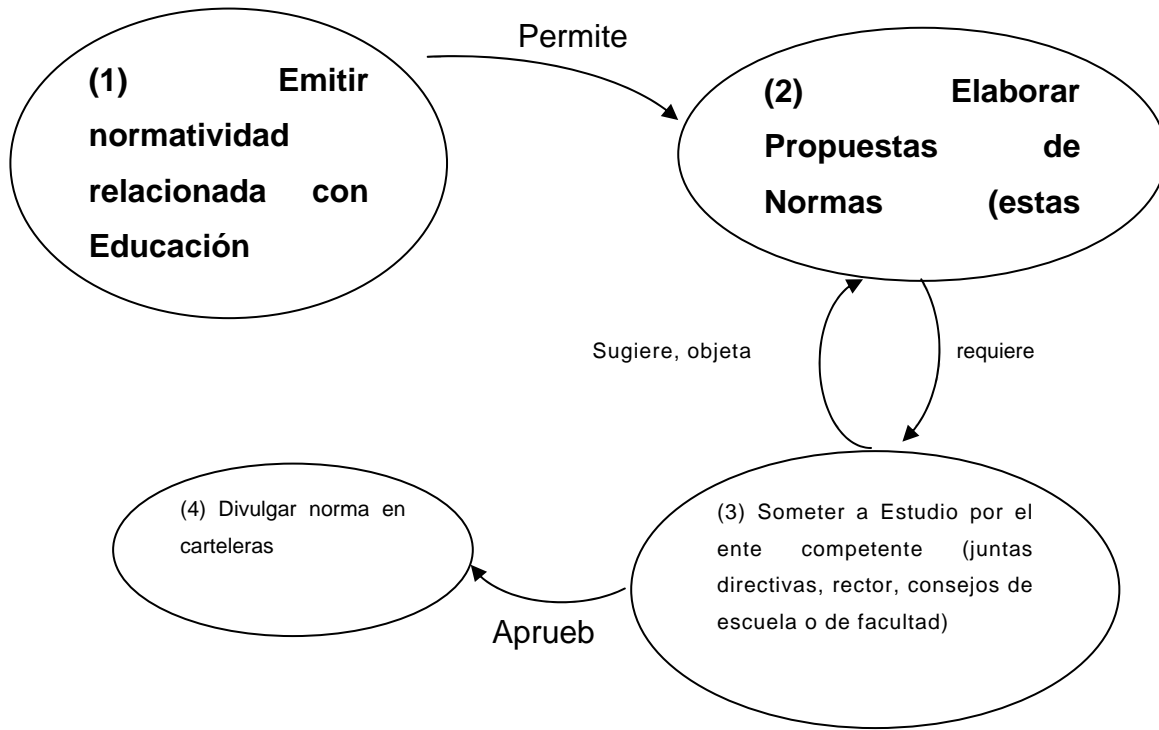


Figura 1.2. Proceso actual de gestión normativa en IES

La tabla 1.2. complementa la situación actual del proceso antes mencionado.

ETAPA	DESCRIPCION
Elaboración de la propuesta de norma.	La(s) persona(s) autorizada(s) por la Institución para elaborar propuestas de norma deben basarse en las normas que rigen la Educación Superior en Colombia.
Estudio por parte del ente encargado	Una vez elaborada la propuesta es presentada al ente competente para su aprobación.
Divulgación de la norma	Una vez se estudia y ratifica la norma, ésta es publicada, normalmente en carteleras.
Cumplimiento.	Actualmente esta fase no tiene ningún control interno.

Tabla 1.2. Proceso actual de gestión normativa en IES

1.2.3 Proceso propuesto para la gestión normativa en IES

Teniendo en cuenta las deficiencias que posee el sistema existente, surge el presente proyecto como una alternativa de solución que pretende no solo optimizar el manejo de la información normativa, sino generar un cambio en el proceso actual, haciendo de este más que una simple forma de solucionar los problemas que surgen al interior de las instituciones, un mecanismo que genere espacios para la participación democrática, el desarrollo de un espíritu reflexivo, de compromiso y de pertenencia. En las figuras 1.3 y 1.4, se ilustra dicho proceso.

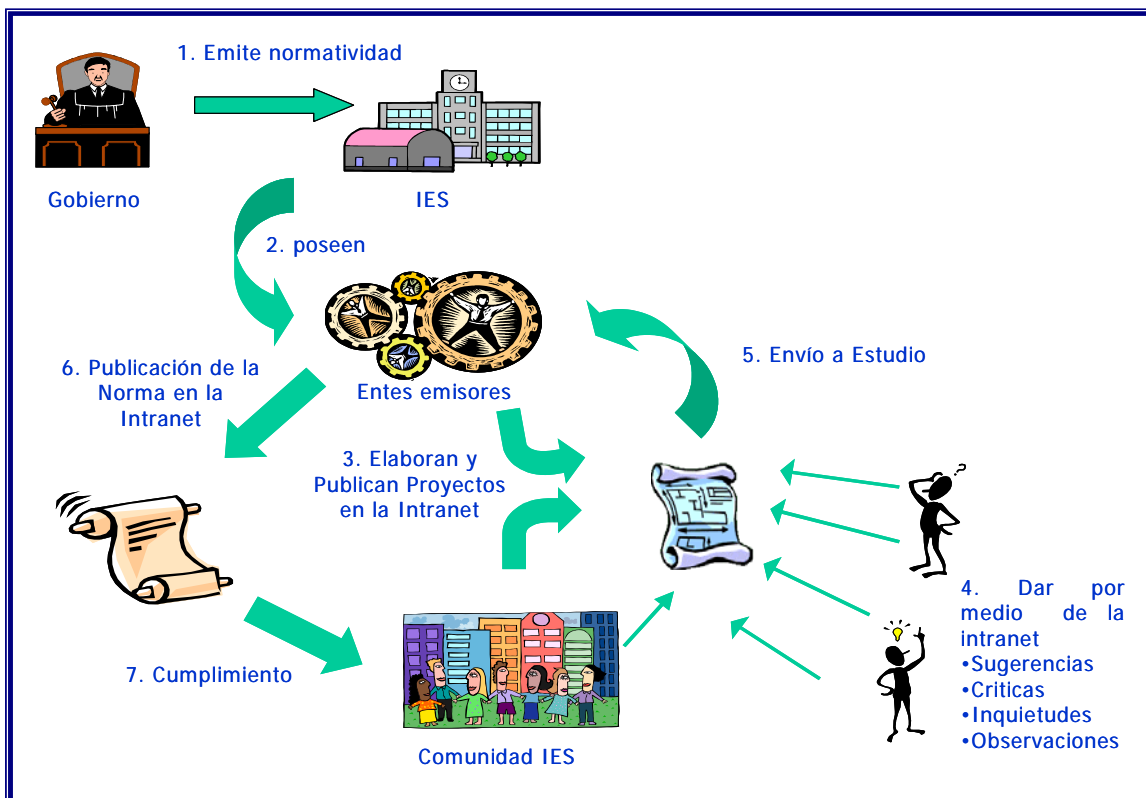


Figura 1.3. Figura enriquecida del proceso propuesto para apoyar la gestión normativa en IES

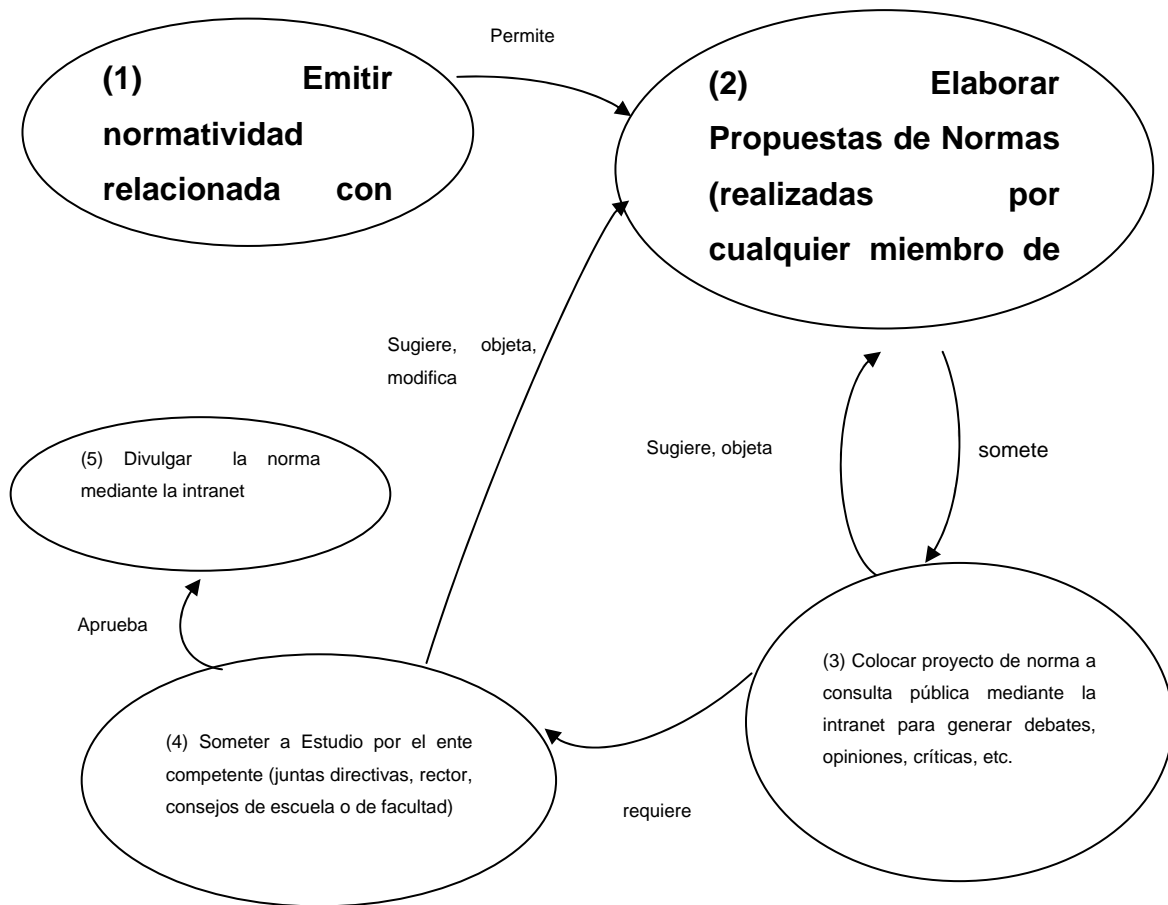


Figura 1.4. Proceso propuesto para apoyar la gestión normativa en IES

ETAPA	DESCRIPCION
Elaboración de la propuesta de norma.	Para elaborar un proyecto de norma la comunidad universitaria tiene a su disposición información sobre las normas que rigen la Educación Superior en Colombia, dentro de las cuales pueden elegir las relacionadas con el tema tratado, esto con el fin de tener un soporte legal sobre el cual basarse. Adicionalmente puede contar con unas pautas básicas que deben tener en cuenta para que dicha norma tenga validez.
Consulta pública	Este es un periodo máximo de tiempo asignado por la institución para mantener publicado el proyecto de norma en la intranet, con el fin de someter a la opinión de la comunidad universitaria dicho proyecto. Si durante este periodo se reciben observaciones el/los autor(es) del proyecto analizan estas observaciones para decidir si aplican o no y por ende realizar los cambios pertinentes.
Estudio por parte del ente encargado	Una vez terminada la consulta pública, la propuesta es enviada al ente competente para su aprobación.
Divulgación de la norma	Una vez se ratifica la norma, ésta es publicada en la intranet y se notifica mediante correo electrónico a las partes involucradas y/o afectadas por la misma.
Cumplimiento.	Es importante recordar esta máxima “una buena norma es una norma que se aplica y se cumple”. Así una normalización no sería completa y útil, para ello la comunidad universitaria puede mediante la intranet conocer las normas y por ende ejercer control sobre el cumplimiento de las mismas.

Tabla 1.3. Proceso propuesto para el apoyo de la gestión normativa en IES

1.3. OBJETIVOS

Objetivo General

Desarrollar un Sistema de Información soportado en tecnologías Intranet, para apoyar la gestión normativa en Instituciones de Educación Superior, empleando la metodología del proceso unificado para el desarrollo de software.

1.3.1 Objetivos Específicos

Desarrollar un sistema de Información Intranet que permita a las Instituciones de Educación Superior:

- ☑ Personalizar la presentación de las páginas del Sistema de Información Intranet, mediante la asignación del nombre de la Institución y su logo, el nombre de la Intranet, así como la edición de colores y tipo de letra de dichas páginas.

- ☑ Contribuir al ejercicio de la autonomía universitaria, mediante la generación en el sistema de los diferentes entes que emiten Información Normativa, además de los diferentes tipos de documentos normativos que ellos emiten.

- ☑ Manejar los documentos normativos, esto incluye, formatos para la entrada de información, diferentes criterios de búsqueda y esquemas de representación para la recuperación de documentos.

- ☑ Publicar los actos administrativos a los usuarios pertinentes de la comunidad universitaria, promoviendo así, la transparencia de la gestión de los diferentes entes, así como de los actos emitidos por estos.

- ☑ Fomentar la participación democrática y el espíritu reflexivo mediante la generación de un espacio donde la comunidad universitaria pueda publicar, modificar y recibir sugerencias que otros miembros de dicha comunidad realicen a sus proyectos de actos administrativos, además de consultar la condición en la que se encuentran dichos proyectos, permitiendo de esta manera la participación en la elaboración de los actos administrativos y por ende el velar porque dichos actos se cumplan a cabalidad.

- ☑ Consultar e incluir información normativa relacionada con Educación Superior que expide el gobierno nacional, esto con el fin de ofrecer un soporte legal en el cual basarse para la elaboración de los actos administrativos de las IES.

- ☑ Establecer comunicación entre los diferentes entes emisores de actos administrativos mediante correo electrónico, esto con el fin de notificar la emisión de actos administrativos relacionados con ellos y demás funcionalidades que permite dicho correo.
- ☑ Construir los modelos de Casos de Uso, de Análisis y de Diseño, para el Sistema de Información Intranet, siguiendo la metodología del Proceso Unificado de Desarrollo de Software.
- ☑ Desarrollar el Sistema de Información Intranet utilizando software de libre distribución y código abierto, tales como el sistema operativo Linux, el Servidor Apache, el lenguaje PHP y el manejador de base de datos Postgres.
- ☑ Especificar un conjunto de procedimientos para el manejo de la herramienta, administración y actualización de la información contenida en los servidores del sistema.

1.4. IMPACTO

El desarrollo de este proyecto traerá beneficios a la comunidad universitaria de la Institución que lo adopte, ya que podrán aprovechar las ventajas que ofrece como son:

- ☑ Manejar Eficientemente la Información Normativa, ya que se utilizaran los recursos que las IES poseen, junto con las facilidades que ofrecerá el sistema en aspectos como búsqueda y publicación de la información.
- ☑ En cuanto a la publicación y almacenamiento de la Información Normativa, el sistema ofrecerá formatos para hacer una estandarización en la entrada de datos y garantizar que las búsquedas ofrezcan resultados satisfactorios.

- ☑ Permitir que las IES otorguen a quien consideren conveniente un espacio para publicar los proyectos de Actos Administrativos, de esta manera la comunidad universitaria podrá interactuar tanto en la elaboración de los diferentes actos administrativos como velar por el cumplimiento de dichos actos promoviendo así el ejercicio de la participación democrática y el desarrollo del espíritu reflexivo.

- ☑ Reducción del tiempo en la adaptabilidad ya que la mayoría de la comunidad universitaria esta familiarizada con el ambiente(exploradores, buscadores, etc.) en que se desarrollará el sistema.

Es de anotar que el proyecto no está dirigido a una institución de educación superior en particular, sino que pretende ser una herramienta estándar que pueda ser adoptada por diferentes IES, y por lo tanto se podrá personalizar las plantillas de presentación de la Intranet según la IES que lo implante.

El desarrollo de este proyecto es un esfuerzo conjunto del Grupo de Investigación de Sistemas y Tecnologías de Información -STI² por incrementar sus líneas de acción y la aplicación de las tecnologías Intranet para dar solución a problemas organizacionales.

Además, El uso del Proceso Unificado como metodología para el desarrollo del sistema, se constituye en una guía para el grupo de investigación STI, ya que, aunque no es el primer proyecto desarrollado con esta metodología dentro del

² Colectivo de estudiantes, egresados y profesores de la escuela de Ingeniería de Sistemas e Informática de la UIS que desarrollen un proyecto intelectual alrededor de la problemática de la aplicación de la tecnología de la información de las organizaciones, especialmente en lo referente a su pertinencia integrando bajo un enfoque sistémico cuatro líneas de acción: Gerencia y Organizaciones, Planeación, Ingeniería del Software y Auditoría.

grupo, aporta no solo una síntesis del conocimiento teórico de dicha metodología, sino también la experiencia adquirida en la aplicación de la misma.

De esta manera se pretende sentar las bases para avances posteriores en el grupo STI, en cuanto a la cobertura del sistema desarrollado, en búsqueda de lograr productos que sean competentes en el mercado.

1.5. RESUMEN CAPITULO 1

En este capítulo se presentaron los aspectos que se han tenido en cuenta para desarrollar este trabajo de grado, desde la problemática identificada sobre el manejo de información normativa en las Instituciones de Educación Superior –IES, hasta la materialización del proyecto como una alternativa de solución a la situación dada. Los tópicos que se han tenido en cuenta para la explicación del desarrollo del proyecto están relacionados con trabajos realizados en las áreas de intranets, manejo de información normativa y los beneficios que el proyecto traerá a las IES.

CAPITULO 2

2. MARCO TEÓRICO

2.1. INTRODUCCION

En este capítulo se presenta un marco conceptual básico sobre las diferentes tecnologías utilizadas en el desarrollo del presente trabajo de grado. Es de aclarar que el objetivo de este capítulo no es el de profundizar en los conceptos manejados, sino mostrar los aspectos teóricos generales para la comprensión del proyecto. Adicionalmente y con el fin de aclarar al lector el concepto de Acto Administrativo bajo el cual se trabaja en el presente proyecto, se incluye una breve descripción del mismo, sus partes y funciones.

2.2. TECNOLOGÍAS INTERNET

Las tecnologías Internet representan un nuevo método para llegar a la gente, teniendo en cuenta este criterio, dichas tecnologías se clasifican según su nivel de cobertura en Internet, Extranet e Intranet.

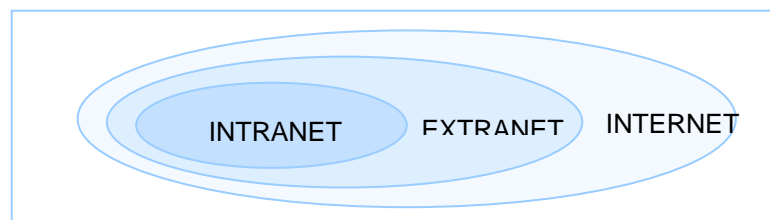


Figura 1.5. Alcance de Intranet, Extranet e Internet.

Las tecnologías Internet están entrando en nuestra cultura y están cambiando la forma de comunicación entre las personas, un ejemplo claro es cuando nos comunicamos con el resto de gente por medio del correo electrónico, chat, o nos suscribimos a un servicio de noticias en línea de la misma manera cuando navegamos a través del World Wide Web. Pero no solo tenemos la opción de ver información también podemos ser parte activa de estas tecnologías cuando colocamos algo en la Red para que los demás puedan observarlo, de la misma manera cualquier corporación puede estar interesada en utilizar una Intranet y/o Extranet para mejorar la comunicación de su organización.

¿Y Como funcionan las Tecnologías Internet? Todos los computadores que se encuentran conectados a una red necesitan tener un lenguaje y unas reglas comunes con el fin de poder comunicarse, a este lenguaje se le denomina protocolo. Existen dos protocolos básicos que hacen posible la comunicación dentro de Internet que son TCP(Protocolo de Control de Transmisión) e IP (Protocolo Internet).

2.2.1 Internet

Dar una definición exacta sobre qué es Internet es casi imposible, debido a que está en cambio y evolución permanente. Una buena forma de entender Internet es pensando en ella como una gran red mundial de computadores formada por multitud de pequeñas redes y computadores individuales conectados unos con otros de manera que sea posible el intercambio de información entre ellos.

2.2.2 Extranet

Una Extranet es una solución Internet que permite el intercambio de información de manera permanente, confidencial y segura con personas fuera de la organización como socios de negocios, clientes, proveedores, entre otros. y que posee las siguientes características generales:

- ☑ Es un sistema de acceso restringido basado en tecnologías Internet.
- ☑ La comunicación se da solo entre la empresa (dueña del sistema) y su entorno. Las otras empresas usuarias de la Extranet no necesariamente se comunican entre sí.

Como se dijo anteriormente la única diferencia significativa entre una Extranet y una Intranet es que en esta última generalmente sólo acceden empleados de la empresa, mientras que a una Extranet se permite el acceso tanto a empleados como a clientes, proveedores, colaboradores, etc. Por tanto se puede pensar en una Extranet como una extensión de la Intranet a usuarios externos autorizados.

2.2.3 Intranet

Una Intranet es una red interna y autosuficiente que enlaza múltiples usuario usando las tecnologías internet. Construidas y gestionadas por las organizaciones, estas redes proporcionan comunicaciones interplataforma entre los usuarios autorizados y en tiempo real.

Las intranets han revolucionado los conceptos de trabajo, colaboración y comunicación dentro de las organizaciones, consiguiendo hacer más agradable y productiva las labores a las que se dedica la mayor cantidad de tiempo y dinero, este es el caso de la búsqueda de información, la comunicación y la colaboración entre personas dentro de la organización, la publicación y distribución de la información.

Las Intranets dentro de las organizaciones, no sólo deben limitarse a proveer información corporativa, sino deben poseer las cualidades de volver eficiente y productivo el trabajo dentro de las empresas. Deben ser parte de los procesos de

automatización de los negocios y organizaciones, convirtiéndose en plataformas que permiten una interacción permanente entre los que laboran dentro de ellas.

2.3. TECNOLOGÍAS DISPONIBLES PARA EL DESARROLLO

Para desarrollar aplicaciones y dotar a las páginas web de funcionalidad se puede trabajar tanto en el lado del cliente como en el lado del servidor, las variantes son:

Programación en el cliente:

- ☑ El navegador envía un request.
- ☑ El servidor envía un response que contiene código que el navegador entiende.
- ☑ El navegador interpreta el código enviado por el servidor y realiza una determinada acción.

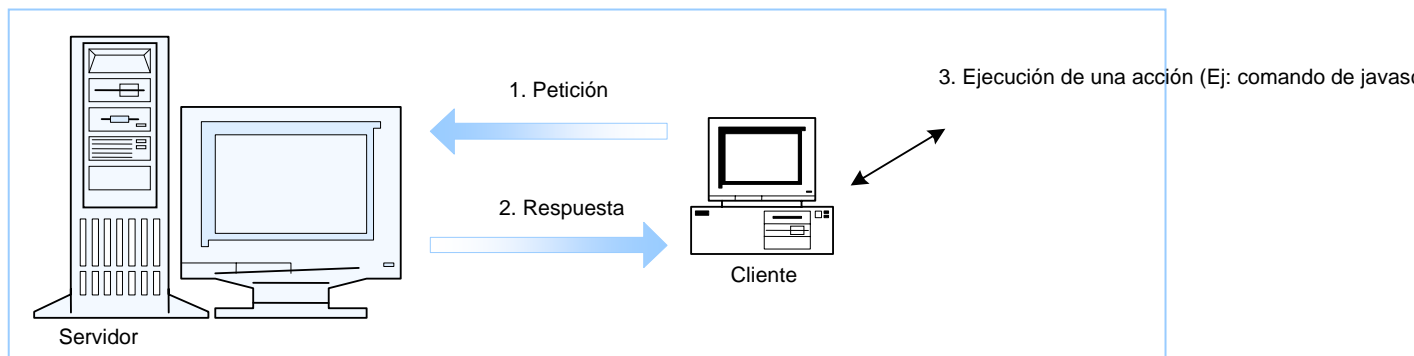


Figura 1.6. Programación en el lado del cliente

Programación en el servidor:

- ☑ El navegador envía un request.
- ☑ El servidor ejecuta una aplicación que realiza una determinada acción.
- ☑ El servidor envía el resultado de dicha aplicación al cliente.

- ☑ El navegador muestra el resultado recibido del servidor.

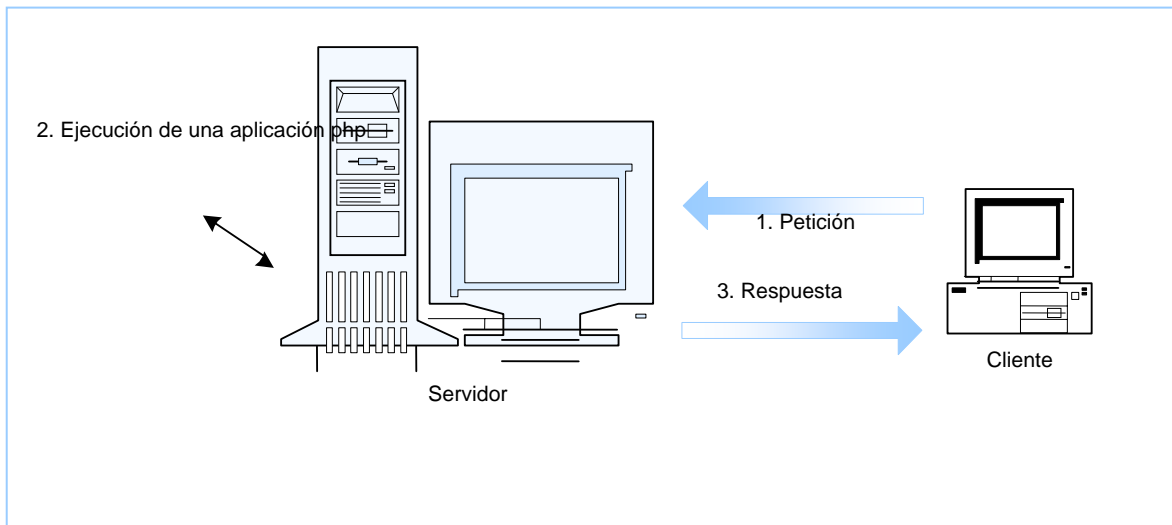


Figura 1.7 Programación en el lado del Servidor

Esquema mixto: (programación en el cliente y en el servidor)

- ☑ El navegador envía un request.
- ☑ El servidor ejecuta una aplicación que realiza una determinada acción.
- ☑ El servidor envía el resultado de dicha aplicación al cliente conteniendo código a interpretar por el navegador.
- ☑ El navegador interpreta el código enviado por el servidor y realiza una determinada acción.

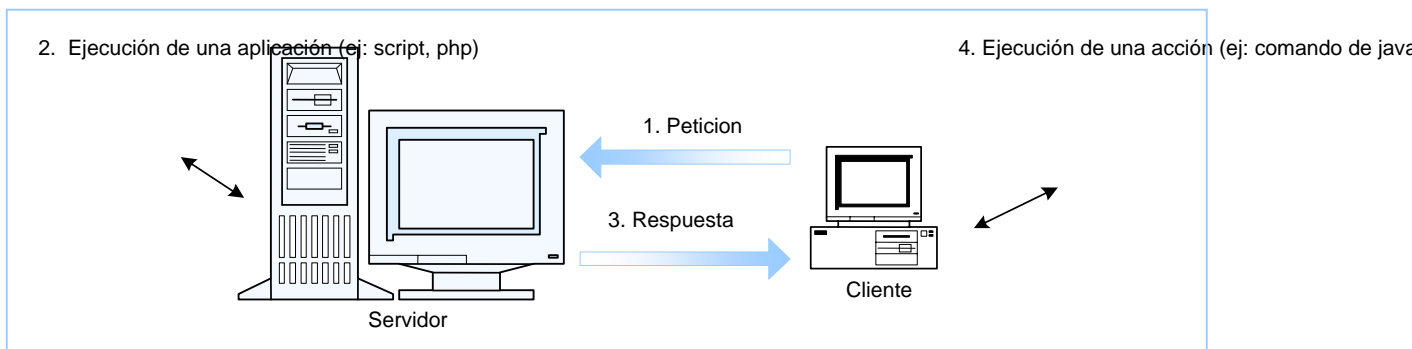


Figura 1.8. Esquema mixto. Programación en el cliente y en el servidor.

La programación del lado del cliente tiene como principal ventaja que la ejecución de la aplicación se delega al cliente, con lo cual se evita recargar al servidor de trabajo. El servidor sólo envía el código, y es tarea del navegador interpretarlo. La gran desventaja de esta metodología es que el código que el servidor envía es “sensible” a qué cosas puede o no hacer el navegador. El usuario puede, por ejemplo, decidir deshabilitar una funcionalidad del navegador que es necesaria para que se ejecute un determinado servicio o peor aún, navegadores distintos pueden interpretar el mismo código de distintas formas. Típicamente Netscape y Microsoft, que producen los dos navegadores más usados del mercado, no están de acuerdo sobre como se implementan diversas tecnologías en el cliente.

Programar del lado del servidor tiene como gran ventaja que cualquier cosa puede hacerse sin tener en cuenta el tipo de cliente, ya que la aplicación se ejecuta en el servidor que es un ambiente controlado. Una vez ejecutada la aplicación, el resultado que se envía al cliente puede estar en un formato “normalizado” que cualquier cliente puede mostrar. La desventaja reside en que el servidor se sobrecarga de trabajo ya que además de servir páginas, es responsable de ejecutar aplicaciones. A menudo esto redundante en requisitos de hardware mayores a medida que el servidor ejecuta más y más servicios.

Sin embargo, debido a las incompatibilidades existentes y a la posibilidad de que el usuario controle que cosas se ejecutan y cuales no, la programación del lado del cliente no es muy recomendable y debe limitarse a código altamente estándar que pueda interpretarse de cualquier forma en cualquier navegador, lo cual obliga a ejecutar la gran mayoría de las aplicaciones y servicios del lado del servidor.

2.3.1 2.3.1. Programación del lado del cliente

A continuación se presenta la tabla 1.4 que resume los principales lenguajes para la programación de los sitios web en el lado del cliente.

LENGUAJE	DESCRIPCIÓN
HTML	Hypertext Markup Language. Lenguaje que se utiliza para la creación de páginas web. Consta de un conjunto de elementos, denominados etiquetas o marcas, que se utilizan para incluir texto, dibujos, tablas, cabeceras, etc.
DHTML	Extensiones de HTML que cubre el uso de hojas de estilo en cascada (CSS), uso de layers o capas para el posicionamiento y visibilidad de elementos en el navegador, programación con lenguajes de scripts y modelado de objetos de documentos (DOM).
JavaScript	Lenguaje interpretado incrustado en las paginas web para permitir ejecutar código en el cliente a través del manejo de objetos y la captura de eventos producidos en el navegador.
VBScript	Lenguaje interpretado similar a JavaScript proporcionado por Microsoft.
Applets de Java	Pequeña aplicación accesible en un servidor Internet, que se transporta por la red, se instala automáticamente y se ejecuta como parte de un documento web.
Actives	Tecnología de Microsoft que permite ejecutar un programa que siga este estándar en los equipos de los usuarios al ser vinculado como objeto en una página web. Los programas ActiveX dependen del sistema operativo Windows, por lo tanto no funcionan fuera de este ambiente.
Templates	Un template o plantilla consiste en un archivo que contiene un trozo de código HTML, por ejemplo el cuerpo principal del web. Entre este código se encuentran insertadas varias palabras entre llaves {} que harán de variables de la plantilla. Lo que se hacen entonces es asignar valores a estas variables y generar el código resultante. La ventaja principal de utilizar plantillas es la separación de los datos que se quieren presentar, del código. Así, se puede modificar una plantilla sin preocuparse por si corrompe el código del script. Además el desarrollador no necesita tratar con HTML en absoluto.

Tabla 1.4. Lenguajes del lado del cliente

En este proyecto serán utilizados los tres primeros lenguajes para la programación del lado del cliente.

2.3.2 2.3.2. Programación del lado del Servidor

Para el desarrollo de aplicaciones del lado del servidor existen 3 grandes metodologías, utilizar el protocolo CGI, utilizar una API provista por el web-servidor o bien utilizar un “módulo” del web servidor.

- **El protocolo CGI:**

El protocolo CGI (Common Gateway Interface) fue creado para establecer un protocolo estándar de comunicación entre el servidor web y cualquier lenguaje de programación de forma tal que desde el lenguaje “x” puedan recibirse datos que el usuario envía usando el método “POST” o “GET” y además el resultado de la aplicación sea enviado por el servidor al navegador. Típicamente para recibir datos se usa alguna biblioteca o módulo del lenguaje elegido que implementa el protocolo CGI y para enviar datos simplemente se envían a la salida estándar desde el lenguaje elegido y el servidor web se encarga de redireccionar esto al navegador.

De esta forma pueden realizarse aplicaciones para un sitio web en casi cualquier lenguaje. Los lenguajes interpretados rápidamente ganaron terreno ya que tienen un ciclo de desarrollo en tiempo inferior a los lenguajes compilados y son más fáciles de depurar dentro del ambiente CGI.

Los lenguajes no interpretados (C, C++) tienen como ventaja que requieren menos recursos del servidor al generarse el proceso CGI (no hace falta un interprete) y además suelen ser mucho más veloces en su ejecución (no se necesita interpretar nada), sin embargo el desarrollar y depurar suelen ser tareas muy complejas y no siempre se justifica el esfuerzo si la aplicación es pequeña.

La desventaja de las aplicaciones CGI consiste en que el servidor debe realizar un proceso, y ejecutar la aplicación o bien el interprete de la aplicación, y este ciclo,

que se cumple cada vez que se ejecuta la aplicación, CGI consume muchos recursos y en general es costoso en tiempo para el servidor.

- **Uso de una API del servidor:**

Otra técnica factible consiste en utilizar una API (application programming interface) provista por el servidor web para desarrollar aplicaciones, es decir que el servidor provee un lenguaje en el cual se pueden desarrollar aplicaciones. Este esquema, como podemos apreciar, es mucho más eficiente que el anterior ya que el servidor web es el encargado de ejecutar las aplicaciones en forma directa sin necesidad de crear un proceso. Las desventajas son sin embargo importantes: En primer lugar las aplicaciones creadas en este marco no son portables ya que sólo pueden ejecutarse en un servidor web determinado, esto es una gran desventaja frente a las aplicaciones CGI que podían una vez desarrolladas ejecutarse en cualquier servidor. La segunda gran desventaja es que frecuentemente un error de programación de una aplicación podría ocasionar que el servidor deje de funcionar, genere un error, pierda memoria u otros problemas. Esto ocasiona que este tipo de aplicación no sea confiable.

- **Uso de un Módulo del Servidor Web**

La tecnología más reciente para la ejecución de aplicaciones consiste en anexar a un servidor web “módulos” que le permitan interpretar un determinado lenguaje. De esta forma se logra eficiencia ya que el servidor no necesita crear un nuevo proceso por cada aplicación que ejecuta. Las aplicaciones son portables ya que son desarrolladas en un lenguaje estándar que no depende del servidor web. Las aplicaciones son confiables ya que si bien pueden producir un error en el lenguaje en que están diseñadas, si el módulo es sólido, dichos errores no pueden comprometer al servidor.

En las siguientes tablas (1.5 y 1.6) se presenta un resumen de los principales lenguajes con una descripción de su uso en programación del lado del servidor así como una breve comparación de los mismos.

LENGUAJES	DESCRIPCIÓN
Perl	Practical Extraction and Report Language. Lenguaje interpretado creado con el objetivo principal de simplificar las tareas de administración de un sistema UNIX. Hoy en día se ha convertido en un lenguaje de propósito general.
Python	Es un lenguaje interpretado que permite escribir programas pequeños, utilizado en desarrollo web para la creación de CGI
C, C++	Utilizado para la creación de CGI
PHP	PHP es un lenguaje interpretado diseñado para el desarrollo de sitios dinámicos. La distribución más popular de PHP es como módulo para el servidor Apache, aunque puede funcionar como un intérprete para ejecutar aplicaciones CGI.
ASP	Active Server Pages. Tecnología creada por Microsoft destinada a la creación de sitios web.
JSP	Java Server Pages (JSP). Es un lenguaje interpretado insertado en páginas web y basada en Java para el desarrollo de sitios dinámicos.
Mod_perl	Módulo de Perl para el servidor web
Mod_python	Módulo Python para el servidor web

Tabla 1.5. Lenguajes del lado del servidor

	CGI (interpretado)	CGI (compilado)	API del servidor	Módulo del servidor
Ejemplos	Perl, Python	C, C++	Netscape Enterprise	PHP, ASP, JSP, Mod_perl, Mod_python, FastCGI
Tiempo de Desarrollo	Corto	Largo	Medio	Corto
Depuración	Sencilla	Compleja	Compleja	Sencilla
Confiabilidad	Alta	Alta	Baja	Alta

Eficiencia	Baja	Media	Alta	Alta
------------	------	-------	------	------

Tabla 1.6. Comparación de lenguajes del lado del servidor

Para el desarrollo de este proyecto se ha escogido PHP para el desarrollo del lado del servidor por tanto a continuación se realiza una pequeña introducción a este lenguaje.

PHP (Hypertext Preprocessor)

El PHP es un lenguaje embebido en páginas HTML y que se ejecutan en el servidor, La mayoría de su sintaxis es similar a C, Java y Perl, con solamente un par de características PHP específicas. La meta de este lenguaje es permitir a los desarrolladores web, escribir páginas dinámicas de una manera rápida y fácil.

Productos similares a PHP son Active Server Pages (ASP) de Microsoft, ColdFusion de Allaire y Java Server Pages (JSP) de Sun.

PHP es fácil de aprender comparado con otros mecanismos para obtener la misma funcionalidad. A diferencia de JSP o CGI basados en C, PHP no requiere un conocimiento exhaustivo del lenguaje de programación. A diferencia de Perl, PHP tiene una sintaxis muy fácil de comprender y a diferencia de ASP, no requiere conocer más de un lenguaje de programación o de la instalación de módulos externos o comerciales para realizar tareas más complicadas no previstas en el lenguaje más usado (Visual Basic Script).

La mayoría de las funciones más útiles están predefinidas en PHP:

- ✓ Acceso a bases de datos: ODBC, Oracle, Postgres, SQL Server, MySQL, Informix, Interbase, SyBase, mSQL, dBase entre otros.
- ✓ Conectividad: HTTP, FTP, COM, YP/NIS, SNMP, Sockets, CORBA, LDAP.
- ✓ Servicios Correo y Noticias: POP, IMAP, SMTP, NNTP.
- ✓ Manejo de Textos y Gráficos: XML, HTML, PDF, GD, Flash.
- ✓ Funciones Matemáticas.
- ✓ POSIX: semáforos, memoria compartida, acceso a ficheros, expresiones regulares, cronómetros, etc.
- ✓ Comercio Electrónico: Cybercash, Verisign.
- ✓ Formularios.
- ✓ Encriptación y Compresión: MD5, Gzip, Bzip2, OpenSSL, etc.

Las instrucciones PHP están embebidas en HTML. Una página PHP es una página normal HTML que con unas marcas especiales le indican al servidor que deben interpretarse. Las principales ventajas de PHP son las siguientes:

- ✓ PHP puede ser agregado rápidamente al código HTML producido por editores HTML interactivos.
- ✓ PHP facilita la interacción entre diseñadores y programadores.
- ✓ No se necesita re-escribir cada línea de código HTML en un lenguaje de programación.
- ✓ PHP reduce costes y aumenta la eficiencia.

2.4. ARQUITECTURA DE SOFTWARE

La arquitectura de un sistema es la vista conceptual de la estructura de este. Toda aplicación contiene código de presentación, código de procesamiento de datos y código de almacenamiento de datos. Por tanto su arquitectura se define según como este distribuido este código.

En los últimos años la arquitectura de los sistemas de software desarrollados ha evolucionado junto con el resto de la tecnología informática. Hace no muchos años, cuando los mainframes eran una novedad, se desarrollaban sistemas monolíticos en donde un sistema no conocía nada más allá de su entorno de operación.

Cuando la tecnología de redes aparece y se difunde, la industria no tardó en darse cuenta de la ventaja de desarrollar sistemas que contaran con la capacidad de interactuar con otros sistemas residentes en otras máquinas dentro de la red. De aquí surge el modelo Cliente-Servidor donde un "Cliente" solicita servicios de un "Servidor" el cual gestiona las solicitudes de varios clientes a la vez.

Luego llegó el auge de Internet y el desarrollo de nuevas tecnologías para software por componentes. Con estas se puede construir una aplicación distribuida que reside en uno o más servidores en la red y además se disminuyó la necesidad de software cliente ya que generalmente se utiliza uno ya estandarizado: los navegadores de internet tales como Netscape o Internet Explorer, entre otros.

Las organizaciones están aprovechando estas tecnologías que permiten a sus usuarios el tener acceso sencillo y casi universal a sus aplicaciones corporativas sin la necesidad de incurrir en todos los gastos de mantenimiento que conlleva el modelo cliente-servidor como fue planteado en sus inicios.

Para que todo esto funcione, se requiere de incrementar lógica de programación del lado del servidor ya que la funcionalidad del cliente es mínima. Aquí es donde surge el concepto de sistema multicapa o de "n" capas como una metodología para el diseño de sistemas distribuidos. Esta arquitectura se hizo popular a principios de los años 90 y en la actualidad se ha afianzado como la arquitectura de software de aplicación empresarial.

2.4.1 Cliente – Servidor

Esta es una versión simplificada de la arquitectura en capas, donde la capa de aplicación se encuentra repartida entre el cliente y el servidor. Los sistemas basados en la arquitectura cliente / servidor están formados por dos partes lógicas o capas: un servidor que proporciona servicios, y un cliente que solicita servicios del servidor o servidores. Los dos, juntos, forman un sistema de computación completo con una clara división de responsabilidades.

Por lo general, un proceso cliente transmite una solicitud (a través de la red) de conexión al servidor y luego pide algún tipo de servicio a través de la conexión. Los procesos cliente normalmente gestionan la porción de interfaz de usuario de la aplicación, validan los datos introducidos por el usuario, realizan las solicitudes a los servidores y, a veces, ejecutan cierta lógica de negocio.

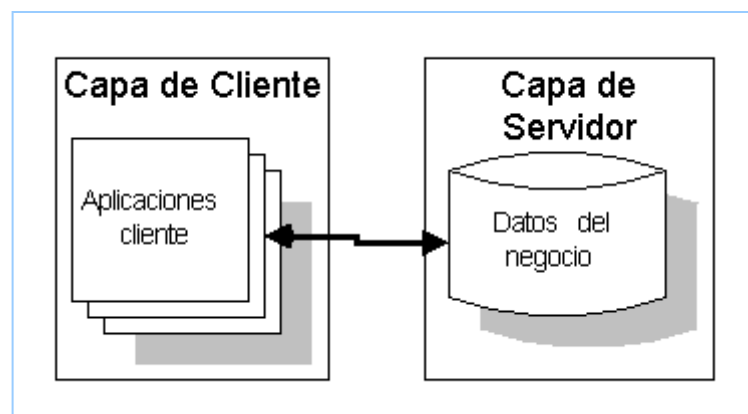


Figura 1.8. Arquitectura Cliente - Servidor.

Una aplicación servidor (o proceso servidor) casi siempre se arranca con ella misma y luego permanece latente esperando la solicitud del cliente para proporcionar un servicio específico. Los clientes gestionan recursos compartidos como ficheros, impresoras, enlaces de comunicación, bases de datos, etc.

2.4.2 Múltiples capas

Toda aplicación contiene código de presentación, código de procesamiento de datos y código de almacenamiento de datos. La arquitectura de las aplicaciones difiere según como está distribuido este código. Una arquitectura multicapa consiste en dividir la funcionalidad del sistema total en capas lógicas que pueden ser encapsuladas como componentes que interactúan entre ellos a alto nivel y supone una mayor escalabilidad de la aplicación, un mantenimiento menor y un incremento de la reutilización de componentes.

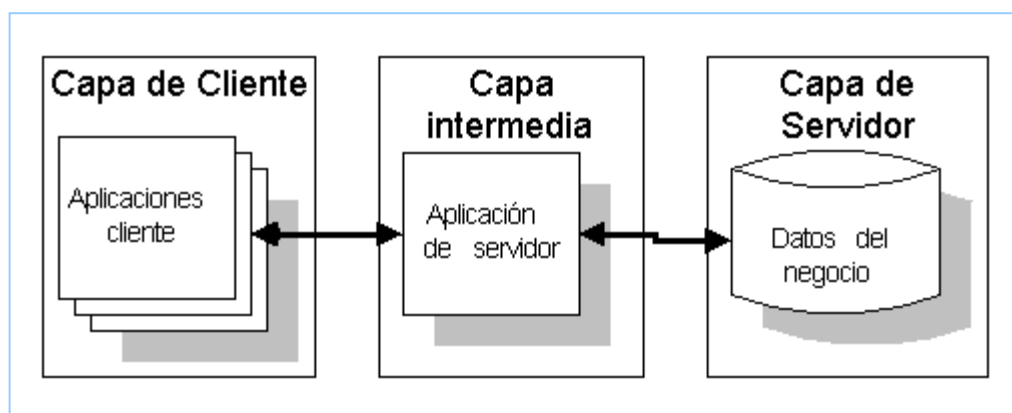


Figura 1.9. Arquitectura de Tres capas. Caso mas general de arquitectura multicapas.

El número de capas típico que se utiliza en esta arquitectura son tres, capa de datos, de negocio y de presentación. Así en la capa de datos encapsulamos todas las funciones de base de datos, las funciones de interfaz de usuario en la capa de "Presentación" y finalmente toda la lógica de operación en la capa de "Negocios".

Esta encapsulación tiene como objeto ofrecer una colección más simple de funciones que permiten desarrollar aplicaciones a alta velocidad y bajo costo. El costo de mantenimiento del sistema disminuye notablemente ya que una modificación en una capa no debe afectar a las demás.

- **Capa de Datos**

El nivel de servicios de datos es responsable de: Almacenar, recuperar y mantener los datos así como verificar la integridad de los mismos.

Los servicios de datos tienen una variedad de formas y tamaños, incluyendo los sistemas de administración de bases de datos relacionales, servidores de correo electrónico y sistemas de archivos.

- **Capa de Aplicación o de Negocios**

La capa de aplicación es el “puente” entre un usuario, representado en la capa de presentación, y los servicios de datos, que se encuentran en la capa de datos. Los servicios de esta capa responden a peticiones del usuario (u otros servicios de negocios) para ejecutar una tarea de este tipo. Cumplen con esto aplicando procedimientos formales y reglas de negocio a los datos relevantes. Esto aísla al usuario de la interacción directa con la base de datos.

El nivel de servicios de negocios es responsable de: Recibir la entrada del nivel de presentación, interactuar con los servicios de datos para ejecutar las operaciones de negocios para los que la aplicación fue diseñada a automatizar (por ejemplo, la preparación de puestos por ingresos, el procesamiento de ordenes y así sucesivamente) y de enviar el resultado procesado al nivel de presentación.

- **Capa de Presentación**

Los servicios de presentación proporcionan la interfaz necesaria para presentar información y reunir datos. También aseguran los servicios de negocios necesarios para ofrecer las capacidades de transacciones requeridas e integrar al usuario con la aplicación para ejecutar un proceso de negocios.

Los servicios de presentación generalmente son identificados con la interfaz de usuario, y normalmente residen en un programa ejecutable localizado en la estación de trabajo del usuario final. El cliente proporciona el contexto de presentación, generalmente un navegador como Microsoft Internet Explorer o Netscape, que permite ver los datos remotos a través de una capa de presentación HTML, u otro tipo de aplicación.

Mediante el uso de capas, se separa la programación que da acceso a los datos en las bases de datos de otros contenidos del sistema. Esto ayuda a asegurar que los desarrolladores estén libres para enfocarse en escribir su lógica de negocios en componentes sin preocuparse acerca de cómo se muestra la salida. Recíprocamente, esto da libertad a los diseñadores de usar herramientas familiares para modificar la interfaz.

La capa de servicios de presentación es responsable de: Obtener información del usuario, enviar la información del usuario a los servicios de negocios para su procesamiento, recibir los resultados del procesamiento de los servicios de negocios y presentar estos resultados al usuario.

Aunque el diseño de las interfaces depende exclusivamente de quien lo hace y hacia quien va dirigido, existen ciertos lineamientos básicos a tener en cuenta que determinan que tan utilizable va a ser el sistema, por ejemplo, si la interfaz de usuario es pobre o no tiene estándares claros, el usuario no se sentirá cómodo al usar la aplicación, o peor aún, dejará de usarla.

Existen unos lineamientos básicos que se deben tener en cuenta a la hora de diseñar y desarrollar interfaces. En la siguiente tabla se presentan algunos de dichos lineamientos, una buena referencia es “Los siete pecados mortales de un

sitio Web (y por qué deben de evitarse a toda costa)³ de Jesse Berst que describe los grandes errores que se pueden cometer en el diseño de un sitio Web.

LINEAMIENTO	DESCRIPCIÓN
Consistencia:	Todas las pantallas de la aplicación Web deben de tener una distribución consistente de imágenes, texto y controles gráficos. Ante acciones de usuario comunes (elegir un elemento de una lista desplegable, dejar un campo en blanco donde no es permitido) el sistema debe mostrar mensajes con la misma estructura sin importar en que parte del sistema ocurra el evento.
Flujo de pantallas	El paso de una pantalla a otra debe de ser coherente con el trabajo que intente realizar el usuario. Por ejemplo, no es nada conveniente hacer que el usuario tenga que pasar por una pantalla de configuración de colores en pantalla para llegar a la pantalla de configuración de márgenes de impresión.
No sobre-poblar las pantallas	Al presentar gran cantidad de controles gráficos en una misma pantalla dificultan la comprensión de la misma. Si para realizar una tarea determinada se requiere de la obtención de mucha información por parte del usuario, es muy recomendable dividir en diferentes pantallas sucesivas la captura de información y/o establecimiento de opciones
Agrupar elementos relacionados	Una buena práctica en la creación de interfaces de usuario es la agrupación de elementos que estén relacionados entre sí. Por ejemplo, delimitando por un rectángulo o algún otro tipo de imagen los datos comunes.
Navegación inconsistente.	A veces se hace clic en una barra de menú lateral, otras es un menú despegable. La ubicación de los títulos, gráficos, vínculos, etc. debe de ser consistente en todas las páginas del sitio
Vínculos rotos.	Es muy importante verificar que todos los vínculos apunten a la página que deben, de lo contrario se hace difícil la navegación
Sitios que requieren de un navegador específico	Es muy molesto toparse con sitios que están diseñados para un navegador específico, así como los que requieren que se baje un plug-in ⁴ determinado sin ofrecer una versión del documento que no requiera de dicho plug-in.

³ Jesse Berst; -Seven Deadly Website Sins-. http://www.zdnet.com/anchordesk/story/ story_1716.html. 17 Abril 2001

⁴ Componente de software que agrega funcionalidad a un navegador sin necesidad de recom-pilarlo. Un ejemplo típico de plug-in es el de Macromedia Flash, el cual es usado para visualizar animaciones vectoriales en páginas Web. Éste plug-in puede ser instalado para el Internet Explorer y Netscape Navigator.

No poner información de contacto:	dirección de la entidad, números telefónicos, etc
Sitios que abren otras ventanas de navegación.	Es molesto para el navegador ingresar a un sitio y que éste abra nuevas ventanas de navegación con vínculos a otros sitios.
Símbolos de “En construcción”.	La mayoría de los sitios en el Web están en constante evolución, Es poco profesional indicar que el sitio está en construcción.

Tabla 1.7. Lineamientos básicos para diseñar y desarrollar Interfaces

2.5. ACTOS ADMINISTRATIVOS

Con el fin de aclarar al lector el concepto de Acto Administrativo bajo el cual se trabaja en el presente proyecto, se incluye este apartado. Por definición el Acto Administrativo, es una declaración de voluntad de una entidad administrativa que, produce efectos jurídicos sobre intereses, obligaciones o derechos de los administrados dentro de una situación concreta.

Con esta definición, se precisan los más importantes rasgos que la buena doctrina administrativa atribuye al Acto Administrativo:

Su carácter público, por ser emitidos por una Administración Pública, en ejercicio de sus potestades administrativas –*ius Imperium*- y, por tanto, sometido a reglas de Derecho Público. De ahí su carácter imperativo y obligatorio.

Su carácter de acto jurídico especial, pues requiere de una declaración de voluntad, aun cuando ésta pueda ser expresa o presunta. Se deja de lado, por ello, cualquier comportamiento o actividad material o por vías de hecho de las entidades de la Administración Pública.

Su carácter de acto definitivo y externo, pues solo éste puede trascender la esfera interna de la Administración Pública y afectar intereses, obligaciones y derechos de los administrados. Se deja de lado, en este sentido, los actos preparatorios o de administración interna destinados a organizar o hacer funcionar los servicios y actividades de la Administración Pública.

Su carácter de acto determinado y concreto, pues las prestaciones que impone a la vez que deben ser precisas, deben recaer sobre sujetos conocidos y determinados. La definición no comprende, en consecuencia, a los actos abstractos o generales, como las directivas y reglamentos.

2.5.1 El objeto o contenido de los actos administrativos

Todo acto administrativo debe tener un objeto. No se admitiría un acto administrativo sin él, esto porque que el objeto es aquello que se decide, declara o certifica por la autoridad en cada caso. Vale decir que, en términos sencillos y tratándose de resoluciones, es lo que aparece en la parte resolutive de las mismas.

El objeto del acto administrativo es, así, aquella prestación obligatoria que se ordena realizar en dicho acto, y como tal debe ser, cuando menos, preciso, física y jurídicamente posible, y congruente con la motivación, conforme al siguiente detalle:

- ✓ El Objeto del Acto Administrativo es preciso y claro cuando puede determinarse inequívocamente sus efectos jurídicos.

- ✓ El objeto del Acto Administrativo es físicamente posible cuando la prestación que contiene, esto es la orden de dar, hacer o no hacer que conlleva, es factible de realizarse por el obligado.

- ✓ El Objeto del Acto Administrativo es jurídicamente posible cuando la prestación que contiene no contraviene alguna prohibición legal o no afecta algún derecho o interés legítimo y no viola norma legal alguna.
- ✓ El objeto del Acto Jurídico es congruente con la motivación cuando lo que se decide encuentra sustento en los hechos probados que aparecen en la motivación y amparo en los fundamentos jurídicos que allí mismo se exponen. Asimismo, este principio, obliga a pronunciarse sobre todas las cuestiones planteadas por los administrados.

2.5.2 Las Resoluciones

Se ha decidido profundizar un poco más en este tema, por ser las resoluciones el tipo de acto administrativo más utilizado en la mayoría de IES. Las resoluciones tienen las siguientes partes:

Expositiva, que empieza con el término “VISTOS” y contiene una breve síntesis del origen del acto que se emite.

Considerativa, que empieza con la frase “CONSIDERANDO” y contiene una relación de los argumentos de hecho y de derecho que fundamentan la decisión.

Resolutiva, que empieza con el término “SE RESUELVE” y contiene lo que se ordena o se decide.

Dispositiva, que se expresa con las frases: “CÚMPLASE, PUBLIQUESE Y ARCHIVESE” o fórmulas parecidas, que expresa el carácter imperativo de la resolución.

2.6. RESUMEN CAPITULO 2

El marco conceptual presentado en este capítulo no profundiza en las diferentes temáticas ya que no es el propósito del presente proyecto, por el contrario, solo se proporcionan al lector los aspectos básicos necesarios para la comprensión de dicho proyecto. Inicialmente se presentaron algunos conceptos relacionados con las tecnologías de Internet, así como la relación existente entre internet – Intranet –extranet, posteriormente se trató sobre las diferentes tecnologías y la arquitectura para el desarrollo de software; por último se presentó una breve teoría sobre los actos administrativos.

RESUMEN PARTE I

Para tener una idea clara de la magnitud del trabajo de grado que se plantea, es necesario enunciar las bases fundamentales que permitan visualizar los objetivos que se desean lograr. Es por eso que esta parte marca el comienzo de un proceso completo en donde cada etapa se realiza con el fin de dar los parámetros necesarios para seguir avanzando hasta cumplir todas las metas propuestas.

Inicialmente y con base tanto en los principios y objetivos que el Gobierno Nacional tiene para Educación Superior, como en una investigación realizada en diferentes Instituciones de Educación Superior de Bucaramanga, se presentó la problemática existente en dichas Instituciones, posteriormente se planteó la alternativa de solución junto con los beneficios que trae tanto a la UIS como a la Institución que implante el sistema y por último se presentó el marco conceptual en el cual se enmarca el proyecto.

3. NOTAS BIBLIOGRAFICAS

- ☑ Greer, Tyson. **Así son las Intranets**. Primera Edición. McGraw Hill. España, 1998. Este libro aportó la conceptualización básica que se necesita para diseñar una Intranet, adicionalmente tocó temas de gran importancia como la seguridad y los riesgos que se deben tener en cuenta a la hora de desarrollarla. Otro aspecto que destaca el libro y fue de gran importancia en esta parte, es el beneficio que obtiene una organización al implantar una Intranet.

- ☑ David Garret y otros. **Intranets al descubierto**. Editorial Prentice Hall. España, 1997. Este libro complementó la información que se necesitó para fundamentar esta parte, porque profundiza en aspectos técnicos y características tanto de hardware como de software que se deben tener en cuenta.

- ☑ Bobadilla, Jesús. Alcocer, Alejandro y otros. **Superutilidades para Webmasters**. Primera Edición. Editorial McGraw Hill. España, 1999. Una vez recopilado y estudiado el material sobre intranets, fue necesario buscar información que permitiera implementar el sistema deseado, para ellos se tuvo en cuenta este libro ya que proporciona una completa explicación sobre algunas de las tecnologías más conocidas y utilizadas en las aplicaciones web, abarcando desde la creación de páginas estáticas hasta la seguridad en la web. Además muestra cómo instalar y administrar servidores web, resaltando la importancia de servidores y navegadores seguros.

- ☑ Danesh, Arman. **Aprendiendo Javascript en una semana**. Prentice Hall. México, 1996. Debido al tipo de tecnologías que se planea manejar, en esta parte el libro dio una idea general de el tipo de cosas que se puede hacer con Javascript.

- ☑ Instituto Colombiano para el Fomento de Educación Superior –ICFES. **Educación Superior Compendio de Normas.** División de procesos editoriales del ICFES. Santa Fe de Bogotá 1995. En este compendio de normas se pudo apreciar la forma como están clasificadas las normas de educación superior, tema importante a la hora de implementar el manejo de ellas.

- ☑ Las siguientes tesis de grado se tomaron en cuenta ya que aportaron una idea del proceso evolutivo que han tenido los sistemas de información intranets desarrollados al interior de la UIS, de igual manera permiten ver los aportes que estos sistemas han hecho y los aspectos que han destacado; esto permitió restringir un poco el ámbito de las intranets al papel que deben jugar cuando se implementan para organizaciones como Instituciones de Educación Superior.

- ☑ Forero, Oscar. Gereda, Reymarx. Mariano Nikolai. **Desarrollo de un Sistema de Intranet para la Universidad Industrial de Santander.** Bucaramanga, 1998. Se propuso la creación de un sistema informático que aproveche las tecnologías de la red Internet a través de la infraestructura de la red local de la universidad.

- ☑ Suárez Alvarez, Edwin Ramón. **Sistema Intranet - Extranet para la Seccional de la Universidad Industrial de Santander en Barrancabermeja.** Bucaramanga, 2001. Se desarrollo un sistema que posea mecanismos de comunicación interna y externa de la seccional de la Universidad Industrial de Santander en Barrancabermeja, para apoyar el tratamiento de la información tanto dentro de la institución como el intercambio de ésta con la sede de la universidad en Bucaramanga.

- ☑ Velez Ramos, Jeimy Beatriz. Vergara Laurens, Idalides José. **Sistema de Información Intranet para la Escuela de Ingeniería de Sistemas e**

Informática de la Universidad Industrial de Santander. Bucaramanga, 2000. Se propone el desarrollo de un sistema de información, tecnología Internet, orientado a apoyar la gestión del consejo y comité de trabajos de grado, apoyar el proceso de elaboración, registro y publicación de información referente a los docentes y miembros de la Escuela de Ingeniería de Sistemas e Informática de la Universidad Industrial de Santander.

PARTE II – FASE DE INICIO⁵

En esta parte se presenta conceptualmente el proyecto. Inicialmente se identifican las diferentes características o requerimientos que el sistema demanda y que están relacionados tanto con los aspectos técnicos a tener en cuenta como con las facilidades que el sistema debe ofrecer, posteriormente y con base en dichos requerimientos se halla el modelo del dominio, éste permite comprender un poco más el contexto del sistema ya que permite ver cómo se relacionan las diferentes entidades básicas. Paralelamente al modelo del dominio se identifican los casos de uso más importantes, éstos incluyen los actores que intervienen en el sistema y cómo interactúan con el mismo. Una vez identificados los casos de uso, éstos se analizan y se agrupan en paquetes que conformarán la arquitectura candidata, dicha arquitectura junto con los riesgos críticos identificados permiten vislumbrar tanto el impacto como la viabilidad del proyecto a desarrollar.

Esta fase se desarrolla a través de una sola iteración, con un tiempo estimado en 51 días, fecha que marcará el primer hito en el desarrollo de este proyecto.

⁵ Si desea más información acerca de esta fase, remítase al anexo A

CAPITULO 3

4. RECOPIACIÓN DE REQUISITOS E IDENTIFICACIÓN DE CASOS DE USO

4.1. INTRODUCCION

Este capítulo permite tener más claridad acerca del contexto global en el cual se enmarca el proyecto con base en tres aspectos vitales como son: el aporte del sistema a la solución de la problemática en las IES, los tipos de usuarios que interactuarán con el sistema y por último la identificación de los diferentes riesgos con el fin de controlarlos para que no afecten el desarrollo del proyecto.

Inicialmente se identifican las características generales que el sistema requiere para dar solución a la problemática presentada, posteriormente se presenta el modelo del dominio y con base en éste, se identifican los actores y casos de uso relevantes para el sistema, finalmente se hace una clasificación de los principales riesgos identificados.

4.2. LISTA DE CARACTERÍSTICAS

En la tabla 2.1. se presentan las principales características que proporcionan una descripción general del sistema.

Tipo	Descripción	Justificación
Aspectos Técnicos	Sistema implementado en red	En las Instituciones de Educación Superior pueden existir diferentes fuentes que emiten documentación normativa así como varios usuarios interesados en tener acceso a la misma, pero la forma en que se produce, procesa y distribuye actualmente dicha información tiende a ser obsoleta, además de agredir uno de los principios fundamentales de la Educación Superior, como es el de incentivar en su comunidad y principalmente en sus educandos un espíritu reflexivo respetando la libertad de pensamiento y el pluralismo ideológico; es por esto que se hace necesario generar un cambio en este proceso utilizando tecnologías y herramientas que faciliten dicho cambio.
	Sistema desarrollado teniendo en cuenta los recursos hardware y software que la institución posea.	Para que el sistema tenga el impacto que se desea, debe además de brindar apoyo a la gestión normativa, estar económicamente al alcance de las IES, esto implica que se desarrolle teniendo en cuenta los recursos que las instituciones poseen para su implantación.
Facilidades del Sistema	Asignar identificación de la institución y de la intranet en el sistema.	Es de anotar que el sistema a desarrollar, es un producto estándar, es decir, puede ser implantado en cualquier institución de educación superior, por este motivo debe permitir la asignación de características propias de dicha institución.
	Elegir tipo de letra y el color deseado para las diferentes partes de las presentaciones	
	La generación en el sistema de: ✓ entes emisores de documentación normativa. ✓ Tipo de Actos administrativos.	
	Uso de cuentas de usuario	
	Inclusión de documentos normativos en la base de datos	

	Consulta eficiente de la información normativa.	Esta característica es muy importante ya que influye en la utilización del sistema, por tanto dicha consulta debe devolver la información principal de los actos administrativos como: título, autores, síntesis general, tema, entre otros
	Publicación de Actos Administrativos, proyectos de dichos actos y eventos de interés general	Este aspecto permite que los usuarios puedan conocer las normas existentes en la institución así como las propuestas de actos administrativos; de esta manera la comunidad universitaria puede vincularse activamente en dicho proceso.
	Impresión de los documentos normativos	Esto por si se desea tener una copia en archivo físico de los actos administrativos o en el caso de que cualquier usuario desee tener una copia impresa ya sea de un acto administrativo o proyecto del mismo.

Tabla 2.1. Características del sistema

4.3. MODELO DEL DOMINO

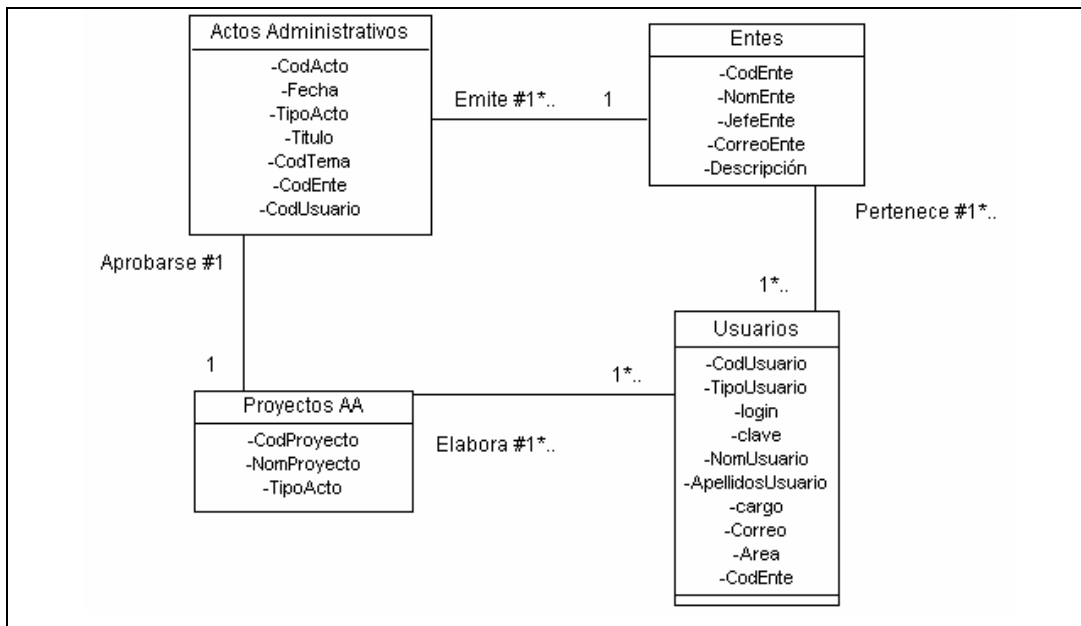


Figura 2.1. Modelo del dominio

El sistema utilizará las tecnologías internet para publicar, tanto los actos administrativos que emiten los diferentes entes encargados de gestionar dicha documentación como los proyectos de actos administrativos que los miembros de la comunidad universitaria deseen proponer a la institución. Esta

información puede ser consultada y actualizada por los usuarios autorizados para realizar dichas funciones.

4.4. ACTORES

En la tabla 2.2. se presentan los actores que interactúan en el sistema.

Actor	Descripción
Comunidad Universitaria	La conforman todas aquellas personas que tienen alguna relación directa con la institución; por ejemplo estudiantes, profesores, decanos, secretarías, personal administrativo, empleados entre otros. Entre las principales acciones que efectúan con el sistema están consultar e imprimir actos administrativos, dar sugerencias sobre proyectos de actos administrativos y consultar eventos.
Administrador	Es aquella persona que hace parte de la Comunidad Universitaria y cuya función principal es el administrar el Sistema de Información Intranet.
Secretaría Del Ente	Como su nombre lo indica, es la persona que ocupa el cargo de secretaria del ente emisor, y es a quién se le ha delegado la función de guardar y divulgar a las personas y/o entes pertinentes, los actos administrativos que el ente aprueba, rechaza o aplaza.
Autores de Proyectos de Actos Administrativos	Son miembros de la comunidad universitaria, a quienes la institución otorga el derecho a elaborar proyectos de actos administrativos, estos es, si se identifica una problemática que puede ser solucionada mediante un acto administrativo, este autor elabora una propuesta de dicho acto, la cual, después de ser publicada para recibir críticas, sugerencias, etc. Puede ser enviada a estudio al ente pertinente para su aprobación, en caso que se apruebe, pasa automáticamente de ser proyecto de AA, a ser Acto Administrativo como tal.

Tabla 2.2. Actores

4.5. MODELO DE CASOS DE USO

4.5.1 Casos de Uso Comunidad Universitaria

El siguiente diagrama de UML, figura 2.2 ilustra las funciones básicas que desempeñan los miembros de la comunidad universitaria en el sistema.

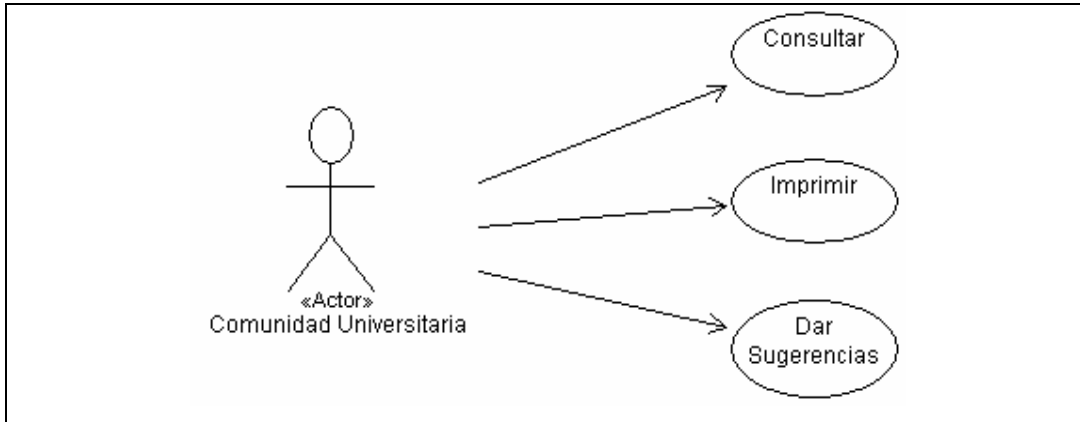


Figura 2.2. Casos de uso de la comunidad universitaria

En la tabla 2.3. se presentan los casos de uso correspondientes a la comunidad universitaria, con su respectiva descripción.

Actor	Casos de uso	Descripción caso de uso
Comunidad Universitaria	Consultar Acto Administrativo	<p>Este caso de uso es utilizado por todos los actores para consultar información sobre los Actos Administrativos emitidos. Una vez iniciado el caso de uso, los pasos a seguir tienen relación con las restricciones que se deseen colocar a las búsquedas y son los siguientes:</p> <ul style="list-style-type: none"> ✓ Seleccionar el Ente o Entes dentro de los cuales se va a buscar la información ✓ Seleccionar el acto o actos administrativos pertinentes. ✓ Seleccionar el autor. ✓ Incluir (si se conoce) otra información como el tema, la fecha, el título y las palabras claves.

	Consultar Proyectos de AA	<p>Este caso de uso permite hacer búsquedas teniendo aspectos como:</p> <ul style="list-style-type: none"> ✓ Tipo de Acto Administrativo ✓ Autor de dicho documento. ✓ Tema ✓ Título ✓ Palabras clave <p>Así mismo se pueden enviar sugerencias al autor de dicho acto administrativo.</p>
	Imprimir	Este caso de uso permite a los actores imprimir el cuerpo de determinado acto administrativo.
	Dar sugerencias	Este caso de uso permite a la comunidad universitaria dar sugerencias sobre los diferentes proyectos de actos administrativos que estén publicados

Tabla 2.3. Casos de Uso Comunidad Universitaria

4.5.2 Casos de Uso Administrador

El siguiente diagrama de UML, figura 2.3 ilustra las funciones básicas que desempeña el administrador en el sistema.

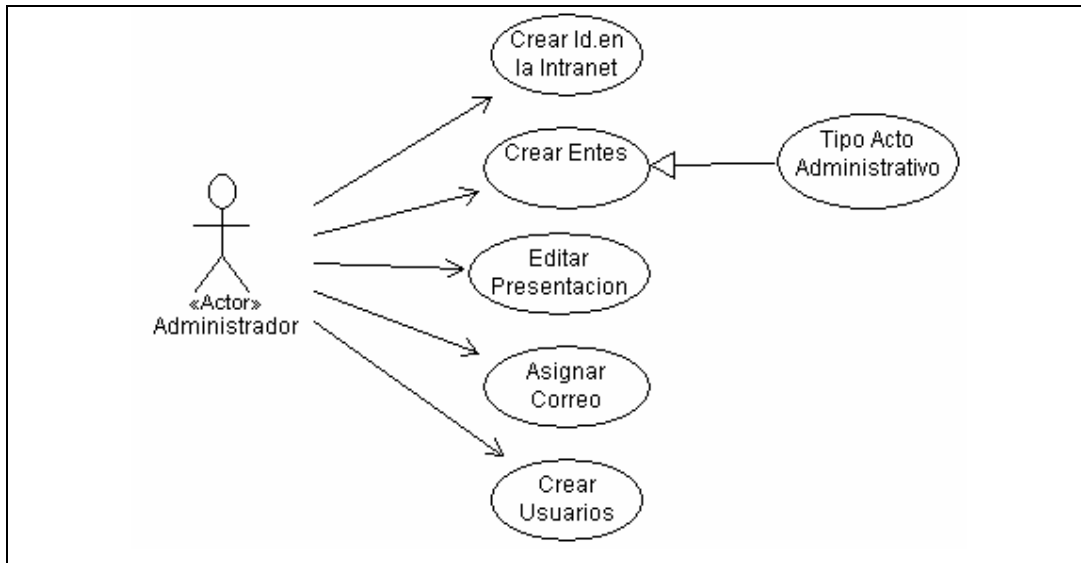


Figura 2.3. Casos de uso del Administrador

En la tabla 2.4. se presentan los casos de uso correspondientes al administrador respectiva descripción.

Actor	Casos de uso	Descripción caso de uso
Administrador	Crear Identificación	<p>El administrador utiliza este caso de uso para incluir los datos que identifican tanto a la Institución como al Sistema de Información Intranet. Una vez iniciado el caso de uso crear identificación, las actividades a realizar son los siguientes:</p> <p>Asignar la dirección IP y el Nombre que va a llevar el Sistema de Información Intranet</p>

	<p>Crear Ente</p>	<p>El administrador utiliza el caso de uso crear ente para incluir jerárquicamente en el Sistema Intranet de Información, los nombres de los diferentes entes de la Institución que emiten documentación normativa. Los pasos que se deben seguir son:</p> <p>Incluir el nombre del Ente correspondiente</p> <p>Definir qué autoridades conforman los diferentes entes.</p> <p>Incluir la dirección IP del servidor para dicho Ente.</p> <p>Elegir los documentos que este Ente emite.</p> <p>Asignar el correo electrónico para dicho Ente.</p>
	<p>Crear Acto Administrativo</p>	<p>El administrador utiliza este caso de uso para crear los nombres de los actos administrativos para cada ente. Una vez que el caso de uso comience, ya ha sido creado el Ente (por otro caso de uso llamado Crear Ente) al cual se le asignaran los actos administrativos correspondientes. Los pasos son los siguientes:</p> <p>Elegir el acto administrativo que va a incluir.</p> <p>Dar nombre al acto administrativo elegido</p>
	<p>Editar presentación</p>	<p>El administrador utiliza el caso de uso editar presentación para seleccionar el color de fondo y el tipo de letra para las plantillas de presentación. Una vez se ha iniciado el caso de uso, los pasos a realizar son los siguientes:</p> <p>Seleccionar el color de cada una de las partes (frames) de las páginas del Sistema de Información Intranet.</p> <p>Elegir el tipo de letra para el título, texto y demás hipervínculos.</p> <p>Incluir el Nombre y Logo de la Institución.</p> <p>Incluir datos Institucionales generales como Misión, visión, Reseña Histórica.</p>

	Crear usuario	Por medio de este caso de uso, el administrador puede asignar los diferentes niveles de acceso que tienen los diferentes usuarios(Administrador, Secretaria del Ente, Autor de Proyectos y Comunidad Universitaria) al Sistema de Información Intranet.
	Asignar Correo Electrónico	El administrador por medio de este caso de uso asignará correo electrónico a aquellas personas y/o entes pertenecientes a la institución que la institución crea necesario para llevar a cabo de forma eficiente la comunicación y notificación de actos administrativos.

Tabla 2.4. Casos de Uso Administrador

4.5.3 Casos de Uso Secretaría del Ente

El siguiente diagrama de UML, figura 2.4 ilustra las funciones básicas que desempeña la secretaria del ente emisor en el sistema.

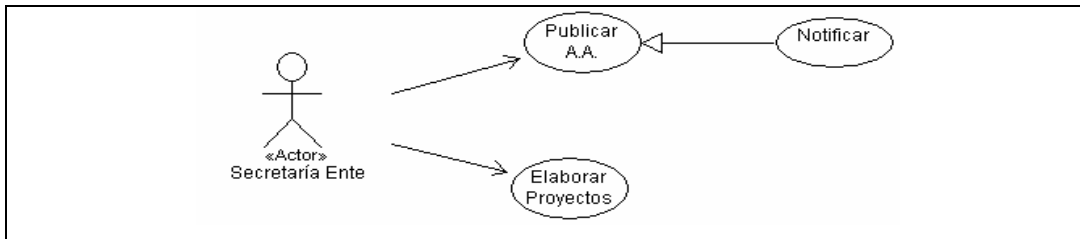


Figura 2.4. Casos de Uso Secretaría del Ente

En la tabla 2.5. se presentan los casos de uso correspondientes a la secretaria de ente, con su respectiva descripción.

Actor	Casos de Uso	Descripción Caso de Uso
	Publicar Acto Administrativo	Este caso de uso permite publicar y guardar automáticamente los actos administrativos emitidos por los diferentes Entes de la Institución. Los pasos que se deben seguir una vez se inicia el caso de uso son los siguientes: Seleccionar el nivel de acceso que se le va aplicar al Acto Administrativo involucrado. Notificar a los Entes involucrados en dicho acto administrativo.
	Elaborar Proyecto	La secretaria del ente elabora los Actos Administrativos que son guardados en el sistema como proyectos de Actos Administrativos hasta cuando sean firmados por la autoridad correspondiente.

	Notificar	Una vez emitidos los actos administrativos, la secretaria debe notificar de dicha emisión a los entes involucrados en este.
--	-----------	---

Tabla 2.5.. Casos de Uso Secretaría de Ente

4.5.4 Casos de Uso Autores de Proyecto

El siguiente diagrama de UML, figura 2.5 ilustra las funciones básicas que desempeña la secretaria del ente emisor en el sistema.

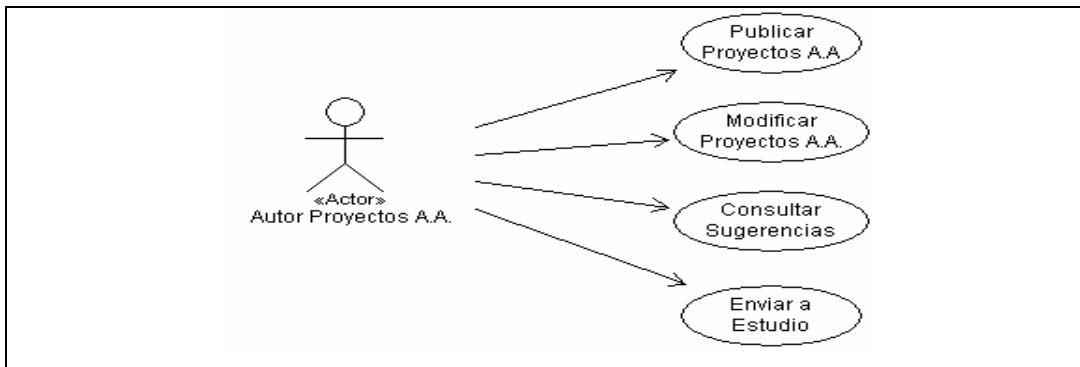


Figura 2.5. Casos de uso Autor Proyectos A.A.

En la tabla 2.6. se presentan los casos de uso correspondientes al Autor Proyectos AA, con su respectiva descripción.

Actor	Casos de Uso	Descripción Caso de Uso
Autor Proyectos AA	Publicar Proyecto	Este caso de uso permite a los autores de proyectos de A.A. publicar y guardar automáticamente los proyectos de actos administrativos que estén en construcción.
	Modificar Proyecto AA	Este caso de uso es utilizado por los autores de proyectos de A.A. cuando deseen hacer cambios a sus proyectos que tienen en la etapa de elaboración.
	Consultar Sugerencias	Por medio de este caso de uso, los autores de proyectos de Actos Administrativos, pueden consultar las sugerencias que la comunidad universitaria les hace respecto a los proyectos que tienen publicados.
	Enviar a Estudio	Este caso de uso es utilizado por los Autores de Proyectos cuando consideran que su Acto Administrativo está listo para ser sometido a estudio para su aprobación.

Tabla 2.6. Casos de Uso Autor Proyectos AA

4.6. LISTA INICIAL DE RIESGOS

Para dar a conocer el significado de riesgo tomaremos la definición que hace Robert Charette en su libro sobre análisis y gestión de riesgo

“En primer lugar, el riesgo afecta los futuros acontecimientos. El hoy y el ayer están más allá de lo que nos pueda preocupar, pues ya estamos cosechando lo que sembramos previamente con nuestras acciones del pasado. La pregunta es, podemos, por tanto, cambiando nuestras acciones actuales, crear una oportunidad para una situación diferente y, con suerte, mejor para nosotros en el futuro. Esto significa, en segundo lugar, que el riesgo implica cambio, que puede venir dado por cambio de opinión, de acciones, de lugares... [En tercer lugar,] el riesgo implica elección y la incertidumbre que entraña la elección. Por tanto el riesgo como la muerte y los impuestos, es una de las pocas cosas inevitables en la vida.”⁶

Al terminar la fase de inicio, tenemos identificados los principales riesgos que pueden afectar el proyecto. Para una mejor organización, hemos seguido la clasificación dada por Pressman⁷.

4.6.1 Riesgos del tamaño del producto

NOMBRE DEL RIESGO	¿CÓMO SE VA A MANEJAR?
Tamaño de la Base de Datos creada o empleada por el producto.	Para evitar el riesgo de que el espacio asignado por la Institución para almacenar la información que se va a manejar no sea suficiente, se realizará una estimación en kb del espacio físico que necesita cada Acto Administrativo y se sugerirá realizar copias de seguridad cada determinado tiempo o tamaño de la base de datos.
Número de usuarios que utilizan el sistema de forma	Con el fin de evitar la congestión en el sistema, se proporcionará un cuadro donde se establezcan las especificaciones tanto de red como de servidor

⁶ CHARETTE, R. N. Building Bridges over Intelligent Rivers. American Programmer, Vol. 5, Nº7, septiembre 1992 Págs. 2-9.

⁷ PRESSMAN, Roger S. Ingeniería del Software – un enfoque práctico. Cuarta Edición. McGraw Hill. España 1998. Págs. 90-94.

simultánea.	para un número máximo de usuarios;
-------------	------------------------------------

Tabla 2.7. Riesgos del Tamaño del Producto

4.6.2 Riesgos del impacto en el negocio

NOMBRE DEL RIESGO	¿CÓMO SE VA A MANEJAR?
Fecha Límite de Entrega	Para evitar retrasos en la entrega del producto y por ende la generación de un sobre costo que en nuestro caso sería ocasionado por el pago de un semestre adicional, además del costo de dirección, se elaboró un cronograma, en donde se presentan las diferentes actividades a realizar, teniendo en cuenta para cada una de ellas, un intervalo adecuado de tiempo.
Interoperatividad con otros productos o sistemas	Previendo la posibilidad de que el sistema se aislé y quede obsoleto, a causa de la imposibilidad de interactuar con otros sistemas, se hace necesario realizarlo con tecnologías que permitan la comunicación entre estos sistemas.
Riesgo de Mercado	Dado que en algunas IES, el área normativa no se considera de vital importancia, y por ende no se le asignan los suficientes recursos para el manejo de su información, el producto puede presentar problemas al comercializarlo. Para mitigar este riesgo se incluyen en el producto aspectos en el área normativa, que pueden resultar de interés para estas instituciones.
Riesgo Estratégico	Según la entidad interesada, existen diferentes opciones: UIS. Por ser esta institución, quien posee los derechos para la comercialización del producto, es responsabilidad de ella establecer las estrategias de mercadeo necesarias para obtener el máximo beneficio económico. IES. Para estas instituciones, el producto está enfocado hacia el apoyo organizacional de las mismas y no hacia un beneficio económico; esto puede traer como consecuencia que no tenga la acogida esperada. Para disminuir este riesgo es muy importante hacer notar el beneficio que en el ámbito organizacional puede traer la implantación de este producto.
Cantidad y calidad de la documentación del producto	Con el fin de evitar la desactualización del sistema y problemas generados por la administración del mismo, se elaborará la documentación que oriente y facilite la realización de estas actividades.

Tabla 2.8. Riesgos del Impacto en el Negocio.

4.6.3 Riesgos relacionados con el cliente

NOMBRE DEL RIESGO	¿CÓMO SE VA A MANEJAR?
Necesidades del cliente	Por la naturaleza estándar del producto, existe la posibilidad de que las necesidades identificadas en la investigación realizada en las IES, no refleje la veracidad de dichas necesidades. Para disminuir este riesgo, se hace necesario consultar otras fuentes de información.

Tabla 2.9. Riesgos Relacionados con el Cliente.

4.6.4 Riesgos del Proceso

NOMBRE DEL RIESGO	¿CÓMO SE VA A MANEJAR?
Nuevos Métodos de Análisis, Diseño o Pruebas	La metodología a seguir, es el proceso unificado de desarrollo de software. En nuestro medio, el uso de dicha metodología es relativamente reciente, esto ocasiona que la asimilación de la metodología tenga un cierto grado de dificultad y exista la posibilidad de cometer errores que se podrían evitar si hubiera tanto experiencia como suficiente documentación específica. Para minimizar este riesgo se han establecido estrategias de estudio grupal en el grupo de investigación.

Tabla 2.10. Riesgos del Proceso.

4.6.5 Riesgos del entorno de desarrollo

NOMBRE DEL RIESGO	¿CÓMO SE VA A MANEJAR?
Calidad y Elección del software	Dada la falta de experiencia en el manejo de las nuevas tecnologías software y hardware, se puede presentar deficiencias en la calidad del producto. Para mitigar este riesgo se han consultado fuentes relacionadas con el software a utilizar a fin de elegir la opción más adecuada para realizar el software de nuestro proyecto. Por otro lado, se han trazado estrategias para minimizar el tiempo de aprendizaje de dichas herramientas.

Tabla 2.11 Riesgos del Entorno de Desarrollo.

4.7. RESUMEN CAPITULO 3

En este capítulo se presentó a grandes rasgos la alternativa de solución que se plantea, no solo para suplir las necesidades de herramientas para el manejo

de información que en el área normativa poseen las IES, sino para sentar las primeras bases de un cambio que se requiere en el proceso de democratización en dicha área.

Con el fin de ilustrar claramente lo expuesto anteriormente, se siguió un proceso de desarrollo que inicia con la identificación de los requisitos y el ámbito del sistema, con base en ellos se construyeron los modelos de casos de uso con sus respectivos actores, sin olvidar la influencia que sobre ellos tienen los riesgos.

CAPITULO 4

5. ANÁLISIS

5.1. INTRODUCCION

Tomando como punto de partida los productos desarrollados en el capítulo anterior, se pasa a realizar un análisis de los mismos, esto es, se toman las principales funcionalidades del sistema, se descomponen en los diferentes procesos que interactúan para llevar a cabo dichas funcionalidades, posteriormente se analizaron dichos procesos para agrupar los casos de uso que tuvieran procesos con características similares, adicionalmente se elige el software a utilizar para el desarrollo del sistema, que en este caso son herramientas de libre distribución.

Los paquetes de análisis y los requisitos no funcionales proporcionan la arquitectura candidata sobre la cual se va a desarrollar el sistema.

5.2. MODELO DE ANALISIS.

5.2.1 Diagrama de Colaboración para la realización del Caso de Uso Búsqueda.

El diagrama de colaboración presentado en la figura 2.6 muestra el proceso que se lleva a cabo para realizar la consulta de los diferentes actos administrativos.

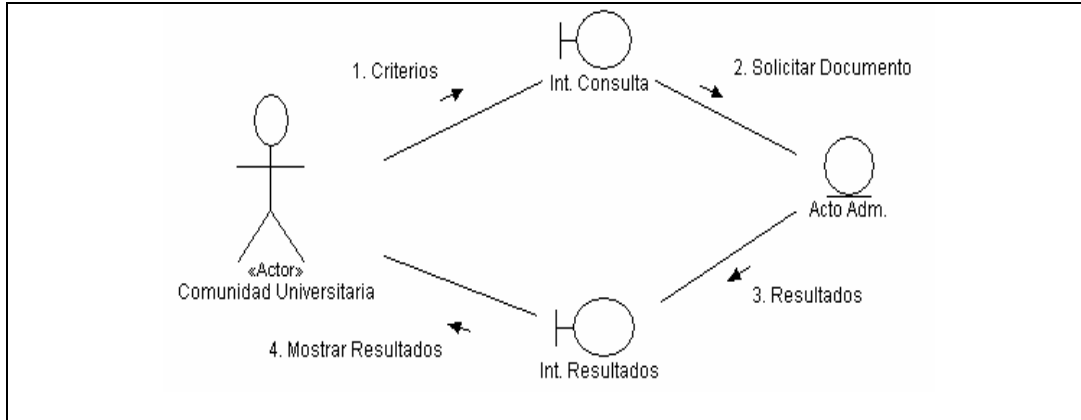


Figura 2.6. Diagrama de Colaboración: Consultar Actos Administrativos.

5.2.2 Diagrama de Colaboración para la realización del Caso de Uso Publicar Acto Administrativo.

En el diagrama de colaboración presentado en la figura 2.7. muestra el proceso llevado a cabo por la secretaría de ente para **publicar un acto administrativo**. En dicho diagrama se puede observar que todo acto administrativo pasa por una fase inicial en la cual se almacena como **proyecto de acto administrativo** y sólo hasta cuando es aprobado por el ente correspondiente pasa a publicarse y almacenarse como acto administrativo transfiriéndose desde la entidad **Proyectos AA** a la entidad **Actos Administrativos**

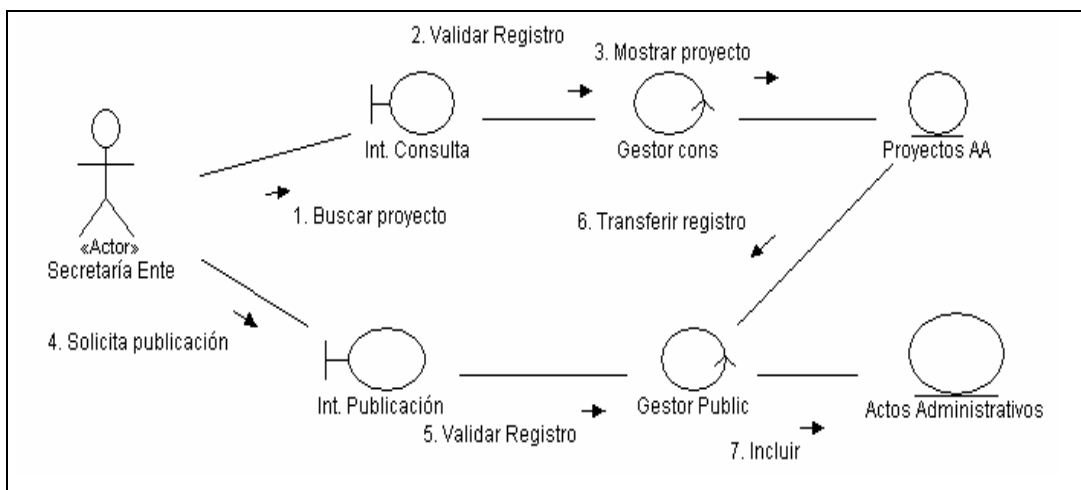


Figura 2.7.. Diagrama de Colaboración: Publicación de Actos Administrativos.

5.3. ARQUITECTURA CANDIDATA

5.3.1 Identificación Paquetes de Análisis.

Paquete de Análisis Crear usuario

En este paquete de análisis están los casos de uso implicados en la creación de los diferentes usuarios(actores) que interactuaran con el sistema.

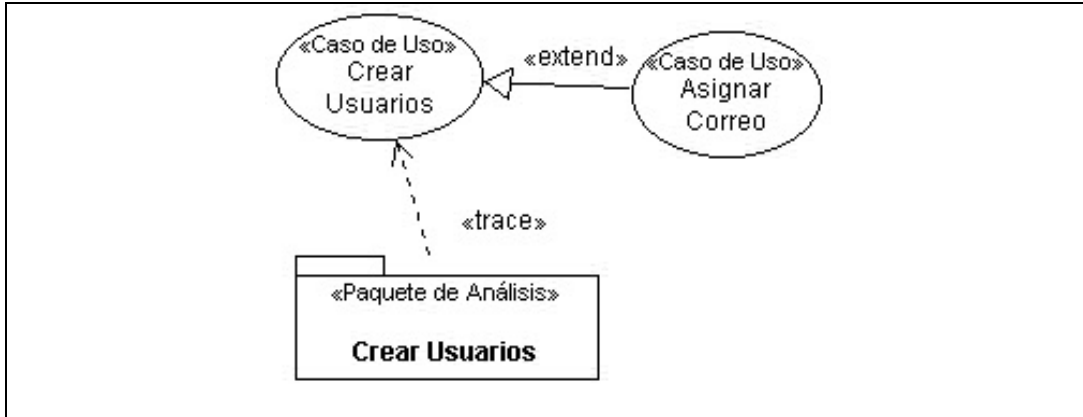


Figura 2. 8. Paquete de Análisis: Gestión Crear Usuario

Paquete de Análisis Configuración.

En el paquete de análisis Configuración se contemplan los casos de uso implicados en el proceso de configuración inicial de la intranet, en este caso particular solo el administrador es el actor involucrado en este proceso.

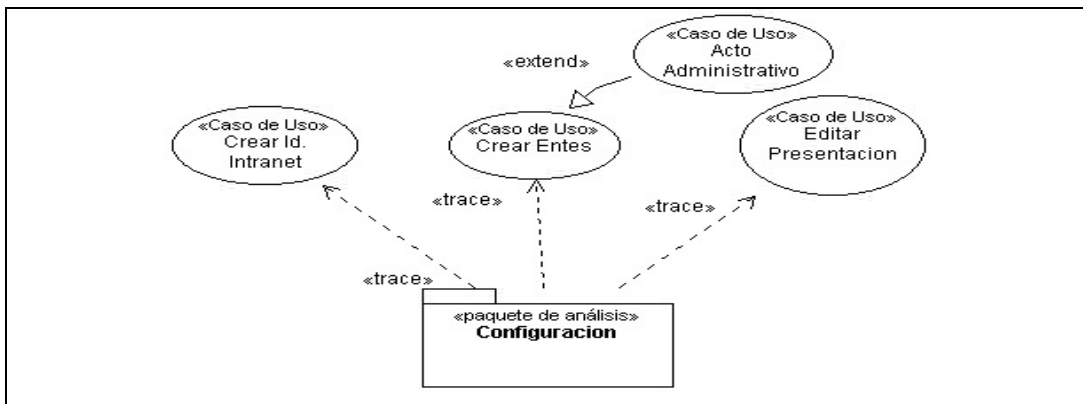


Figura 2.9. Paquete de Análisis: Configuración.

Paquete de Análisis Gestión de Publicación.

Este paquete de análisis contiene los casos de uso implicados con el proceso de publicación ya sea de actos administrativos por parte de algún ente o un proyectos de este tipo, por parte de algún autor de proyectos.

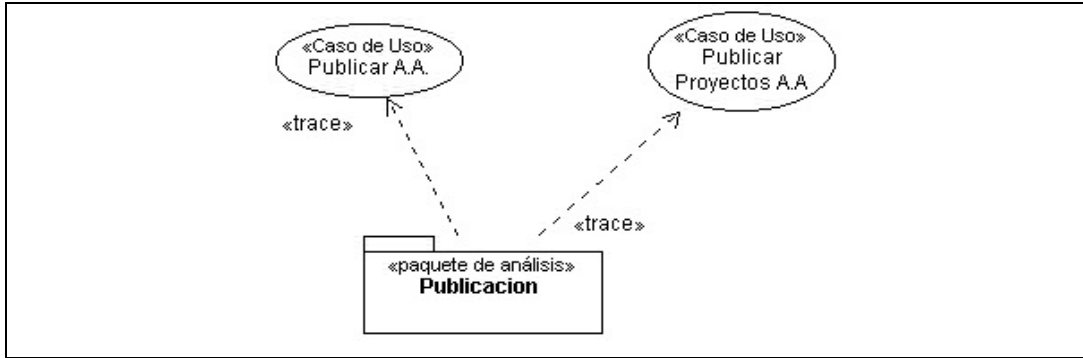


Figura 2.10. Paquete de Análisis: **Publicación.**

Paquete de Análisis Gestión de Consulta.

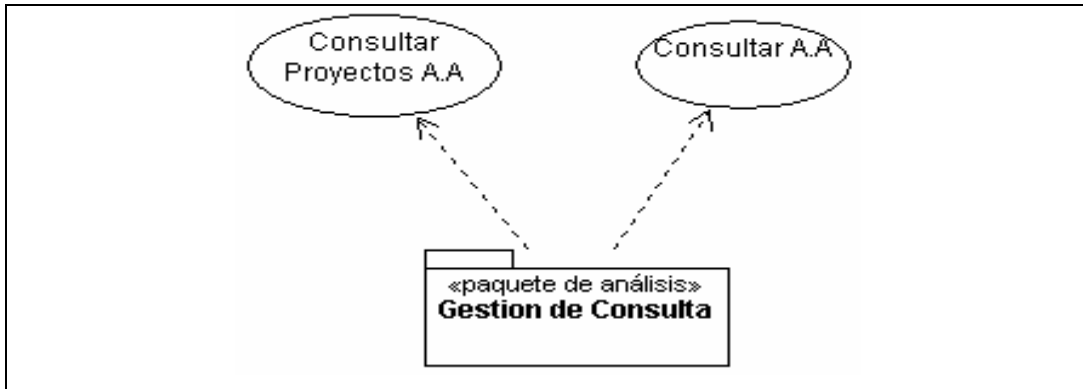


Figura 2.11. Paquete de Análisis: **Gestión de Consulta.**

En este paquete de análisis están los casos de uso implicados en la consulta de los documentos normativos almacenados en la base de datos.

5.3.2 Requisitos Especiales Comunes.

Para el desarrollo del sistema se han elegido las siguientes características:

- Sistema Operativo Servidor: Linux

- Sistema operativo clientes: Linux o Windows
- Lenguaje para aplicaciones web: PHP
- Base de Datos: PostgreSQL o MySQL.

5.4. RESUMEN CAPITULO 4

Con el flujo de trabajo de análisis se pudo apreciar con más detalle los procesos más importantes involucrados en el sistema, para esto se elaboró el modelo de análisis que en este punto permite ver el funcionamiento de los casos de uso más importantes, se agruparon dichos casos de uso en los paquetes de análisis con el fin de facilitar el manejo del sistema y por último se identificaron los requisitos especiales comunes bajo los cuales se va a desarrollar el sistema.

RESUMEN PARTE II

Siguiendo la metodología del Proceso Unificado de Desarrollo de Software, en este momento se finalizó la fase de inicio, de ella se puede concluir que el proyecto a desarrollar no solo es viable sino que contribuye a sentar bases no solo en el grupo de investigación STI sino en la Escuela de Ingeniería de Sistemas y a nivel más general de la Universidad Industrial de Santander – UIS sobre la importancia de la concordancia entre los objetivos de las Instituciones de Educación Superior – IES y la situación actual de las mismas, en especial lo relacionado con Educación Integral, participación democrática, desarrollo del espíritu reflexivo y demás temas que de alguna forma están relacionados con el proceso de elaboración, procesamiento y distribución de las normas en dichas Instituciones.

Dada la gran importancia de los casos de uso para el desarrollo del proyecto, se presenta a modo de resumen la Tabla 2.12 con los casos de uso que hasta el momento han sido identificados. Para cada uno de ellos, se define el grado de conocimiento que se tiene de él y si esto es suficiente para su total comprensión.

Casos de Uso Identificados	Identificado	Descrito	Analizado	Diseñados e implementados	¿Se comprende Completamente?
Crear Identificación	F. Inicio	F. Inicio			No
Crear Ente	F. Inicio	F. Inicio			No
Crear Tipo Acto Administrativo	F. Inicio	F. Inicio			No
Editar Presentación	F. Inicio	F. Inicio			Si
Crear usuario	F. Inicio	F. Inicio			No
Asignar Correo Electrónico	F. Inicio	F. Inicio			No
Consultar Acto Administrativo	F. Inicio	F. Inicio	F. Inicio		Si
Consultar Proyectos de A.A.	F. Inicio	F. Inicio			Si

Imprimir	F. Inicio	F. Inicio			Si
Dar sugerencias	F. Inicio	F. Inicio			No
Publicar Acto Administrativo	F. Inicio	F. Inicio	F. Inicio		Si
Notificar	F. Inicio	F. Inicio			No
Elaborar Proyecto	F. Inicio	F. Inicio			No
Publicar Proyecto	F. Inicio	F. Inicio			No
Modificar Proyecto	F. Inicio	F. Inicio			No
Consultar sugerencias	F. Inicio	F. Inicio			No
Enviar a estudio.	F. Inicio	F. Inicio			No

Tabla 2.12 Casos de Uso identificados durante la fase de Inicio

Por último, como ya han sido presentados los diagramas que describen la vista de la arquitectura en cada uno de los flujos desarrollados en esta fase, en la tabla 2.13 solo se especificará las secciones donde se encuentra dicha información.

Flujo	Sección	Vista	Comentario
Captura de Requisitos	Casos de uso en Detalle	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Figura 2.2. Casos de uso de la comunidad universitaria <input checked="" type="checkbox"/> Tabla 2.3 Casos de Uso de la comunidad universitaria <input checked="" type="checkbox"/> Figura 2.3. Casos de uso del Administrador. <input checked="" type="checkbox"/> Tabla 2.4. Casos de Uso Administrador <input checked="" type="checkbox"/> Figura 2.4. Casos de Uso Secretaría del Ente. <input checked="" type="checkbox"/> Tabla 2.5.. Casos de Uso Secretaría de Ente <input checked="" type="checkbox"/> Figura 2.5. Casos de uso Autor Proyectos A.A. <input checked="" type="checkbox"/> Tabla 2.6. Casos de Uso Autor Proyectos AA 	<p>Presenta los actores y casos de uso mas importantes del sistema, además de proporcionar una vista de la arquitectura a través de los casos de uso relevantes para la creación de la arquitectura candidata.</p>
Análisis	Diagramas de colaboración y Paquetes de Análisis.	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Figura 2.6. Diagrama de Colaboración: Consultar Actos Administrativos. <input checked="" type="checkbox"/> Figura 2.7.. Diagrama de Colaboración Publicación de Actos Administrativos <input checked="" type="checkbox"/> Figura 2.8. Paquete de Análisis Crear Usuarios. <input checked="" type="checkbox"/> Figura 2.9. Paquete de Análisis Configuración. <input checked="" type="checkbox"/> Figura 2.10. Paquete de Análisis Publicación. <input checked="" type="checkbox"/> Figura 2.11. Paquete de Análisis Gestión de Consulta. 	<p>Principales diagramas de colaboración y paquetes de análisis encontrados.</p>

Tabla 2.13. Diagrama: Vista de la arquitectura contenidas en laFase de Inicio.

6. NOTAS BIBLIOGRAFICAS

- ☑ JACOBSON, Ivar. BOOCH, Grady. RUMBAUGH, James. **El Proceso Unificado de Desarrollo de Software**. Primera Edición. Addison Wesley. España, 2000. Este libro fue de gran importancia en esta parte ya que sirvió como guía de aprendizaje y de comprensión de la metodología, en especial lo relacionado con la identificación de actores y casos de uso, elaboración y conceptualización de los modelos de casos de uso, del dominio y de análisis con sus respectivos diagramas
- ☑ TAPSCOTT Don. **La era de los negocios electrónicos**. Primera Edición. McGraw-Hill. Colombia 2000. En la conceptualización de la idea del proyecto y sobretodo de su importancia en el contexto social actual, fue interesante el papel de este libro en lo que respecta al segmento que trata el tema de “Gobierno y gobernabilidad en el mundo de las redes” ya que menciona aspectos interesantes sobre cómo las políticas de gobernabilidad de las organizaciones afectan el desarrollo y aplicación de las tecnologías de información y cómo a su vez dichas tecnologías transforman la estructura, funciones y alcance del gobierno o administración en estas organizaciones .
- ☑ Pressman, Roger. **Ingeniería del Software –Un Enfoque Práctico**. McGraw Hill. Mexico, 1998.

PARTE III- FASE DE ELABORACIÓN

En esta parte se presenta una estructuración más formal de las funcionalidades del sistema, para esto se toma como base la conceptualización que se llevó a cabo en la fase de inicio. En primer lugar se identifican casi en su totalidad los requisitos restantes, con base en estos, se complementan los diferentes modelos presentados en la parte I.

En este momento el modelo del dominio ofrece una estructura sobre la cual el sistema se va a apoyar, ya que se han identificado claramente tanto las entidades que lo conforman como las relaciones existentes entre dichas entidades. Igualmente el modelo de casos de uso se ha complementado con base en los requisitos identificados, posteriormente se realiza el proceso de análisis, en este, se agrupan los casos de uso, -en este caso- según su funcionalidad y se toman los más importantes para mostrar su estructura y funcionamiento interno, complementando de esta manera el modelo de análisis.

Con base en el análisis, se inicia el proceso de diseño, en el cual se modela una estructura que soporte tanto las acciones que el sistema va a realizar, como las propiedades y restricciones del mismo, esto es, se especifica qué recursos hardware y software se van a utilizar, así como las diferentes relaciones entre los subsistemas identificados.

En este punto, la implementación se enfoca solo a instalar y configurar el software del sistema así como de asignar los componentes necesarios a cada uno de los nodos detallados en el diagrama de despliegue.

Esta fase se desarrolla a través de una sola iteración, con un tiempo estimado en 72 días, fecha que marcará el segundo hito en el desarrollo de este proyecto.

CAPITULO 5

7. RECOPIACIÓN DE REQUISITOS FALTANTES, Y NUEVA VERSIÓN DE LOS MODELOS

7.1. INTRODUCCION

En este capítulo se presenta la mayoría de los requisitos faltantes que aportan funcionalidad al sistema, con base en ellos se realiza una nueva versión del modelo del dominio, éste junto con una descripción detallada de las entidades que lo componen, muestran la estructura bajo la cual es manejada la información. Posteriormente se expone la nueva versión del modelo de casos de uso, para este se tomó como base los paquetes de análisis de la fase de inicio, adicionándoles los nuevos casos de uso identificados.

7.2. LISTADO DE CARACTERÍSTICAS RESTANTES

En la tabla 3.1. se presentan las características restantes identificadas que proporcionan una descripción más completa del sistema.

Tipo	Descripción	Justificación
Facilidades del sistema	Publicación de la legislación sobre educación superior que expide el gobierno colombiano	Estas características son muy importantes porque permiten colocar y por ende consultar en línea el marco legal existente sobre educación superior, de esta manera se ofrece a la comunidad universitaria una forma cómoda y mas asequible de tener acceso a esta información tan importante a la hora de elaborar propuestas de actos administrativos o emisión de los mismos.
	Consulta de legislación que ha sido emitida sobre educación superior en Colombia	
	Publicación de eventos institucionales de interés general	El sistema además de manejar la información normativa, deja un espacio a los eventos institucionales con el fin de incentivar el acceso a la intranet..
	Consulta de eventos institucionales que se realizan en la institución	
	Consultar ayuda para la elaboración de documentos normativos	Este aspecto es de gran importancia, ya que ofrece un pequeño marco conceptual que orienta acerca de los aspectos que se deben tener en cuenta a la hora de elaborar documentos normativos, además de presentar un modelo de la estructura que debe tener dichos documentos.
	Notificar al administrador la existencia de proyectos nuevos.	Este requisito permite tener un control sobre la información que se publica en la red, ya que por las características del sistema, los usuarios pueden publicar sus propuestas de actos administrativos.

Tabla 3.1. Características restantes del sistema.

7.3. MODELO DEL DOMINIO

En la tabla 3.2 se presentan las diferentes entidades involucradas en el modelo del dominio, así como una breve descripción del contenido de las mismas.

ENTIDAD	DESCRIPCIÓN
inoinstitucion	Todos los datos correspondientes a la identificación de la institución y de la Intranet, como nombre, lugar donde se almacenarán los documentos, etc.
tipoletras	A esta entidad pertenecen los tipos de letra con sus respectivas características, que se utilizarán en la intranet
emblemas	Las imágenes que permiten identificar la institución dentro de la Intranet
colormarcos	Se almacenan el nombre y correspondiente color de los tres marcos a utilizar en la intranet
tipoaa	A esta entidad pertenecen los tipos básicos de actos administrativos como son, acuerdos, actas y resoluciones
actosente	Los diferentes tipos de actos administrativos que se manejan en la Institución interesada, estos pueden pertenecer a cualquiera de los tres tipos mencionados en tipoaa, pero pueden variar tanto en cantidad y como en nombre.
entes	Los diferentes organismos pertenecientes a la Institución y que pueden

	emitir Actos Administrativos.
usuarios	Son los diferentes miembros de la comunidad universitaria que pertenecen a un tipo de usuario diferente a “comunidad universitaria”.
tipousuario	Son los diferentes tipos de usuario que la el sistema maneja.
proyectosaa	Se almacenan los proyectos de actos administrativos que han sido elaborados por diferentes miembros de la Institución correspondientes a los actores “autor de proyecto”.
proyestudio	Se almacenan los proyectos de actos administrativos que se han enviado al ente correspondiente para estudio, dentro de ellos están los que han sido aplazados y los que aún no se han revisado.
proyrechazados	Como su nombre lo indica, a esta entidad corresponden aquellos proyectos que han sido rechazados por el ente.
temas	Los diferentes temas dentro de los cuales se pueden clasificar los actos administrativos.
normases	Aquí se almacenan las normas de educación superior expedidas por el gobierno nacional.
tipones	Son los tipos de normas que maneja el sistema, esta clasificación se hizo teniendo en cuenta la clasificación del compendio de normas de educación que expidió ICONTEC.
eventos	Son los eventos que los entes desean publicar a fin de que sean conocidos por la comunidad universitaria.

Tabla 3.2. Entidades del modelo del dominio.

El modelo del dominio, sus objetos y relaciones se encuentran en la figura 3.1.

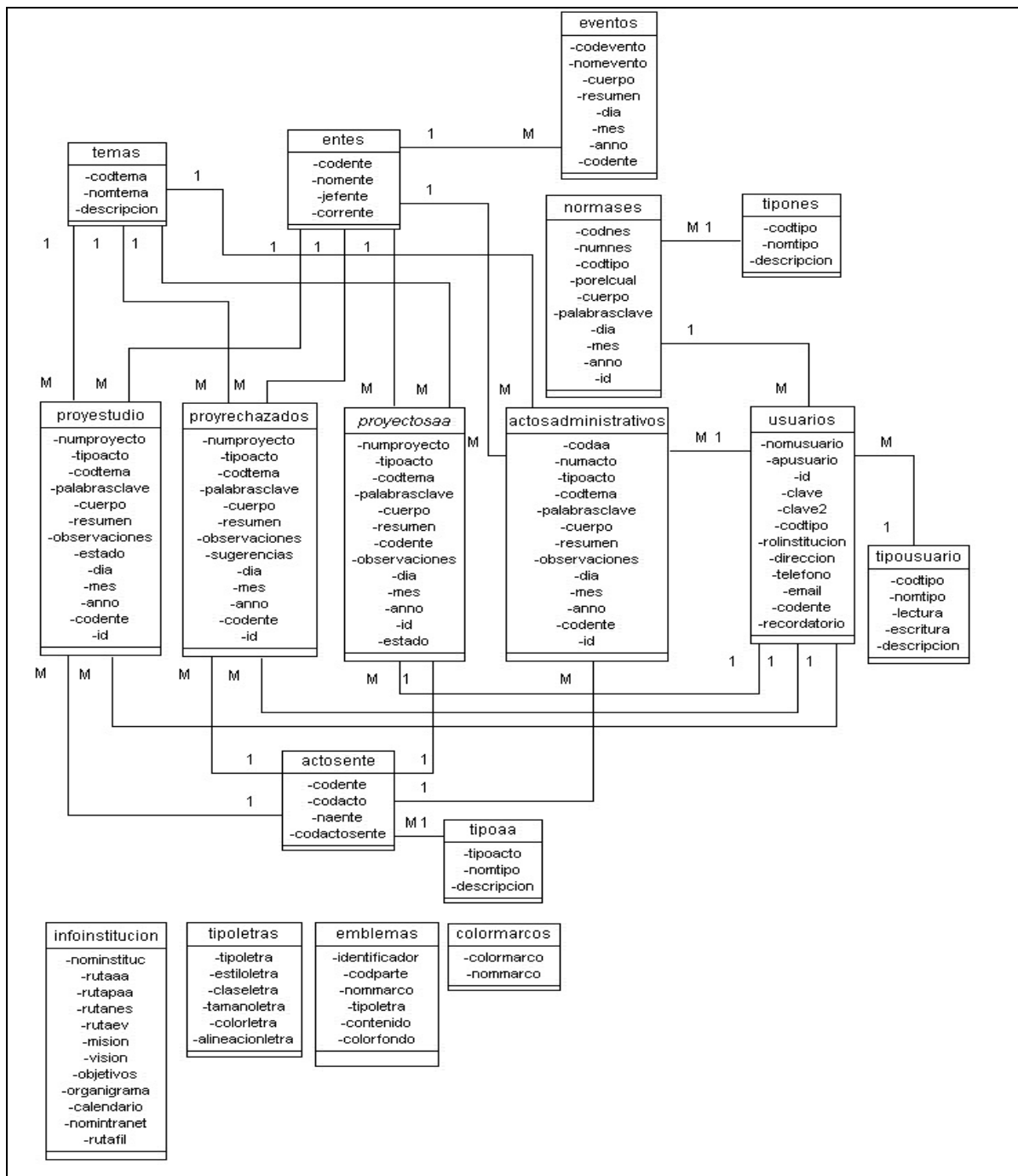


Figura 3.1. Modelo del dominio

7.4. NUEVA VERSIÓN MODELOS DE CASOS DE USO

7.4.1 Modelo de Casos de Uso: Gestión de Configuración

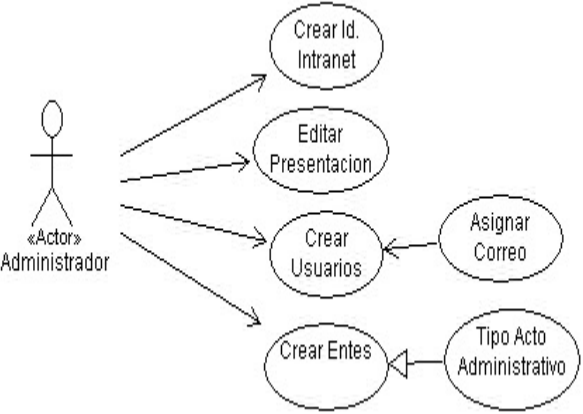
DIAGRAMA UML	DESCRIPCIÓN DEL MODELO
 <p>Figura3.2. Modelo de Casos de Uso: Gestión de Configuración</p>	<p>El sistema permite al Administrador incluir información propia de cada institución y que le permite identificarse. Esta información está relacionada con la identificación de la intranet, la edición de la presentación que se va a mostrar en las plantillas de la intranet, así mismo incluir los entes emisores de actos administrativos que la institución posee, los actos administrativos que se manejan, crear los diferentes tipos de usuarios y asignarles correo si es necesario.</p>

Tabla 3.3. Modelo de casos de uso: Gestión de Configuración.

7.4.2 Modelo de Casos de Uso: Gestión de Publicación

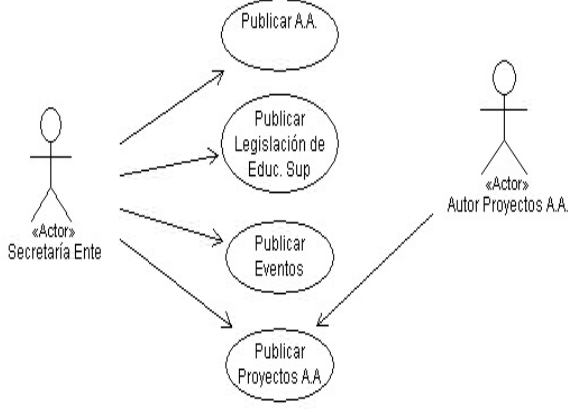
DIAGRAMA UML	DESCRIPCIÓN DEL MODELO
 <p>Figura 3.3. Modelo de Casos de Uso: Gestión de Publicación</p>	<p>El sistema permite publicar tanto los Actos Administrativos que son emitidos por los diferentes entes como la legislación que sobre Educación Superior emite el gobierno nacional, también permite publicar los eventos que se van a realizar y que son de interés general, esto con el fin de darlos a conocer a la comunidad universitaria, Adicionalmente, el sistema permite publicar por determinado tiempo los proyectos de actos administrativos que han sido elaborados por cualquier miembro de la comunidad universitaria.</p>

Tabla 3.4. Modelo de Casos de Uso: Gestión de Publicación

7.4.3 Modelo de Casos de Uso: Gestión de Proyectos

DIAGRAMA UML	DESCRIPCIÓN DEL MODELO
<p>Figura 3.4. Modelo de Casos de Uso: Gestión de Proyectos</p>	<p>La elaboración, modificación, consulta de sugerencias y envío a estudio de proyectos es una función que el sistema permite. Para la elaboración de proyectos, los actores pueden invocar el caso de uso consultar ayudante, el cual les brinda una orientación sobre los aspectos básicos que debe tener para realizar esta tarea. Finalmente, el resto de comunidad universitaria puede ver los diferentes proyectos que han sido publicados y dar sugerencias u opiniones de dichos proyectos a los autores de los mismos.</p>

Tabla 3.5. Modelo de Casos de Uso: Gestión de Proyectos

7.4.4 Modelo de Casos de Uso Gestión de Consulta

DIAGRAMA UML	DESCRIPCIÓN DEL MODELO
<p>Figura 3.5. Modelo de Casos de Uso: Gestión de Consulta</p>	<p>El sistema permite realizar consultas sobre los actos administrativos, proyectos de dichos actos, legislación sobre educación superior, es decir, aquellas normas emitidas por el gobierno nacional y por último permite consultar los diferentes eventos a realizar y que son de interés general. Estas funciones que el sistema ofrece pueden ser realizadas por los diferentes actores: Autor de proyectos, Secretaría de Ente, Comunidad Universitaria.</p>

Tabla 3.6. Modelo de Casos de Uso: Gestión de Consulta

7.5. RESUMEN CAPITULO 5

En este capítulo se tuvieron en cuenta nuevos aspectos de la funcionalidad del sistema que ofrecen soporte a la gestión normativa, facilitando de esta manera el conocimiento del marco legal que se debe tener en cuenta para la concepción y regulación de la normatividad en la institución, adicionalmente y con el fin de incentivar la interacción de la comunidad con el sistema, se permitió la publicación de eventos institucionales. Con base en los aspectos anteriormente expuestos, se construyeron tanto la versión definitiva del modelo del dominio como las nuevas versiones de los modelos de casos de uso, proporcionando de esta manera una base más sólida sobre la cual trabajar para llevar a cabo el desarrollo del proyecto.

CAPITULO 6

8. ANÁLISIS

Las actividades que aquí se efectúan son muy similares a la etapa de análisis de la fase de inicio, solo que se hace una profundización mayor, debido a que se tiene una mayor complejidad sobre la labor que el sistema va realizar.

Al identificar nuevos casos de uso se hace indispensable analizarlos individualmente y su interacción con los ya existentes. Esto generará una nueva versión de los paquetes de análisis. Una vez obtenidos dichos paquetes, se procede a tomar los casos de uso más importantes de cada paquete, para descomponerlos en sus procesos internos, con la finalidad de comprender los diferentes aspectos a tener en cuenta para llevar a cabo dicha funcionalidad.

8.1. MODELO DE ANALISIS

A continuación se presentarán el análisis de la arquitectura, de los casos de uso, de las clases de análisis y de los paquetes del sistema.

8.1.1 Análisis de la Arquitectura

El agregar nuevos casos de uso a cada uno de los paquetes existentes, conlleva a que se rediseñen o se creen nuevos paquetes de acuerdo con los criterios que se plantearon en la fase de inicio. A continuación se presenta la nueva versión de dichos paquetes.

☑ **Paquete de Análisis: Gestión de Configuración**

DIAGRAMA DE UML	DESCRIPCIÓN
<p style="text-align: center;">Figura 3.6. Paquete: Gestión de Configuración</p>	<p>En el paquete de análisis Gestión de Configuración se contemplan los paquetes de análisis involucrados en el proceso de configuración inicial de la intranet. En este caso particular solo el administrador es el actor involucrado en este proceso</p>

Tabla 3.7. Paquete de Análisis: Gestión de Configuración

☑ **Paquete de Análisis: Gestión de Publicación**

DIAGRAMA DE UML	DESCRIPCIÓN
<p style="text-align: center;">Figura 3.7. Paquete: Gestión de Publicación</p>	<p>Este paquete de análisis contiene los casos de uso implicados con el proceso de publicación de la información. En este caso el administrador, la Secretaría del y el autor son los que pueden intervenir en este proceso.</p>

Tabla 3.8. Paquete de Análisis: Gestión de Publicación

Paquete de Análisis: Gestión de Proyectos de Actos Administrativos

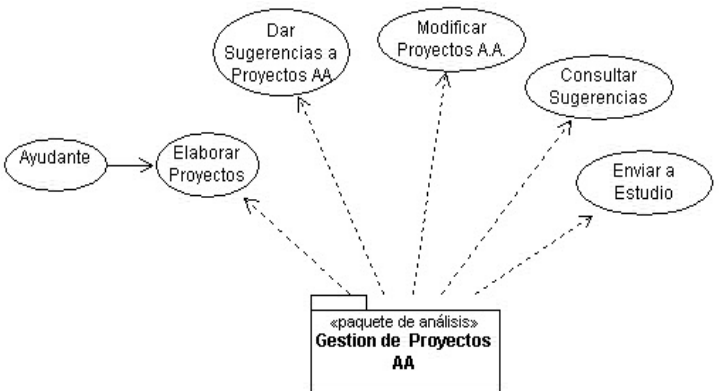
DIAGRAMA DE UML	DESCRIPCIÓN
 <p data-bbox="316 934 868 966">Figura 3.8. Paquete: Gestión de Proyectos A.A.</p>	<p data-bbox="971 567 1292 903">El paquete de análisis gestión de Proyectos AA, esta conformado por los casos de uso que llevan a cabo esta actividad, El Autor, la secretaría del ente y la comunidad Universitaria son los actores que están involucrados en este proceso.</p>

Tabla 3.9. Paquete de Análisis: Gestión de Proyectos A.A.

Paquete de Análisis: Gestión de Consulta

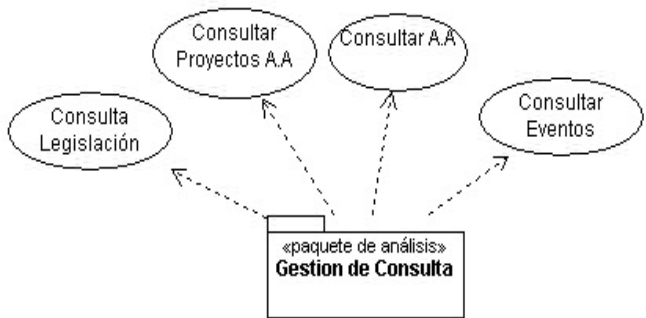
DIAGRAMA DE UML	DESCRIPCIÓN
 <p data-bbox="324 1627 803 1659">Figura 3.9. Paquete: Gestión de consulta</p>	<p data-bbox="920 1333 1292 1585">En este paquete de análisis están los casos de uso implicados en la consulta de los documentos normativos almacenados en la base de datos, este proceso involucra a todos los actores del sistema.</p>

Tabla 3.10. Paquete de Análisis: Gestión Consulta.

8.1.2 Análisis de Casos de Uso

Muchos casos de uso no son claramente comprensibles tal y como están descritos en el modelo de casos de uso, por tanto deben ser mejorados en función de las clases del análisis, pero los casos de uso que deben ser mejorados son solo aquellos importantes para la comprensión de los requisitos.

Para realizar este análisis primero se identifican las clases del análisis (entidad, control e interfaz) y como se relacionan entre si para llevar a cabo el flujo de sucesos del caso de uso. Para este fin se han utilizado diagramas de colaboración, ya que el objetivo es identificar requisitos y responsabilidades sobre las clases y no tanto identificar secuencias de interacción detalladas y ordenadas cronológicamente, caso en el cual se utilizaría diagramas de secuencia.

Dado que ya se tienen unos paquetes de análisis, se continuaran utilizando durante el transcurso de esta fase. Para cada uno de estos paquetes se toman los casos de uso que son de importancia desde el punto de vista de la arquitectura y que aportan para crear la línea base de la misma.

Paquete de Gestión de Configuración

De este paquete se han seleccionado los casos de uso: Crear Ente, Crear Acto Administrativo y crear ID de la intranet.

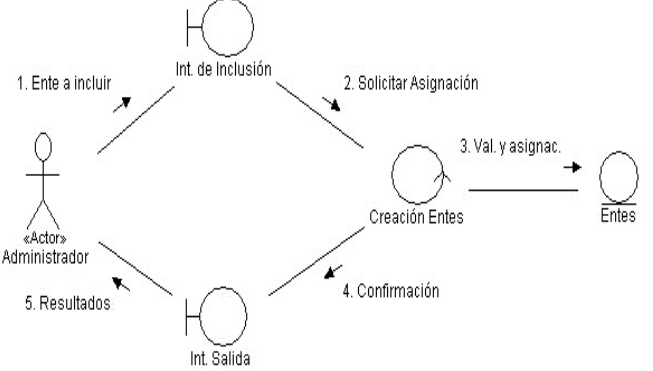
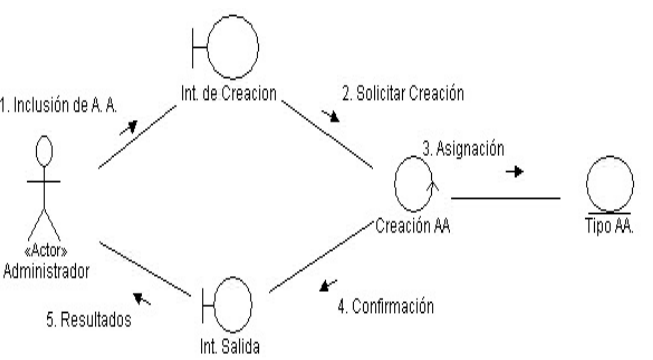
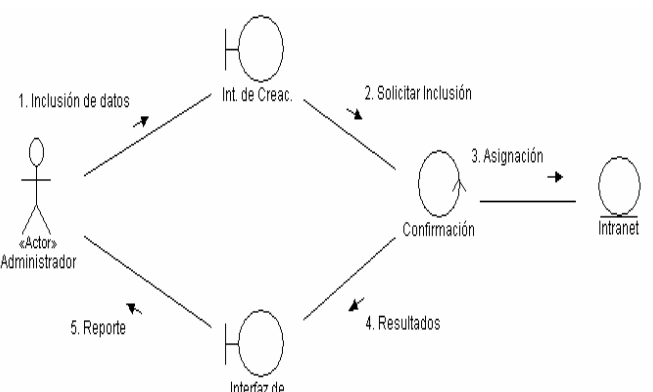
DIAGRAMA DE COLABORACION	DESCRIPCION
 <p>Figura3.10. Diagrama de Colaboración: CU Crear Ente</p>	<p>Este caso de uso muestra el proceso necesario para llevar a cabo la inclusión de información relacionada con las características de los diferentes entes designados por la institución para la elaboración y aprobación de actos administrativos existentes en la Institución.</p>
 <p>Figura3.11. Diagrama de Colaboración: CU Crear tipo Acto Administrativo</p>	<p>En la figura 3.11. se ilustra el proceso que se realiza para crear los diferentes tipos de actos administrativos, esta información es almacenada en la clase de entidad: tipo AA, que permite ingresar las características de los diferentes tipos de actos administrativos que se manejan al interior de la Institución.</p>
 <p>Figura3.12. Diagrama de Colaboración: CU Crear ID de la intranet</p>	<p>Este diagrama de colaboración muestra el proceso a seguir para crear la identificación de la intranet, aspecto importante porque caracteriza de manera individual cada sistema.</p>

Tabla 3.11. Casos de uso significativos del paquete de análisis: Gestión de Configuración

☑ **Paquete Gestión de Publicación**

En este paquete se ha seleccionado el caso de uso Publicar Acto Administrativo.

DIAGRAMA DE COLABORACION	DESCRIPCION
<p>Figura3.13. Diagrama de Colaboración: CU Publicar Acto Administrativo</p>	<p>Es de anotar que todo acto administrativo pasa por una fase inicial en la cual se almacena como proyecto de acto administrativo y sólo hasta cuando es aprobado por el ente correspondiente pasa a publicarse y almacenarse como acto administrativo transfiriéndose desde la entidad Proyectos AA a la entidad Actos Administrativos</p>

Tabla 3.12. Casos de uso significativos del paquete de análisis: Gestión de Publicación

☑ **Paquete Gestión de Proyectos**

En este paquete se ha seleccionado el caso de uso Elaborar Proyectos.

DIAGRAMA DE COLABORACION	DESCRIPCION
<p>Figura 3.14. Diagrama de Colaboración: CU Elaborar</p>	<p>El diagrama de colaboración presentado en la figura 3.14. ilustra el proceso para elaborar un proyecto, desde la elección del tipo de acto administrativo involucrado hasta la elección del tema y los usuarios involucrados en la realización del mismo. Finalmente es almacenado en la entidad proyectos AA desde donde se publica para someterlo a comentarios, sugerencias y/o críticas.</p>

Proyecto	
-----------------	--

Tabla 3.13. Casos de uso significativos del paquete de análisis: Gestión de Proyectos

Paquete Gestión de Consulta

De este paquete se ha seleccionado el caso de uso: Consultar Acto Administrativo.

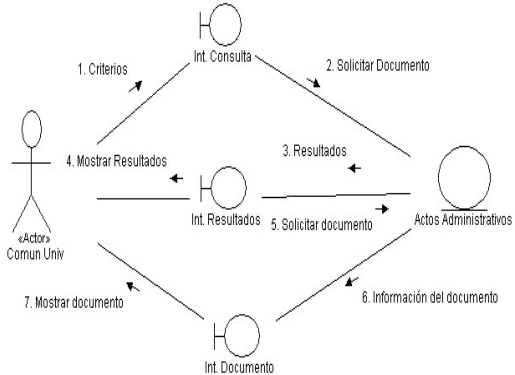
DIAGRAMA DE COLABORACIÓN	DESCRIPCION
 <p>Figura3.15. Diagrama de Colaboración: CU Consultar Acto Administrativo</p>	<p>Este caso de uso es generalmente el más utilizado, como su nombre lo indica permite consultar los actos administrativos existentes teniendo en cuenta ciertos criterios de búsqueda como el tema, autor, ente emisor, fecha, etc.</p>

Tabla 3.14. Casos de uso significativos del paquete de análisis: Gestión de Consulta

8.2. RESUMEN CAPITULO 6

En este capítulo se presentó una nueva versión de los paquetes de análisis, en ellos se incluyeron los nuevos casos de uso identificados, esto con el fin de facilitar la comprensión de la estructura del sistema. Adicionalmente de cada paquete se eligieron los casos de uso más relevantes para realizar el análisis de su funcionamiento, este análisis se realizó mediante diagramas de colaboración ya que éstos permiten observar los procesos internos necesarios para llevar a cabo dicho caso de uso.

A partir de este momento se tiene una estructura prácticamente completa del proyecto y con base en ella empieza el trabajo fuerte en la parte de diseño e implementación.

CAPITULO 7

9. DISEÑO

En este capítulo se presenta el modelo de diseño, que en esta fase tiene en cuenta (casi en su totalidad) los requisitos y restricciones necesarias para alcanzar las metas del proyecto; específicamente se establece el diseño de la arquitectura, el cual tiene en cuenta aspectos como los nodos y configuraciones de red, así mismo se identifican los subsistemas necesarios para soportar los requisitos; paralelamente se desarrollan las interfaces necesarias tanto para el usuario como entre los mismos subsistemas; por último se definió el modelo de arquitectura en capas, sobre el cual se va a trabajar.

9.1. MODELO DE DISEÑO

9.1.1 Diseño de la arquitectura

El objetivo del diseño de la arquitectura es esbozar los modelos de diseño y despliegue y su arquitectura mediante la identificación de los siguientes elementos:

- Nodos y sus configuraciones de red (diagrama de despliegue).
- Subsistemas y sus interfaces.
- Arquitectura en capas.

Nodos y configuraciones de red

Las configuraciones físicas de red suelen tener una gran influencia sobre la arquitectura del software. Las configuraciones habituales utilizan un patrón de tres

capas en el cual, como se dijo en la parte I⁸, los clientes (las interacciones de los usuarios) se dejan en una capa, la funcionalidad de base de datos en otra y la lógica del negocio o de la aplicación en una tercera, este tipo de configuración es la utilizada en el presente proyecto.

El Sistema SIGNO se ejecutará sobre un nodo servidor y uno o varios nodos clientes. En el Nodo servidor se encuentra la capa de datos y la de lógica o aplicaciones, aquí se encuentran almacenados los datos del sistema. En los nodos clientes se ejecuta la capa de presentación. La comunicación entre el nodo del cliente y del servidor se realiza mediante el protocolo TCP/IP dentro de la Intranet propia del Campus Universitario. En la figura 3.15. se observa el diagrama de despliegue correspondiente.

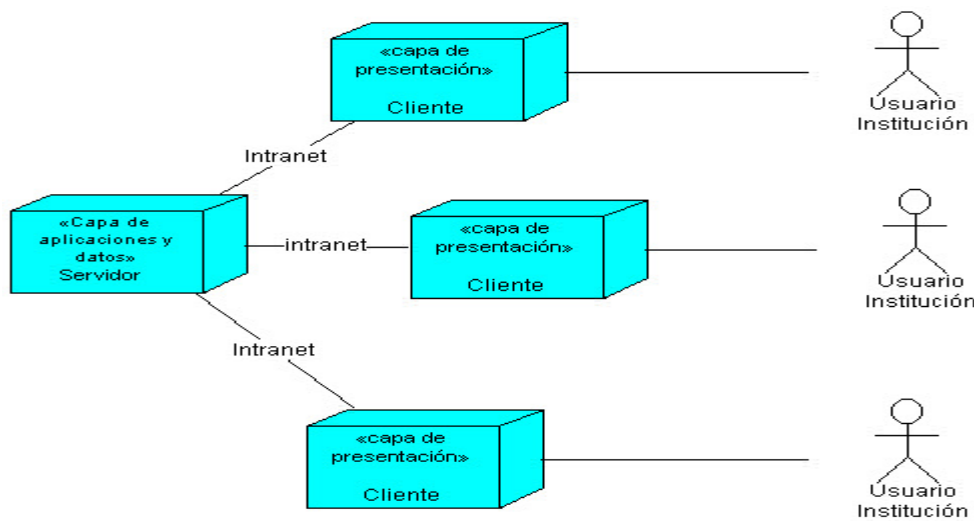


Figura 3.16. Diagrama de despliegue del sistema SIGNO

⁸ Escribir página en donde se hable de esto.

Los requerimientos de hardware de cada uno de estos nodos se describen a continuación:

Nodo Servidor

Para la fase de desarrollo se utilizará un computador con las siguientes características:

- Procesador Pentium III de 500MHz o superior
- Memoria RAM de 128MB o más
- Disco Duro de 40G
- Tarjeta de Red

Es de anotar que para el uso del sistema dentro de la Institución se deben aprovechar los recursos hardware y software que dicha Institución posea.

Nodo Cliente

Los requerimientos para los equipos clientes son:

- Procesador Pentium de 133MHz o superior
- Memoria RAM de 32MB
- Tarjeta de Red

Además, el nodo cliente debe contar con acceso a la red de área local (LAN) de la Institución.

9.1.2 Diseño de Subsistemas

Subsistemas de aplicación

Primero se identifican subsistemas a partir de los paquetes de análisis. En el flujo de trabajo de análisis se obtuvieron 4 paquetes. Dado que durante el análisis se realizó una descomposición del sistema en paquetes, estos se utilizan como base para los subsistemas de diseño. De esta forma se cuenta con los subsistemas de gestión de configuración, gestión de publicación, gestión de proyectos y gestión de consulta.

Se identificaron dos nuevos subsistemas. El primero: **Gestión de Sesiones** se encarga de controlar el acceso al sistema por parte de los usuarios. Aunque aun no ha sido tratado este aspecto, se dará una breve explicación: Todos los actores para poder acceder a cada uno de estos subsistemas deben estar en sesión, es decir deben haber validado su ingreso al sistema, y deben tener permiso para acceder al subsistema y a un caso de uso en particular. Por lo tanto, como todos los subsistemas comparten esta funcionalidad, se agrupó esta característica en este nuevo subsistema. El segundo subsistema es el de **Gestión de Interfaz** que permitirá dividir la capa lógica de la capa de presentación, razón por la cual es utilizado por todos los subsistemas que necesiten comunicarse con algún actor.

El conjunto de subsistemas se pueden observar en la figura 3.17 y una breve descripción de cada uno de ellos se encuentra en la tabla 3.15. Dado que estos dos últimos subsistemas no son accedidos directamente por los usuarios, sino que son utilizados por los demás subsistemas, se denominan subsistemas generales de aplicación y por esta razón han sido ubicados bajo la línea horizontal resaltando esta diferencia.

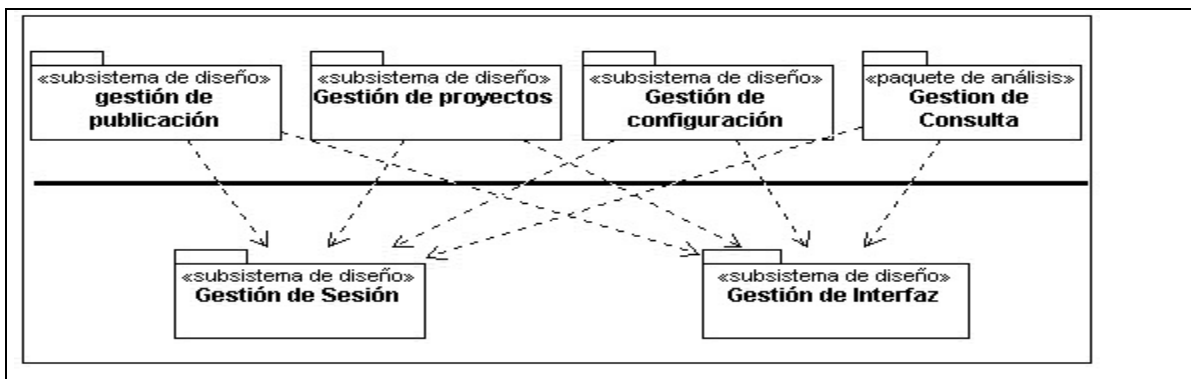


Figura 3.17. Subsistemas de diseño

Subsistema	Descripción
Gestión de Publicación	Permite la difusión mediante el Sistema de Información Intranet de los proyectos de actos administrativos, actos administrativos, eventos, normatividad de Educación Superior.
Gestión de Proyectos	Este subsistema brinda a la comunidad universitaria la opción de elaborar proyectos de actos administrativos, además de recibir sugerencias sobre los mismos.
Gestión de Configuración	Esta funcionalidad, permite incluir la información que identifica a la institución.
Gestión de Consulta	Este subsistema permite que la comunidad universitaria tenga acceso a los actos administrativos, proyectos de actos administrativos, eventos y normatividad de educación superior.
Gestión de Sesión	Es aquí donde se controla el acceso a las diferentes funcionalidades del sistema, según el tipo de usuario.
Gestión de Interfaz	Este subsistema permite la comunicación entre los demás subsistemas y el usuario.

Tabla 3.15. Descripción de Subsistemas de Diseño.

- **Diseño del subsistema Gestión de Sesión**

En el protocolo http las peticiones individuales no están relacionadas unas con otras, por tanto el servidor web (y por ello PHP) no pueden distinguir fácilmente entre los usuarios y no sabe nada de las sesiones de los usuarios. Para solucionar este inconveniente, es necesario encontrar un mecanismo que se encargue de las validaciones de dichos usuarios. Como se mencionó anteriormente, el subsistema

Gestión de sesión realiza esta tarea: Valida el ingreso al sistema por parte de un actor, según tenga permiso para acceder a una zona o subsistema determinado, así como a un caso de uso en particular.

Para explicar con mas detalle el funcionamiento de este subsistema se hará uso del siguiente diagrama de secuencia. Sin embargo este diagrama no contiene todos los detalles, restricciones y caminos alternativos, ya que su objeto es mostrar al lector de una manera sencilla en la forma como se realizará la validación de ingreso al sistema y como otros subsistemas accederán al sistema de Gestión de Sesión para a validar el acceso a los casos de uso que solicita el actor.

Como se observa en el diagrama de secuencia de la figura 3.16., cuando un usuario quiere ingresar al sistema debe primero solicitar su ingreso al mismo; el subsistema de gestión de sesión se encargará de solicitar al usuario la información necesaria (login y password) que será verificada en la base de datos. Si el ingreso es correcto el gestor de sesiones le enviará un identificador de sesión⁹ al usuario, que consta de un numero aleatorio que lo identifica y de ciertos datos propios del usuario.

Cuando el usuario, desea hacer uso de un proceso para llevar a cabo un caso de uso de cualquier subsistema, se debe verificar que dicho usuario tenga permiso para su ejecución y para acceder o modificar a la información que este maneja. Esto es de especial importancia, dado que para la comunicación entre el cliente y servidor se utiliza tecnología internet, exponiendo los problemas de seguridad propios de esta tecnología.

⁹ El termino Sesión para una instancia de un usuario que visita un sitio donde se mira una o mas paginas. Secuencia de paginas que un usuario visita en un sito web desde que entra hasta que lo abandona.

En el siguiente diagrama de secuencia, se observan este tipo de validaciones cuando un autor de proyecto o una secretaria de ente desea llevar a cabo el caso de uso **elaborar proyecto**. Para este caso de uso, se necesita realizar dos validaciones: la primera se realiza en el momento en que el usuario ingresa al sistema. La segunda cuando el autor o la secretaria de ente envía los datos del nuevo proyecto, reforzando la seguridad del sistema, ya que evita el problema de que un usuario no autorizado, trate de insertar nuevos proyectos. Esto se podría realizar dado que comúnmente los navegadores web luego de visitar una pagina, pueden mantener localmente información sobre esta. Este usuario, por tanto podría aprovechar estas paginas temporales y a través del formulario de envío utilizado por algún usuario, correctamente validado, enviar la información para la creación de un nuevo proyecto.

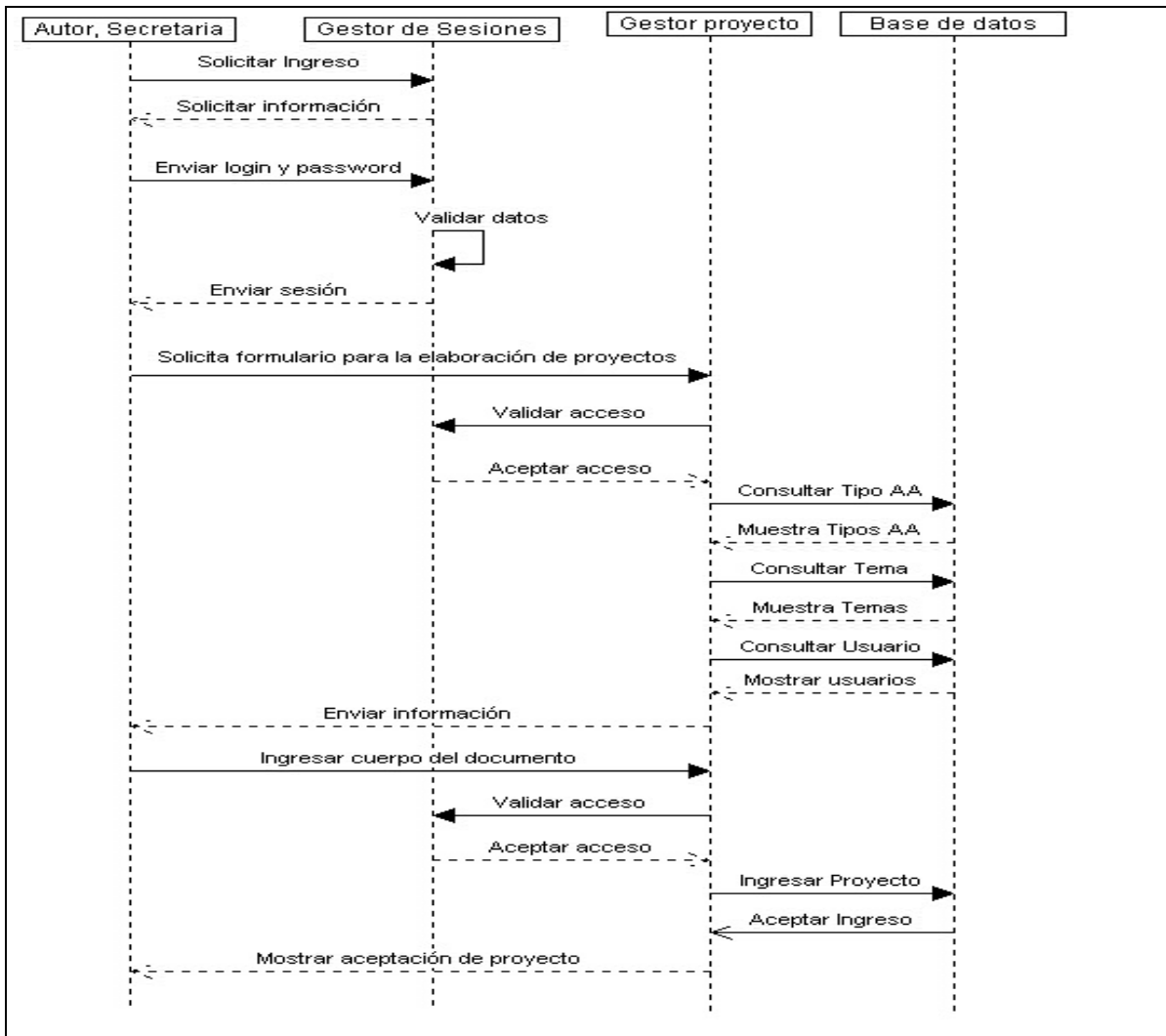


Figura3.18. Diagrama de Secuencia de las interacciones entre subsistemas.

En la etapa de implementación, estas sesiones serán realizadas con ayuda del modulo de PHP para el manejo de las mismas. Esto se explicará con mas detalle en la fase de construcción cuando se implemente este subsistema.

- **Diseño de la interfaz**

Para el desarrollo de la interfaz se siguieron los lineamientos básicos cuyo principal objetivo es facilitar la interacción del usuario con el sistema, estos aspectos están relacionados con consistencia, flujo de pantallas, no sobre

población de pantallas, agrupación de elementos relacionados, navegación inconsistente, sitios que requieren un navegador específico, uso de marcos, etc., y se describieron en la tabla 1.7

Igualmente se tuvieron en cuenta los diferentes roles que puede desempeñar un usuario en el sistema, tanto en el instalador de SIGNO como en el Sistema de Información Intranet generado. Es de anotar que cada actor tiene acceso a ciertos casos de uso de un subsistema.

Para continuar con la descripción de la interfaz, se describe en primer lugar la estructura que va a tener el instalador SIGNO. Vemos en la figura 3.19 , que esta dividida en tres secciones: En la sección superior (1) se encuentra el logotipo del sistema, junto con las opciones de ayuda e información del proyecto y grupo de investigación. En la sección izquierda (2) estará cada uno de los pasos que se deben seguir para generar el Sistema de Información Intranet. Por ultimo en la sección derecha (3) de la interfaz estará una ayuda que orienta al administrador en la instalación del sistema.

(1) Logo, ayuda e información de Signo y del grupo de investigación STI	
Paso de Instalación (2)	Ayuda (3)

Figura 3.19. Diseño General de la interfaz del sistema SIGNO.

De otro lado, una vez generado el Sistema de Información Intranet para la Institución interesada, se muestra dicho Sistema, el cual va a tener la estructura mostrada en la figura 3.20, en la cual se pueden observar tres secciones: en la

superior (1) se encuentran el escudo y nombre de la Institución que implante el sistema, en la parte izquierda (2) se encuentran diferentes hipervínculos u opciones que permiten a los usuarios interactuar con el sistema, y por último en la sección derecha (3) se encuentra el resultado de las acciones que se realicen en las secciones 1 y 2, además de un menú que facilita la gestión normativa.

(1) Logo y nombre de la Institución	
Hipervínculos (2)	Resultados (3)

Figura 3.20. Diseño General de la interfaz del Sistema de Información generado

Subsistemas intermedios y de software del sistema

- **Software del sistema y de la capa intermedia.**

El software del sistema y la capa intermedia constituyen los cimientos de un sistema. Ya que toda la funcionalidad descansa sobre software como sistemas operativos, sistemas de gestión de bases de datos, software de comunicaciones etc. La elección de productos software es objetivo de la fase de inicio y elaboración.

El sistema Intranet de información SIGNO se desarrolla bajo el sistema operativo **LINUX**, que se caracteriza por su robustez y estabilidad, así como por ser un sistema operativo de libre distribución, multiusuario y de mayor difusión en la actualidad.

El sistema estará basado en **PHP (Hypertext Preprocessor)**, que es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor, y con el apoyo de **POSTGRES** para el manejo de base de datos.

Se utilizará el servidor para páginas **Web Apache**, muy utilizado en la actualidad por su eficiencia, funcionalidad y velocidad.

Todas las herramientas mencionadas anteriormente son productos “hechos en Internet”, lo cual se traduce en costo nulo y con un grupo de desarrolladores de excelente calidad, logrando una solución económica sin que el sistema pierda su robustez y seguridad.

9.2. MODELO DE IMPLEMENTACIÓN

9.2.1 Software del sistema

Para la configuración del servidor se instaló el software descrito en el flujo de trabajo de diseño¹⁰:

Sistema operativo **LINUX**, distribución RedHat 8.0.

Como servidor páginas Web se instaló y configuró Apache¹¹; **PHP**¹²(**Hypertext Preprocessor**), como lenguaje de scripts que se ejecuta en el servidor y para el manejo de la base de datos se instaló y configuró **POSTGRES**¹³.

¹⁰ Sección 3.1. - Modelo del diseño .

¹¹ En esta página encontrará mas información sobre la instalación y configuración del servidor apache:
<http://quark.fe.up.pt/ApachES/manual-es/install.html>

¹² En la página <http://www.php.net/manual/es/installation.php> encontrará mas información sobre la instalación y configuración de php como módulo de apache.

La base de datos diseñada en el flujo de análisis fue implementada en este motor de base de datos según el diagrama de clases de la Figura 3.1.

9.3. RESUMEN CAPITULO 7

Este capítulo planteó el diseño casi en su totalidad, aportando las bases necesarias para la implementación del sistema. Específicamente, se organizó la estructura tanto física como conceptual que soportará dicho sistema. Se tuvo en cuenta los nodos y configuración de red sobre las cuales funcionará éste, además del diseño de las interfaces, aspecto muy importante porque de este depende parte del funcionamiento y de la usabilidad de la Intranet, por último, se describió cómo se va a aplicar el modelo de arquitectura de tres capas a fin de optimizar el funcionamiento y mantenimiento del software.

¹³ En la página <http://lucas.hispalinux.es/Postgresql-es/web/> se encuentra mas información sobre el procedimiento de instalación y configuración de Postgres.

RESUMEN PARTE III

En esta parte se identificaron gran parte de los requisitos del sistema, y con base en estos se complementó el modelo del dominio, identificando por tanto las entidades y relaciones necesarias para soportar el sistema; se detalló además gran parte de los casos de uso, de ellos se analizaron los más importantes a fin de tener una visión global de dicho sistema; posteriormente

se realizó el proceso de diseño, en el cual se tuvo en cuenta los recursos hardware y software a utilizar, el modelo de arquitectura en capas, los subsistemas identificados y las interfaces a utilizar. Por último, en esta parte se realizó una parte de la implementación relacionada con la instalación y configuración del software del sistema además de la asignación de los componentes necesarios a cada uno de los nodos especificados en el diagrama de despliegue mostrado en la figura 3.16

De esta manera se concluye la fase de elaboración, al final de la cual se pueden apreciar los siguientes casos de uso:

Casos de Uso Identificados	Identificado	Descrito	Analizado	Diseñados e implementados	¿Se comprende Completamente ?
Crear Identificación	F. Inicio	F. Inicio	F. Elaboración		Si
Crear Ente	F. Inicio	F. Inicio	F. Elaboración		Si
Crear Tipo Acto Administrativo	F. Inicio	F. Inicio	F. Elaboración		Si
Editar	F. Inicio	F. Inicio			Si

Presentación					
Crear usuario	F. Inicio	F. Inicio			Si
Asignar Correo Electrónico	F. Inicio	F. Inicio			Si
Publicar Proyecto	F. Inicio	F. Inicio			Si
Publicar Acto Administrativo	F. Inicio	F. Inicio	F. Inicio		Si
Publicar Legislación de Educación Superior	F. Elaboración	F. Elaboración			Si
Publicar Eventos	F. Elaboración	F. Elaboración			Si
Consultar Acto Administrativo	F. Inicio	F. Inicio	F. Inicio		Si
Consultar Proyectos de A.A.	F. Inicio	F. Inicio			Si
Consultar sugerencias	F. Inicio	F. Inicio			No
Consultar legislación de Educación Superior	F. Elaboración	F. Elaboración			Si
Consultar Evento	F. Elaboración	F. Elaboración			Si
Elaborar Proyecto	F. Inicio	F. Inicio	F. Elaboración		Si
Modificar Proyecto	F. Inicio	F. Inicio			No
Enviar a estudio.	F. Inicio	F. Inicio			No
Dar sugerencias	F. Inicio	F. Inicio			Si
Consultar ayudante	F. Elaboración	F. Elaboración			
Imprimir	F. Inicio	F. Inicio			Si
Notificar	F. Inicio	F. Inicio			No

Tabla 3.16 Casos de Uso al final de la fase de Elaboración

Adicionalmente, como ya han sido presentados los diagramas que describen la vista de la arquitectura en cada uno de los flujos desarrollados en esta fase, en la

tabla 3.17 solo se especificará las secciones donde se encuentra dicha información

Flujo	Sección	Vista	Comentario
Captura de Requisitos	Casos de uso en Detalle	<p>Tabla 3.3. Modelo de casos de uso: Gestión de Configuración.</p> <p>Tabla 3.4. Modelo de Casos de Uso: Gestión de Publicación</p> <p>Tabla 3.5. Modelo de Casos de Uso: Gestión de Proyectos.</p> <p>Tabla 3.6. Modelo de Casos de Uso: Gestión de Consulta.</p>	<p>Presenta los actores y casos de uso mas importantes del sistema, estos incluyen tanto los identificados en la fase de inicio, como los que se hallaron en la fase de elaboración, agrupados según los procesos existentes.</p>
Análisis	Diagramas de colaboración y Paquetes de Análisis.	<p>Tabla 3.7. Paquete de Análisis: Gestión de Configuración.</p> <p>Tabla 3.8. Paquete de Análisis: Gestión de Publicación</p> <p>Tabla 3.9. Paquete de Análisis: Gestión de Proyectos A.A.</p> <p>Tabla 3.10. Paquete de Análisis: Gestión Consulta.</p> <p>Figura3.10. Diagrama de Colaboración: CU Crear Ente</p> <p>Figura3.11. Diagrama de Colaboración: CU Crear tipo Acto Administrativo</p> <p>Figura3.12. Diagrama de Colaboración: CU Crear ID de la intranet.</p> <p>Figura3.13. Diagrama de Colaboración: CU Publicar Acto Administrativo</p> <p>Figura 3.14. Diagrama de Colaboración: CU Elaborar Proyecto.</p> <p>Figura3.15. Diagrama de Colaboración: CU Consultar Acto Administrativo.</p>	<p>Con base en los actores y casos de uso identificados, se realizaron los paquetes de análisis y se elaboraron los diagramas de colaboración a fin de entender los principales casos de uso.</p>
Diseño	Diseño de la Arquitectura	<p>Figura 3.15. Diagrama de despliegue del sistema SIIN.</p> <p>Figura Subsistemas de diseño.</p> <p>Figura3.16. Diagrama de Secuencia de las interacciones entre subsistemas.</p> <p>Tabla 3.2. Entidades del modelo del dominio</p> <p>Figura 3.1. Modelo del dominio</p>	<p>Aquí se presentan los diagramas que ilustran el diseño de la arquitectura proyectada para el sistema.</p>

Tabla 3.17. Diagramas de la vista de la arquitectura contenidas en esta parte.

10. NOTAS BIBLIOGRAFICAS

- ☑ JACOBSON, Ivar. BOOCH, Grady. RUMBAUGH, James. **El Proceso Unificado de Desarrollo de Software**. Primera Edición. Addison Wesley. España, 2000. Al igual que para la parte II, en esta parte fue de gran importancia, ya que sirvió como guía en el desarrollo y maduración del proyecto.

PARTE IV- FASE DE CONSTRUCCION

En este momento, los requisitos y la arquitectura de la herramienta software son estables. La mayor parte del trabajo de identificación de características y análisis de funcionalidad fue realizada en las fases de inicio y elaboración. Ahora en la fase de construcción, el énfasis estará en la implementación y pruebas del software, hasta conseguir una versión ejecutable de la herramienta con capacidad operativa lo suficientemente satisfactoria como para permitir la primera entrega del producto a los usuarios. Adicionalmente, durante la fase de construcción se crearán diversos documentos que complementan la herramienta, tales como la guía del usuario final. La construcción del sistema se podrá considerar terminada cuando se pueda garantizar la realización de todos los casos de uso y además, que no surgirán riesgos incontrolables al utilizar la aplicación.

En pocas palabras, en la fase de elaboración, se analizaron los casos de uso, y paquetes que eran significativos desde el punto de vista de la arquitectura y al final se logró obtener un modelo del análisis casi completo al que se le añadirán simplemente actualizaciones menores en la fase de construcción. Con respecto al modelo de diseño se tiene una situación semejante, ya que no se añadirán nuevos subsistemas sino que se conservarán los que ya existen en forma de esqueleto en la línea base de la arquitectura. Por estas razones y para no presentar información repetitiva, el lector podrá remitirse a la tabla 3.15. para obtener una descripción de los subsistemas de diseño

CAPITULO 8

11. SUBSISTEMAS ESPECÍFICOS DE APLICACION

En la fase de elaboración se identificaron los subsistemas específicos de aplicación. Durante esta fase se implementan estos subsistemas con el fin de asegurar que cumplen su papel en cada construcción. Posteriormente se presentan las pruebas que permiten verificar el funcionamiento de los mismos.

Para no extender el documento en exceso, solo se presenta la implementación y pruebas de una funcionalidad de subsistema. La totalidad de las pruebas se encuentran en el Anexo B

11.1. GESTION DE CONFIGURACION

Para llevar a cabo la realización de este subsistema, se utiliza el lenguaje PHP que permite realizar los procesos necesarios para la consulta y actualización de la información contenida en la base de datos, específicamente en las tablas: *infoinstitucion, entes, tipoaa, actosente, colormarcos, tipoletras, emblemas*

El proceso antes mencionado requiere la información que Gestión de Interfaz le proporciona y utiliza dicho subsistema para mostrar al usuario el resultado del proceso de configuración.

11.1.1 Implementación

El código PHP utilizado para implementar este subsistema está inmerso en 7 páginas web:

- informaciongeneral.php
- crearentes.php
- creatipoaa.php
- editarcolormarcos.php
- emblemas.php
- editarfuente.php
- resultadosinstalador.php

Se ha elegido en este caso, los procesos realizados para implementar la generación de los diferentes entes autorizados por la Institución para emitir documentación normativa¹⁴. En la tabla 4.1. se encuentra un segmento de php que permite la actualización de la base de datos *entes* y en la figura 4.1. se encuentra la página que permite el ingreso de la información necesaria para la creación de dichos entes.

Código PHP	Descripción
<pre><?php if(isset(\$btncreaentes)) { include('conex.php'); \$conexion=conectarse(); \$insertar = "insert into entes(codente, nomeno, jefente, corrente) values ('\$txtcodente','\$txtnomeno','\$txtjefente','\$txtcorrente)"; \$insertar1 = mysql_query(\$insertar,\$conexion); if (!\$insertar1) \$mensaje .= "Error en la inserción" ; mysql_close(\$conexion); } ?></pre>	<p>Este segmento de código php realiza la inserción de la información del ente en la tabla entes si se ha presionado el botón btncreaentes, teniendo en cuenta que previo a la inserción se ha llamado a la pagina conex.php que contiene la función conectarse(), la cual permite hacer la conexión con la base de datos.</p>

Tabla4.1 . Código PHP para la creación de los Entes

¹⁴ ROJAS Juan Pablo, SUAREZ Martha Ligia. SIGNO CD-R Bucaramanga 2003

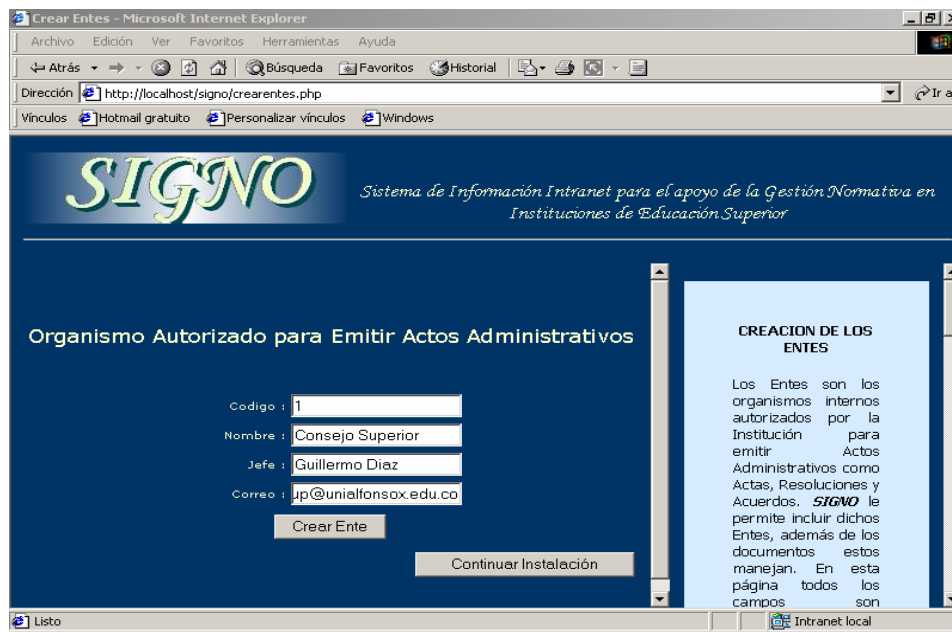


Figura 4.1. Interfaz que permite la inclusión de la información de cada Ente

11.1.2 Pruebas

- **Casos de prueba**

- Correcto almacenamiento de información en la base de datos
- Conexión y selección correcta de la base de datos¹⁵
- Evitar ambigüedad en los datos
- Integridad transaccional

En esta parte solo se mostrará el caso de prueba: Correcto almacenamiento de información en la base de datos.

¹⁵ Este caso de prueba es común a todos los subsistemas generales de aplicación, por ende solo se realizara prueba en este subsistema.

- **Procedimiento de Prueba**

El procedimiento para realizar esta prueba es bastante sencillo, por medio de la interfaz se incluye la información que permite identificar al ente, y después se procede a verificar en la base de datos que dicha información corresponda con los tabla y los campos correspondientes.

Datos Entrada	Estos datos corresponden al ente: Código : 1 Nombre : Consejo Superior Jefe : Guillermo Díaz Correo : consup@unialfonsox.edu.co								
Registro en la base de datos	<table border="1"> <thead> <tr> <th data-bbox="612 800 769 842">codente</th> <th data-bbox="769 800 912 842">nomete</th> <th data-bbox="912 800 1055 842">jefente</th> <th data-bbox="1055 800 1198 842">corrente</th> </tr> </thead> <tbody> <tr> <td data-bbox="612 842 769 898">1</td> <td data-bbox="769 842 912 898">Consejo Superior</td> <td data-bbox="912 842 1055 898">Guillermo Díaz</td> <td data-bbox="1055 842 1198 898">consup@unialfonsox.edu.co</td> </tr> </tbody> </table>	codente	nomete	jefente	corrente	1	Consejo Superior	Guillermo Díaz	consup@unialfonsox.edu.co
codente	nomete	jefente	corrente						
1	Consejo Superior	Guillermo Díaz	consup@unialfonsox.edu.co						

Tabla4.2. Prueba para la creación de los entes

Como se puede observar en la tabla 4.2., la información se ha almacenado correctamente en la tabla indicada.

11.2. GESTION DE PUBLICACION

Como se ha explicado desde la fase de elaboración, este subsistema tiene como finalidad el difundir tanto todo lo concerniente a los actos administrativos y proyectos de los mismos, como normatividad que rige la educación superior y eventos que se quiera dar a conocer a la comunidad universitaria.

11.2.1 Implementación

Este subsistema al igual que el anterior, se implementa en lenguaje PHP, para el manejo de la información en las tablas involucradas, tales como

actosadministrativos, proyectosaa, normases, eventos, entes, codactosente, usuarios y temas, el código esta embebido en las páginas:

- publicarproyecto.php
- publicaractoadministrativo.php
- publicares.php
- publicarevento.php

Se ha elegido presentar la implementación de los procesos para publicar proyecto de acto administrativo. En la tabla 4.3 se encuentra un segmento de php que permite la inclusión (en la tabla: *proyectosaa*), de la información relacionada con los proyectos de actos administrativos, por ejemplo, nombre de los autores, tema, tipo de acto administrativo, fecha, entre otros. Adicionalmente, en la figura 4.2. se encuentra la página que permite el ingreso de la información antes mencionada.

Código PHP	Descripción
<pre> <?php if(isset(\$btncrearproyecto)) { \$consulta = "select codtema from temas where nomtema='\$stema'"; \$consulta1=mysql_query(\$consulta,\$conexion); \$infotemas=mysql_fetch_array(\$consulta1); \$ctema=(\$infotemas["codtema"]); \$insertar = "insert into proyectosaa(tipoacto,dia,mes,anno,codtema,pala brasclave,cuerpo,resumen,id,codente) values ('\$caa','\$sdia','\$smes','\$sanno','\$ctema','\$txtpalcla ve','\$textcuerpo','\$textresumen','\$txtid','\$centa')"; \$insertar1 = mysql_query(\$insertar,\$conexion); if(\$insertar1) { print("<SCRIPT language=JavaScript>alert(\"EL proyecto de Acto Administrativo se ha creado satisfactoriamente\"); </script>"); print("<SCRIPT language=JavaScript> window.location=\"publicarproyecto.php\" </script>"); } else { </pre>	<p>Una vez se ha hecho click en el botón de “Publicar Proyecto” cuyo nombre es btncrearproyecto, se realiza una consulta que permite buscar el código del tema seleccionado y posteriormente se hace la inserción de los datos en la tabla proyectosaa, y se verifica que se haya creado el proyecto satisfactoriamente. Es de aclarar que la conexión a la base de datos se ha hecho en otra sección de la página para efectos de poder utilizarla en otras validaciones.</p>

```
print("<SCRIPT
language=JavaScript>alert(\"Error en la
Inserción\"); </script>");
print("<SCRIPT language=JavaScript>
window.location=\"publicarproyecto.php\"
</script>");
}
}
?>
```

Tabla 4.3. Código PHP que permite la publicación de un proyecto

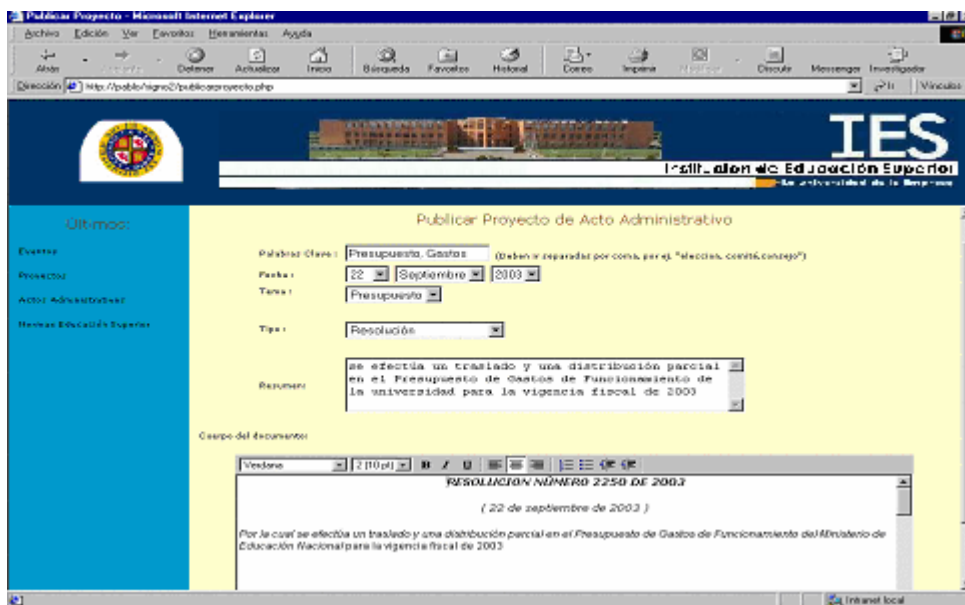


Figura 4.2 . Interfaz que permite la publicación de un proyecto

11.2.2 Pruebas

- **Casos de Prueba**

- ☑ Correcto almacenamiento de información en la base de datos
- ☑ Conexión y selección correcta de la base de datos¹⁶
- ☑ Evitar ambigüedad en los datos
- ☑ Integridad transaccional

¹⁶ Este caso de prueba es común a todos los subsistemas generales de aplicación, por ende solo se realizara prueba en este subsistema.

- **Procedimiento de Prueba**

En este subsistema se muestra el caso de prueba : Evitar ambigüedad en los datos, el cual se puede observar en la tabla 4.4. Existe una situación en la que se debe tener especial cuidado y es cuando un ente aprueba un proyecto de acto administrativo, el proceso debe asegurar que dicho proyecto deje de existir como tal en el sistema y pase a publicarse como acto administrativo. Para realizar la prueba, se parte de un proyecto al cual se le da aprobación. Se espera que inmediatamente sea almacenado en la tabla actosadministrativos y se elimine de la tabla proyectosaa.

Datos Entrada

Registro en la base de datos

Registros antes de efectuar cambios

numproyecto	tipoacto	dia	anio	codtema	palabrasclave	cuerpo	resumen	mes	codente	estado	observaciones	sugerencias	id
8		3	16	2003	2	Presupuesto	(BLOB)	(BLOB)	Septiembre	10 (Blob)	(Blob)	(Blob)	ersanchez

Registros después de efectuar cambios

numproyecto	tipoacto	dia	anio	codtema	palabrasclave	cuerpo	resumen	mes	codente	estado	observaciones	sugerencias	id
18	1	3			2	Presupuesto	(BLOB)		10	16 Septiembre	2003 (BLOB)	(Blob)	ersanchez

Tabla 4.4. Prueba para la aprobación de un proyecto y publicación del mismo como acto administrativo

11.3. GESTION DE PROYECTOS

Por medio de este subsistema, la comunidad universitaria tiene la oportunidad de participar activamente en el proceso de gestión normativa, mediante la elaboración y publicación de proyectos de actos administrativos. Adicionalmente dicha comunidad puede recibir y dar sugerencias sobre los mismos en el lapso que estén publicados. Este subsistema permite además la consulta de los proyectos según el estado en el que se encuentren (En elaboración, En estudio, Aplazados y Rechazados).

11.3.1 Implementación

Para implementar este subsistema se ha construido en lenguaje PHP los procesos que permiten cumplir con las funciones dadas a este subsistema. Estos procesos involucran las tablas de proyectosaa, proyestudio, proyrechazados, tipoaa, actosadministrativos, usuarios, entes, temas.

Este subsistema emplea los subsistemas de sesiones, interfaz y de publicación para realizar todas sus actividades y las páginas que maneja son las siguientes:

- publicarproyecto.php
- consultarproyaa.php
- consultarproyestudio.php
- pmodificarproyaa.php
- verproyectosaa.php
- verproyestudio.php
- verproyrechazados.php

Para este subsistema se ha elegido presentar la implementación de los procesos utilizados para modificar un proyecto de acto administrativo. En la tabla 4.5 se puede observar un segmento de php que permite realizar este proceso, teniendo en cuenta y validando que dicho proceso solo se puede realizar si el usuario esta autorizado, esto es si es el autor del mismo.

Código PHP	Descripción
<pre>elseif((isset(\$btnmodproyaa))&&(\$textcuerpo!="")) { \$sql = "update proyectosaa set resumen=\$textresumen', cuerpo='\$textcuerpo', palabrasclave=\$txtpalclave', dia=\$sdia', mes=\$smes', anno=\$sanno', codtema=\$ctema', tipoacto=\$caa' where numproyecto= '\$codigoproym'; \$result = mysql_query(\$sql); if(\$result){ print("<SCRIPT language=JavaScript>alert(\"El proyecto de acto administrativo se ha actualizado satisfactoriamente\"); </script>"); print("<SCRIPT language=JavaScript> window.location=\"pmodificarproyaa.php\" </script>"); } }</pre>	<p>Este código se ejecuta siempre y cuando se haya echo clic en el botón “actualizar proyecto” (btnmodproyaa). Previo a este código se hacen las validaciones necesarias, posteriormente aquí se procede a realizar la actualización de la tabla proyectosaa y se verifica que el proceso se haya realizado satisfactoriamente.</p>

Tabla 4.5. Código PHP que permite modificar un proyecto

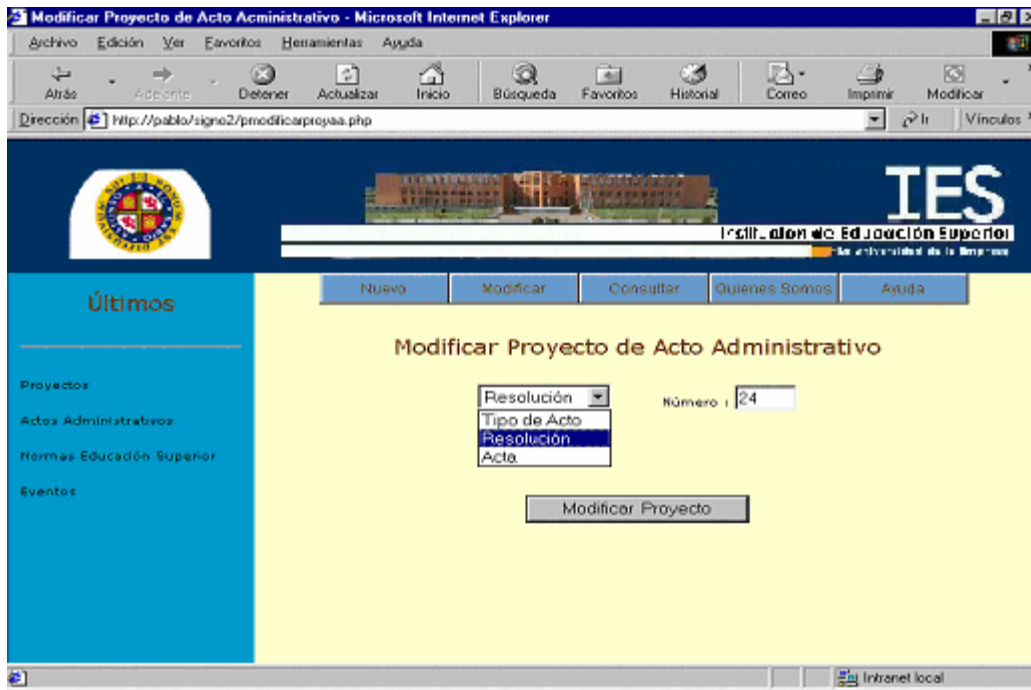


Figura 4.3 . Interfaz que permite Modificar un proyecto

11.3.2 Pruebas

- **Casos de prueba**

- Correcto almacenamiento de información en la base de datos
- Conexión y selección correcta de la base de datos¹⁷
- Evitar ambigüedad en los datos
- Integridad transaccional

- **Procedimiento de prueba**

En este subsistema se muestra el caso de prueba : Mantener la Integridad Transaccional, el cual se puede observar en la tabla 4.6. Para observar este proceso se muestra la tabla proyectosaa antes de modificar el registro, posteriormente se ingresa a la página modificar proyectos mediante el menú “Modificar – Proyecto”, en dicha página se pide seleccionar el tipo de proyectos y el número del mismo, se verifican los datos y se muestra la información del proyecto para que pueda ser modificada, una vez realizados los cambios se oprime el botón “Actualizar Proyecto”, se valida la información ingresada y se actualiza el registro en la tabla antes mencionada, dicha actualización se muestra igualmente en la tabla 4.6.

¹⁷ Este caso de prueba es común a todos los subsistemas generales de aplicación, por ende solo se realizara prueba en este subsistema.

**Datos
Entrada**

Modificar Proyecto de Acto Administrativo

24Resumen : se efectúa un traslado y una distribución parcial en el Presupuesto de Gastos de Funcionamiento de la universidad para la vigencia fiscal de 2003

Palabras Clave : Presupuesto (Deben ir separadas por coma, por ej. "eleccion, comité, consejo")

Fecha : 16 Septiembre 2003

Tema : Presupuesto

Tipo : Resolucion Vicerectoral

Cuerpo del documento :

RESOLUCION
(16 de septiembre de 2003)

Por la cual se efectúa un traslado y una distribución parcial en el Presupuesto de Gastos de Funcionamiento de la universidad para la vigencia fiscal de 2003

EL RECTOR DE LA UNIVERSIDAD PRIVADA ALFONSO X

Actualizar Proyecto Enviar a Estudio Volver

Registro en la base de datos

Registros antes de efectuar cambios

numproyecto	tipoacto	dia	anno	codtema	palabrasclave	cuerpo	resumen	mes	codente	observaciones	sugerencias	id
24		3	22	2003	2 Presupuesto, Gastos	(BLOB)	(BLOB)	Septiembre	10	(BLOB)	(BLOB)	ersa

Registros después de efectuar cambios

numproyecto	tipoacto	dia	anno	codtema	palabrasclave	cuerpo	resumen	mes	codente	estado	observaciones	sugerencias	id
8		3	16	2003	2 Presupuesto	(BLOB)	(BLOB)	Septiembre	10	(BLOB)	(BLOB)	(BLOB)	er

Tabla 4.6. . Prueba para modificar un proyecto y a la vez enviarlo a estudio

11.4. GESTION DE CONSULTA

Este es el subsistema más utilizado por los usuarios, su finalidad es extraer información de la base de datos cumpliendo los criterios de búsqueda elegidos por dichos usuarios a través de la interfaz.

Para ejecutar las búsquedas, se utiliza el lenguaje PHP, por medio del cual se accede a la base de datos y se realiza la sentencia sql encargada de obtener los datos que cumplen los criterios elegidos.

Los resultados serán presentados por medio de una interfaz que muestra el tipo de acto, el número, el ente al que pertenece, la fecha en que fue publicado, el resumen o abstrac del documento¹⁸, y un acceso a la información completa.

11.4.1 Implementación

La implementación se realiza mediante código PHP incluido en 5 páginas web:

- consultaraa.php
- consultarproyaa.php
- consultarproyestudio
- consultarproyrechazados
- consultarnes¹⁹.php
- consultarev.php
- consultaru.php

¹⁸ En este apartado, con la palabra “documento” no solo nos referimos a lo relacionado con actos administrativos sino con la información completa de eventos y normatividad de educación superior

¹⁹ Normas de Educación Superior

Para este subsistema se presenta los procesos realizados para implementar la búsqueda de proyectos de actos administrativos. En la tabla 4.7. se encuentra un segmento de código php que permite la búsqueda de un proyecto. Esta búsqueda se realiza en la tabla proyectosaa. En la figura 4.4. se encuentra la página en la cual se ingresa los datos características del documento o documentos que se buscan.

Código PHP	Descripción
<pre> Código perteneciente a la página consultarproyaa.php switch (\$scriterios) { case 'Ente': \$var1=" \$tabla.codente = ".\$coden."";break; case 'Tipo': \$var1=" \$tabla.tipoacto = ".\$caa."";break; case 'Autor': \$var1=" \$tabla.id=".\$idautor."";break; case 'Palabras Clave':\$var1=" \$tabla.palabrasclave = ".\$txtpalclave."";break; } session_register(\$var1); // Registrar Variable de sesion print("<SCRIPT language=JavaScript> window.location=\"proyencontrados.php\" </script>"); Codigo perteneciente a la página frmproyectos.php \$cacto="select distinct usuarios.nomusuario,actosente.naente, entes.nomente, numacto, dia, mes, anno, resumen from actosadministrativos, actosente, usuarios, entes where entes.codente=actosadministrativos.codente && usuarios.id=actosadministrativos.id && actosente.codactosente=actosadministrativos.tipoacto && actosadministrativos.id='\$txtid' && ".\$var1; \$cacto1=mysql_query(\$cacto,\$conexion); \$caa=mysql_fetch_array(\$cacto1) </pre>	<p>Con esta sección de código se pretende dar una idea de cómo se realiza el proceso de búsqueda. En la página de consultarproyaa.php, se ingresan los criterios de búsqueda, aquí por efectos de espacio solo se muestra cuando se elige un criterio del primer grupo. Según sea la opción elegida la primera parte de la consulta es almacenada en la variable \$var1. La segunda sección de código, pertenece a la página frmproyectos.php, en la cual se presenta la consulta total, aquí se hacen las validaciones que verifican la existencia de datos y se concatena con la variable var1.</p>

Tabla 4.7 . Código PHP que permite consultar un Proyecto de Acto Administrativo

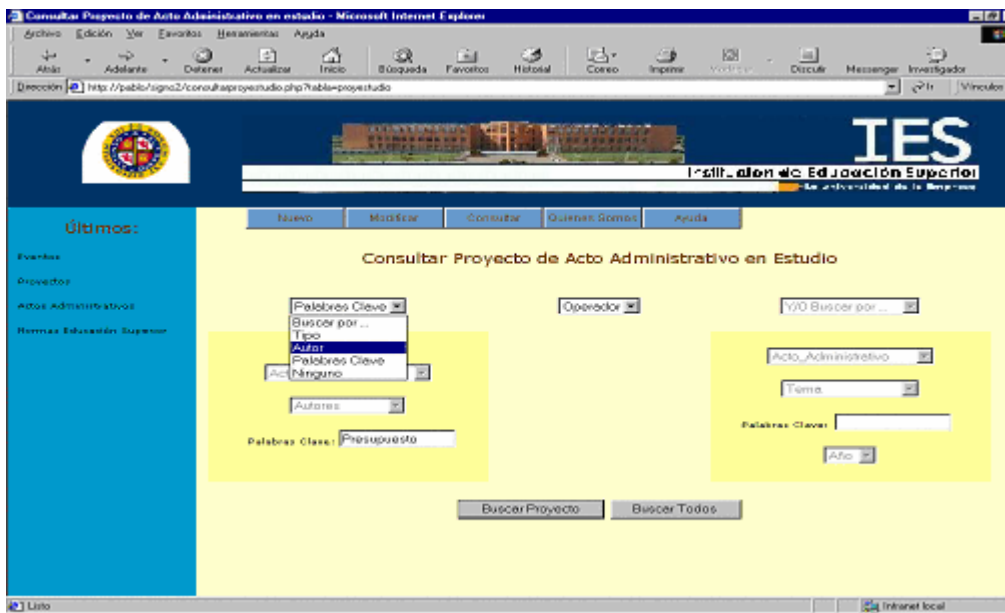


Figura 4.4. . Interfaz que permite consultar un Proyecto de Acto Administrativo

11.4.2 Pruebas

- **Casos de prueba**

- Conexión y selección correcta de la base de datos.
- Congruencia entre los criterios de búsqueda y los resultados de la misma.

En esta parte se muestra el caso de prueba Congruencia entre los criterios de búsqueda y los resultados de la misma.

- **Procedimiento de Prueba**

Como en todos los subsistemas generales de aplicación, la información (en este caso los diferentes criterios de búsqueda) proviene de la interfaz, una vez tomada esta información se procede a realizar la búsqueda de los diferentes documentos que cumplen con las condiciones dadas.

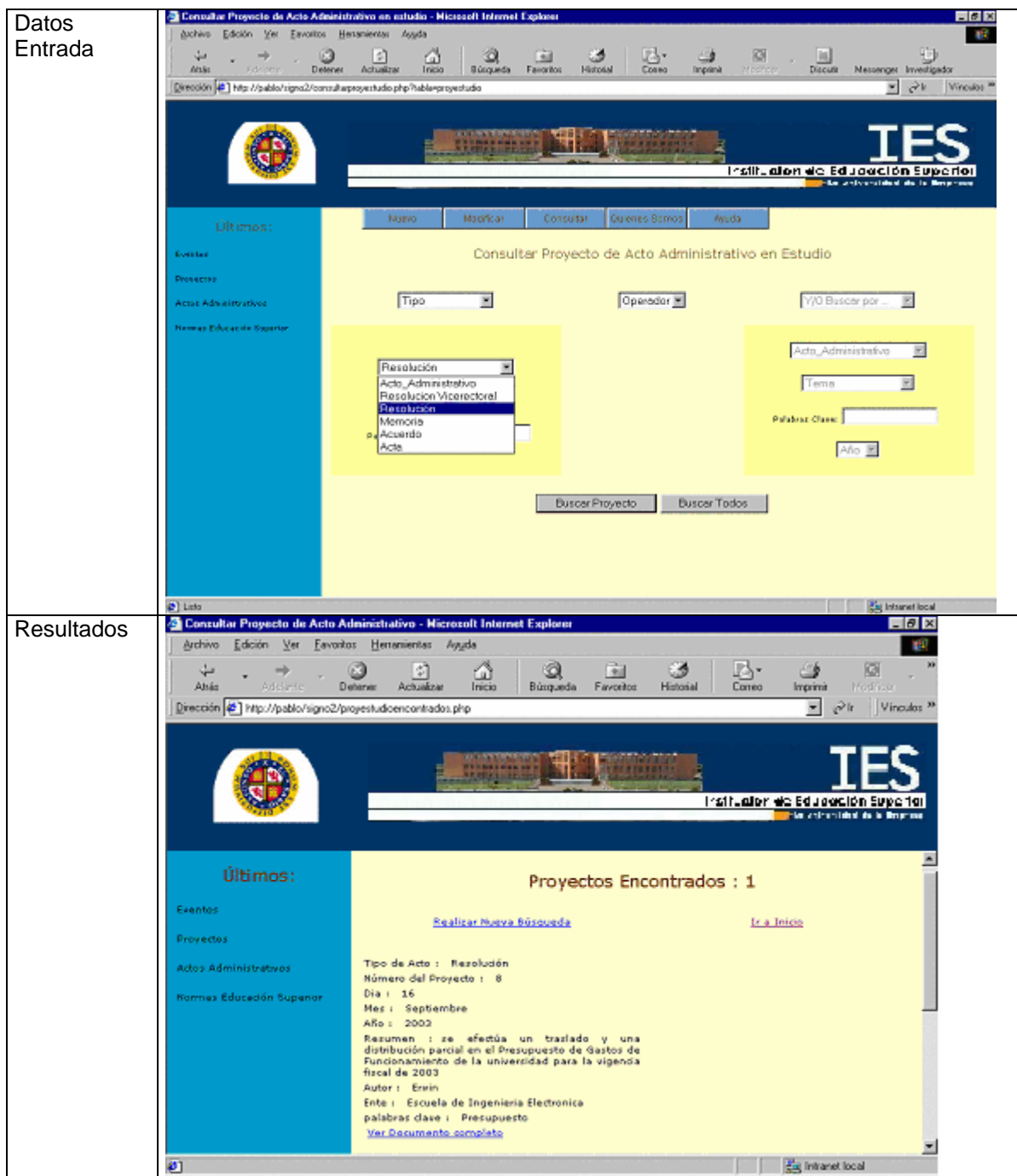


Tabla 4.8. . Prueba para buscar un Proyecto de Acto Administrativo

La prueba ha sido satisfactoria, dado que los resultados que arrojó la búsqueda son consistentes con los criterios elegidos.

11.5. RESUMEN CAPITULO 8

En este capítulo se presentaron los subsistemas que permiten, tanto la configuración del Sistema de Información Intranet, como el posterior manejo de la información normativa. Para cada subsistema se realizó la implementación y pruebas de los procesos involucrados en el funcionamiento de los mismos.

Debido a que no se desea ser repetitivos en el documento, solo se presentaron segmentos de código de una funcionalidad por cada subsistema y una caso de prueba a la misma, las demás pruebas se puede encontrar en el anexo B y la aplicación se puede encontrar en el grupo de investigación STI.

CAPITULO 9

12. SUBSISTEMAS GENERALES DE LA APLICACIÓN

En este capítulo se presenta la implementación y pruebas de los subsistemas cuyo principal objetivo es manejar gran parte de la seguridad del sistema, ellos son los llamados subsistemas generales de la aplicación. El primero es *gestión de sesión* que maneja el ingreso y navegación de los usuarios en el sistema y el segundo es *gestión de interfaz* encargado de recibir los datos dados por el usuario y pasarlos al subsistema involucrado en el proceso que se esté realizando.

12.1. GESTION DE SESION

Como se ha explicado desde la fase de elaboración, cuando el usuario, desea hacer uso de un proceso para llevar a cabo un caso de uso de cualquier subsistema, se debe verificar que dicho usuario tenga permiso para su ejecución y para acceder o modificar a la información que este maneja. Dada la naturaleza del sistema es de vital importancia evitar el ingreso de usuarios no autorizados al sistema. Cuando un usuario desea ingresar al sistema, envía su login y su clave de acceso. El sistema valida esta información y crea una nueva sesión registrando las variables de sesión propias del mismo y que serán preservadas mientras se encuentre dentro del sistema.

12.1.1 Implementación

Para implementar este subsistema se ha construido en lenguaje PHP los procesos que permiten cumplir con las funciones dadas a este subsistema. En esta parte se valida que el id y la clave que el usuario ingresó sean válidos y se registran las variables de sesión necesarias para su identificación. . En la tabla 4.9 se encuentra el segmento de código php que registra dichas variables y en la figura 4.2. se presenta la página que permite el ingreso al sistema

Código PHP	Descripción
<pre>// verifica la existencia del usuario en el sistema \$consulta=("select id, clave from usuarios where id='\$txtid'"); \$consulta1=mysql_query(\$consulta,\$conexion); if(!mysql_fetch_array(\$consulta1)) { print("<SCRIPT language=JavaScript>alert(\"Usuario no registrado\"); </script>"); print("<SCRIPT language=JavaScript> window.location=\"iniciodesesion.php\" </script>"); } else { session_register(txtid,txtclave); // Registra en la sesión el id y la clave de usuario // consulta el perfil del usuario \$codtipo="select usuarios.codtipo from usuarios where usuarios.id='\$txtid'"; \$codtipo1=mysql_query(\$codtipo,\$conexion); \$infotipo=mysql_fetch_array(\$codtipo1); \$tipo=(\$infotipo["codtipo"]) session_register(tipo);</pre>	<p>Con esta sección de código se verifica que el id y la clave de usuario existan en la base de datos y se registran dichas variables, posteriormente se realiza una consulta para conocer el tipo de usuario al que pertenece y se registra esta variable la cual permite el acceso del usuario para efectuar ciertos casos de uso.</p>

Tabla 4.9 . Código PHP que permite validar el ingreso de un usuario al sistema con sus respectivos permisos

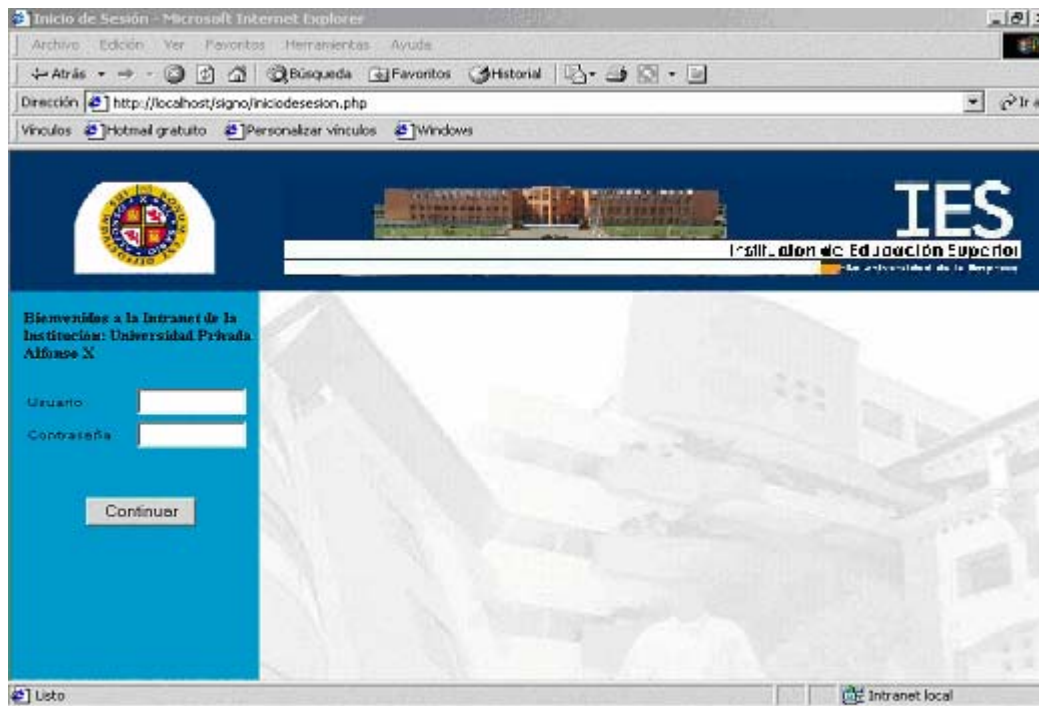


Figura 4.5 . Interfaz que permite ingresar al sistema

12.1.2 Pruebas

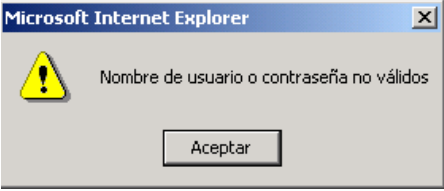

- **Casos de Prueba**

- Validar que el usuario que desea ingresar al sistema exista en la base datos
- Verificar el tipo de usuario al que pertenece para determinar a cuales casos de uso puede tener acceso.

- **Procedimiento de Prueba**

Para este subsistema se va a validar el caso de prueba: Validar que el usuario que desea ingresar al sistema exista en la base datos, para ello se ingresa un nombre

de usuario válido y una contraseña inválida, en este caso debe aparecer un mensaje de error . Posteriormente se ingresan tanto nombre de usuario como contraseña válidos y debe permitir el ingreso al sistema.

<p>Datos Entrada</p>	<p>Caso 1: Usuario: ersanchez (Dato Válido) Contraseña: XXX (Dato inválido)</p> <p>Caso 2 : Usuario: ersanchez (Dato Válido) Contraseña: xxxxxx (Dato Válido)</p>
<p>Resultados</p>	<p>Caso 1:</p>  <p>Caso 2:</p> 

--	--

Tabla 4.10 . Prueba para validar el ingreso de un usuario al Sistema

12.2. GESTION DE INTERFAZ

El principal objetivo de este subsistema es permitir la comunicación entre los subsistemas y estos con los usuarios, para tal fin, se construye un conjunto de páginas web que faciliten estos procesos. La interfaz se construye utilizando php y javascript, lenguajes destinados a proporcionar la comunicación entre los usuarios y las aplicaciones así como las validaciones de la información que el usuario ingresa.

12.2.1 Implementación

Para este subsistema se construyeron 2 grupos de páginas, el primero corresponde a la configuración del sistema de información intranet y el segundo corresponde a el sistema generado.

Primer grupo de páginas

- ☑ Bienvenida (bienvenida.php): Es la página inicial del instalador, en ella se da las características básicas que se debe tener en cuenta para realizar la instalación del sistema.
- ☑ Información General (informaciongeneral.php) : Es la página donde se ingresa la información principal de la Institución, como el nombre de la IES, misión, visión, objetivos, organigrama, calendario académico además del nombre de la Intranet.
- ☑ Crear entes (crearentes.php) : Como su nombre lo indica, esta página permite ingresar los datos que identifican en el sistema, al organismo emisor de actos administrativos.
- ☑ Crear tipo de Actos Administrativos (creartipoaa.php): En esta página se asignan en el sistema los tipos de actos administrativos pertinentes a cada ente.
- ☑ Editar el Color de los Marcos (editarcolormarcos.php) : Por medio de ella, se asigna el color de los diferentes marcos de las páginas de la Intranet.
- ☑ Editar Fuente (editarfuente.php): Esta página permite configurar las características de los 6 tipos de letra que se permiten manejar en la intranet.
- ☑ Emblemas (emblemas.php): Es la página que permite incluir las imágenes insignias de la IES, para las paginas de la Intranet.
- ☑ Resultados Instalador (resultadosinstalador.php): En esta página muestra los resultados de la instalación y permite acceder a las paginas de modificación.

Segundo grupo de páginas:

- Inicio de sesion (iniciodesesion.php):
- consultaraa.php: esta página permite buscar los actos administrativos.
- consultarproyaa.php: por medio de ella se buscan los proyectos de actos administrativos.
- consultarev.php: utilizando esta pagina se pueden consultar los diferentes eventos.
- consultarnes.php : permite consultar las normas de educación superior
- consultaru.php: se emplea para buscar a un usuario determinado
- publicarproyaa.php: permite elaborar proyectos de actos administrativos.
- publicarev.php: permite incluir los eventos que posteriormente se publicarán
- publicaractoadministrativo.php: Por medio de esta página se puede elaborar y publicar un acto administrativo
- crearusuario.php: permite ingresar los datos de un nuevo usuario
- publicarnes.php: por medio de esta página se incluye la normatividad que rige las IES y es básica para la elaboración de actos administrativos.
- Páginas para modificar proyectos, actos administrativos, eventos, usuarios, normatividad IES

Para este subsistema se presenta los procesos realizados validar el ingreso de la información de un ente al sistema. En la tabla 4.7. se encuentra un segmento de código php y javascript que permite hacer dichas validaciones y mostrar los errores cometidos. En la figura 4.6. se encuentra la página en la cual se ingresa los datos del ente en mención.

Código javascript	Descripción
<pre> <script language="JavaScript"> function validaremail() { if (frmentes.txtcorrente.value.indexOf('@', 0) == -1 frmentes.txtcorrente.value.indexOf('.', 0) == -1) { alert("Dirección de e-mail inválida"); frmentes.txtcorrente.focus(); } } } </script> </pre>	<p>Esta función valida que el correo incluya el símbolo @ y el punto, características que tiene todo correo electrónico. En caso contrario muestra el mensaje de error y el cursor se sitúa en el campo destinado para ello (txtcorrente)</p>

Tabla 4.11 . Código PHP que permite validar el ingreso de un ente al sistema



Figura 4.6 . Interfaz que permite Ingresar la información del ente

12.2.2 Pruebas

- **Casos de prueba**

Para este subsistema se utilizan pruebas negativas, es decir, pruebas que intentan provocar que el sistema falle.

- ☑ Validar que los datos ingresados correspondan en la base datos
- ☑ Verificar la no existencia de campos nulos donde es necesaria la información.

- **Procedimiento de prueba**

Aquí se va a mostrar el caso de prueba: Validar que los datos ingresados correspondan con el formato asignado. Este proceso se puede observar en la tabla 4.12. El procedimiento para realizar esta prueba consiste en ingresar datos que no correspondan al formato de los campos, por ejemplo, ingresar letra en campos numéricos , no incluir el símbolo @ en los correos, incluir caracteres no válidos, etc.

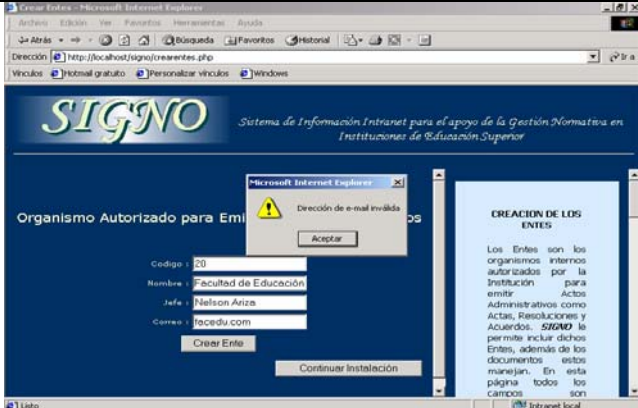
<p>Datos Entrada</p>	<p>Código : 20 Nombre : Facultad de Eduación Jefe : Nelson Ariza Correo : facedu.com</p>
<p>Resultados</p>	

Tabla 4.12. Prueba para validar el ingreso de un Ente en el sistema

12.3. PLANIFICACIÓN DE LA FASE DE TRANSICIÓN.

Esta sección ofrece una guía para la realización de la fase de transición del proyecto software enmarcado dentro de la propuesta de investigación titulada - *Sistema de Información Intranet para el apoyo de la gestión Normativa en Instituciones de Educación Superior* -. Aquí se exponen los aspectos generales que se deben tener en cuenta para desarrollar dicha fase.

El objetivo en esta parte del ciclo de desarrollo consiste principalmente en implantar el producto en su ambiente de operación. Al comenzar esta fase se cuenta con una versión operativa inicial, que al superar las diversas pruebas de la fase de construcción, ofrece la confianza suficiente para ser distribuida a los usuarios como una versión beta. Sin embargo, dado que existe una cantidad de trabajo por realizar que variará en función de los resultados que arrojen las pruebas beta, resulta imposible planificar por adelantado y en forma detallada los pasos que marcarán el camino en esta fase. De hecho, lo que es factible es ofrecer algunas orientaciones con respecto a la preparación de las pruebas de aceptación, la selección de los usuarios y los artefactos que se deben entregar para dar por terminada la primera versión de la herramienta.

12.4. RESUMEN CAPITULO 9

En este capítulo se presentó la implementación y pruebas de los subsistemas generales de aplicación. Estos son de gran importancia ya que interactúan con los demás subsistemas para validar tanto el ingreso de usuarios al sistema como la entrada de datos al mismo. Esta parte es primordial en cualquier aplicación, ya que involucra gran parte de la seguridad que se debe manejar para hacer del sistema un sistema confiable en el cual se pueda garantizar la veracidad de los datos que se manejan.

13. ELABORACIÓN DE MANUALES DEL SISTEMA

El manual de usuario, que se entrega con el software consta de dos partes, la primera está dirigida al administrador del sistema, y contendrá las recomendaciones básicas para la instalación del sistema así como las pautas que permitan el mantenimiento de la información ya sea a través del sistema o directamente en la base de datos que lo soporta; la segunda parte está dirigida a los usuarios del sistema, y por tanto orientada a los casos de uso que realizan los diferentes usuarios..

RESUMEN PARTE IV

Es en esta fase fue donde se realizó la mayoría de implementación y pruebas al sistema, consiguiendo una versión ejecutable de la herramienta. Por cuestiones de espacio en el documento solo se presentó algunas funcionalidades específicas de cada subsistema para verificar que la capacidad operativa del sistema cumplió con los objetivos planteados en el presente trabajo de grado. La totalidad de las pruebas se puede encontrar en el anexo E y la herramienta en el grupo de investigación STI.

Para continuar con el esquema llevado durante las fases de inicio y elaboración, se incluye el resumen de los casos de uso y la fase en las que se identificaron, describieron, analizaron, diseñaron e implementaron, dicho resumen se puede observar en la tabla 4.13. , en ella se puede observar que aparecen los casos de uso *editar colores*, *ingresar emblemas*, *editar fuentes*, los cuales hacen parte del caso de uso Editar presentación.

Casos de Uso	Identificado	Descrito	Analizado	Diseñados e implementados
Crear Identificación	F. Inicio	F. Inicio	F. Elaboración	F. Construcción
Crear Ente	F. Inicio	F. Inicio	F. Elaboración	F. Construcción
Crear Tipo Acto Administrativo	F. Inicio	F. Inicio	F. Elaboración	F. Construcción
Editar colores	F. Construcción	F. Construcción	F. Construcción	F. Construcción
Ingresar emblemas	F. Construcción	F. Construcción	F. Construcción	F. Construcción
Editar fuentes	F. Construcción	F. Construcción	F. Construcción	F. Construcción
Crear usuario	F. Inicio	F. Inicio		F. Construcción
Asignar Correo Electrónico	F. Inicio	F. Inicio		F. Construcción

Publicar Proyecto	F. Inicio	F. Inicio	F. Construcción	F. Construcción
Publicar Acto Administrativo	F. Inicio	F. Inicio	F. Inicio	F. Construcción
Publicar Legislación de Educación Superior	F. Elaboración	F. Elaboración		F. Construcción
Publicar Eventos	F. Elaboración	F. Elaboración		F. Construcción
Consultar Acto Administrativo	F. Inicio	F. Inicio	F. Inicio	F. Construcción
Consultar Proyectos de A.A.	F. Inicio	F. Inicio		F. Construcción
Consultar sugerencias	F. Inicio	F. Inicio		F. Construcción
Consultar legislación de Educación Superior	F. Elaboración	F. Elaboración		F. Construcción
Consultar Evento	F. Elaboración	F. Elaboración		F. Construcción
Elaborar Proyecto	F. Inicio	F. Inicio	F. Elaboración	F. Construcción
Modificar Proyecto	F. Inicio	F. Inicio		F. Construcción
Enviar a estudio.	F. Inicio	F. Inicio		F. Construcción
Dar sugerencias	F. Inicio	F. Inicio		F. Construcción
Consultar ayudante	F. Elaboración	F. Elaboración		F. Construcción
Imprimir	F. Inicio	F. Inicio		F. Construcción
Notificar	F. Inicio	F. Inicio		F. Construcción

Tabla 4.13. Resumen de Casos de Uso

Para finalizar se presenta la tabla 4.14 que contiene un paralelo entre los subsistemas hallados en la fase de elaboración y sus equivalentes en la fase de construcción.

Subsistema del diseño	Subsistema de la construcción
Gestión de configuración	informacióngeneral.php crearentes.php creartipoaa.php editarcolormarcos.php

	editarfuente.php emblemas.php resultados.php
Gestión de publicación	publicarproyaa.php publicaractoadministrativo.php publicarnes.php publicarev.php
Gestión de proyectos	modificarproyaa.php. Esta página junto con paginas auxiliares permiten modificar los proyectos en elaboración, en estudio, aplazados o rechazados.
Gestión de consulta	consultarproyaa.php consultaraa.php consultarnes.php consultaru.php consultarev.php
Gestión de sesión	
Gestión de interfaz	Se encuentran todas las páginas web

Tabla 4.14. Equivalencia entre los subsistemas hallados en las fases de elaboración y construcción

14. NOTAS BIBLIOGRAFICAS

- JACOBSON, Ivar. BOOCH, Grady. RUMBAUGH, James. **El Proceso Unificado de Desarrollo de Software**. Primera Edición. Addison Wesley. España, 2000. Al igual que para la parte II, en esta parte fue de gran importancia, ya que sirvió como guía en el desarrollo y maduración del proyecto.

- DANESH, Arman. **Aprendiendo Javascript en una Semana**. Prentice Hall. México, 1996. Este libro aportó información sobre el manejo de este lenguaje para implementar aplicaciones relacionadas con la interfaz.

- <http://quark.fe.up.pt/apachES/manual-Es/install.html> . Esta página contiene información sobre instalación y configuración del servidor Apache.

PARTE V – CONCLUYENDO EL TRABAJO DE GRADO

CAPITULO 10

15. REFLEXIONES FINALES

15.1. INTRODUCCION

Este capítulo está destinado a mostrar las conclusiones obtenidas después de haber realizado el presente trabajo de investigación. Esto incluye, la descripción del proceso que vivieron los autores en la concepción, planeación y desarrollo de este proyecto, la presentación de los principales logros alcanzados con la investigación, y un conjunto de recomendaciones relacionadas a fin de darle continuidad al trabajo de grado.

15.2. NARRANDO LA EXPERIENCIA

Cada experiencia que se vive es fuente continua de aprendizaje y crecimiento personal. Hemos considerado enriquecedor compartir con ustedes el proceso que vivimos durante el desarrollo de este trabajo de investigación, proceso que relatamos a continuación.

...Cuando comenzamos a buscar una idea de proyecto, indagamos en varias áreas en las que se podría trabajar y tuvimos la oportunidad de ir a la UTS, Institución de Educación Superior de Bucaramanga, en la cual pudimos observar que el manejo de documentación general era bastante rudimentario y por ende tedioso e inseguro. Decidimos hablar con el profesor Luis Carlos Gómez, para presentarle la problemática y él nos expuso cómo se podría limitar el proyecto a trabajar en la documentación normativa por ser ésta de gran importancia para las IES, debido que les proporciona el marco legal sobre el cual se soportan. Se trató entonces de lograr un convenio entre la UIS y la UTS, pero no fue posible por causas económicas, entonces el profesor Luis Carlos nos propuso construir una herramienta que se pudiera adaptar a cualquier IES, teniendo en cuenta que su documentación normativa tiene el mismo formato y lo único que cambia son los nombres de los Actos Administrativos y el Ente que puede emitirlos. La idea nos llamó bastante la atención y mas aún teniendo en cuenta que se plantearía un cambio en el proceso de Gestión Normativa en las IES, con el fin de involucrar en dicho proceso a la comunidad universitaria por medio de un Sistema de Información Intranet que permitiría a dicha comunidad elaborar proyectos de actos administrativos, publicarlos para someterlos a observaciones y críticas de la comunidad universitaria y posteriormente enviarlos a estudio en donde el ente decide si lo aprueba, rechaza o aplaza.

Con esta idea presentamos el tema ante la EISI en donde hubo un problema por el título, se arregló y fue aprobado. Fue entonces cuando empezamos, por una parte a estudiar sobre diferentes tipos de actos administrativos, decidiendo trabajar sobre Resoluciones, Acuerdos y Actas por ser estos los mas utilizados y por otra parte a realizar una investigación en las principales IES de Bucaramanga con el fin de indagar sobre el manejo que ellas daban al área normativa.

Con base en esta información empezamos la elaboración del plan. La metodología que decidimos seguir para la realización del Sistema de Información Intranet fue el

Proceso Unificado de Desarrollo de Software, debido a que al interior del grupo STI, quien auspició este proyecto, ésta era la metodología vigente.

La elaboración del plan fue un proceso de mucho trabajo, ya que por la naturaleza “estándar”²⁰ del proyecto debíamos ser muy cuidadosos a la hora de definir los objetivos, adicionalmente, el área que estábamos investigando era un área que no manejábamos ya que en la carrera no se trata el tema de la Gestión Normativa. Adicionalmente debíamos aprender bien la metodología a utilizar así como analizar cuáles herramientas software serían las adecuadas para desarrollar el proyecto, que para el caso fueron linux, apache, PHP, Postgres, html, javascript.

Después de haber hecho varias reuniones con el Director para revisar el plan, en las cuales surgían ideas, se corregían posibles errores, y avanzaba el desarrollo del mismo, se llegó a un punto en el cual se consideró que se podía entregar el plan a la escuela para su aprobación. El plan fue evaluado por un profesor quien consideró que aparentemente no era lo suficientemente para un trabajo de grado de dos personas he hizo algunas observaciones adicionales. Hubo entonces la necesidad de pasar una carta para aclarar la importancia del proyecto y finalmente el plan fue aprobado.

A partir de ese momento trabajamos arduamente para desarrollar el Sistema de Información Intranet –SIGNO - que culminamos con esta parte, sentando de esta manera las primeras bases de investigación en el área de Gestión Normativa. Con esto pretendemos que este sea un comienzo en la investigación de dicha área, en la que hay mucho por aprender, aplicar y mejorar, no solo en Instituciones de Educación Superior sino en todos los organismos que manejan normatividad.

²⁰ En este apartado mencionamos “Estándar” para referirnos a que el sistema debía adaptarse a cualquier Institución de Educación Superior.

15.3. CONCLUSIONES

El presente trabajo de grado ha sido un proceso de aprendizaje y enriquecimiento personal, del cual se puede concluir que:

- ✓ Aunque la investigación se realizó sólo en las principales IES de Bucaramanga, se pudo observar que el manejo que actualmente dan dichas Instituciones al proceso de Gestión Normativa no cumple a cabalidad con los principios y objetivos que sobre Educación Superior tiene Colombia. Adicionalmente no existen herramientas que permitan optimizar dicho proceso.
- ✓ El Proceso Unificado de Desarrollo de Software es una metodología que permite el desarrollo gradual, aspecto que se pudo observar durante el desarrollo del presente trabajo de grado, en el cual, el sistema se construyó progresivamente a lo largo del tiempo incluyendo requisitos, complementando los modelos, implementando y probando las facilidades, que dieron como resultado el Sistema de Información Intranet SIGNO.
- ✓ Se debe incentivar más el proceso de investigación para detectar áreas en las que la aplicación de Ingeniería de Sistemas, no solo brinde apoyo para optimizar procesos, sino promueva cambios metodológicos que aporten al desarrollo Institucional y organizacional.
- ✓ No se puede pretender que la aplicación del sistema desarrollado genere de inmediato un cambio en la cultura con respecto a la normatividad, pero sí, que se den los primeros pasos hacia un proceso que facilite el aprendizaje e involucre a toda la comunidad universitaria en el desarrollo de la Gestión Normativa.

- ✓ Un aspecto para destacar del presente trabajo de grado es su adaptabilidad a cualquier IES, debido a los aspectos comunes que en Actos Administrativos poseen dichas instituciones, por ende, es importante el identificar características comunes a diversas organizaciones, con lo cual se puede evitar el desarrollo de varias herramientas software que finalmente hacen lo mismo.

15.4. RECOMENDACIONES

La culminación del presente trabajo de grado no significa que la labor de investigación haya terminado, por el contrario marca el inicio de una etapa en la cual se debe avanzar ya que existe mucho por explorar, aprender y mejorar. A continuación se exponen una serie de recomendaciones a tener en cuenta a fin de darle continuidad al proyecto.

- ✓ Es imperioso iniciar la etapa de Transición en el Consultorio Jurídico de la Universidad Industrial de Santander, ya que el presente trabajo de grado brinda apoyo a los proyectos de maestría que pretenden iniciar una transformación organizacional en dicho estamento de la Universidad, mediante la autorregulación y aprendizaje organizacional entre otros.
- ✓ Se propone darle continuidad al presente proyecto, mediante la ampliación del sistema de Información Intranet, en la cual, el sistema maneje la jerarquía de los diferentes entes y actos administrativos, con el fin de orientar de una forma más acertada a los usuarios sobre la gestión normativa según los organismos encargados de dicho proceso. Adicionalmente se debe tener en cuenta mejorar la interfaz (vista preliminar de la configuración, en la medida en que se ingrese la información, que se incluyan otros estilos de letras, etc).

- ✓ Expandir la aplicación del sistema a otras Instituciones donde la gestión normativa sea similar, por ejemplo, en los consejos municipales, en las gobernaciones, Alcaldías, colegios, escuelas, etc. Teniendo en cuenta tipos adicionales de actos administrativos, según las necesidades que dichas instituciones tengan.

BIBLIOGRAFÍA

- ☑ BOBADILLA, Jesús. ALCOCER, Alejandro y otros. Superutilidades para Webmasters. Primera Edición. Editorial McGraw Hill. España, 1999.
- ☑ DAVID Garret y otros. Intranets al descubierto. Editorial Prentice Hall. España, 1997.
- ☑ DANESH, Arman. Aprendiendo Javascript en una semana. Prentice Hall. México, 1996.
- ☑ GREER, Tyson. Así son las Intranets. Primera Edición. McGraw Hill. España, 1998.
- ☑ INSTITUTO COLOMBIANO PARA EL FOMENTO DE EDUCACIÓN SUPERIOR –ICFES. Educación Superior Compendio de Normas.
- ☑ JACOBSON, Ivar, BOOCH, Grady, RUMBAUGH, James. El Proceso Unificado de Desarrollo de Software. Primera edición. Addison Wesley. España, 2000.
- ☑ JACOBSON, Ivar, BOOCH, Grady, RUMBAUGH, James. El Lenguaje Unificado de Modelado. Primera edición. Addison Wesley. España, 1999.
- ☑ MCCONNELL, Steve. Desarrollo y Gestión de Proyectos Informáticos. Primera Edición. McGraw-Hill. España, 1997

- PRESSMAN, Roger. Ingeniería del software. Un enfoque práctico. Cuarta edición. McGraw Hill. España, 1998.

- www.apache.org. Sitio oficial de Apache.

- www.php.net. Sitio oficial de PHP

- www.postgres.com. Sitio oficial de Postgres

- www.redhat.com. Sitio oficial de RedHat

ANEXO A

EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE Y UML

El siguiente anexo resume el marco metodológico utilizado en el presente trabajo de investigación. Inicialmente, se presentan diferentes modelos de procesos para la ingeniería del software, también llamados “ciclo de vida de desarrollo software” –CVDS-, y que constituyen el conjunto de procesos sistemáticos que tienen lugar durante la existencia del producto, desde su concepción inicial hasta que la organización decide no continuar manteniéndolo. Seguidamente, se expone con más detalle El Proceso Unificado de Desarrollo de Software, indicando las características, fases, flujos de trabajo y artefactos generados por el proceso. Se muestran algunos ejemplos de los modelos elaborados en este proyecto de grado. Adicionalmente se presenta una sección en la cual se da una visión general de UML.

Antes de caracterizar, de manera más amplia, cada uno de los procesos, se presenta en la figura A.1. un esquema de clasificación de los diferentes modelos de ciclos de vida y en la tabla A.1. una síntesis de las características comunes de cada uno de los modelos, esto con el ánimo de tener una visión general inicial que se desglose a lo largo del documento, para terminar con una síntesis más

compleja, la cual permita discernir entre las diferentes posibilidades a la hora de definir el paradigma apropiado para un determinado proyecto software.

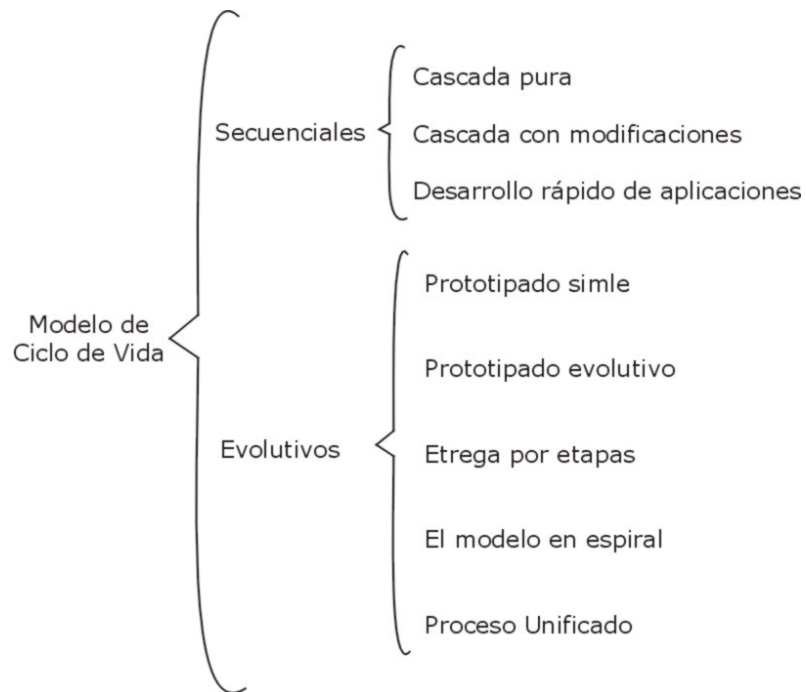
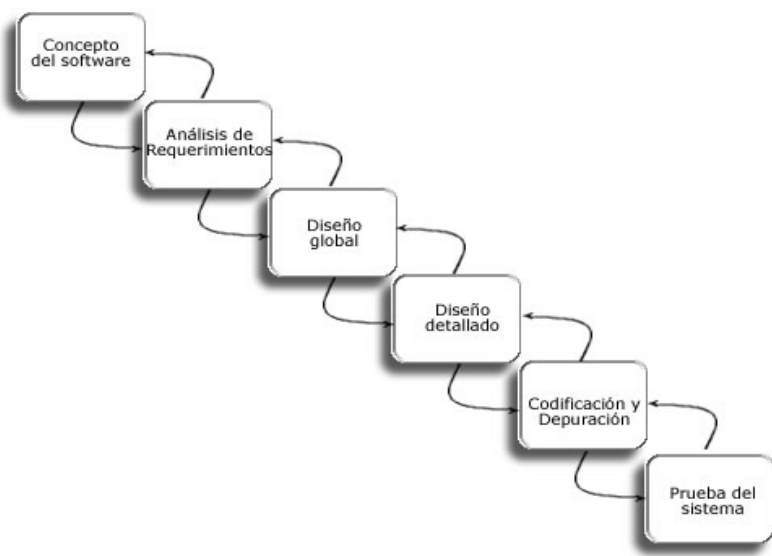


Figura A.1. Clasificación de los modelos de ciclo de vida del producto software

Ciclo de vida	Topología	Descripción
---------------	-----------	-------------

El modelo lineal secuencial ó Cascada Pura



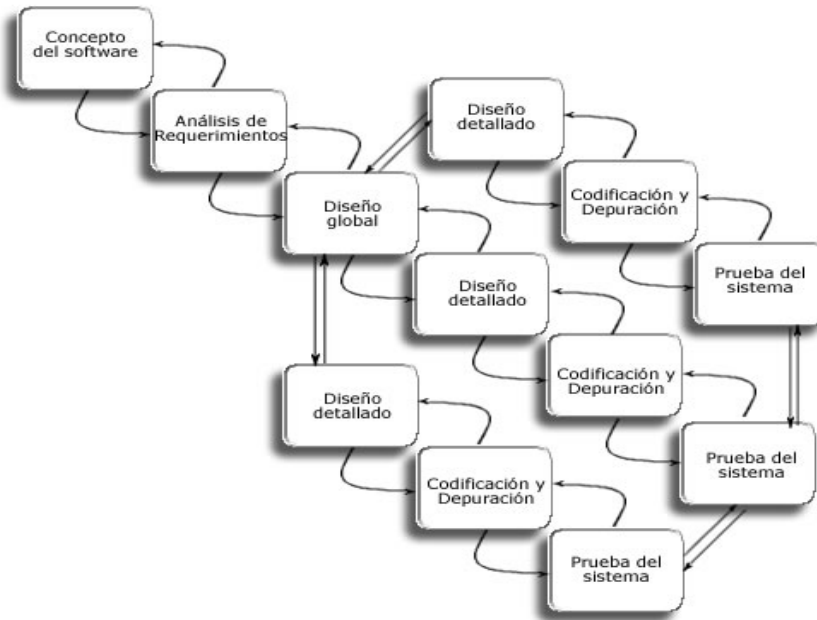
Es el predecesor de todos los modelos de ciclo de vida. Se utiliza correctamente para proyectos en los que la definición del producto es estable. Está dirigido por documentos, pues éstos son los resultados de la finalización de cada etapa. Una vez se inicia una etapa es difícil devolverse a una anterior.

Modelo de Cascadas con fases solapadas



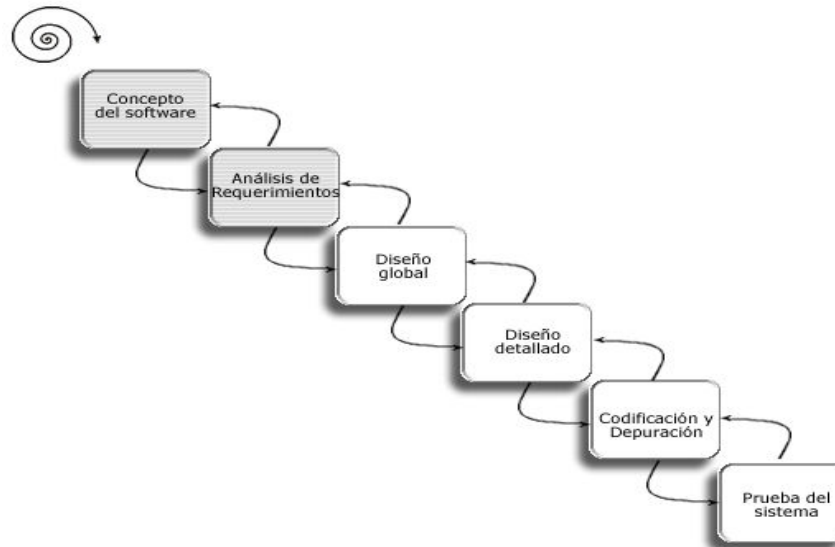
Presenta las mismas etapas que Cascada Pura pero permite un mayor solapamiento entre etapas; debido a esto los hitos son más ambiguos por lo que es más difícil determinar el progreso del proyecto. Se reduce la documentación.

Modelo de Cascadas
con Subproyectos



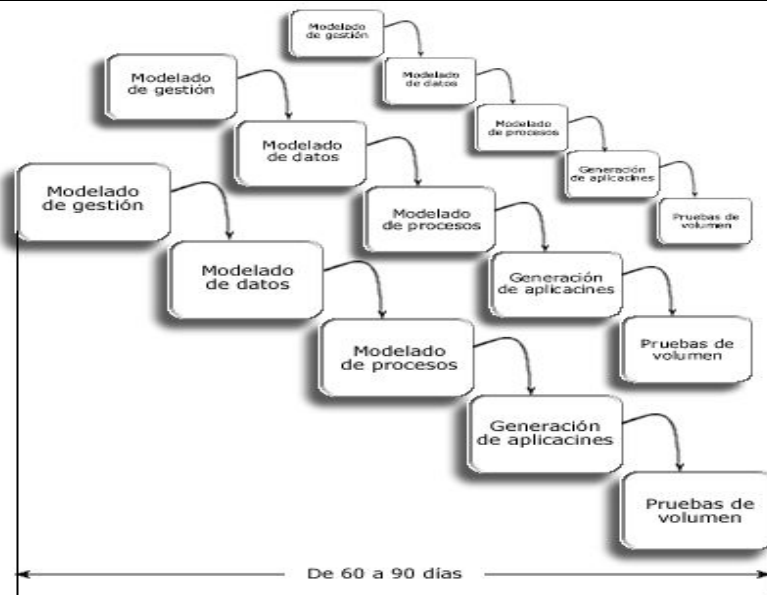
Requiere una planificación cuidadosa que permita la ejecución de algunas de las tareas de la cascada en paralelo. El principal riesgo es la presencia de interdependencias imprevistas.

Modelo de Cascadas
con Reducción de
Riesgos



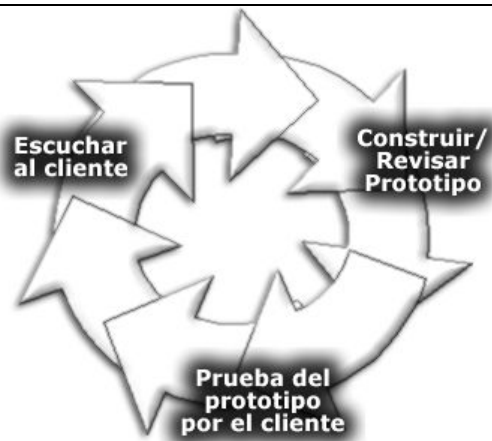
En esta modificación se usa un espiral en lo alto de la cascada para controlar el riesgo de los requerimientos que en la cascada pura deben estar completamente definidos antes de empezar el diseño global.

El modelo RAD
(Rapid Application
Development)



Es una adaptación a alta velocidad del modelo en cascada en el que se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes. Si se comprenden bien los requisitos y se limitan el ámbito del proyecto es posible lograr un sistema completamente funcional en un periodo corto.

Prototipado simple



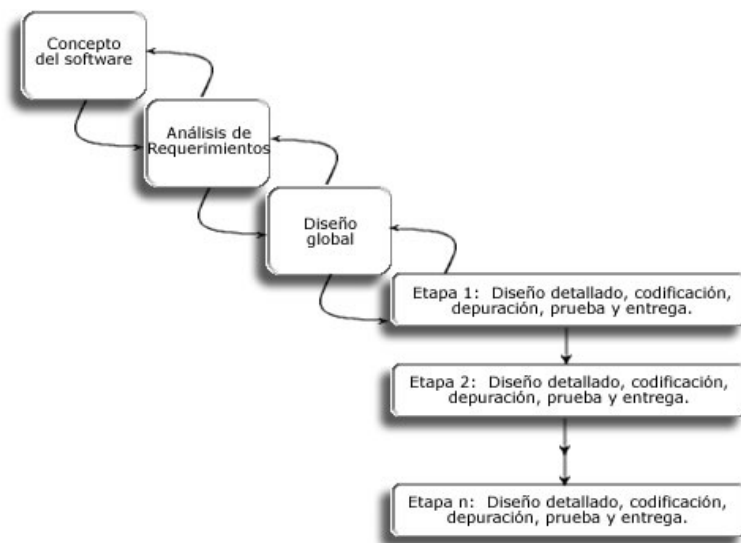
Este modelo es usado cuando los objetivos y requisitos del software no están definidos completamente desde el inicio, por lo que se construye un prototipo rápido basado en un diseño rápido, el cual es evaluado por el cliente y se utiliza para refinar los requisitos.

Prototipado evolutivo



En este modelo el concepto del sistema es desarrollado a medida que avanza el proyecto. Se comienza por diseñar e implementar las partes más importantes del programa en un prototipo, que es ampliado y refinado hasta que se termine. El prototipo se convierte en el producto final.

Modelo de entrega por etapas



La entrega por etapas evita el problema del modelo en cascada de no terminar ninguna parte del sistema que se está realizando hasta que esté finalizado completamente. Una vez finalizado el diseño, se puede implementar y entregar el sistema en etapas.

Tabla A.1. Características generales de los modelos de ciclo de vida del software

A.1. MODELOS DE CICLO DE VIDA SECUENCIALES

A.1.1. Cascadas

Cascada Pura

En el modelo de cascada pura un proyecto progresa a través de una secuencia ordenada de pasos llamadas etapas, las cuales van desde el concepto inicial de software hasta la prueba del sistema. Sus etapas no se solapan y una vez iniciada una de ellas, es difícil devolverse a la anterior. Al final de cada etapa se realiza una evaluación para determinar si se puede pasar a la siguiente o se continua en la actual hasta su cumplimiento.

El modelo en cascada no proporciona resultados tangibles en forma de software hasta que finaliza su ciclo. Sin embargo cuando se está familiarizado con el modelo, la documentación generada en cada etapa proporciona suficientes indicaciones del progreso del proyecto. La topología de este modelo puede observarse en la figura A.2.

A pesar de que este modelo presenta serios inconvenientes para su uso, como se muestra en la Tabla A.4. Resumen de los principales Modelos de Ciclo de Vida de Desarrollo del Software y en la Tabla A.4. Guía para la selección del ciclo de vida a utilizar en un proyecto, sirve de base o plantilla para otros modelos de ciclo de vida mas efectivos y realistas.

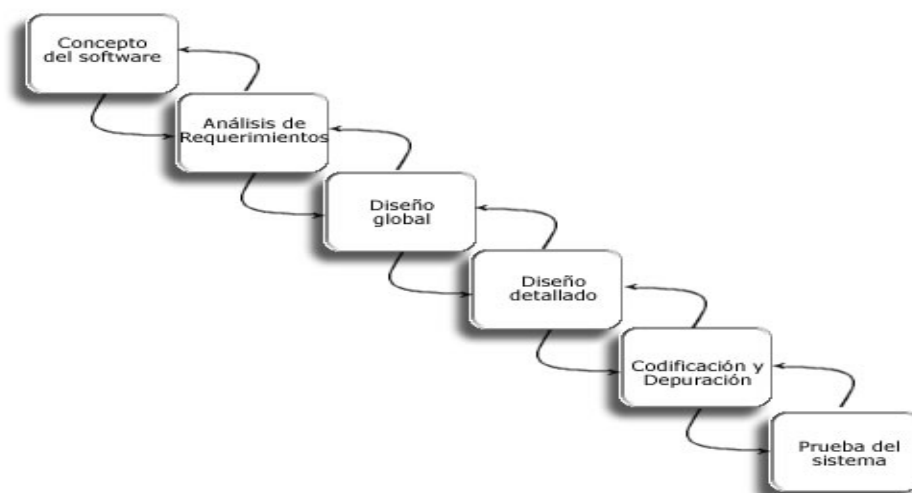


Figura A.2. Modelo de Cascada pura.

Etapa	Descripción	Actividades	Resultados
Concepto de software	Como el software siempre forma parte de un sistema más grande (o empresa), el trabajo comienza estableciendo qué requisitos de todos los elementos del sistema y asignando al software algún subgrupo de estos requisitos.	Interconexión con otros elementos como hardware, personas y bases de datos. Recolección de requisitos en todos los niveles. Pequeña porción de análisis y diseño.	Requisitos en el nivel de sistema, de empresa estratégico y en el nivel del área de negocio.
Análisis de requerimientos	Se intensifica la reunión de requisitos, centrados en el software.	Estudio de: La naturaleza de los programas a construirse, con base en la comprensión del dominio de información del software.	Comprensión de la naturaleza de los programas que han de construirse.
Diseño global y detallado	El diseño del software es un proceso que encierra muchos pasos y se centra en cuatro atributos distintos de programa: estructura de datos, arquitectura de software, representaciones de interfaz y detalle procedimental (algoritmo).	Traducir requisitos en una representación del software que permitan evaluar su calidad antes de comenzar la codificación.	Representación del software.
Codificación y depuración	El diseño se debe traducir en una forma legible por la máquina. El paso de generación de código lleva a cabo esta tarea.	Generar código.	Código legible por la máquina.
Prueba del sistema	Proceso centrado en los procesos lógicos internos del software, para asegurar la ejecución de sentencias y procesos externos funcionales.	Realización de pruebas para la detección de errores y aseguramiento de resultados.	Listado de errores y sugerencias de cambios.

Tabla A. 2. Etapas del ciclo de vida en cascada pura.

☑ **Cascadas modificadas.**

La mayoría de los inconvenientes del modelo de cascada pura son ocasionados por el tratamiento secuencial de sus etapas, sin permitir que una de ellas pueda iniciarse sin terminar la anterior, es decir, no permite etapas solapadas en su modelo. Las siguientes variaciones del método buscan solucionar este problema introduciendo pequeñas modificaciones al mismo.

La primera variación es llamada **Cascada con Fases Solapadas** como muestra la Figura A.3. La cual sugiere un mayor grado de solapamiento. Por ejemplo, se debería tener el diseño global y parte del detallado antes de considerar completo el análisis de requerimientos. El problema con este modelo es que debido al solapamiento entre sus capas, los hitos son más ambiguos, haciendo más difícil la gestión del progreso. Además la realización de actividades en paralelo unido a una mala comunicación puede llevar a suposiciones incorrectas e ineficacia en el desarrollo. Por lo tanto ésta variación es aplicable básicamente, a proyectos pequeños y bien definidos.



Figura A.3. Modelo de Cascada con Fases Solapadas.

La segunda variación es **Cascada con subproyectos** (ver Figura A.4.). En ella, se trata de dividir el sistema en subsistemas lógicamente independientes que se puedan tratar como proyectos separados y puedan proseguir a su propio ritmo. Esto evitaría el problema del modelo cascada pura que debe esperar a concluir la etapa de diseño detallado de todas las partes del sistema antes de poder empezar a implementar. El problema principal de este nuevo enfoque es la presencia de interdependencias imprevistas entre los módulos.

La tercera modificación es llamada **Cascada con Reducción de Riesgos**, la cual elimina el problema de tener que comprender la totalidad de requerimientos antes de continuar con el diseño de la arquitectura. Modificando el modelo de cascada pura se puede colocar una espiral antes iniciar sus etapas, para reducir el riesgo de los requerimientos. Durante esta espiral se controlan los riesgos principalmente del análisis de requerimientos y diseño de la arquitectura a través de prototipos de interfaz de usuario o de un sistema antiguo donde el usuario pueda exponer y aclarar sus requerimientos. Esta modificación se usa cuando el producto depende del desarrollo de un núcleo de alto riesgo para el sistema. Una imagen de este modelo se presenta en la figura A.5.

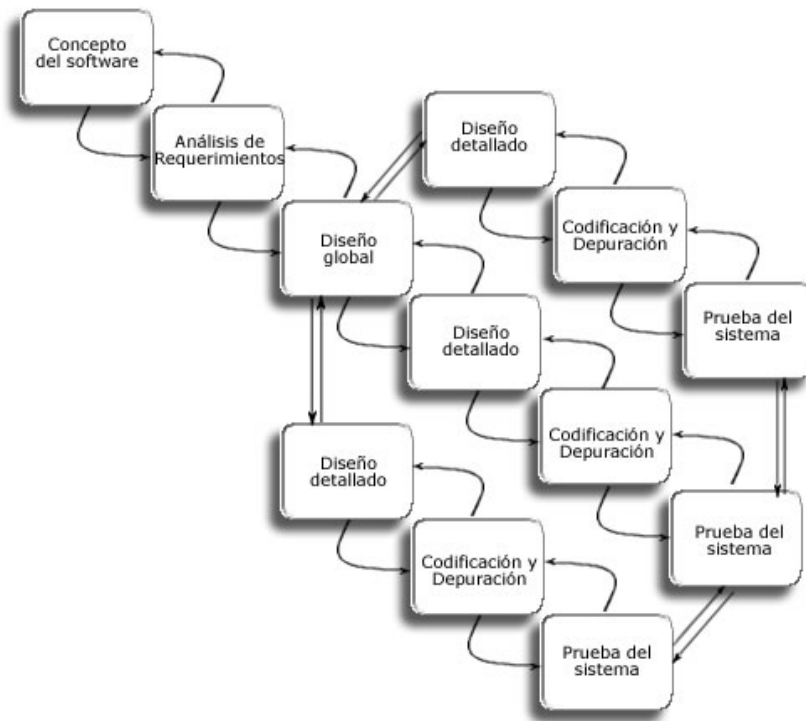


Figura A.4. Modelo de Cascada con subproyectos.

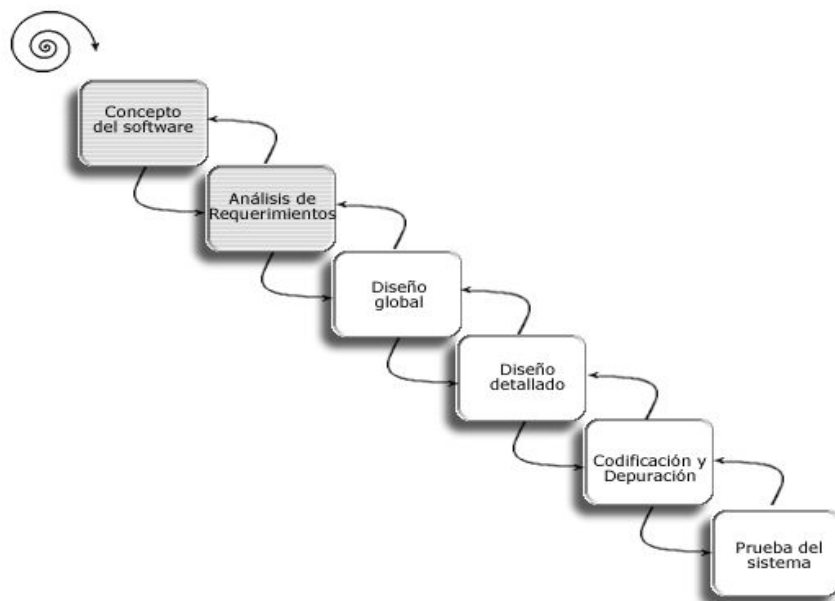


Figura A.5. Modelo de Cascada con Reducción de riesgos.

A.1.2. DRA (Desarrollo Rápido de Aplicaciones)

El modelo DRA es una adaptación a alta velocidad del modelo en cascada en el que se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes reutilizables y utilizando técnicas de cuarta generación en lugar de software con lenguajes de programación de tercera generación. En la figura 7 se muestran las fases que comprende este modelo.

En proyectos donde se comprenden bien los requisitos, se limita correctamente el ámbito del proyecto, y el sistema se puede dividir en módulos, el proceso DRA permite al equipo de desarrollo crear un sistema completamente funcional dentro de periodos cortos de tiempo (60-90 días). En la figura A.6 se muestra su topología y en la tabla A.3. se describen tanto las etapas como las actividades y resultados de cada una de ellas.

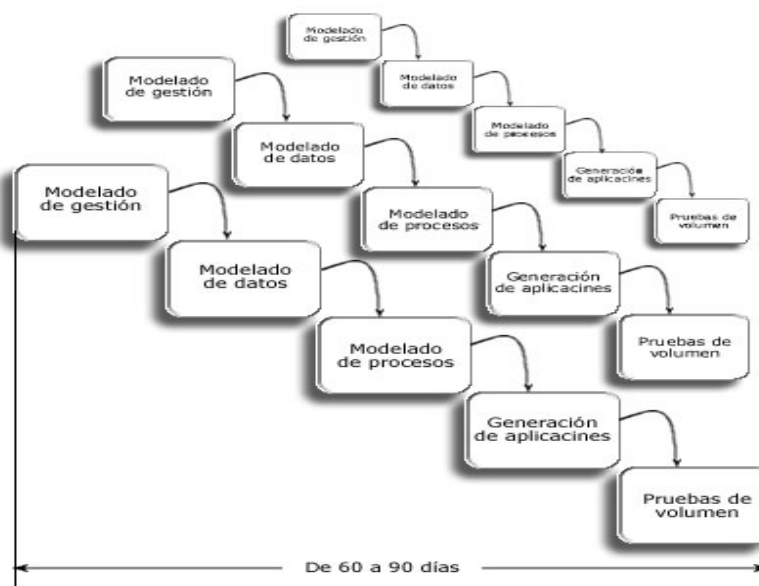


Figura A.6. Modelo de Desarrollo de Aplicaciones Rápidas (DRA).

Etapa	Descripción	Actividades	Resultados
Modelado de gestión	El flujo de información entre las funciones de gestión se modela de forma que responda a interrogantes alrededor de la gestión de la información.	Estudio de las fuentes, flujos y destino de la información.	Definición del flujo de información.
Modelado de datos	El flujo de información se refina como un conjunto de objetos de datos necesarios para apoyar la empresa. Se definen los atributos de los objetos y las relaciones entre estos objetos.	Definición de los atributos de los objetos y las relaciones entre ellos.	Atributos de los objetos y relaciones entre éstos.
Etapa	Descripción	Actividades	Resultados
Modelado del proceso	Se logra el flujo de información necesario para implementar una función de gestión. Las descripciones del proceso se crean para añadir, modificar, suprimir, o recuperar un objeto de datos.		
Generación de aplicaciones	En ésta fase se trabaja para utilizar componentes de programas ya existentes o para crear componentes reutilizables.	Generar código.	Código.
Pruebas y entrega	En esta etapa se prueban todos los componentes nuevos y se deben ejercitar todas las interfaces a fondo.		Listado de errores y cambios sugeridos.

Tabla A.3. Etapas del modelo de ciclo de vida DRA.

A.2. MODELOS DE CICLO DE VIDA EVOLUTIVOS

A.2.1. Prototipado

Prototipado simple

En este modelo, también llamado prototipo desechable, se construyen prototipos o acercamientos al sistema que sirven para identificar los requisitos del cliente. El prototipo se convierte en el primer sistema después de varias iteraciones cuando se estabilizan los requisitos. Sin embargo, este sistema debe ser nuevamente reconstruido dado que en su construcción no se tienen en cuenta aspectos de calidad y facilidad de mantenimiento. Ver Figura A.7.



Figura A.7. Modelo de Prototipo simple.

El paradigma de construcción de prototipos comienza con la recolección de requisitos conocidos, se realiza un diseño rápido centrado en los aspectos visibles para el cliente y se construye el primer prototipo. Este será evaluado por el cliente y se utilizará para refinar los requisitos del software a desarrollar. Se construye un nuevo prototipo y se continúa esta secuencia hasta que se satisfacen las necesidades del cliente y el desarrollador comprende lo que necesita hacer.

Para el desarrollo de este modelo se debe tener presente que este será solo un sistema inicial y el cliente debe estar consiente de esto. Este modelo es útil cuando no se definen los requisitos fácilmente. Además cuando se necesita probar la eficacia de un algoritmo, la capacidad de adaptación de un sistema operativo, o la forma como debería ser la interfaz de usuario.

Prototipado Evolutivo

El prototipado evolutivo es un modelo que toma sus bases del prototipado simple pero posee mayores controles sobre la calidad y desarrolla primero las áreas de mayor riesgo del sistema, de tal forma que el prototipo pueda ser tomado como producto final una vez se llegue a su fin. Es decir, en este modelo se desarrolla el concepto del sistema a medida que avanza el proyecto. El prototipo evolutivo es un enfoque donde se desarrolla primero las partes seleccionadas del sistema y luego el resto a partir de estas partes. A diferencia de otros tipos de prototipado, en el evolutivo no se descarta el código del prototipo; lo transforma en el código entregado finalmente. El desarrollo de prototipos continúa hasta que se decide que el prototipo es lo suficientemente bueno y se puede entregar como producto final.

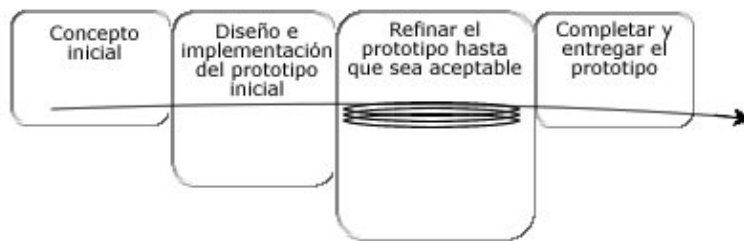


Figura A.8. Modelo de Prototipo Evolutivo.

Este modelo genera signos visibles de progreso, pero se puede correr el riesgo de caer en el esquema de codificar y corregir; sin ninguna planificación, ni gestión.

A.2.2. Entrega por etapas

En este modelo no se entrega el producto total al final del proyecto sino que se muestra al cliente en etapas refinadas sucesivamente proporcionando una funcionalidad útil antes de entregar el 100% del proyecto. Primero se realiza la definición del concepto del software, el análisis de requerimientos y la creación del diseño global de una arquitectura como en el modelo cascada. A continuación se

procede a realizar el diseño detallado, la codificación, depuración y prueba dentro de cada etapa.

Este modelo no funciona sin una planificación adecuada tanto para niveles técnicos como para niveles de gestión. En un nivel de gestión, se debe asegurar que las etapas que se planifican son significativas para el cliente y que el trabajo se distribuye entre el personal del proyecto de tal forma que pueden completar su trabajo a tiempo. En un nivel técnico, hay que asegurarse de que se han tenido en cuenta todas las dependencias técnicas entre diferentes componentes de un producto para evitar retrasos por dependencias no previstas.

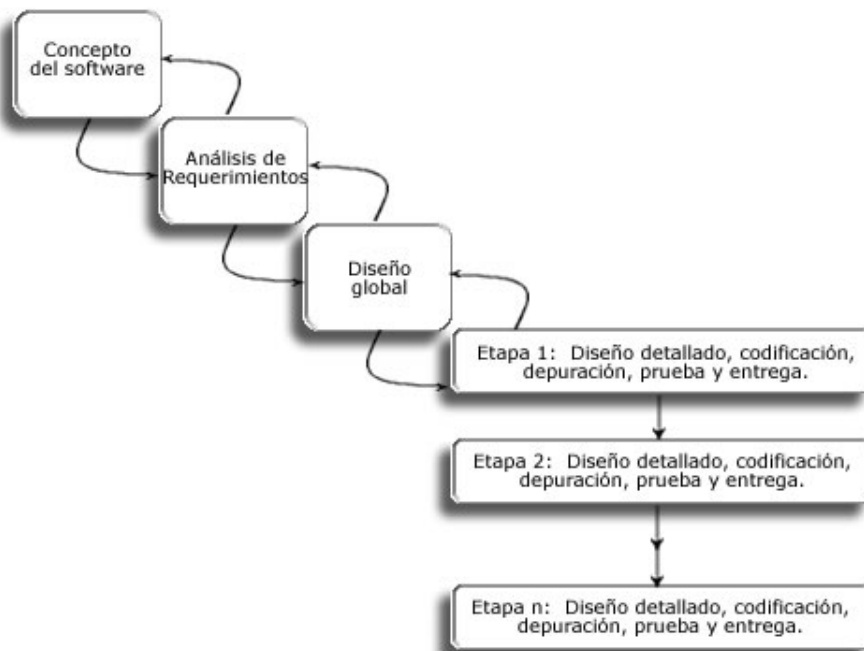


Figura A.9. Modelo de Entrega por etapas.

A.3. ESPIRAL

“El modelo de espiral es un modelo de ciclo de vida orientado a riesgos que divide un proyecto software en miniproyectos. Cada miniproyecto se centra en uno o mas riesgos importantes hasta que todos éstos estén controlados.“. El concepto “riesgo” puede referirse a requerimientos o arquitecturas poco comprensibles, problemas de ejecución importantes o con la tecnología subyacente.

El método parte de una escala pequeña en medio de la espiral, se localizan los riesgos, se genera un plan para manejarlos, y a continuación se establece una aproximación a la siguiente iteración. Después de controlar todos los riesgos mas importantes, el modelo en espiral finaliza con un ciclo de vida en cascada, con prototipado o con otro modelo. También se pueden incorporar otros modelos de ciclo de vida en sus iteraciones.

Este método requiere una mayor planificación y un mayor seguimiento del proyecto, pero a cambio se obtiene una gran mejora en la visibilidad del progreso y en la reducción del riesgo.

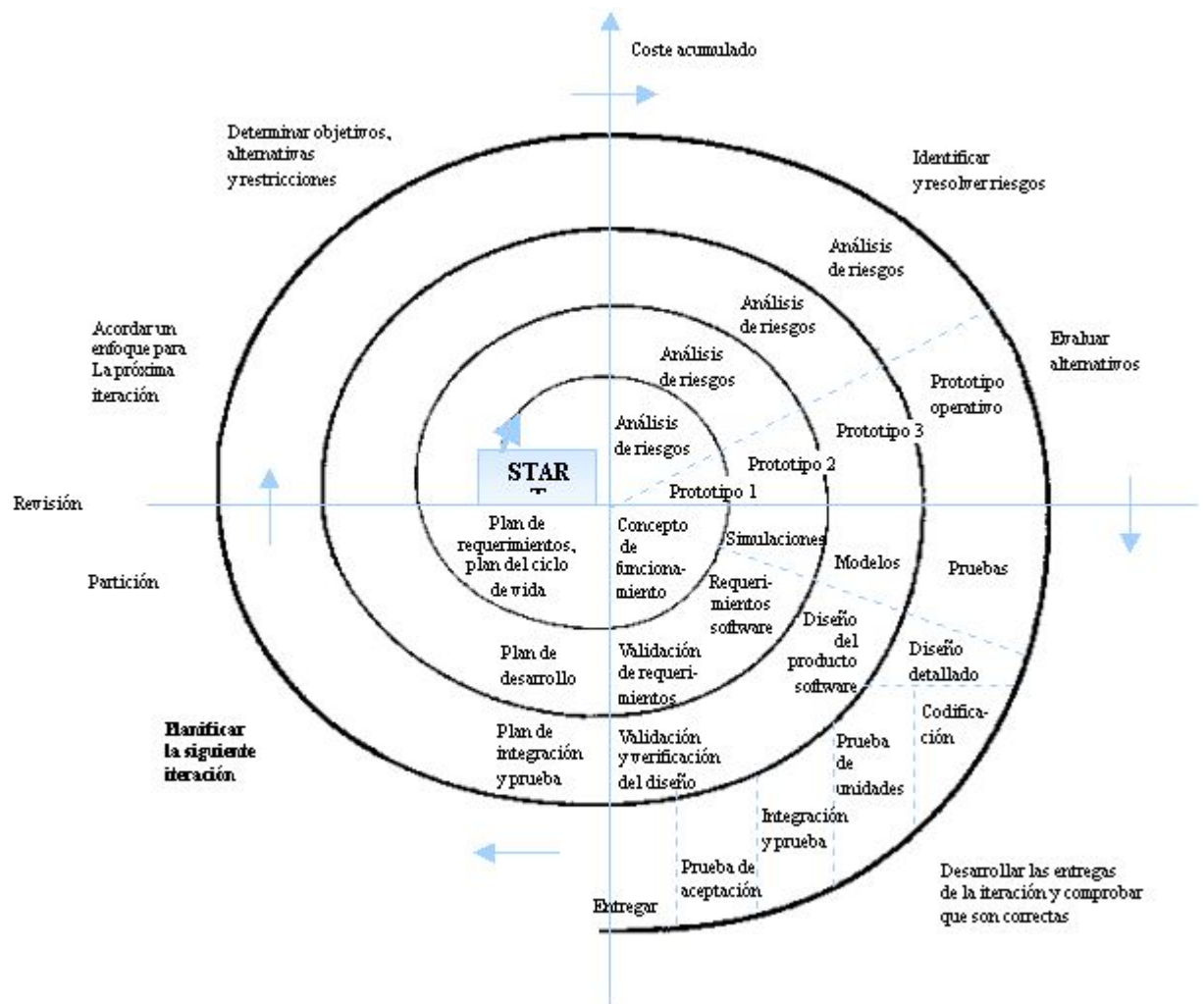


Figura A.10. Modelo de Espiral.

A.4. PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE

Según Jacobson(2000): “ El proceso unificado es un proceso de desarrollo de software. Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un sistema software. Sin embargo, el Proceso Unificado es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto”.

El objetivo de esta metodología es permitir la producción de un software que satisfaga las necesidades del usuario, de forma eficiente y predecible, dentro de los planes y presupuestos preestablecidos.

El proceso unificado está basado en componentes y se sostiene sobre tres ideas básicas: casos de uso, arquitectura y desarrollo iterativo e incremental. Para hacer que estas ideas funcionen, se necesita un proceso, que tenga en cuenta ciclos, fases, flujos de trabajo, gestión del riesgo, control de calidad, gestión de proyecto y control de la configuración.

Un desarrollo iterativo, guiado por los casos de uso y centrado en la arquitectura, construye un software mediante pequeños incrementos, y añade cada incremento a la acumulación previa de incrementos de tal forma que siempre se tenga una construcción ejecutable. La arquitectura proporciona la estructura sobre la cual guiar las iteraciones mientras que los casos de uso definen los objetivos y dirigen el trabajo de cada iteración.

De esta manera el proceso reduce el riesgo de grandes retrasos en la entrega de un producto, se fijan metas mas inmediatas por lo cual se puede controlar mejor el avance del proyecto.

El Proceso Unificado divide el proceso de desarrollo en ciclos, donde se obtiene una nueva versión del producto al final de cada ciclo. Cada ciclo se divide en cuatro Fases: Inicio, Elaboración, Construcción, y Transición. Cada una de estas fase concluye con un hito bien definido donde deben tomarse decisiones respecto al proyecto como la reestructuración del cronograma de trabajo. Cada una de estas fases se divide a su vez en iteraciones.

Cada iteración sigue la estructura de un pequeño ciclo de vida en cascada, pasando a través de los cinco flujos de trabajo fundamentales: requisitos, análisis, diseño, implementación y prueba. En la iteración también incluye la planificación que precede a los flujos de trabajo y la evaluación que va detrás de ellos.

“...Estos conceptos – los de desarrollo dirigido por los casos de uso, centrado en la arquitectura, iterativo e incremental – son de igual importancia. La arquitectura proporciona la estructura sobre la cual guiar las iteraciones, mientras que los casos de uso definen los objetivos y dirigen el trabajo de cada iteración. La eliminación de una de las tres ideas reduciría drásticamente el valor del Proceso Unificado. Es como un taburete de tres patas, sin una de ellas el taburete se cae...”
(Jacobson(2000))

Características del proceso

Las características de esta metodología se muestran en la figura A.1

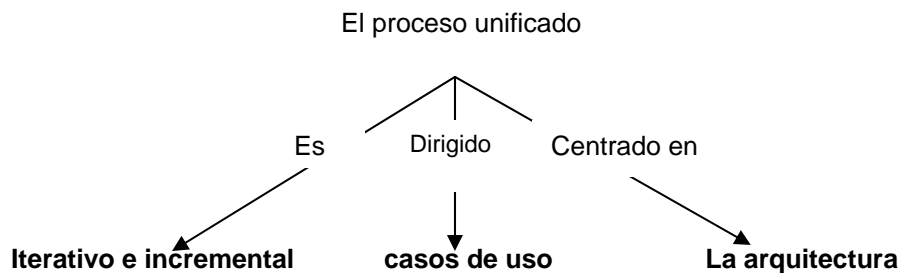


Figura A.11. Características del proceso unificado de desarrollo de software

Tipo de modelo	Ventajas	Inconvenientes	Se utiliza cuando...	Etapa de Detección de errores	Contacto con el cliente	Variantes y/o relación con otros modelos
Cascada Pura	<p>Presenta el proyecto con una estructura que ayuda a minimizar el esfuerzo inútil.</p> <p>Ayuda a localizar errores en las primeras etapas del proyecto a un bajo costo (Si se tiene una definición estable del producto)</p> <p>Permite enfrentarse a la complejidad de forma ordenada (En Proyectos complejos comprendidos totalmente)</p>	<p>No permite flexibilidad en los cambios, es decir, no permite volver atrás para corregir errores.</p> <p>No proporciona resultados tangibles en forma de software hasta que finaliza su ciclo.</p> <p>El cliente tiene contacto con el producto solo hasta el final del desarrollo.</p> <p>Dificultad para especificar exactamente los requerimientos al comienzo del proyecto.</p> <p>Discontinuidad de sus etapas, es decir, no se solapan.</p>	<p>Se construye de una versión de mantenimiento de un producto existente o migración de un producto a una nueva plataforma.</p> <p>Los requerimientos se pueden definir fácil y rápidamente a partir del conocimiento del área en estudio.</p> <p>Se tiene una definición estable del producto, con variaciones mínimas de los requisitos.</p> <p>Se utiliza tecnologías conocidas.</p> <p>Los requerimientos de calidad dominan sobre los de costes y planificación.</p> <p>Se dispone de personal inexperto en el manejo de</p>	Las pruebas para detección de errores se hacen al final del ciclo.	El cliente tiene contacto con el producto solo hasta el final del desarrollo	<p>Variantes: Cascada con Fases Solapadas</p> <p>Cascada con subproyectos.</p> <p>Cascada con Reducción de Riesgos</p> <p>Este tipo de metodología es una plantilla para otros métodos</p>

			una metodología en particular.			
DRA	<p>Desarrollo de proyectos en periodos cortos de tiempo (60-90 días)</p> <p>Enfatiza en el uso de componentes reutilizables.</p>	<p>Demanda gran cantidad de personas para proyectos grandes.</p> <p>Requiere un grupo de desarrolladores comprometidos.</p> <p>No es recomendable su uso, cuando los riesgos técnicos son altos.</p>	<p>Hay comprensión de los requisitos del cliente.</p> <p>Es posible separar el desarrollo del software en módulos.</p> <p>No se requiera un alto rendimiento en el proyecto.</p> <p>Hay buen conocimiento de las herramientas de desarrollo</p>	<p>Al concluir un modulo se entrega un producto de prueba al cliente.</p>	<p>Al final de cada módulo</p>	<p>Es una adaptación a alta velocidad del modelo en cascada en el que se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes</p>
Prototipado	<p>El producto final cumple con todos los requisitos del cliente.</p>	<p>Imposibilidad de conocer al comienzo del proyecto, el tiempo de duración.</p> <p>Puede caer fácilmente en desarrollo sin ninguna planeación.</p>	<p>Los requisitos cambian con rapidez</p> <p>El cliente no especifica los requerimientos.</p> <p>No se identifica fácilmente el área de aplicación</p> <p>No hay seguridad en la utilización de un algoritmo o plataforma de</p>	<p>Cada vez que el cliente prueba el prototipo</p>	<p>Permanentemente, hasta que se acepte el producto final</p>	<p>Variantes: Prototipado simple Prototipado evolutivo</p> <p>Es una plantilla para otros métodos</p>

			desarrollo.			
Espiral	<p>Las primeras iteraciones son menos costosas.</p> <p>Mientras los costos suben los riesgos bajan.</p> <p>Proporciona bastante control de gestión.</p> <p>Reducción continua del nivel de riesgo en el proyecto.</p> <p>Mejor visibilidad de progreso.</p>	<p>Modelo complicado.</p> <p>Puede ser difícil definir hitos y objetivos de comprobación que indiquen si esta preparado para pasar al siguiente nivel.</p> <p>El cliente no siempre esta predispuesto a ser parte integral del desarrollo de software</p>	<p>Es aplicable a todo tipo de desarrollo de software, especialmente en aquellos donde no se conocen bien los requerimientos del sistema.</p>	<p>Las pruebas se hacen al final de cada ciclo.</p>	<p>Hasta que se complete todo el numero especificado de iteraciones requeridas para completar el producto</p>	<p>Puede finalizar con un modelo en cascada, prototipado u otro modelo</p> <p>Puede incorporar otros modelos en sus iteraciones</p>
Incremental	<p>Presenta signos tangibles de progreso en el proyecto.</p> <p>Proporciona una funcionalidad útil al cliente antes de entregar el 100% del proyecto.</p>	<p>No funciona sin una planificación adecuada tanto para niveles técnicos como para niveles de gestión</p>	<p>Se conoce exactamente lo que se va a construir.</p> <p>El personal no esta disponible para una implementación completa.</p>	<p>Las pruebas se hacen al final de cada etapa</p>	<p>Cuando se hace las entregas parciales del producto software en cada etapa</p>	<p>Combina elementos del modelo en cascada con la filosofía iterativa de construcción por prototipos</p>

	Se pueden entregar las prestaciones más importantes al inicio.					
Proceso unificado	<p>Busca eliminar los riesgos críticos primero.</p> <p>Reduce el costo de riesgos.</p> <p>Presenta signos visibles de progreso rápidamente.</p> <p>Proporciona funcionalidad al cliente antes de entregar todo el proyecto.</p> <p>Se entregan prestaciones importantes al inicio.</p>	<p>Requiere mayor planificación.</p> <p>Requiere mayor control de gestión.</p> <p>Se necesita definir su arquitectura antes de dar inicio a las iteraciones de construcción.</p>	Es un marco de trabajo genérico.	Las pruebas se hacen al final de cada iteración. Además al final de cada fase se realiza una evaluación del plan de trabajo	Cuando se hace entrega de los incrementos al final de cada iteración, en especial en la fase de construcción.	<p>En cada una de sus fases e iteraciones utiliza el método en cascada.</p> <p>Al inicio se puede utilizar prototipos de prueba.</p>

Tabla A.4. Resumen de los principales Modelos de Ciclo de Vida de Desarrollo del Software.

◆ FASES E ITERACIONES

El Proceso Unificado se repite a lo largo de una serie de ciclos que constituyen la vida de un sistema. Cada ciclo consta de cuatro fases: inicio, elaboración, construcción y transición, cada una de ellas concluye con un hito bien definido donde deben tomarse decisiones respecto al proyecto, como la asignación y estimación de recursos, tiempo, número de iteraciones, etc. Igualmente cada fase se divide en iteraciones, éstas siguen la estructura de un pequeño ciclo de vida en cascada, pasando a través de los cinco flujos de trabajo fundamentales: requisitos, análisis, diseño, implementación y prueba (Ver figura A.12).

Como ya se ha dicho, el Proceso Unificado consta de las cuatro fases siguientes:

1. Inicio Establecer la planificación del proyecto.
2. Elaboración Establecer un plan para el proyecto y una arquitectura correcta.
3. Construcción Desarrollar el sistema.
4. Transición Proporcionar el sistema a sus usuarios finales.

Durante la fase de **inicio**, la meta es identificar las características principales con las que contará el sistema. El objetivo, habiendo terminado la fase de inicio, es tener una arquitectura candidata para la aplicación a desarrollar e igualmente un plan para controlar los diversos riesgos críticos identificados. En este punto se habrán delimitado los alcances del proyecto y se podrá justificar su puesta en marcha. Igualmente se tendrá un bosquejo del plan de trabajo. En este proyecto, la formulación de requisitos estuvo basada en un estudio preliminar sobre el manejo de los actos administrativos en las diferentes Instituciones de Educación Superior y la interacción por parte de la comunidad universitaria en la elaboración, ejecución y control de dichos actos. al igual que los beneficios que la herramienta ofrece a las Instituciones de Educación Superior para enriquecer su acción educadora.

El propósito de la fase de **elaboración**, es analizar el espacio del problema, establecer una base arquitectónica sólida, desarrollar el plan del proyecto y eliminar los elementos de más alto riesgo. Las decisiones arquitectónicas deben tomarse con una comprensión del sistema global. Esto implica que se deben describir la mayoría de los requisitos del sistema.

Durante la fase de **construcción**, se desarrolla de forma iterativa e incremental un producto completo, en su versión beta, que esté preparado para ofrecerlo a la comunidad de usuarios. Esto implica describir los requisitos restantes y los criterios de aceptación, refinando el diseño y completando la implementación del sistema y las pruebas del software. En este proyecto, al final de la fase de construcción se obtuvo el Sistema de Información Intranet para el apoyo de la Gestión Normativa en Instituciones de Educación Superior -**SIGNO**

Durante la fase de **transición**, el software se despliega en la comunidad de usuarios. Una vez que el sistema ha sido puesto en manos de los usuarios finales, a menudo aparecen cuestiones que requieren un desarrollo adicional para ajustar el sistema, corregir algunos problemas no detectados o finalizar algunas características que habían sido pospuestas. Esta fase demanda una considerable cantidad de tiempo, razón por la cual se omitió para los propósitos del presente proyecto de grado.

Al final de la fase de transición se decide si se han satisfecho los objetivos del ciclo de vida del proyecto, y se determina la necesidad de comenzar otro ciclo de desarrollo. Este es también un punto en el que se asimilan las lecciones aprendidas en el proyecto para mejorar el proceso de desarrollo, que será aplicado al próximo proyecto.

❖ FLUJOS DE TRABAJO DEL PROCESO

El Proceso Unificado consta de cinco flujos de trabajo. Cada uno de ellos está constituido por *Actividades*, que describen las tareas que llevan a cabo los *trabajadores* para crear o modificar los *artefactos*. Un *trabajador* es un puesto de trabajo al cual se le asigna una o varias personas que tienen los conocimientos y habilidades necesarias para desempeñarse eficazmente dentro del equipo de desarrollo del proyecto. Un *artefacto* representa cualquier tipo de información creada, producida, cambiada o utilizada por los trabajadores en el desarrollo del sistema.

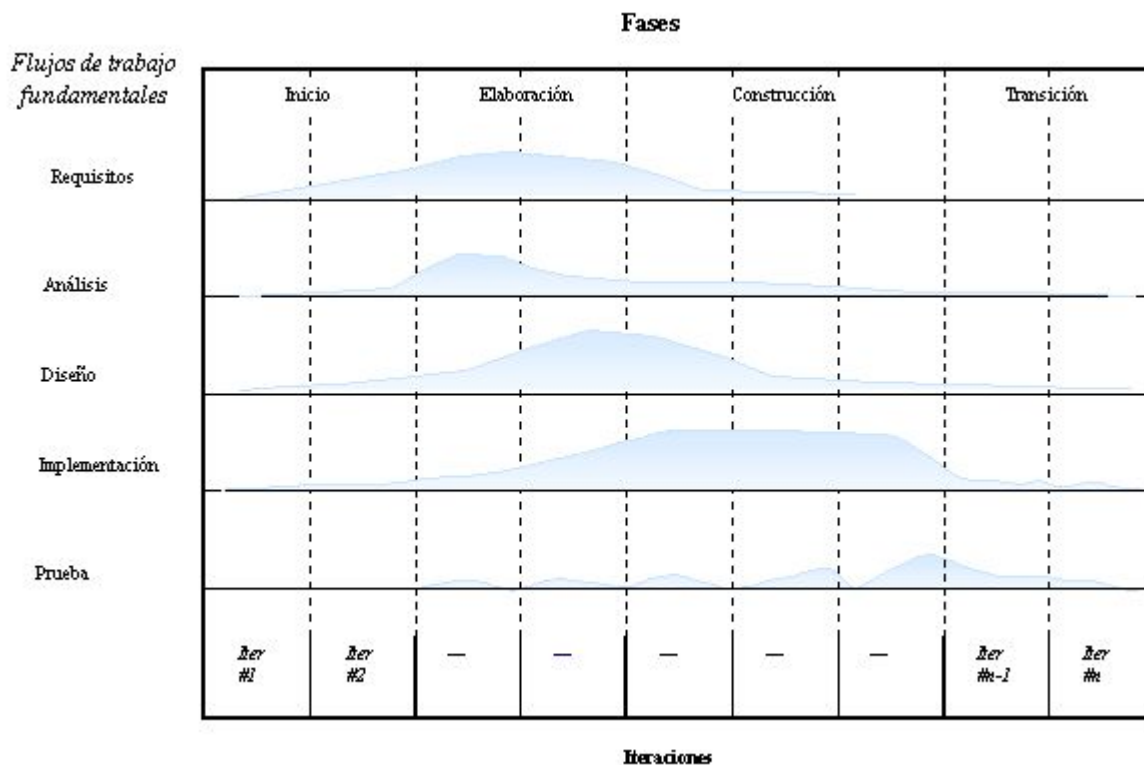


Figura A.12. Características del proceso unificado de desarrollo de software

La tabla A.12²¹ resume las principales actividades que se realizan en cada flujo de trabajo durante las diferentes fases, así como una descripción de los aspectos más importantes a tener en cuenta en cada actividad de acuerdo a la fase donde se desarrolle. Al final de la tabla, se anexan los productos resultantes de cada fase del ciclo de desarrollo del proyecto.

⊕ **MODELOS**

Los modelos son el tipo de artefacto más importante en el Proceso Unificado. Un modelo es una simplificación de la realidad, creada para comprender mejor el sistema que se está elaborando. En el Proceso Unificado, hay nueve modelos que en conjunto cubren todas las decisiones importantes implicadas en la visualización, especificación, construcción y documentación de un sistema con gran cantidad de software, estos son: modelo del negocio, del dominio, de casos de uso, de análisis, de diseño, de procesos, de despliegue, de implementación y de pruebas.

La tabla A.3 especifica los modelos que fueron utilizados en la elaboración del proyecto, igualmente se muestran algunas vistas de estos artefactos desarrollados durante la fase de inicio.

²¹ Estas tablas sobre el Proceso Unificado fueron tomadas de la tesis de grado de Mora(2002).

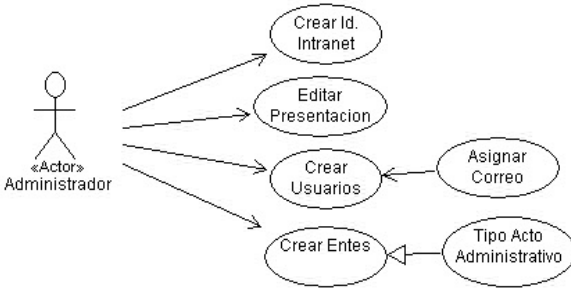
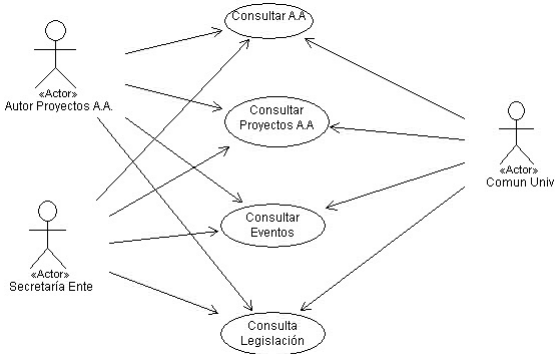
Flujo de Trabajo	ACTIVIDAD	Inicio	Elaboración	Construcción
Requisitos	Enumerar los requisitos candidatos	Las características provenientes de todas las fuentes de información se convierten en requisitos candidatos.	Se capturan la mayoría de los requisitos restantes para poder estimar el tamaño del proyecto.	Se concluye la recopilación de requisitos
	Comprender el contexto del sistema	Se usa el modelo de negocio para entender los procesos realizados dentro del ámbito del proyecto.		
	Capturar requisitos funcionales	Estos se representaran como casos de uso (CU)		
	Capturar requisitos no funcionales	Especifican propiedades del sistema, como dependencias de la plataforma, restricciones del entorno o de la implementación, etc.		
	Encontrar los actores y casos de uso	Solo los necesarios para describir el ámbito del sistema o la arquitectura candidata, ignorando las alternativas o caminos dentro de cada CU.	Su objetivo es comprender el 80% de los casos, buscando aquello que sea significativo desde el punto de vista de la arquitectura.	Conclusión de la descripción de los CU.
	Priorizar los casos de uso	Se realiza en la medida que se encuentran los CU. Permite planificar iteraciones futuras.	Se prioriza de acuerdo al nuevo conjunto de CU, según los riesgos percibidos y el orden en que se decida seguir el desarrollo.	A medida que identificamos CU, los añadimos a la clasificación con el objetivo de establecer su prioridad.
	Detallar los caso de uso	Solo aquellos necesarios para comprender el ámbito del sistema y los que sean de riesgo para la viabilidad del proyecto. Su detalle evitará pasar por alto los requisitos necesarios y críticos del sistema.	No se detallan en su totalidad CU seleccionados, sino que se limita el detalle a los CU y escenarios que necesitamos para esta fase.	Terminar el detalle de los CU faltantes.
	Prototipar la interfaz de usuario			Este prototipo se convierte en la especificación de la interfaz de usuario del sistema.

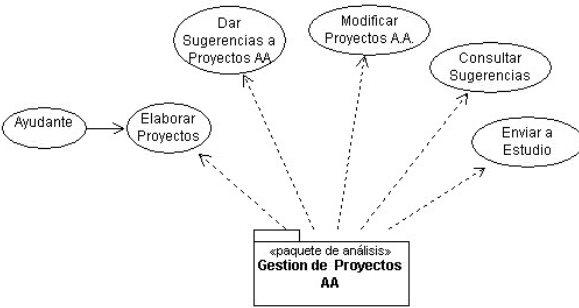
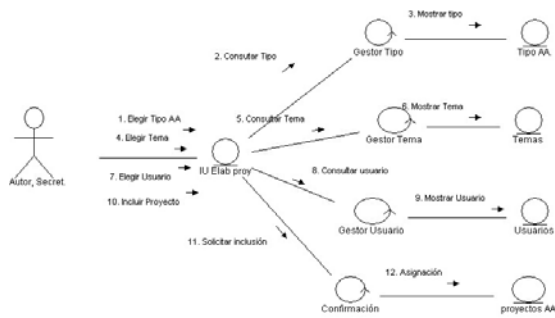
	Estructurar el modelo de casos de uso		En esta fase se buscan similitudes, simplificaciones y oportunidades para mejorar la estructura del modelo de CU.	Se pueden introducir mejoras sobre la estructura del modelo de CU, pero de aquellos que no se han desarrollados.
Análisis	Análisis de la arquitectura	Utilizando el conjunto inicial de CU detallado, se construye la primera versión del modelo de análisis para estas partes del sistema.	Se extiende este análisis que puede servir de base para la arquitectura. Se realiza una partición del sistema en paquetes de análisis empleando la arquitectura de capas.	
	Análisis de caso de uso	Se busca determinar los recursos compartidos entre los CU de relevancia en esta fase.	Solo se analiza los CU que aporten a la arquitectura del sistema y aquellos que sean dependientes entre si.	En cada iteración de esta fase se ampliara el modelo de análisis con los CU que sean incluidos en esta.
	Análisis de paquetes		Se busca garantizar que cada paquete de análisis cubran los CU analizados hasta esta fase. Así como que sean independientes entre si, tanto como sea posible.	Se refinan los paquetes que se identificaron en la anterior fase para acomodar los CU faltantes.
Diseño	Diseño de la arquitectura	Se desarrolla un esbozo inicial de la vista de la arquitectura del modelo de diseño, que realice los CU de interés como colaboraciones entre subsistemas. Se elige el software del sistema.	Se incluye en el modelo de diseño la arquitectura en capas, los subsistemas, interfaces y realizaciones de CU de aquellos arquitectónicamente significativos.	Se determina la conveniencia de realizar subsistemas completos a pesar de que alguna de sus partes pertenezcan a CU de baja prioridad
	Diseño de un caso de uso		Los CU importantes para esta fase se diseñan como subsistemas y se describen las interacciones entre estos.	Se diseñan los CU que surgen en esta fase como parte de los subsistemas existentes..
	Diseño de subsistemas		Se diseñan los subsistemas resultantes del diseño de la arquitectura. Si es necesario, se actualiza la vista de la arquitectura del modelo de diseño	Normalmente no se añaden nuevos subsistemas pero si se incluyen mejoras a estos.
Implementación	Implementación de la arquitectura		Se realiza la implementación de la arquitectura del sistema y se describe en la vista de arquitectura del modelo de implantación.	

	Integrar el sistema		Se integran las construcciones de los CU implementados en esta fase.	Se realizan e integran las capas inferiores de la arquitectura y luego las superiores.
	Implementar un subsistema		Se inicia la implementación de las partes cruciales para la arquitectura del sistema.	Se implementan todos los subsistemas.
	Realizar pruebas de unidad		Se realizan estas pruebas sobre los casos implementados.	Se realizan estas pruebas y se corrige el diseño y la implementación de cada componente de ser necesario.
Prueba	Planificar prueba		Se seleccionan los objetivos que evaluarán la línea base de la arquitectura.	Se establecen objetivos para cada construcción resultante de una iteración y luego para el sistema total
	Diseñar prueba		Con base en estos objetivos y en los CU implementados se diseña la prueba	Determinar como probar los requisitos del usuario en el programa.
	Realizar prueba de integración		se verifica las interfaces entre componentes implementados y que trabajen bien conjuntamente.	Se realizan pruebas de integridad añadiendo cada construcción y probando que todos los componentes trabajen correctamente de forma conjunta.

PRODUCTOS	<p>Una lista de características.</p> <p>Un esbozo de los modelos que representan la primera versión del modelo de casos de uso, el modelo de análisis y el modelo de diseño.</p> <p>Un primer esquema de la descripción de una arquitectura candidata.</p> <p>Un primer borrador de un plan para el proyecto en su totalidad.</p>	<p>Nueva versión de todos los modelos: casos de uso, análisis, diseño, despliegue e implementación.</p> <p>Una línea base de la arquitectura.</p> <p>Una descripción de la arquitectura, incluyendo vistas de los modelos de casos de uso, análisis, diseño, despliegue e implementación.</p> <p>El plan de proyecto para las fases de construcción.</p> <p>Versión inicial del manual de usuario.</p>	<p>El sistema software ejecutable.</p> <p>Todos los modelos del sistema</p> <p>La descripción de la arquitectura.</p> <p>Manuales de usuario, técnicos y administrativos</p>
-----------	---	--	--

Tabla A.5 Productos y Actividades de cada uno de los flujos de trabajo a través de las fases del Proceso Unificado.

MODELOS	DESCRIPCIÓN	REPRESENTACIÓN
Modelo del negocio	Establece una abstracción de la organización. Describe los procesos de negocio de una empresa en términos de casos de uso del negocio y actores del negocio.	<p>El modelo del negocio se representa en términos de casos de uso y actores</p> <p>Ejemplo:</p>  <pre> graph LR Admin[«Actor» Administrador] --> CId[Crear Id. Intranet] Admin --> EP[Editar Presentación] Admin --> CU[Crear Usuarios] Admin --> CE[Crear Entes] AC[Asignar Correo] --> CU TAA[Tipo Acto Administrativo] --> CE </pre> <p>Este modelo de negocio representa los procesos llevados a cabo en la configuración del Sistema de Información Intranet. Esto es, se incluyen las características propias de la Institución, de sus entes y la documentación.</p>
Modelo de casos de uso	Establece los requisitos funcionales del sistema.	<p>Contiene actores, casos de uso y sus relaciones</p> <p>Ejemplo:</p>  <pre> graph LR APAA[«Actor» Autor Proyectos A.A.] --> CA[Consultar A.A.] APAA --> CPA[Consultar Proyectos A.A.] APAA --> CE[Consultar Eventos] APAA --> CL[Consulta Legislación] SE[«Actor» Secretaría Ente] --> CA SE --> CPA SE --> CE SE --> CL ACU[«Actors» Comun Univ] --> CA ACU --> CPA ACU --> CE ACU --> CL </pre> <p>Este diagrama de casos de uso describe los casos de uso utilizados en la gestión de consulta.</p>

<p>Modelo de análisis</p>	<p>Establece un diseño de las ideas.</p> <p>El modelo del análisis es una jerarquía de paquetes del análisis que contienen clases del análisis²² y realizaciones de casos de uso.</p>	<p>Contiene clases del análisis, realización de casos de uso-análisis y paquetes del análisis.</p> <p>Ejemplo:</p>  <p>paquetes del análisis</p>  <p>clases de análisis y realización de un caso de uso</p> <p>Existe un paquete de análisis llamado Gestión de proyectos, que contiene los casos de uso relacionados con las funcionalidades que ofrece esta zona de la herramienta.</p>
---------------------------	--	--

²² Las clases del análisis se dividen en: clases de interfaz, que se utilizan para modelar la interacción entre el sistema y sus actores; clases de entidad: para modelar información persistente y clases de control, que se usan para encapsular el control de un caso de uso en concreto.

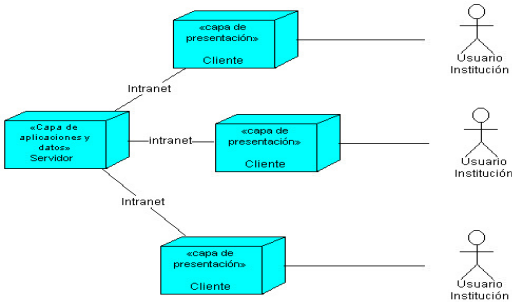
Modelo de diseño	<p>Establece el vocabulario del problema y su solución.</p> <p>El modelo de diseño es un modelo de objetos que describe la realización física de los casos de uso centrándose en cómo los requisitos, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar.</p> <p>Es una abstracción de la implementación del sistema.</p>	Modelo no realizado para el sistema
Modelo de despliegue	Establece la topología hardware sobre la cual se ejecutará el sistema.	<p>El modelo de despliegue contiene nodos.</p>  <p>Esta gráfica muestra los nodos físicos en los que se ejecutará la aplicación SIGNO</p>
Modelo de implementación	Establece las partes que se utilizarán para ensamblar y hacer disponible el sistema físico.	
Modelo de pruebas	Establece las formas de validar y verificar el sistema.	Contiene casos de prueba, procedimientos de prueba y/o componentes de prueba.

Tabla A.6 Modelos en el proceso unificado

A.5. VISIÓN GENERAL DEL UML

El siguiente anexo da una breve noción general del Lenguaje Unificado de Modelado. La primera parte está dedicada a exponer las principales características de UML, luego, se presenta de manera organizada, información acerca de su sintaxis. Se utilizan ejemplos de este proyecto como apoyo en la comprensión de la estructura básica del lenguaje. Para mayor información acerca del tema, se aconseja consultar las siguientes fuentes bibliográficas: Jacobson(1999) y Fowler(1999) .

❖ ¿Qué es el UML?

El lenguaje Unificado de Modelado es el sucesor de la oleada de métodos de análisis y diseño orientados a objetos que surgió a finales de la década de los ochenta y principios de la siguiente. El UML unifica, sobre todo los métodos de Booch, Rumbaugh(OMT) y Jacobson²³ . Esta unificación fue promovida por el OMG de tal manera que UML se convirtió en la notación estándar para la descripción de métodos software.

Según su definición, UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software (ver figura A.1), desde una perspectiva orientada a objetos:

Es un lenguaje, porque ofrece una serie de reglas y vocabulario para la representación conceptual y operativa de un sistema.

²³ Para mayor información, consultar Fowler(1999)

Es un lenguaje para visualizar, porque permite a los desarrolladores representar gráficamente, el proceso de modelado aplicado a sus proyectos software. UML combina unos símbolos gráficos con una semántica bien definida, lo que permite elaborar modelos totalmente comprensibles.

Es un lenguaje para especificar, Cubre la especificación de todas las decisiones de análisis, diseño e implementación que deben realizarse al desarrollar y desplegar un sistema con gran cantidad de software.

Es un lenguaje para construir, aunque UML no es un lenguaje de programación, tiene una relación directa con las plataformas de desarrollo como Java, C++, Visual Basic, inclusive, con bases de datos relacionales y/o orientada a objetos. Es posible la generación de código a partir de un modelo en UML.

Es un lenguaje para documentar, porque documenta toda clase de artefactos que produce una organización software además de código ejecutable. Estos artefactos incluyen: requisitos, arquitectura, diseño, código fuente, planificación de proyectos, pruebas, prototipos y versiones.

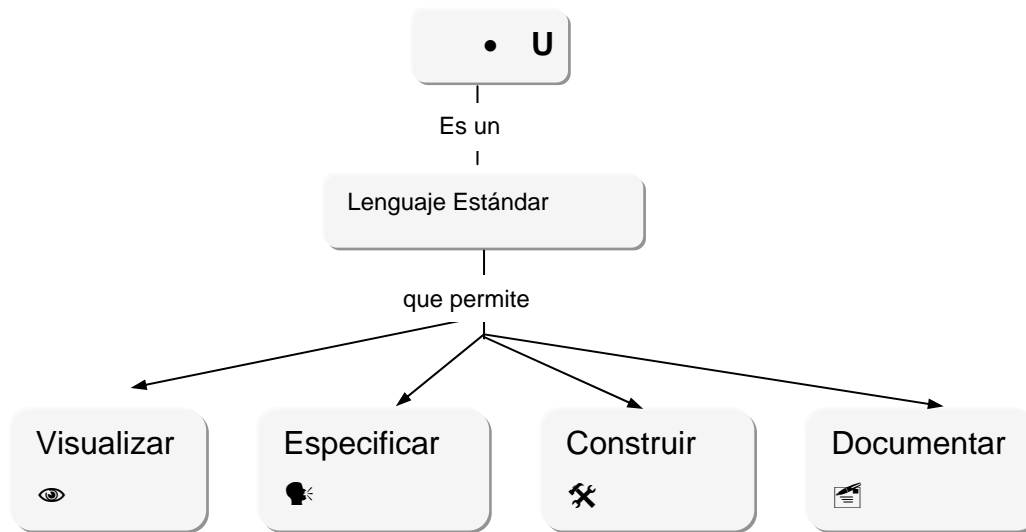
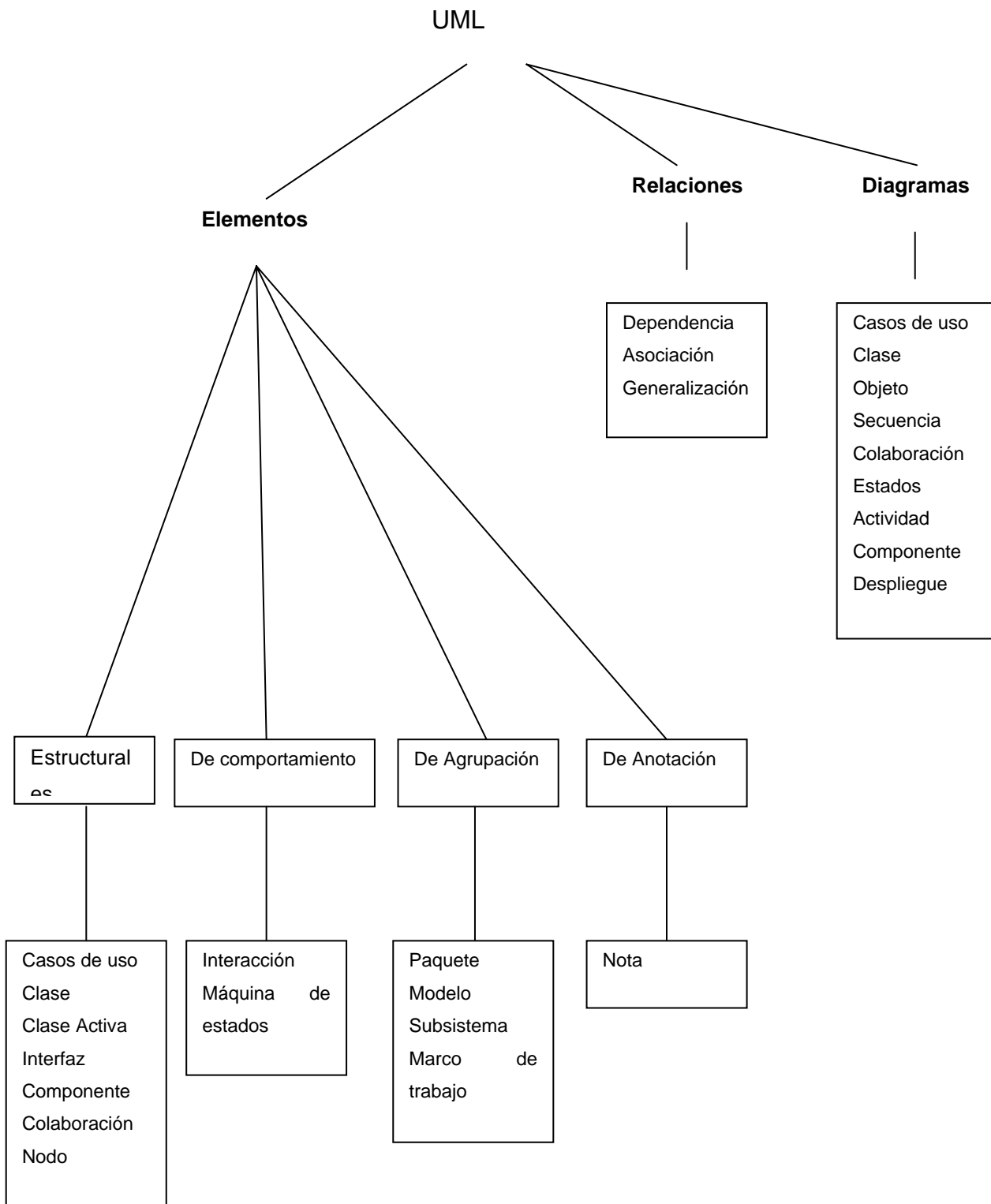


Figura A.13 Definición de UML

⊕ Estructura básica del lenguaje

UML proporciona a los desarrolladores un vocabulario, que incluye tres categorías:

- Elementos
- Relaciones
- Diagramas



La figura A.14 especifica el vocabulario de UML:

⊕ **Elementos en UML:** son los bloques básicos de construcción orientados a objetos de UML. Se utilizan para escribir modelos bien formados. Como se observa en la figura A.6, estos se clasifican en:



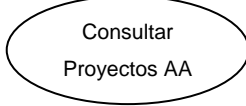

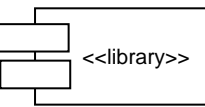

✕ **Elementos estructurales:** Son los nombres de los modelos UML. En su mayoría son las partes estáticas de un modelo, y representan cosas que son conceptuales o materiales. En total, hay siete tipos de elementos estructurales, como se resume en la tabla A.7.

✕ **Elementos de comportamiento:** Son las partes dinámicas de los modelos UML. Representan comportamiento en el tiempo y el espacio. Semánticamente éstos están conectados normalmente a diversos elementos estructurales como clases, objetos y colaboraciones.

Hay dos tipos principales de elementos de comportamiento, como se expone en la tabla A.8.

✕ **Elementos de agrupación:** son las partes organizativas de los modelos UML. (Ver tabla A.9).

✕ **Elementos de anotación:** Son comentarios que se pueden aplicar para describir, clarificar y hacer observaciones sobre cualquier elemento de un modelo. (Ver la tabla A.9) .

ELEMENTOS ESTRUCTURALES						
Elemento	Descripción	Representación				
CLASE	Es una descripción de un conjunto de objetos que comparten los mismos atributos, operaciones, relaciones y semántica. Una clase implementa una o más interfaces.	<table border="1"> <tr><td>Nombre</td></tr> <tr><td>Atributos</td></tr> <tr><td>Operaciones</td></tr> <tr><td>Comportamiento Extra</td></tr> </table>	Nombre	Atributos	Operaciones	Comportamiento Extra
Nombre						
Atributos						
Operaciones						
Comportamiento Extra						
INTERFAZ	<i>Colección de operaciones que especifican un servicio de una clase o componente.</i>					
COLABORACIÓN	Define una interacción y es una sociedad de roles y otros elementos que colaboran para proporcionar un comportamiento cooperativo mayor que la suma de los comportamientos de los elementos.					
CASO DE USO	<i>Descripción de un conjunto de secuencias de acciones que un sistema ejecuta y que produce un resultado observable de interés para un actor particular. Se utiliza para estructurar los aspectos de comportamiento de un modelo.</i>					
CLASE ACTIVA	Es una clase cuyos objetos tienen uno o más procesos o hilos de ejecución y por lo tanto pueden dar origen a actividades de control. Una clase activa es igual que una clase, excepto en que sus objetos representan elementos cuyo comportamiento es concurrente con otros elementos.					
COMPONENTE	<i>Es una parte física y reemplazable de un sistema que representa el empaquetamiento físico de diferentes elementos lógicos, como clases, interfaces y colaboraciones.</i>					
NODO	<i>Es un elemento físico que existe en tiempo de ejecución y representa un recurso computacional, que por lo general dispone de algo de memoria y, con frecuencia, capacidad de procesamiento.</i>					

MAQUINA ESTADOS	DE	Comportamiento que especifica la secuencia de estados por las que pasa un objeto o interacción durante su vida	Validan
--------------------	----	--	---------

Tabla A.7. Elementos estructurales


ELEMENTOS DE COMPORTAMIENTO		
ELEMENTO	DESCRIPCION	REPRESENTACIÓN
INTERACCION	Comportamiento que comprende un conjunto de mensajes intercambiables entre un conjunto de objetos, dentro un ambiente en particular.	Enviar 

Tabla A.8 Elementos de comportamiento

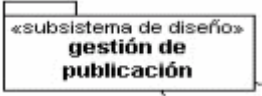
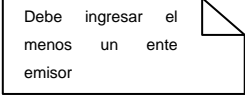
ELEMENTOS DE AGRUPACIÓN Y ANOTACIÓN		
ELEMENTO	DESCRIPCION	REPRESENTACIÓN
PAQUETE (AGRUPACIÓN)	Los paquetes se utilizan para organizar los elementos de modelado en partes mayores que se pueden manipular como un grupo. Pueden emplearse para representar diferentes vistas de la arquitectura del sistema.	
NOTA (ANOTACIÓN)	Las notas se utilizan para especificar cosas como requisitos, observaciones, revisiones y explicaciones, además de representar restricciones.	

Tabla A.9. Elementos de agrupación y anotación

⊕ **Relaciones en UML:** Son los bloques básicos de construcción para relaciones de UML. Se utilizan para escribir modelos bien formados. Como se observa en la figura A.10, estos se clasifican en:

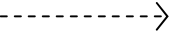
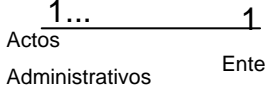
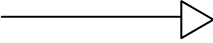
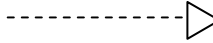
RELACIONES		
RELACIÓN	DESCRIPCIÓN	REPRESENTACIÓN
DEPENDENCIA	Relación semántica entre dos elementos, en la cual un cambio a un elemento (el elemento independiente) puede afectar a la semántica del otro (el elemento dependiente).	
ASOCIACIÓN	Relación estructural que describe un conjunto de enlaces, los cuales son conexiones entre objetos. A menudo incluye la multiplicidad y los nombres de rol.	
GENERALIZACIÓN	Relación de especialización / generalización en la cual los elementos del elemento especializado (el hijo) pueden sustituir a los objetos del elemento general (el padre).	
REALIZACIÓN	Relación semántica entre clasificadores, en donde un clasificador (clases, interfaces, señales, nodos, casos de uso y subsistemas) especifica un contrato que otro garantiza que cumplirá.	

Tabla A.10. Relaciones en UML.

✚ **Diagramas en UML:** Los diagramas se dibujan para visualizar un sistema desde diferentes perspectivas, puede contener en teoría cualquier combinación de elementos y relaciones. Como se observa en la figura A.11, estos se clasifican en:

DIAGRAMAS		
DIAGRAMA	DESCRIPCIÓN	EJEMPLO
De casos de uso	Representa las funciones del sistema desde el punto de vista del usuario. Se utilizan para la comunicación con los usuarios y para expresar de forma clara y sencilla los requisitos.	Este diagrama especifica las funcionalidades que ofrece al autor de proyectos la parte de gestión de proyectos. Entre estos servicios, se encuentra la Publicación, Consulta, Modificación de proyectos de actos administrativos.
De clase	Bosquejan la arquitectura estática del sistema en función de clases y sus asociaciones.	No utilizado
De Dominio	Captura los tipos más importantes de objetos en el contexto del sistema. Los objetos del dominio representan las "cosas" que existen o los eventos que suceden en el entorno en el que trabaja el sistema	Este diagrama es presentado en la sección 7.3.

<p>De Secuencia</p>	<p>Representan un conjunto de elementos de un sistema que interaccionan entre ellos organizados en secuencias de tiempo.</p>	<p>Este diagrama destaca la ordenación temporal del paso de mensajes entre los objetos que interactúan al realizar el caso de uso Ejecutar jugada.</p>
<p>De Colaboración</p>	<p>Son parecidos a los diagramas de secuencia pero dan mayor libertad para distribuir los objetos.</p> <p>Representan la interacción entre elementos de un sistema y se organiza con respecto a espacio y tiempo. Es una representación espacial de los objetos, sus enlaces y sus interacciones.</p> <p>Junto con los de secuencias se denominan, diagramas de interacción.</p>	<p>La anterior figura corresponde al diagrama de colaboración del caso de uso: Crear Identificación.</p> <p>El administrador, haciendo uso de la interfaz de usuario, proporciona la configuración para la generación de la intranet. Posteriormente, Confirmación valida los datos y procede a almacenar los datos en Intranet. Por último se muestra la confirmación de la realización correcta de la transacción por medio de la Interfaz de Resultados.</p>
<p>De estado</p>	<p>Representan el estado de un elemento del sistema, las condiciones y respuestas que da a ciertos eventos a los que es expuesto.</p>	<p>No utilizado</p>

De despliegue	Muestra un conjunto de nodos y sus relaciones. Representa el despliegue de los componentes y subsistemas sobre los dispositivos físicos.	No utilizado
De objetos	Muestra un conjunto de objetos y sus relaciones. Representan instantáneas de instancias de los elementos encontrados en los diagramas de clase.	No utilizado
De actividades	Tipo especial de diagrama de estados que muestra un flujo de actividades dentro de un sistema. Cubren la vista dinámica y son importantes al modelar el funcionamiento de un sistema y resaltan el flujo de control entre objetos.	No utilizado
De componentes	Muestra la organización y las dependencias entre un conjunto de componentes. Cubren la vista de implementación estática de un sistema.	No utilizado

Tabla A.11 Diagramas de UML

ANEXO B

PRUEBAS DEL SISTEMA

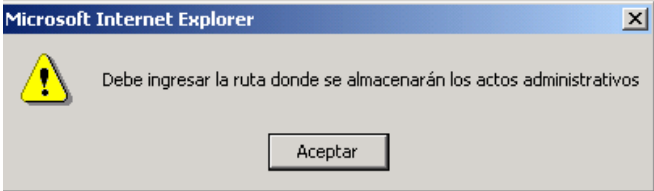
Este anexo presenta las principales pruebas realizadas para comprobar el correcto funcionamiento del sistema. Las pruebas se encuentran organizadas por subsistemas y según el orden en que fueron presentadas en los capítulos 8 y 9 de la fase de construcción.



Para el desarrollo de estas pruebas la base de datos ha sido poblada con datos ficticios. Para cada una de estos casos también se realizaron pruebas de validación de todos los datos solicitados al usuario.

SUBSISTEMA GESTION DE CONFIGURACIÓN

Casos de prueba

- Caso 1: Validar campos no vacíos
- Caso 2: Validar tipo de archivos permitidos
- Caso 3: Validar correo
- Caso 4: Verificar que no se creen duplicados
- Caso 5: Correcto almacenamiento en la base de datos

Página: Información General		
<p>Para esta página se realizarán los casos 1, 2, 4 y 5 por su pertinencia con la misma; el caso 3 no se realiza pues aquí no se maneja correo.</p> <p style="text-align: center;">Caso 1: Validar campos no vacíos</p> <p>Este caso de prueba se realiza en todas las páginas donde el usuario ingresa información, pero solo se muestra para esta página ya que el proceso es el mismo y no es relevante repetirlo.</p>		
Datos de entrada	Nombre de la Institución:	Universidad Privada Alfonso X:
	Nombre de la Intranet:	Intranet.unialfonsox.edu.co
	Actos Administrativos:	
	Proyectos de Actos Administrativos	
	Normas de Educación Superior	
	Eventos:	
	Información de la Institución:	
	Misión:	
	Visión:	
	Objetivos:	
	Organigrama:	
Calendario Académico		
Resultados	<p>En esta página, los últimos 5 campos son opcionales ya que no son de vital importancia para el sistema, pero los campos restantes son obligatorios y en caso de no ingresar la información se presenta un mensaje de error como el siguiente que ocurre cuando se deja el campo correspondiente a la ruta donde se almacenan los Actos Administrativos y se muestra la misma página.</p> <div style="text-align: center;">  </div>	
Caso 2 : Validar tipo de archivos permitidos		
Datos de Entrada	Nombre de la Institución:	Universidad Privada Alfonso X
	Nombre de la Intranet:	Intranet.unialfonsox.edu.co
	Actos Administrativos:	/var/www/html/signo/actos
	Proyectos de Actos Administrativos	/var/www/html/signo/proyectos
	Normas de Educación Superior	/var/www/html/signo/nes
	Eventos:	/var/www/html/signo/eventos
	Información de la Institución:	/var/www/html/signo/info
	Misión:	C:\Mis documentos\bibliografia\document.zip
	Visión:	
	Objetivos:	
Organigrama:		
Calendario Académico		
Resultados	En este caso para Ingresar la misión, visión , objetivos, organigrama y Calendario	

	académico solo se aceptan documentos html o .doc, en caso contrario presenta el siguiente mensaje de error y se continúa en la misma página.	
		
Caso 4 : Verificar que no se creen duplicados		
Datos de Entrada	Nombre de la Institución:	Universidad Privada Alfonso X
	Nombre de la Intranet:	Intranet.unialfonsox.edu.co
	Actos Administrativos:	/var/www/html/signo/actos
	Proyectos de Actos Administrativos	/var/www/html/signo/proyectos
	Normas de Educación Superior	/var/www/html/signo/nes
	Eventos:	/var/www/html/signo/eventos
	Información de la Institución:	/var/www/html/signo/info
	Misión:	C:\Mis documentos\bibliografia\document.zip
	Visión:	
	Objetivos:	
Organigrama:		
Calendario Académico		
Resultados	<p>Como esta información se incluye solo una vez y para evitar que se creen duplicados, antes de insertar el registro se hace una consulta para verificar que no existan registros, de existir se presenta el siguiente mensaje de error y continúa en la misma página</p> <div style="text-align: center;">  </div>	
Caso 5 : Correcto almacenamiento en la base de datos		
Datos de Entrada	Nombre de la Institución:	Universidad Privada Alfonso X
	Nombre de la Intranet:	Intranet.unialfonsox.edu.co
	Actos Administrativos:	/var/www/html/signo/actos
	Proyectos de Actos Administrativos	/var/www/html/signo/proyectos
	Normas de Educación Superior	/var/www/html/signo/nes
	Eventos:	/var/www/html/signo/eventos
	Información de la Institución:	/var/www/html/signo/info
	Misión:	
	Visión:	
	Objetivos:	
Organigrama:		
Calendario Académico		
Resultados		


Host	Database	Table	Data	Query							
SIIGNO / infoinstitucion: 1 Records (1 retrieved)											
nominsti	rutaaa	rutapaa	rutanes	rutaev	mission	vision	objetivos	organigrama	calendario	nomintranet	
Privada Alfonso	/var/www/html/sign				/var/www/html/sign				/var/www/html/sign		unialfonsox.edu.co

En caso de no ocurrir errores se hace la inserción y permite el paso a la pagina "Crear Entes"


Pagina: Crear Entes

Para esta página solo se realizan los casos 3 y 4, pues los demás casos o ya se realizaron en la fase de construcción o no son pertinentes en este caso, razones que se mantendrán a lo largo de las pruebas realizadas en el presente anexo.

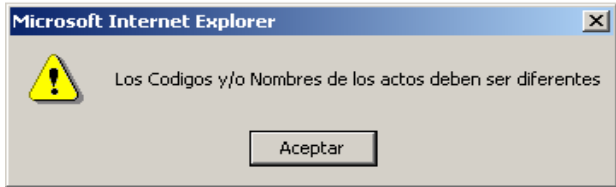

Caso 3 : Validar Correo

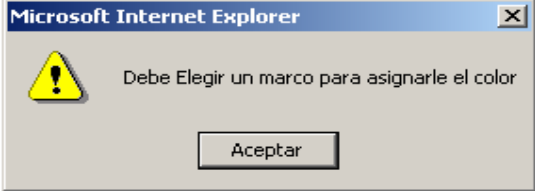
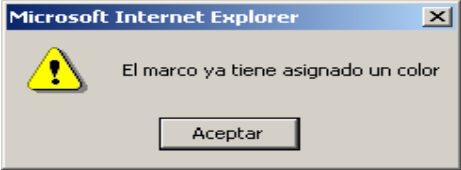

Datos de Entrada	Código	1911
	Nombre	Consejo Directivo
	Jefe	Manuel Pérez
	Correo	condirunialfonsox.edu.co
Resultados	<p>En esta prueba se pretende ingresar un ente con cuyo correo no contiene símbolo @, carácter indispensable para que sea valido, el resultado es el siguiente mensaje de error:</p>  <p>Igualmente se produce el mensaje de error cuando no se digita el dominio.</p>	



Caso 4 : Verificar que no se creen duplicados

Datos de Entrada	Código	1
	Nombre	Consejo Directivo
	Jefe	Manuel Pérez
	Correo	mperez@unialfonsox.edu.co
Resultados	<p>En esta prueba se pretende ingresar un ente con un código que ya existe y los demás campos nuevos, el resultado es el siguiente mensaje de error:</p> 	

	Igualmente se producen mensajes de error cuando se ingresa un nombre que ya existe en la base de datos, sin importar que el código ingresado sea diferente, ya que no pueden existir dos entes con el mismo nombre y diferente código. Para los campos correo y jefe del ente no se hacen estas verificaciones, pues puede darse el caso en que se repita cualquiera de los dos campos.
En caso de no ocurrir errores la navegación continua con la página "Crear tipo de Actos Administrativos"	

Página : Crear Tipo de Actos Administrativos		
Por la misma razones expuestas anteriormente para esta página sólo se realizan los casos 4 y 5		
Caso 4 : Verificar que no se creen duplicados		
Datos de Entrada	Tipo de Acto	Acta
	Código	1
	Nombre	Memoria
	Tipo de Acto	Resolución
	Código	2
	Nombre	Memoria
Resultados	<p>En este caso se tratan de incluir 2 tipos de actos administrativos con formato diferente, código diferente pero mismo nombre y eso sería crear un duplicado, por ende se debe validar y mostrar el siguiente mensaje de error</p> <div style="text-align: center;">  </div>	
Caso 5 : Correcto almacenamiento en la base de datos		
Datos de Entrada	Tipo de Acto	Acta
	Código	1
	Nombre	Memoria
	Tipo de Acto	Resolución
	Código	2
	Nombre	Resolución
Resultados	<div style="text-align: center;">  </div>	
Al validar la información e incluir los registros en la base de datos puede continuar a la página "Editar el color de los Marcos"		

Página : Editar el Color de los Marcos		
<p>Para esta página se realizan un caso particular de validación en el cual se verifica que antes de realizar la inserción, la casilla desplegable tenga seleccionado un ítem diferente al del título inicial, adicionalmente se realizan los casos 4 y 5</p>		
Caso * : Verificar que la casilla desplegable tenga seleccionado un ítem válido		
Datos de Entrada	Color Marco	#ffffff
	Nombre	Elegir Marco
Resultados	<p>De la casilla de selección "color marco" debe haber seleccionado un marco (superior, derecho o izquierdo), en caso contrario se despliega el siguiente mensaje de error:</p> <div style="text-align: center;">  </div>	
Caso 4 : Verificar que no se creen duplicados		
Datos de Entrada	Color Marco	#ffffcc
	Nombre	Marco Derecho
Resultados	<p>En este caso el marco derecho ya tiene asignado el color #ffffcc y por tanto se despliega el mensaje de error:</p> <div style="text-align: center;">  </div>	
Caso 5 : Correcto almacenamiento en la base de datos		
Datos de Entrada	Color Marco	#0099cc
	Nombre	Marco Izquierdo
Resultados	<div style="text-align: center;">  </div>	
<p>Al validar la información e incluir los registros en la base de datos puede continuar a la página "Editar Fuente"</p>		

Página : Editar Fuente		
<p>Para esta página se realizan un caso particular de validación en el cual se verifica que antes de realizar la inserción, las casillas desplegadas tengan seleccionado un ítem diferente al del título inicial, adicionalmente se realizan los casos 4 y 5</p>		
Caso * : Verificar que las casillas desplegadas tengan seleccionado un ítem válido		
Datos de Entrada	Color	#ffffff
	Estilo	Estilo
	Fuente	Arial
	Tamaño	1
	Alineación	center
Resultados	<p>Esta verificación se hace para todas las casillas de selección, pero aquí solo se muestra la prueba para el caso en el que la casilla Estilo se le deja el título original: "Estilo", en estos casos se presenta un mensaje de error dependiendo de la casilla que se haya dejado un ítem inválido.</p> <div style="text-align: center;">  </div>	
Caso 4 : Verificar que no se creen duplicados		
Datos de Entrada	Color	#ffffff
	Estilo	Normal
	Fuente	Arial
	Tamaño	1
	Alineación	center
Resultados	<p>En este caso el Estilo de letra "Normal" ya tiene asignadas las características correspondientes, y por tanto se despliega el mensaje de error:</p> <div style="text-align: center;">  </div>	
Caso 5 : Correcto almacenamiento en la base de datos		
Datos de Entrada	Color	#000000
	Estilo	Normal
	Fuente	Verdana
	Tamaño	1
	Alineación	justify
Resultados		


Host	Database	Table	Data	SQL Query	
SIIGNO / tipoletras: 6 Records (6 retrieved)					
tipoletra	estilo	clase	tamaño	color	alineación
2	Titulo 2	Verdana	3	#000000	Center
1	Titulo 1	Verdana	4	#663300	Center
3	Titulo 3	Verdana	2	#000000	center
4	Titulo 4	Verdana	1	#000000	center
5	Normal	Verdana	1	#000000	Justify
6	Hipervinculo	Verdana	1	#000000	Justify

Al validar la información e incluir los registros en la base de datos puede continuar a la página "Emblemas"

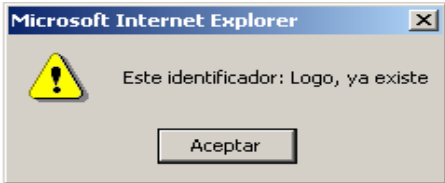
Página : Emblemas

Para esta página se realizan un caso particular de validación en el cual se verifica que antes de realizar la inserción, las casillas desplegadas tengan seleccionado un ítem diferente al del título inicial, adicionalmente se realizan los casos 4 y 5

Caso * : Verificar que las casillas desplegadas tengan seleccionado un ítem válido

Datos de Entrada	identificador	Elegir Identificador
Resultados	Esta validación es para el caso en el que se desee continuar sin haber ingresado los emblemas necesarios.	
		

Caso 4 : Verificar que no se creen duplicados

Datos de Entrada	Identificador	Logo
	Archivo Adjunto	C:\Mis documentos\logo.gif
Resultados	En este caso el logo ya se ha ingresado en la base de datos y por tanto se despliega el mensaje de error:	
		

Caso 5 : Correcto almacenamiento en la base de datos

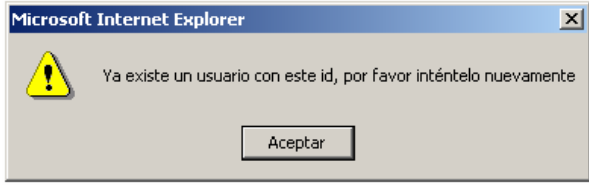

Datos de Entrada	Identificador	Logo
	Archivo Adjunto	C:\Mis documentos\logo.gif
Resultados		

Host Database Table Data Query						
SIIGNO / emblemas: 5 Records (5 retrieved)						
identificador	codparte	nommarco	tipoletra	contenido	colorfondo	
escudo		1 Marco Superior		(BLOB)		
ninstitucion		2 Marco Superior		(BLOB)		
▶ menu		4 Marco Superior	Titulo 1	(BLOB)	#6699cc	
imginicioesion		0 Marco Derecho		(BLOB)		
logo		3 Marco Superior		(BLOB)		

Al validar la información e incluir los registros en la base de datos puede continuar a la página "Resultados Instalador"

Página : Resultados Instalador

Esta página muestra la información que se ingresó en la base de datos y permite acceder a páginas que permiten modificar dicha información o continuar a la página "Crear Usuario". De esta página no se presentan pruebas porque el usuario no ingresa información y lo único que se permite es continuar la navegación.

Página : Crear Usuario		
Para esta página se realizan los casos 4 y 5		
Caso 4 : Verificar que no se creen duplicados		
Datos de Entrada	ID	ersanchez
	Los datos adicionales que permiten identificar al usuario no validan el que se dupliquen registros y por tanto no se muestran aquí, pero para la prueba en el sistema se llenan todos los demás campos	
Resultados	Este ID corresponde a un usuario en la base de datos, por tanto se despliega el siguiente error:	
		
Caso 5 : Correcto almacenamiento en la base de datos		
Datos de Entrada	ID	ersanchez
	Clave	*****
	Digitar nuevamente la clave	*****
	Recordatorio	Profesor
	Tipo	Autor de Proyectos
	Rol dentro de la Institución	Profesor
	Ente al que pertenece	Escuela de Ingeniería Electrónica
	Nombres	Erwin
	Apellidos	Sánchez Florez
	E-mail	ersanchez@unialfonsox.edu.co
	Dirección	Of. 301 Edificio Amarillo
	Teléfono	6481000
Resultados		
Al validar la información e incluir los registros en la base de datos puede ingresar a la intranet		

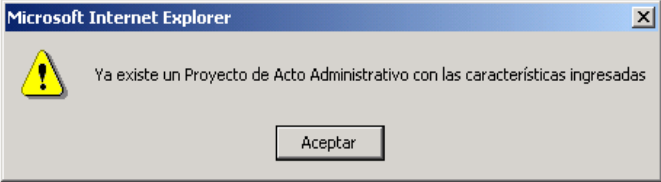
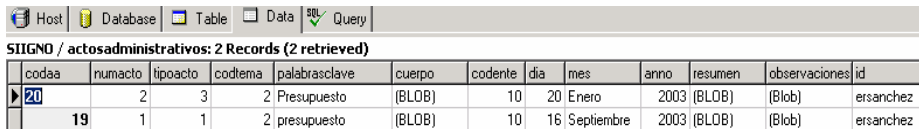
SUBSISTEMA GESTION DE PUBLICACION

Para este solo se presentan los casos de prueba aplicados en dos páginas, debido a que son procesos iguales y no es relevante extender en repeticiones.

Casos de prueba

- Caso 1: Verificar los permisos del usuario
- Caso 2: Verificar que no se creen duplicados
- Caso 3: Correcto almacenamiento en la base de datos

Página: Publicar Acto Administrativo		
Para esta página se realizarán los casos 1, 2, y 3. El caso 1 es común para todas las páginas luego solo se realizara en esta página.		
Caso 1 : Verificar los permisos del usuario		
Datos de Entrada	Usuario:	Magoes
	Contraseña:	*****
Resultados	Se consultó en la base de datos y se halló que el tipo de usuario es 2 : Secretaria General y tiene para todo excepto manejar usuarios.	
Caso 3 : Verificar que no se creen duplicados		
Datos de Entrada	Numero :	20
	Ente:	Escuela de Ingeniería Electrónica
Resultados		
Caso 4 : Correcto almacenamiento en la base de datos		
Datos de Entrada	Tipo :	Acta
	Tema:	Reglamentaciones
	Palabras clave :	Manual
	Fecha :	01 /01/2003
	Resumen:	Por el tamaño de estos campos no se presenta esta información
Cuerpo :		
Resultados		

Página: Publicar Proyecto																																								
<p>Para esta página se realizarán los casos 1, 3 y 4 ya que el caso 2 se realiza cuando se permite adjuntar archivos y este no es el caso. El caso 1 es común para todas las páginas del Sistema de Información Intranet, luego solo se presentará esta prueba en esta página.</p>																																								
Caso 3 : Verificar que no se creen duplicados																																								
Datos de Entrada	Tipo :	Acta																																						
	Tema:	Reglamentaciones																																						
	Palabras clave :	Manual																																						
	Fecha :	01 /01/2003																																						
	Resumen:	Por el tamaño de estos campos no se presenta esta información																																						
Cuerpo :																																								
Resultados																																								
Caso 4 : Correcto almacenamiento en la base de datos																																								
Datos de Entrada	Tipo :	Acta																																						
	Tema:	Reglamentaciones																																						
	Palabras clave :	Manual																																						
	Fecha :	01 /01/2003																																						
	Resumen:	Por el tamaño de estos campos no se presenta esta información																																						
Cuerpo :																																								
Resultados	 <table border="1"> <thead> <tr> <th>codaa</th> <th>numacto</th> <th>tipoacto</th> <th>codtema</th> <th>palabrasclave</th> <th>cuerpo</th> <th>codente</th> <th>día</th> <th>mes</th> <th>año</th> <th>resumen</th> <th>observaciones</th> <th>id</th> </tr> </thead> <tbody> <tr> <td>20</td> <td>2</td> <td>3</td> <td>2</td> <td>Presupuesto</td> <td>(BLOB)</td> <td>10</td> <td>20</td> <td>Enero</td> <td>2003</td> <td>(BLOB)</td> <td>(Blob)</td> <td>ersanchez</td> </tr> <tr> <td>19</td> <td>1</td> <td>1</td> <td>2</td> <td>presupuesto</td> <td>(BLOB)</td> <td>10</td> <td>16</td> <td>Septiembre</td> <td>2003</td> <td>(BLOB)</td> <td>(Blob)</td> <td>ersanchez</td> </tr> </tbody> </table>	codaa	numacto	tipoacto	codtema	palabrasclave	cuerpo	codente	día	mes	año	resumen	observaciones	id	20	2	3	2	Presupuesto	(BLOB)	10	20	Enero	2003	(BLOB)	(Blob)	ersanchez	19	1	1	2	presupuesto	(BLOB)	10	16	Septiembre	2003	(BLOB)	(Blob)	ersanchez
codaa	numacto	tipoacto	codtema	palabrasclave	cuerpo	codente	día	mes	año	resumen	observaciones	id																												
20	2	3	2	Presupuesto	(BLOB)	10	20	Enero	2003	(BLOB)	(Blob)	ersanchez																												
19	1	1	2	presupuesto	(BLOB)	10	16	Septiembre	2003	(BLOB)	(Blob)	ersanchez																												

SUBSISTEMA GESTION DE PROYECTOS

Las paginas que están agrupadas dentro de este subsistema pertenecen a uno u otro subsistema por eso para el subsistema gestión de proyectos no fue necesario presentar las pruebas. Es de aclarar que el hecho de que no se muestren en el presente documento no significa en ningún momento que no se realizaran, sino que no se cree necesario extender innecesariamente el documento.

SUBSISTEMA GESTION DE CONSULTA

Casos de prueba

- Caso 1: Congruencia entre los criterios de búsqueda y los resultados de la misma.

Pagina: Consultar Acto Administrativo		
Caso 1 : Congruencia entre los criterios de búsqueda y los resultados de la misma		
Datos de Entrada	Ente :	Escuela de Ingeniería Electrónica
	Conector:	y
	Año :	2003
Resultados	 <p>The screenshot shows a web browser window titled 'Consultar Proyecto de Acto Administrativo - Microsoft Internet Explorer'. The address bar shows 'http://pablo/signo2/proyectorizados.php'. The page content includes the IES logo and a search results section titled 'Proyectos Encontrados : 1'. The results show a resolution from the Escuela de Ingeniería Electrónica, dated September 22, 2003, regarding a partial transfer and distribution of expenses for the 2003 fiscal year. The author is listed as Erwin.</p>	

SUBSISTEMA GESTION DE SESION

Los casos de prueba para este subsistema se han presentado en el capítulo 9 y en algunas páginas del presente anexo, por tanto no se exponen en esta parte.