

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

**“SISTEMA DE GESTIÓN DE CORRESPONDENCIA – VÍA WEB
CASO: ÁREA CIENCIAS ECONÓMICAS, FINANCIERAS Y
ADMINISTRATIVAS – UNIVERSIDAD PÚBLICA DE EL ALTO”**

Para optar al título de Licenciatura en Ingeniería de Sistemas

Mención: INFORMÁTICA Y COMUNICACIONES

Postulante:	Univ. Santos Wilmer Yana Tarqui
Tutor Metodológico:	Ing. Marisol Arguedas Balladares
Tutor Especialista:	Ing. Isaac Magno Quispe Mamani
Tutor Revisor:	Ing. Enrique Flores Baltazar

**EL ALTO – BOLIVIA
2020**

Dedicatoria

At mi madre Rosalia Jarqui Mamani y a mi padre Rufino Yana Jito, por ser el pilar fundamental en todo lo que soy, en toda mi educación, tanto académico, como en la vida, por su apoyo incondicional y dedicación día tras día y también a mi hija Melany Belen por ser mi fuente de motivación e inspiración para poder superarme cada día más y así poder luchar para que la vida nos depara un futuro mejor

Agradecimiento

Agradecemos a Dios por bendecirnos la vida, por guiarnos a lo largo de nuestra existencia, ser el apoyo y fortaleza en aquellos momentos de dificultad y de debilidad.

Agradezco a mi familia porque ellos estuvieron en los días más difíciles de mi vida como estudiante.

Al Ing. Marisol Arguedas Balladares, Tutor Metodológico por emplear su experiencia en este proyecto indicándome los pasos a seguir, con su firme compromiso desde el inicio hasta el fin.

Al Ing. Isaac Magno Quispe Mamani, Tutor Especialista por sus incansables correcciones en este proyecto, por ayudarme a cumplir los objetivos de este proyecto y su perseverancia para perfeccionar el proyecto.

Al Ing. Enrique Flores Baltazar, Tutor Revisor por sus correcciones y colaboración en puntos precisos e indicados, en el desarrollo del proyecto.

Finalmente agradecer a mis amigos que son parte de la comunidad universitaria

GRACIAS.....

RESUMEN

En un mundo globalizado, donde los avances tecnológicos han tenido transformaciones tanto en la banca, como en el sistema financiero, comercial y productivo en todo el mundo. La educación no queda al margen de estas transformaciones. En la educación superior, cada vez son más las universidades que utilizan y promueven el empleo del software libre entre la comunidad universitaria y la sociedad.

El proyecto sistema de Gestión de Correspondencia vía web se desarrolló para el Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, dedicada a formar profesionales competentes con vocación investigativa y generadora de conocimiento científico y tecnológico, en ámbitos de gestión, económico-financiero, de servicio y en comercio internacional con calidad y calidez, buscando impulsar el desarrollo integral y sostenible de la sociedad.

El presente proyecto plantea un sistema de Gestión de Correspondencia vía web de gestión administrativa para los servicios de atención del solicitante de dicha correspondencia, además contienen módulos de administración de usuarios, por lo cual permiten al usuario satisfacer la necesidad de consultas, generar reportes y entablar una interfaz amigable para los usuarios.

La metodología adoptada para el desarrollo del software será la ingeniería web basada en UML (UWE por sus siglas en Ingles UML web engineering), que tiene como base en Lenguaje de Modelado Unificado UML, el cual modela las distintas etapas de evolución del nuevo sistema de software. Las herramientas utilizadas para el desarrollo e implementación, lenguaje de programación web php utilizando framework Codeigniter y como gestor de base de datos Mysql.

ÍNDICE DE CONTENIDO

CAPITULO I: MARCO PRELIMINAR	Pág.
1.1 INTRODUCCIÓN.....	10
1.2 ANTECEDENTES	11
1.2.1 Antecedentes Institucionales	11
1.2.2 Antecedentes Académicos Internacionales	16
1.2.3 Antecedentes Académicos Nacionales.....	17
1.3 PLANTEAMIENTO DEL PROBLEMA	18
1.3.1 Problema General.....	18
1.3.2 Problemas Específicos	18
1.4 OBJETIVOS.....	19
1.4.1 Objetivo General.....	19
1.4.2 Objetivos Específicos	19
1.5 JUSTIFICACIÓN	20
1.5.1 Justificación Técnica.....	20
1.5.2 Justificación Económica.....	21
1.5.3 Justificación Social	21
1.6 METODOLOGÍA	21
1.6.1 Método de Ingeniería	22
1.6.2 Metodología de Ingeniería Web.....	24
1.7 HERRAMIENTAS.....	24
1.8 LÍMITES Y ALCANCES	29
1.8.1 Límites	29
1.8.2 Alcances	29
1.9 APORTES	30
CAPITULO II: MARCO TEÓRICO.....	32
2.1 CONCEPTOS BÁSICOS	32

2.1.1 Concepto de Sistema	32
2.1.2 Definición de Gestión.....	33
2.1.3 Que significa Gestionar.....	33
2.1.4 Definición Gestión, poder, conducción.....	35
2.1.5 Las estrategias de gestión	37
2.2 CORRESPONDENCIA	38
2.2.1 Definición de correspondencia.....	38
2.2.2 Importancia de la Correspondencia	39
2.2.3 Clasificación de la correspondencia	39
2.2.4 División de la correspondencia	40
2.3 LA WEB	41
2.3.1 Introducción	41
2.3.2 Conceptos Básicos	42
2.3.3 Analizando el uso	43
2.3.4 Caracterizando la web estructura y visibilidad	43
2.3.5 Estructura	44
2.4 INGENIERÍA DE REQUERIMIENTOS.....	45
2.4.1 Requerimiento/requisito	46
2.4.2 Características de los requerimientos.....	46
2.4.3 Tipos de requisitos.....	48
2.4.4 Actividades para la Ingeniería de Requerimientos.....	51
2.4.5 Personal involucrado en la Ingeniería de Requerimientos.....	53
2.4.6 Análisis comparativo de las técnicas de Ingeniería de Requerimientos.	54
2.4.7 Importancia de la Ingeniería de Requerimientos	56
2.4.8 Gestión de Requisitos. Principales características	57
2.5 PROCESO UNIFICADO ÁGIL (AUP).....	60
2.5.1 Agile Unified Process (AUP)	60
2.5.2 Disciplinas de AUP	61
2.5.3 Fases de AUP	62
2.6 INGENIERÍA WEB BASADO EN UML (UWE)	66
2.6.1 Especificando los requisitos.....	68

2.6.2 Definiendo el contenido	76
2.6.3 Estructura y Modelo de Navegación	77
2.6.4 Modelo de presentación.....	79
2.6.5 Modelo de proceso	81
2.7 MODELO DE CALIDAD ESTABLECIDO POR EL ESTÁNDAR ISO 9126 83	
2.7.1 ISO/IEC 9126: Tecnologías de la Información.....	83
2.7.3 Ámbitos de uso de ISO/IEC 9126	84
2.7.4 Atributos de la Norma para Calidad Externa e Interna	85
2.7.5 Atributos Para Calidad en Uso.....	88
2.7.6 Perfil de Calidad Usando ISO/IEC 9126	89
2.8 MODELO CONSTRUCTIVO DE COSTOS (COCOMO).....	89
2.8.1 Características Generales	90
2.8.2 Modelos de estimación	90
2.8.3 Modelo Básico	91
2.8.4 Modelo intermedio	92
2.8.5 Modelo detallado	96
CAPITULO III MARCO APLICATIVO.....	97
3.1 DIAGNÓSTICO DEL SISTEMA ACTUAL	97
3.1.1 Descripción de funciones.....	98
3.1.2 Identificación de actores del negocio	101
3.1.3 Modelo del Negocio	102
3.1.4 Diagrama de Casos de uso del negocio	102
3.1.5 Diagrama de Actividades del Negocio	105
3.1.6 Diagramas de secuencias del negocio	107
Diagramas de secuencias del negocio: Tramite unidad kardex	107
3.1.7 Ingeniería de requerimientos	109
3.2 ANÁLISIS DEL SISTEMA	110
3.2.1 Modelo de casos de uso del sistema.....	110

3.2.1.1 Diagrama de Casos de uso del Director	110
3.2.1.2 Diagrama de Casos de uso de Técnico académico de kardex,	115
Auxiliar de kardex, Secretaria y Auxiliar de Oficina	115
3.2.1.3 Diagrama de Casos de uso del Administrador	123
3.2.2 Diagrama de secuencia del sistema	130
3.2.2.1 Diagrama de secuencia del sistema: Registrar Usuario	131
3.2.2.2 Diagrama de secuencia del sistema: Registrar Organigrama.....	132
3.2.2.3 Diagrama de secuencia del sistema: Registrar Tipo de Correspondencia	133
3.2.2.4 Diagrama de secuencia del sistema: Registrar Correspondencia Interna	134
3.2.2.5 Diagrama de secuencia del sistema: Registrar Instrucción	135
3.2.3 Diagrama de colaboración del sistema.....	136
3.2.3.1 Diagrama de colaboración: Registrar Usuario	136
3.2.3.2 Diagrama de colaboración: Registrar Organigrama	137
3.2.3.3 Diagrama de colaboración: Registrar Tipo de Correspondencia	138
3.2.3.4 Diagrama de colaboración: Registrar Correspondencia Interna.....	139
3.2.3.5 Diagrama de colaboración: Registrar Instrucción	140
3.3 DISEÑO DEL SISTEMA	141
3.3.1 Modelo conceptual del dominio del sistema	141
3.3.2 Modelo de navegación.....	143
3.3.3 Modelo de Presentación	144

3.3.4 Modelo de Implementación	147
3.4 PRUEBAS DE CALIDAD	156
3.4.1 Factores de calidad ISO 9126	156
3.4.1.1 Funcionalidad	156
3.4.1.2 Confiabilidad.....	161
3.4.1.3 Usabilidad.....	162
3.4.1.4 Facilidad de Mantenimiento.....	164
3.4.1.5 Portabilidad.....	165
3.5 ANÁLISIS DE COSTOS	166
3.5.1 COCOMO II	166
3.6 SISTEMA DE GESTION DE SEGURIDAD DE LA INFORMACION ISO-27002	173
3.6.1 Seguridad lógica	173
3.6.2 Seguridad física	173
3.6.2 Seguridad Organizativa	173
3.6.3 Buenas Prácticas para la Seguridad.....	174
4 CAPITULO IV CONCLUSIONES Y RECOMENDACIONES	176
4.1 CONCLUSIONES	176
4.2 RECOMENDACIONES.....	177
5 BIBLIOGRAFÍA	179

1.1 INTRODUCCIÓN

En la actualidad las tecnologías de información han evolucionado rápidamente, este hecho ha sido aprovechado por las empresas e instituciones para mejorar el manejo de grandes cantidades de información, facilitando su organización y control para obtener mayores beneficios.

Hoy en día los sistemas de información se constituyen en herramientas indispensables para mejorar la administración de los recursos de estas entidades, por lo cual existe una predisposición para que cada vez más procesos sean sujetos de automatización a través de la implementación de nuevos sistemas de información.

Por otra parte, la llegada de internet ha cambiado la forma de desarrollar sistemas de información, pasando de una etapa donde se privilegiaban las aplicaciones de escritorio a otra donde la tendencia es el desarrollo de sistemas de información web que funcionen en internet.

A su vez las bases de datos, surgidas como respuesta al nuevo planteamiento de los sistemas orientados hacia los datos, para mejorar la calidad de las prestaciones de los sistemas informáticos y aumentar su rendimiento, presentan una multitud de ventajas especialmente dirigidas al usuario final, las cuales se pueden resumir en nueve puntos: Versatilidad para representar la información; Desempeño; Mínima redundancia; Capacidad de acceso; Simplicidad; Seguridad de la información; Privacidad; Integridad; Afinación de datos.

Uno de los elementos con carácter fundamental al interior de las Entidades es del manejo de la Correspondencia, la cual provee a todas las dependencias de aquella información diaria requerida para la constante toma de decisiones, porque en ella se basa gran parte del trabajo de carácter administrativo.

A nivel del Área de Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto, el Sistema de Registro y Gestión Correspondencia propuesto permitirá la reducción de tiempos y costo en el manejo de los documentos, teniendo en cuenta la disminución de los tiempos de respuesta y el ahorro de insumos. A su vez, optimizará los procesos y tiempos de respuesta a solicitudes, tanto a usuarios internos como externos de la institución, además de conservar y evitar la pérdida de documentos.

Para el desarrollo se empleara la metodología conocida con el nombre de Proceso Unificado Ágil, herramientas de servidor local xampp, lenguaje de programación php, usando el framework CodeIgniter que permitan de esta manera brindarle al programa de Ingeniería de Sistemas, una solución informática óptima de gestión documental y sin ningún costo de licenciamiento.

1.2 ANTECEDENTES

1.2.1 Antecedentes Institucionales

En fecha 28 de julio de 2010 se lleva a cabo la Asamblea General Docente Estudiantil, convocada por las Carreras de Contaduría Pública, Administración de Empresas y Economía, el cual establece mediante Resolución AGDE N° 001/2010 en su Artículo Primero, Aprobar a partir de la fecha 28 de julio de 2010 la vigencia del Área de Ciencias Económicas, Financieras y Administrativas, conformadas por las Carreras de Contaduría Pública, Administración de Empresas y Economía, dentro de la estructura académica de la Universidad Pública de El Alto. En su Artículo Segundo, determina elegir como autoridades interinas del Área a las siguientes personas que representan a las tres carreras ya citadas: Decano de Área Lic. Ricardo Nogales Quispe (Director de Contaduría Pública); Presidente de la Asociación de Docentes del Área: Lic. Carlos Navia Alanes (Docente de la Carrera de Economía); Secretario Ejecutivo del Centro de Estudiantes del Área: Univ. Juan Tito Calle Mamani (Universitario de la Carrera de Administración de Empresas).

Asimismo, el Honorable Consejo Universitario según la Resolución N° 131/2010

de fecha 22/09/2010, en su Artículo Primero, aprueba la vigencia del ÁREA DE CIENCIAS ECONÓMICAS, FINANCIERAS Y ADMINISTRATIVAS, conformada por las tres Carreras de Economía, Contaduría Pública y Administración de Empresas.

Y a partir de la gestión 2014, se incorporan las Carreras de Comercio Internacional y Gestión Turística y Hotelera, según la Resolución del Honorable Consejo Universitario N° 110-A/2014 de fecha 13/06/2014, que en su Artículo Primero establece: “Las Carreras de Comercio Internacional y Gestión Turística y Hotelera se incorporarán al ÁREA DE CIENCIAS ECONÓMICAS, FINANCIERAS Y ADMINISTRATIVAS”.

Misión

Formamos profesionales competentes con vocación investigativa y generadora de conocimiento científico y tecnológico, en ámbitos de gestión, económico-financiero, de servicio y en comercio internacional con calidad y calidez, buscando impulsar el desarrollo integral y sostenible de la sociedad.

Visión

Ser reconocida y constituirse en un referente dentro del ámbito académico, científico y tecnológico a nivel local, nacional e internacional que contribuya en la formación de profesionales innovadores, competentes y éticos, para coadyuvar al desarrollo integral y sostenible en ámbitos de gestión, económico-financiero, de servicio y comercio internacional”.

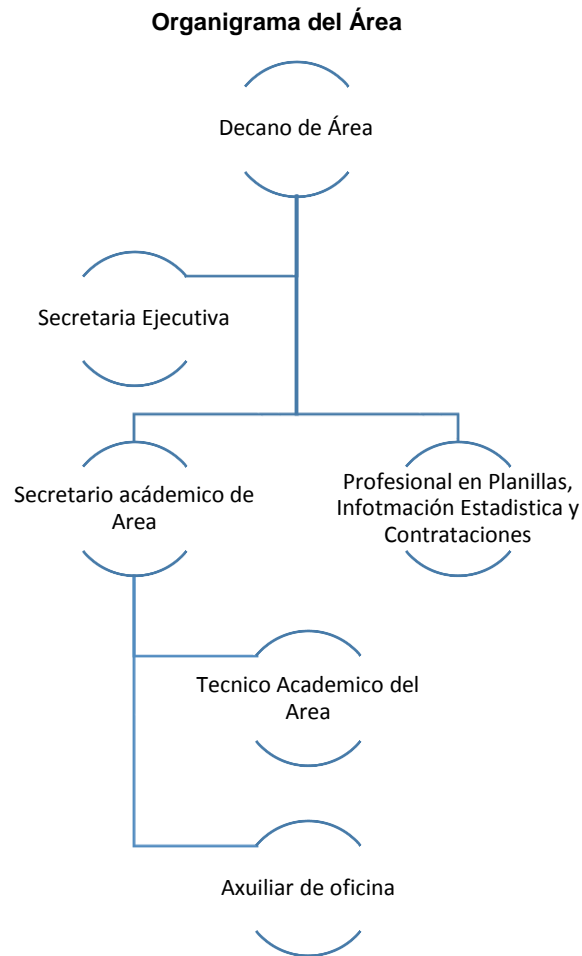
Principios y Valores

Los principios y valores fundamentales del Área de Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto son:

- La práctica y la promoción de los valores sociales e individuales que le son propios, tales como la libertad, solidaridad, responsabilidad, honestidad, transparencia, integridad, fraternidad, justicia social, el pluralismo, el respeto a la otredad y el espíritu crítico reflexivo.
- La sensibilidad y el compromiso de preservar y conservar el medio

ambiente, por un desarrollo sostenible.

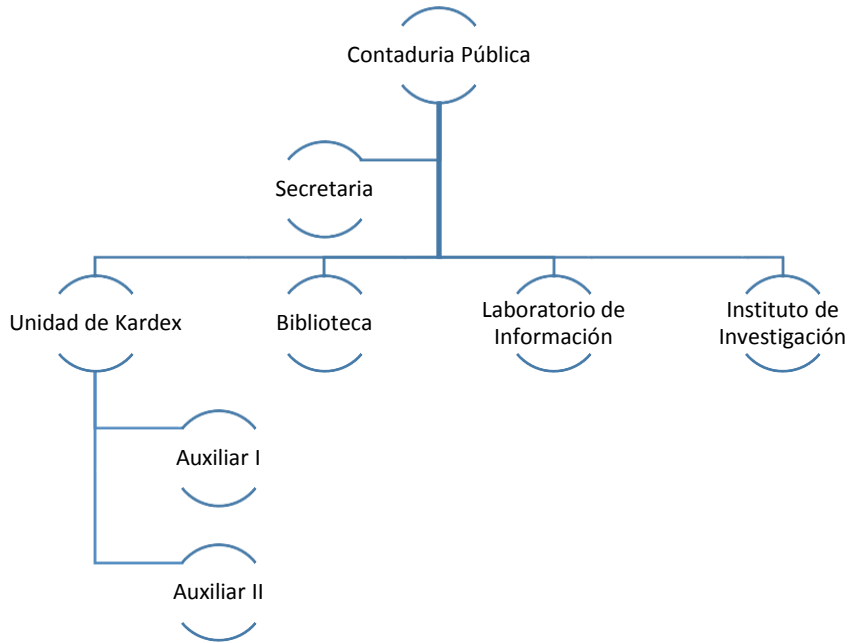
- La responsabilidad individual y colectiva con los distintos sectores de la comunidad universitaria en el gobierno, gestión y control del funcionamiento de la Universidad.
- El trabajo en equipo, como facilitador del proceso formativo, investigador y de gestión.



Fuente: Elaboración Propia

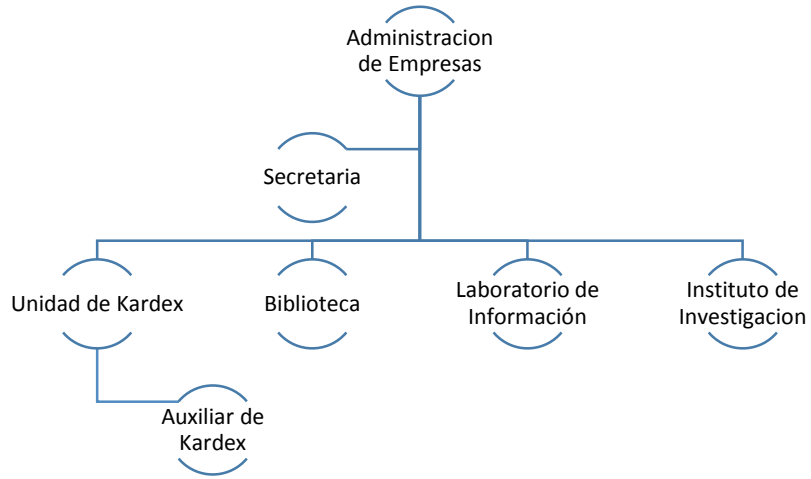
Organigramas de las carreras pertenecientes al “Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto”.

Organigrama Contaduría Pública



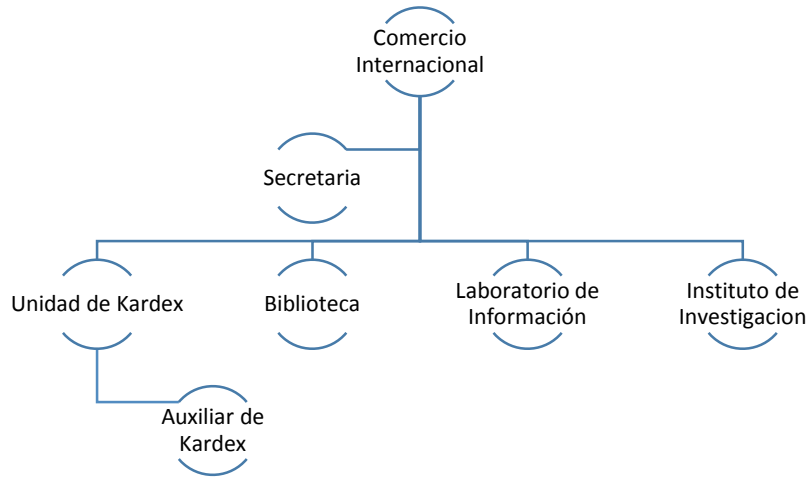
Fuente: Elaboración Propia

Organigrama Administración de Empresas.



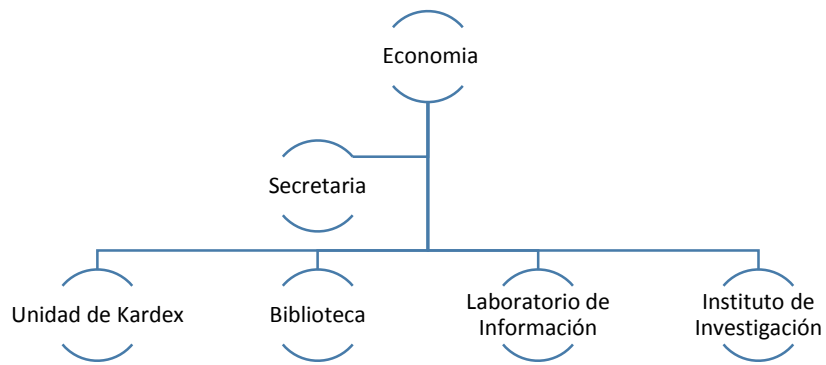
Fuente: Elaboración Propia

Organigrama Comercio Internacional



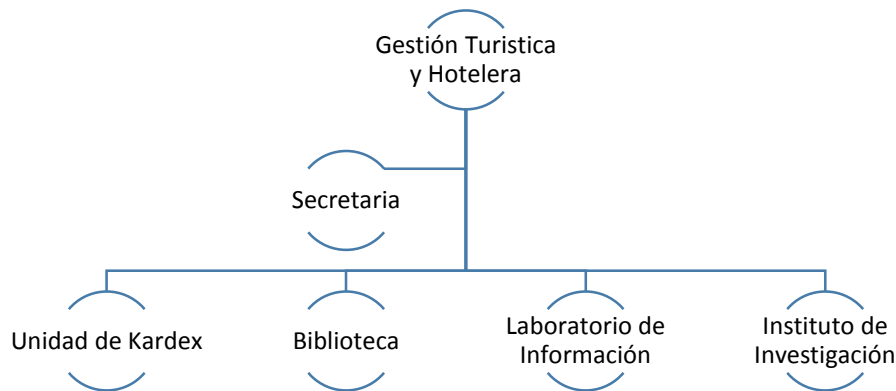
Fuente: Elaboración Propia

Organigrama Economía



Fuente: Elaboración Propia

Organigrama Gestión Turística y Hotelera



Fuente: Elaboración Propia

1.2.2 Antecedentes Académicos Internacionales

Luego de un relevamiento de información, se ha identificado los siguientes trabajos ajenos similares realizados a nivel internacional, que son los siguientes:

- “DISEÑO DE UN SISTEMA DOCUMENTAL PARA EL PROGRAMA DE INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD SAN BUENAVENTURA – SEDE BOGOTA”. El desarrollo del sistema permite la administración y control de documentos de forma sistematizada, mediante el uso de nuevas herramientas y aplicaciones tecnológicas. Para el desarrollo del proyecto se empleó metodologías de desarrollo de aplicaciones XP basadas en procesos ágiles, programación extrema, el modelo de WebUML y herramientas de desarrollo de la base de datos DBDesigner, el lenguaje de programación PHP y el gestor de base de datos MySQL. Se logró disminuir el tiempo de elaboración y la cantidad de errores del proceso de documentación, así mismo, se lleva un registro completo y ordenado de los tramites lo cual a disminuido la desorganización de la información y los reportes generados ayuda a la toma de decisiones sobre inversiones de la empresa (Quinteros, 2007).

- “DISEÑO DE UN SISTEMA DE GESTIÓN DE DOCUMENTOS EN UNA EMPRESA CONSTRUCTORA”, Departamento de Organización Industrial y Gestión de Empresas Escuela Técnica Superior de Ingeniería Universidad de Sevilla, 2017. Por SGD, entendemos el conjunto de elementos y de relaciones entre ellos diseñados con el objetivo de normalizar, controlar y coordinar todos los procesos y actividades que inciden en la producción, recepción, circulación, almacenamiento, organización, conservación, disposición final y accesibilidad de los documentos generados en el transcurso de la actividad de una organización. Las herramientas que se conocen hoy día sobre la gestión de documentos en las organizaciones y sus sistemas de archivos, se desarrollaron a partir de la concepción norteamericana denominada como records management. La gestión documental siempre ha constituido una necesidad y un problema para las organizaciones, pero es en la actual Sociedad de la Información, debido al creciente desarrollo de la información y la generación de contenidos, donde se ha erigido como una función clave del éxito empresarial, estableciéndose como una herramienta básica de soporte para otros procesos gerenciales como la gestión de la calidad, la gestión de riesgos, el control interno y hasta la lucha contra la corrupción. . El proyecto logró alcanzar el objetivo planteando mejorando así el movimiento de los materiales desde el ingreso hasta su egreso, recortando el trabajo de los funcionarios de la constructora. (Pérez, 2017)

1.2.3 Antecedentes Académicos Nacionales

Así también, se ha identificado los siguientes trabajos ajenos similares realizados a nivel Nacional, que son los siguientes:

- SISTEMAS DE INFORMACIÓN ACADÉMICA (SIA)-UPEA, Sistema de Información Académica que facilita a los estudiantes de la Carrera de Ingeniería de Sistemas la inscripción en línea y otros procesos de kardex.

Las herramientas utilizadas, gestor de base de datos Postgres, Servidor web apache, php como lenguaje de programación (Sergio García, 2016).

- SISTEMA DE ADMINISTRACIÓN DE ARCHIVOS ESTUDIANTILES KARDEX-INFORMÁTICA-UMSA, El sistema propuesto prioriza la digitalización de los archivos estudiantiles. El desarrollo del sistema fue guiado por el proceso de desarrollo orientado a objetos que propone Roger R. Pressman (Javier Alex, 2014).

1.3 PLANTEAMIENTO DEL PROBLEMA

1.3.1 Problema General

Actualmente, el Área de Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto, no cuenta con un Sistema de Gestión de Correspondencia que controle el flujo, estado y disposición de los documentos, que regularmente entran y salen de las diferentes dependencias, ocasionando problemas como pérdida de información, tiempo de respuesta lentos a los procesos y deficiencias en la atención al público.

El proceso de gestión de correspondencia está registrada en libros de actas de forma manual, el cual se evidencia por la existencia de retardación en la elaboración de informes y reportes, ocasiona que la asignación, control y supervisión de la documentación, del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, no sea eficiente, lo cual dificulta la toma de decisiones.

1.3.2 Problemas Específicos

- Información descentralizada de entradas y salidas de correspondencia, lo

cual genera demora en el procesamiento de información.

- La búsqueda de información se realiza de forma manual y por la cantidad existente es difícil encontrarla de forma oportuna lo que ocasiona emitir respuestas tardías.
- Existe una enorme cantidad de documentación que se encuentra registrada en cuadernos y Excel lo que conlleva a la pérdida de documentos.
- Falta Información oportuna y ágil respecto a documentación generada, lo que ocasiona no poder emitir informes oportunos

1.4 OBJETIVOS

1.4.1 Objetivo General

Desarrollar un “Sistema de Gestión de Correspondencia - Vía Web Caso: Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto”, que genere información confiable optimizando, a fin de lograr que la respuesta a las solicitudes sea de manera segura y eficiente.

1.4.2 Objetivos Específicos

Los objetivos específicos planteados para el presente proyecto son:

- Realizar un diagnóstico de la situación actual de la documentación de registro de correspondencia a nivel del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.
- Realizar el análisis y diseño para la realización del sistema para el Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.

- Digitalizar y registrar en sistema de Base de Datos los documentos bajo la idea de resguardar y tener a disponibilidad inmediata y fiable.
- Crear opciones de búsqueda avanzadas para diferentes tipos de correspondencias.
- Realizar pruebas de calidad de software.

1.5 JUSTIFICACIÓN

El sistema a desarrollar permitirá al personal del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, llevar un registro de la documentación concerniente a la variedad de correspondencias que llega a las dependencias del Área, donde dicha información servirá para obtener un acceso rápido de la misma, y un mejor control y facilitar la documentación, a toda la comunidad universitaria.

1.5.1 Justificación Técnica

La justificación técnica del presente proyecto es que está se enfoca al hecho de se empleará una metodología para el diseño y desarrollo del sistema de información, ya que las características técnicas del mismo son a exigencias y necesidades que tiene el Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.

Actualmente el Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, cuentan con equipamiento necesario de hardware, conexión a internet, hosting y software que viabiliza el presente proyecto. Por otro lado, se cuenta con la tecnología para poder desarrollar el sistema de gestión de correspondencias y el software necesario para su desarrollo, como lenguaje de programación Php7, como gestor de base de datos MySql y Apache como Servidor Web.

1.5.2 Justificación Económica

El Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, reducirán el uso de material de escritorio debido a que la información estará centralizada en una base de datos, donde los usuarios técnicos y parte administrativa realizarán un seguimiento preciso y oportuno de los movimientos de correspondencia, reduciendo los tiempos de trabajo y esfuerzo.

Por otro lado, mencionamos que el sistema será desarrollado bajo la premisa de software libre, que implica la no erogación de gastos en licencias de uso. Por otra parte, la institución cuenta con cableado de red lo cual facilita su implementación

1.5.3 Justificación Social

El presente trabajo contribuye a mejorar la atención, comunicación y disposición de la información al personal del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, y a la comunidad universitaria, así también instituciones externas, agilizando el servicio para que este, no realice esperas de respuesta de la posible existencia o no de los documentos que se requiera o desee obtener, así obtener información oportuna y segura de forma rápida y en tiempo real.

1.6 METODOLOGÍA

En una reunión celebrada en febrero de 2001 en Utah-EEUU, nace el término "ágil" aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software, incluyendo algunos de los creadores o impulsores de metodologías de software. Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se

pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas. Varias de las denominadas metodologías ágiles ya estaban siendo utilizadas con éxito en proyectos reales, pero les faltaba una mayor difusión y reconocimiento.

Tras esta reunión se creó The Agile Alliance, una organización, sin ánimo de lucro, dedicada a promover los conceptos relacionados con el desarrollo ágil de software y ayudar a las organizaciones para que adopten dichos conceptos. El punto de partida es y fue el Manifiesto Ágil, un documento que resume la filosofía "ágil".

1.6.1 Método de Ingeniería

Las metodologías ágiles son una serie de técnicas para la gestión de proyectos que han surgido como contraposición a los métodos clásicos de gestión como CMMI (Modelo de Capacidad y Madurez). Aunque surgieron en el ámbito del desarrollo de software, también han sido exportadas a otro tipo de proyectos. Todas las metodologías que se consideran ágiles cumplen con el manifiesto ágil que no es más que una serie de principios que se agrupan en 4 valores:

1. Los individuos y su interacción, por encima de los procesos y las herramientas.
2. El software que funciona, frente a la documentación exhaustiva.
3. La colaboración con el cliente, por encima de la negociación contractual.
4. La respuesta al cambio, por encima del seguimiento de un plan.

Proceso Unificado Agil o AUP.- El Proceso Unificado Agil de Scott Ambler o Agile Unified Process (AUP) en inglés es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP. El AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas (test driven

development - TDD), 14 Modelado Agil, Gestión de Cambios Agil, y Refactorización de Base de Datos para mejorar la productividad.

El proceso unificado (Unified Process o UP) es un marco de desarrollo software iterativo e incremental. A menudo es considerado como un proceso altamente ceremonioso porque especifica muchas actividades y artefactos involucrados en el desarrollo de un proyecto software.

Fases. - AUP divide el proceso en 4 fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades.

- **Inicio.** - Esta fase tiene como propósito definir y acordar el alcance del proyecto con los patrocinadores, identificar los riesgos asociados al proyecto, proponer una visión muy general de la arquitectura de software y producir el plan de las fases y el de iteraciones posteriores.
- **Elaboración.** - En la fase de elaboración se seleccionan los casos de uso que permiten definir la arquitectura base del sistema y se desarrollaran en esta fase, se realiza la especificación de los casos de uso seleccionados y el primer análisis del dominio del problema, se diseña la solución preliminar.
- **Construcción.** - El propósito de esta fase es completar la funcionalidad del sistema, para ello se deben clarificar los requisitos pendientes, administrar los cambios de acuerdo a las evaluaciones realizados por los usuarios y se realizan las mejoras para el proyecto.
- **Transición.** - El propósito de esta fase es asegurar que el software esté disponible para los usuarios finales, ajustar los errores y defectos encontrados en las pruebas de aceptación, capacitar a los usuarios y proveer el soporte técnico necesario. Se debe verificar que el producto cumpla con las especificaciones entregadas por las personas

involucradas en el proyecto.

1.6.2 Metodología de Ingeniería Web

También se utiliza la metodología UWE (UML-Based Web Engineering) en las etapas de desarrollo de cada una de las iteraciones del proyecto, ya que esta metodología se especializa en el diseño de aplicaciones Web.

UWE UML (UML-Based Web Engineering) es una herramienta para modelar aplicaciones web, utilizada en la ingeniería web, prestando especial atención en sistematización y personalización (sistemas adaptativos).

UWE es una propuesta basada en el proceso unificado y UML pero adaptados a la web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.

Consiste en una notación y en un método. La notación se basa en UML (OMG, 2003): para aplicaciones Web en general y para aplicaciones adaptativas en particular.

El método consta de seis modelos:

- Modelo de casos de uso para capturar los requisitos del sistema.
- Modelo conceptual para el contenido (modelo del dominio).
- Modelo de usuario: modelo de navegación que incluye modelos estáticos y dinámicos.
- Modelo de estructura de presentación, modelo de flujo de presentación.
- Modelo abstracto de interfaz de usuario y modelo de ciclo de vida del objeto.
- Modelo de adaptación.

1.7 HERRAMIENTAS

En este proyecto de grado el desarrollo estará basado en las herramientas de software libre, puesto que permite la generación de soluciones informáticas.

Los elementos del sistema:

En la creación de un sistema web se trabaja con los elementos como un engranaje, en una página/aplicación web cohabitan lo que es el front-end y el back-end.

El **front-end** y el **back-end** lo que hacen es dividir la capa visual de un sistema y la capa del acceso de datos.

FRONTEND: Es la capa visual que permite a los usuarios/clientes la interacción con el sistema, por lo tanto, es la capa que define las experiencias de los usuarios en el sistema y la única a la que tienen acceso. El frontend son todas esas tecnologías que viven en el navegador y con las cuales se crea el diseño del sistema. Cuando hablamos del diseño de un sistema se trata de la creación de las interfaces, las propiedades que pretendemos que tenga y la interacción de las mismas. Las tecnologías front-end utilizadas en el sistema serán: HTML + CSS + JAVASCRIPT, Bootstrap y otros.

Funciones de cada una de estas tecnologías en el sistema y en el front-end

- **HTML5:** Es un lenguaje de marcado de etiquetas de hipertexto. Es utilizado con la finalidad de ordenar, definir y distribuir el contenido de una interfaz.

Con HTML podemos indicar cuales son las divisiones, secciones de nuestra página y lo que incluyen, ya sea un párrafo, una imagen o un formulario.

- **CSS:** Conocido como la hoja de estilos en cascadas es utilizado con la intención de darle estética y estilos al contenido visual de las páginas.

Por ejemplo: definir el color de fondo de la página, la tipografía a usar, el tamaño de las letras, formas geométricas y una infinidad de funciones.

- **JavaScript:** Es un lenguaje de programación que permite la realización de procesos de cálculos matemáticos y lógicos.
- Tiene una particularidad que lo diferencia de los otros lenguajes de programación web y es que funciona del lado del cliente. Por eso se considera front-end.

Además de procesar datos de operaciones resulta necesario para generar interactividad en las páginas (crear efectos, mostrar/ocultar paneles al hacer click, eventos, acciones), fue esta la razón por la cual está presente en el sistema desarrollado.

✚ **Bootstrap:** es una excelente herramienta para crear interfaces de usuario limpias y totalmente adaptables a todo tipo de dispositivos y pantallas, sea cual sea su tamaño. Además, Bootstrap ofrece las herramientas necesarias para crear cualquier tipo de sitio web utilizando los estilos y elementos de sus librerías.

BACKEND: Son las tecnologías que hacen vida en la capa de acceso de datos, están del lado del servidor por lo tanto los usuarios no tienen acceso a ella.

Un servidor web es un ordenador donde se alojan aplicaciones/páginas en un espacio de memoria con la finalidad de dar respuestas a las peticiones de tipo HTTP que hacen los usuarios. Los servidores son los que dan vida al internet. Las tecnologías back-end con las que se desarrollan sistemas son:

Lenguajes de programación web interpretados de lado del servidor.

- ✓ Frameworks (marcos de trabajo) elaborados en base a un lenguaje de programación.

- ✓ Bases de datos.
- ✓ Servidores HTTP.

Para el desarrollo del backend del sistema se utilizarán las herramientas de CodeIgniter (framework de PHP), phpmyadmin (administrador de base de datos MySQL) y Xampp.

- **CodeIgniter versión 3.1.9:** Es un framework para el desarrollo de aplicaciones en php, que utiliza el MVC. Esto permite a los programadores o desarrolladores Web mejorar su forma de trabajar, además de dar una mayor velocidad a la hora de crear páginas Webs.

MVC.- Es un patrón de desarrollo de software usado para la separación de la lógica un sistema.

El modelo para lo referente a las bases de datos. Consultas, eliminación, actualización y registro en la información.

El controlador es el encargado de recibir y procesar las peticiones de los usuarios, evaluar el tipo de petición, las condiciones de las peticiones para solicitar una respuesta del modelo.

La vista recibe una respuesta del controlador y se encarga de presentar la información solicitada a los usuarios.

Algunas ventajas de usar el framework CodeIgniter:

- Las páginas se procesan más rápido, el núcleo de CodeIgniter es bastante ligero.
- Es sencillo de instalar, basta con subir los archivos al ftp y

tocar un archivo de configuración para definir el acceso a la base de datos.

- Existe abundante documentación en la red.
 - Facilidad de edición del código ya creado.
 - Facilidad para crear nuevos módulos, páginas o funcionalidades.
 - Estandarización del código
 - Separación de la lógica y arquitectura de la web, el MVC.
 - Cualquier servidor que soporte PHP+MySQL sirve para CodeIgniter.
 - CodeIgniter se encuentra bajo una licencia open source, es código libre.
- **Phpmyadmin:** Como todo gran sistema de información se necesita una base de datos donde se pueda almacenar la información y hacer posterior uso de ella. Phpmyadmin es una herramienta que permite el manejo del sistema gestor de base de datos MySQL.

Las tablas de una base de datos son las estructuras donde se guardan la información y los registros de los objetos.

Haciendo una pequeña analogía de una base de datos donde necesitamos almacenar información sobre una biblioteca, biblioteca es el nombre de la base de datos, las tablas serian libros, revistas científicas, tesis y material audiovisual mientras que los campos de la tabla libros son todas las características que posee: textura, nombre, autor, tipo, etc.

- **Xampp:** La institución cuenta con un servidor http (web) pero lo ideal y recomendable es no alojar el sistema en el servidor hasta que no esté listo para su implantación.

Por lo tanto, al utilizar XAMPP se convierte en un servidor local sobre el cual puedo ir desarrollando el proyecto e ir haciendo pruebas.

1.8 LÍMITES Y ALCANCES

1.8.1 Limites

Por requerimientos del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, el desarrollo del sistema esta específicamente orientado a los procesos de gestión de correspondencia.

1.8.2 Alcances

El sistema de información desarrollado tendrá los siguientes alcances:

- Módulo de Administración de Usuarios, se otorgará privilegios de acceso al sistema y se aplicará técnicas de seguridad en cuanto al ingreso al sistema a sólo el personal autorizado como ser los técnicos y administrador.
- Módulo de información Rápida, brindará toda la información concerniente a las correspondencias que tiene el Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.
- Módulo del Personal, ofrece información referente a las al registro del personal administrativo del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, y llevar el manejo de información de manera eficiente.
- Módulo de correspondencia, ofrece información referente a la documentación externa e interna, siendo explícitos en su tipo de correspondencia, derivación y localización de la misma, así proceder a una información precisa.

- Módulo de Gestión, se evalúa la gestión del flujo de correspondencia del sistema, si es óptimo, regular en el Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.
- Módulo de Unidad, se registran el nombre y sigla de las unidades que hacen el trámite correspondiente, también se registra el cargo de la persona que viene a dejar dicha correspondencia del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.
- Módulo de Derivación, se registran el nombre de la instrucción, adjunto especificando el destino de dicha correspondencia del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.
- Módulo de reportes con formatos estandarizado, que coadyuve a la gestión de correspondencias del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.

1.9 APORTES

El sistema de gestión de correspondencias vía web es un importante aporte para manejar la información de los movimientos documentales del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, de este modo el manejo de información transparente, organizada, permitirá la eliminación de tiempos innecesarios y el trabajo exhaustivo.

El desarrollo de sistema de gestión de correspondencia vía web permitirá tener información centralizada en una sola base de datos. Que permitirá tener un incremento de funcionalidad y desempeño dentro del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto,

mediante un adecuado flujo de información, por otro lado el sistema será seguro, escalable, trazable y único.

CAPITULO II: MARCO TEÓRICO

2.1 CONCEPTOS BÁSICOS

2.1.1 Concepto de Sistema

El concepto de sistema en general está sustentado sobre el hecho de que ningún sistema puede existir aislado completamente y siempre tendrá factores externos que lo rodean y pueden afectarlo, por lo tanto podemos referir a Muir citado en Puleo (1985) que dijo: "Cuando tratamos de tomar algo, siempre lo encontramos unido a algo más en el Universo".

Puleo define sistema como "Un conjunto de entidades caracterizadas por ciertos atributos, que tienen relaciones entre sí y están localizadas en un cierto ambiente, de acuerdo con un cierto objetivo".

Una **Entidad** es lo que constituye la esencia de algo y por lo tanto es un concepto básico. Las entidades pueden tener una existencia concreta, si sus atributos pueden percibirse por los sentidos y por lo tanto son medibles y una existencia abstracta si sus atributos están relacionados con cualidades inherentes o propiedades de un concepto.

Los **Atributos** determinan las propiedades de una entidad al distinguirlas por la característica de estar presentes en una forma cuantitativa o cualitativa. Los atributos cuantitativos tienen dos percepciones: La dimensión y la magnitud. La dimensión es una percepción que no cambia y que identifica al atributo, para lo cual se utilizan sistemas de medida basados en unidades o patrones, tales como el CGS, MKS, etc.; ejemplos de dimensión son Kg., tamaño, sexo, color, etc. La magnitud es la percepción que varía y que determina la intensidad del atributo en un instante dado de tiempo, para lo cual se utilizan escalas de medida, tales como: la nominal, la ordinal, la de intervalo y la de razón. (Puleo, 1985)

Las **Relaciones** determinan la asociación natural entre dos o más entidades o entre sus atributos. Estas relaciones pueden ser estructurales, si tratan con la organización, configuración, estado o propiedades de elementos, partes o constituyentes de una entidad y son funcionales, si tratan con la acción propia o natural mediante la cual se le puede asignar a una entidad una actividad en base a un cierto objetivo o propósito, de acuerdo con sus aspectos formales (normas y procedimientos) y modales (criterios y evaluaciones).

El **Ambiente** es el conjunto de todas aquellas entidades, que al determinarse un cambio en sus atributos o relaciones pueden modificar el sistema.

El **Objetivo** es aquella actividad proyectada o planeada que se ha seleccionado antes de su ejecución y está basada tanto en apreciaciones subjetivas como en razonamientos técnicos de acuerdo con las características que posee el sistema. (Puleo, 1985)

2.1.2 Definición de Gestión

Nos encontramos con el término gestión; una palabra que se ha hecho hegemónica, de la mano de concepciones empresariales o administrativas de los procesos sociales, institucionales u organizacionales. Por eso conviene, en primer lugar, aclarar a qué nos referimos con el término gestión y con el verbo gestionar y, en especial, cuál es el alcance de estos términos en el marco de nuestras organizaciones o instituciones públicas. (Paulo, 1973)

2.1.3 Que significa Gestionar

Conviene hacer algunas otras precisiones. En el segundo sentido que se presentó, gestionar no es exclusivamente administrar (o “gerenciar”), ni simplemente organizar y, mucho menos, conducir (en sus sentidos autocrático, carismático, paternalista, etc.). Tampoco es algo que siga la lógica de las “recetas”, tantas veces relacionadas con el marketing estratégico o la producción de imagen corporativa.

Pensar y realizar la gestión es un desafío de cada momento. Porque cambiaron (y cambian permanentemente) las coordenadas de las realidades en las que nos movemos y, al mismo tiempo, nosotros (como instituciones o como personas) somos partícipes de esas transformaciones de los escenarios sociales dentro de los cuales actuamos. El desafío de gestionar, en cuanto actuar creativamente gestando procesos colectivos, nos lleva a adoptar una actitud histórica y analítica: nunca el pasado puede repetirse (sino sólo como farsa, decía Marx) y nunca el futuro como imagen debe llevarnos a negar las condiciones del presente. Forzar las situaciones presentes y a las personas u organizaciones en función de un fantasma del pasado o del futuro, significa desaprovechar las condiciones existentes y destruir a los otros. De lo que se trata, en cambio, es de producir procesos colectivos a partir de las situaciones y las condiciones vividas, que tienen un sentido histórico.

Según expresan Claudia Villamayor y Ernesto Lamas, gestionar es una acción integral, entendida como un proceso de trabajo y organización en el que se coordinan diferentes miradas, perspectivas y esfuerzos, para avanzar eficazmente hacia objetivos asumidos institucionalmente y que deseáramos que fueran adoptados de manera participativa y democrática. En esta línea, gestionar implica una articulación de procesos y resultados, y también de corresponsabilidad y cogestión en la toma de decisiones, en contraposición a la visión empresarial capitalista que se basa en la idea de centralización/descentralización y en el problema de la verticalidad/horizontalidad en las decisiones, sobre la base del derecho que da la propiedad, en forma directa o mediante la delegación que se hace en gerencias y direcciones. Las nuevas formas de gestionar, entonces, toman en cuenta la necesidad de desarrollar procesos de trabajo compartido y asumen la realización personal de quienes participan del proyecto.

Por lo tanto, no podríamos hablar de una gestión “exitosa” si nos estamos refiriendo tan sólo al equilibrio de fuerzas o la estabilidad lograda a partir de las habilidades de quien coordina o conduce. La estabilidad y el equilibrio tienen que basarse realmente en la corresponsabilidad de roles y de tareas diferenciadas,

en el diseño de estrategias participativas que articulen los procesos de trabajo a partir de las búsquedas personales y del proyecto colectivo. Todo ello para concretar los objetivos, con los recursos y las posibilidades de desarrollo del proyecto y de las personas que lo integran.

Cuando se habla de gestionar, entonces, se hace referencia a la forma a través de la cual un grupo de personas establece objetivos comunes, organiza, articula y proyecta las fuerzas, los recursos humanos, técnicos y económicos. En este sentido, la gestión es un proceso de construcción colectiva desde las identidades, las experiencias y las habilidades de quienes allí participan. Esto quiere decir que el proceso de gestión no debe apuntar a la negación o aplanamiento de diferencias, o al acallamiento de conflictos; sino que necesariamente debe articularlos, construyendo procesos colectivos, donde lo “colectivo” no es lo homogéneo, sino una plataforma y un horizonte común, una trama de diferencias articuladas en una concreción social. Lo que implica el reconocimiento y la producción de una cultura colectiva, organizacional o institucional. (Uranga, 2001)

2.1.4 Definición Gestión, poder, conducción

La gestión implica también una concepción y una práctica respecto del poder, de la administración y la circulación del mismo y de las formas de construir consensos y hegemonías dentro de una determinada organización o institución. Vale recalcar que la construcción de hegemonías (según lo expresan los filósofos Ernesto Laclau y Chantal Mouffe) no significa inmediatamente el planteamiento de situaciones de dominio, sino la posibilidad y el proyecto de articulación de fuerzas y de diferencias, a través de un imaginario y un objetivo común.

Por eso, gestionar es más que conducir. La acción de gestionar atraviesa a toda la institución, en sus relaciones inmediatas, en la coordinación interna, en las maneras de establecer o de frustrar lazos de trabajo, comunidades de trabajo, en la selección de determinados medios, en el conjunto de opciones que se adoptan a la hora de interactuar con otras instituciones. No es sólo conducción

o dirección, aunque las implique. Es coordinación de procesos de trabajo en el marco de una organización, donde se dan roles y tareas diferenciadas, que en principio pueden ser articuladas generando niveles de gestión. (Lamas, 1998)

La gestión es un juego de consensos, disensos y transformaciones que implican a toda la institución y a todos sus integrantes. La gestión implica un modo de comprender y de hacer nuestros proyectos desde una cuádruple perspectiva articulada; en nuestro caso: la político-cultural, la sanitario-social, la económica y la organizacional, comunicacional.

1. Político-cultural: comprende el reconocimiento de las escenas y horizontes fundacionales, las memorias de procesos y conflictos, los idearios, los objetivos, las utopías, las misiones, la cultura común.

2. Sanitario-social: comprende el perfil ideológico-conceptual del sector público, la inserción en políticas y programas, la proyección socio comunitario y profesional, la conformación de interlocutores institucionales, sectoriales, personales.

3. Económica: comprende los modelos para el desarrollo de la institución pública y de la administración de la misma, el financiamiento y la proyección económica de la institución.

4. Organizacional-comunicacional: comprende los modos de organización y comunicación internos y externos, los estilos, las formas de organización del trabajo y la participación, la distribución del poder y las estrategias de comunicación, las instancias de corresponsabilidad y cogestión.

La gestión, en este marco, es la manera de llevar adelante la articulación entre las perspectivas, a través de los modos organizacionales que sirvan a la misma y que sean coherentes con los fines y objetivos de la institución. En definitiva, la gestión siempre implica un trabajo de diagnóstico (entendido incluso como reconocimiento de nuestros interlocutores, sus marcos de referencia y sus campos de significación), de construcción de la memoria colectiva (que otorgue sentido histórico a los procesos), de análisis e interpretación del presente común (que permita la articulación de diferencias y la creatividad frente a problemas y desafíos) y la construcción del proyecto institucional (como un camino nunca

prefijado del todo donde confluyen imaginarios, expectativas, quehaceres, prácticas, identidades y experiencias diferentes).

2.1.5 Las estrategias de gestión

Una estrategia es una secuencia general y flexible de acciones a implementar para conseguir un conjunto de objetivos. Según expresa Washington Uranga, las estrategias de gestión deben ser diseñadas, en cada caso, en función de nuestros objetivos, los medios con los que contamos y los obstáculos que van surgiendo. Es por eso que la secuencia diagnóstico-planificación, gestión, así presentada en forma lineal, puede resultar engañosa. En realidad, se trata de una totalidad cuyos elementos no se suceden linealmente sino que forman parte de un todo inseparable. A los fines prácticos, diagnosticamos desde y para la gestión, y planificamos desde y para la gestión.

Ambas actividades -diagnosticar y planificar- son funciones de la gestión. ¿Qué implica la gestión más allá de poner en práctica lo planificado? En la gestión nos enfrentamos a lo imprevisible de los acontecimientos y al desafío de encontrar alternativas sobre la marcha. Los pilotos de los barcos o de los aviones trazan un rumbo sobre la base de una carta de navegación, tomando en cuenta los datos que le aportan los meteorólogos, las experiencias de otros navegantes, etc. Sin embargo, al acometer la travesía surgirán siempre nuevos datos, otros elementos no previstos que requieren de los conocimientos, de la destreza y hasta de la creatividad del navegante para decidir el recorrido en poco tiempo y ante nuevas situaciones. De esta capacidad de gestionar depende que se arribe a puerto seguro.

Gestionar es el arte de hacer posible un rumbo y alcanzar una meta en medio de las dificultades y de la imprevisibilidad de los acontecimientos. No se trata de soslayar las dificultades o de silenciar los conflictos, sino de crear alternativas de viabilidad para los procesos que vive un grupo, una organización o una institución. Para ello es necesario ponderar, negociar, producir siempre nuevos consensos. Porque cuando nos referimos a la gestión, en definitiva, estamos hablando de una acción política. De allí que, de camino, sea necesario prestar

atención a diferentes indicios (que nos anuncian dificultades, o nos confirman el camino recorrido); en este sentido, la acción de gestionar se parece a las del “rastreador” (del Facundo, de Domingo F. Sarmiento) quien, poniendo en juego su observación y su sensibilidad, va reconociendo el territorio a través de pequeñas señas, de indicios que indican las sendas a seguir.

Podemos hacer muy buenos pronósticos en base a un profundo conocimiento de la realidad; podemos trabajar con intensidad para encauzar el futuro en una u otra dirección, pero nunca tendremos la certeza total acerca de lo que va a ocurrir hasta el momento de llevar adelante nuestro proyecto. La interacción con esa novedad es exclusiva de la gestión. Más aún, la incertidumbre, el desorden y el conflicto, en muchos casos, puede ser el comienzo de nuevos órdenes imprevistos, pero que debemos atender y con los que tenemos que interactuar para provocar nuevos procesos, que requieren de nuestra creatividad. (Lamas, 1998) (Andino, 2001)

2.2 CORRESPONDENCIA

2.2.1 Definición de correspondencia

La correspondencia es un elemento importante en las organizaciones diariamente, ya que la misma nos permite enviar y recibir documentos y no sólo se pueden enviar documentos si no también paquetes ahorrando tiempo. Dicho contenido presenta información importante acerca de la correspondencia, la carta y como debe de elaborarse un sobre, el cual es un valioso instrumento en el ámbito empresarial. El sobre y la redacción de la carta juegan un papel muy importante y fundamental en este proceso de comunicación, siguiendo los pasos estipulados se puede lograr hacer un buen uso de la correspondencia.

Es el trato recíproco entre dos personas mediante el intercambio de cartas, tarjetas, telegramas, catálogos, folletos, etc. En las empresas se considera que la correspondencia es el alma del comercio y la industria. Es un medio de comunicación usado por el hombre desde hace muchos años para comunicarse

entre dos personas o individuos que están a larga distancia o cerca con un motivo muy variado. A lo largo del tiempo se ha perfeccionado sus normas y estilos, hasta llegar a nuestros días que existe el e-mail, que es la forma más rápida de enviar y asegurarse que la información llegue al destinatario. (Andino, 2001)

2.2.2 Importancia de la Correspondencia

Es un importante instrumento de comunicación escrita, es la parte intermedia entre las relaciones cliente proveedor y el lazo que une la mayor parte de transacciones comerciales.

De ella depende el desarrollo de las operaciones comerciales; el éxito de un negocio, una venta, por su eficiencia y rapidez han a que las empresas aumenten el volumen de sus ventas

2.2.3 Clasificación de la correspondencia

La correspondencia presenta una variedad de formas con características propias; por tanto, dados su importancia, es necesario realizar una clasificación que permita conocer la finalidad de cada uno.

Por el destino: Públicas y Privadas

- Públicas: contienen información general a través de medios de comunicación (prensa, radio, TV, Cartelera, pancartas.)
- Privadas: Tratan de información de interés y carácter particular comercial y oficial.

Por su contenido: Primera clase y Segunda clase

- Primera clase: es toda correspondencia de carácter actual y personal su contenido es solo de información: cartas, tarjetas, postales, documentos de negocios, periódicos, etc.

- Segunda clase: Es toda correspondencia relacionada con bultos, muestras, paquetes, cuya tramitación se realiza también por correo.

Por su tramitación: Postal y Telegráfica

- Postal: Cuando se acude a la oficina de correos para hacerla llegar a su destino, se divide en:
 - Ordinaria: Es la que sigue curso normal a través del correo
 - Certificada: Cuando el remitente paga una tarifa doble y exige un recibo para su entrega a su destinatario
 - Telegráfica: Es cuando la correspondencia se tramita ante las oficinas del telégrafo

Por su puntuación: Abiertas, Cerradas y Mixtas.

- Abiertas: No llevan signo de puntuación en la fecha, dirección, firma, antefirma, excepto en las abreviaturas.
- Cerradas: Son las que llevan los signos de puntuación
- Mixtas: Son las que combinan las Abiertas y Cerradas

Por su extensión: Largas medianas y cortas

- Largas: Cuando contienen más de doscientas palabras
- Medianas: Cuando contienen doscientas palabras
- Cortas: Cuando contienen cien palabras

Por su Forma: Abarca todos los tipos de correspondencia su diferencia está en redacción y presentación

2.2.4 División de la correspondencia

Correspondencia comercial:

Se refiere a la cruzada entre comerciantes, industrias, banqueros, etc. Y su finalidad es promover y agilizar las diversas transacciones comerciales.

Correspondencia Familiar y amistosa:

La finalidad de esta correspondencia es muy variada cualquier asunto particular entre ella.

Correspondencia Oficial:

Se refiere a la correspondencia cruzada entre los distintos organismos y oficinas de los gobiernos nacionales, municipales, estatales, etc. (Andino, 2001) (Boccaletti, 2006) (Beaza, 2003)

2.3 LA WEB**2.3.1 Introducción**

Crece más rápido que la capacidad de ella misma para detectar sus cambios. Sus conexiones son dinámicas y muchas de ellas quedan obsoletas sin ser nunca actualizadas. El contenido de la Web es hoy de miles de terabytes (un terabyte o También es un billón de megabytes) de texto, imágenes, audio y video. Para aprovechar esta gran base de datos no estructurada es importante poder buscar información en ella, adaptándose al crecimiento continuo de la Web. Al igual que Internet, la red de computadores que interconecta el globo, que ya sobrepasó los 430 millones de computadores conectados en más de 220 países durante 2006, los servidores Web también crecen en forma exponencial desde 1993 (un servidor Web es el software que administra un sitio Web). Lamentablemente nadie sabe su número exacto, pues no es posible a partir de un nombre de dominio saber si es o no un servidor Web (la mayoría comienza con www, pero muchos lugares no siguen esta convención). Además un mismo computador puede manejar distintos servidores y también existen servidores virtuales (un mismo servidor Web puede manejar lógicamente otros servidores). En el año 2000, el número de servidores sobrepasó los 10 millones y en mayo de 2007 ya llegaban a los 120 millones. (Boccaletti, 2006)

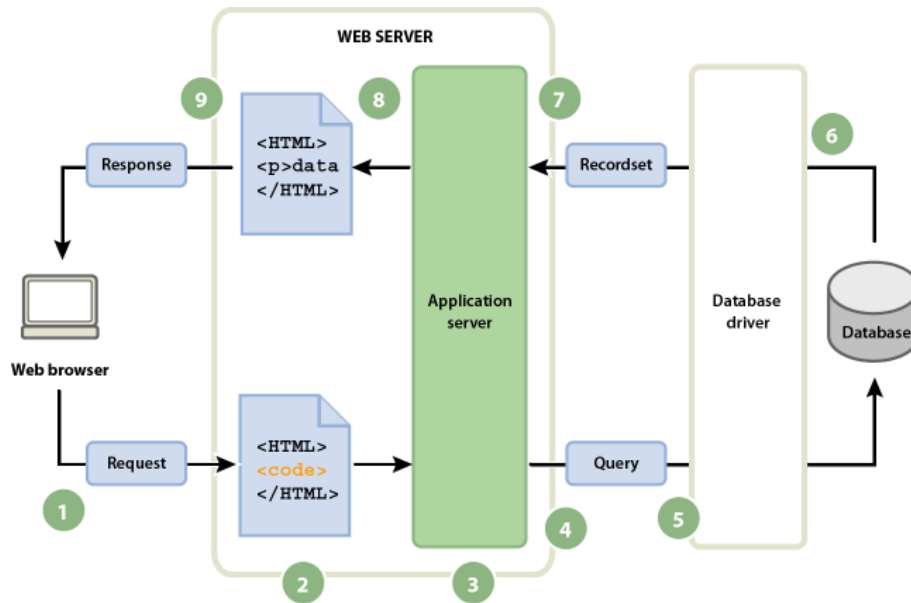


Figura 2.1 Principales características de la Web.

2.3.2 Conceptos Básicos

La Web es compleja: hay páginas estáticas y dinámicas, públicas y privadas, con o sin metadatos, que representan la semántica de la Web, tal como se muestra en la Figura 2.1. Las páginas estáticas son aquellas que existen todo el tiempo en un archivo en algún servidor Web. Las páginas dinámicas son aquellas que se crean cuando una persona interactúa con un servidor Web, por ejemplo la respuesta a una consulta en un buscador o el resultado de rellenar un formulario en un sitio de comercio electrónico. Actualmente, la mayor parte de la Web es dinámica, y como en algunos sitios se puede generar un número no acotado de páginas dinámicas (por ejemplo, un calendario), la Web que podemos crear es infinita. Las páginas públicas son las que todas las personas pueden ver y las privadas son las que están protegidas por una clave o se encuentran dentro de una Intranet. Como cada persona tiene acceso a distintas páginas privadas, la Web pública depende del observador. En particular cada buscador refleja una Web pública distinta. Algunos sitios tienen información semántica que ayuda a los buscadores y se estima que un 5% de ellos tiene

información fidedigna. Sin embargo, más son los sitios que tienen información falsa, lo que se llama spam de Web. (Boccaletti, 2006)

2.3.3 Analizando el uso

Analizar las bitácoras de acceso (logs) a un sitio Web es lo más interesante desde el punto de vista comercial. Por ejemplo, una página que nunca es visitada tal vez no tiene razón de ser, o si páginas muy visitadas no están en los primeros niveles, esto sugiere mejorar la organización y navegación del sitio. Por lo tanto, es importante detectar patrones de acceso y sus tendencias. Esta detección puede ser genérica o para un usuario específico (lo que permite personalizar sitios en forma dinámica) y los resultados pueden ser usados para recomendar servicios o productos. El problema principal en este caso es poder diferenciar a los usuarios y cuándo se conectan o desconectan (determinar sesiones). (Beaza, 2003)

2.3.4 Caracterizando la web estructura y visibilidad

Cuántas referencias tiene una página HTML?, (HTML es un acrónimo para Hyper Text Markup Language; el lenguaje usado para estructurar páginas Web). Más del 75% de las páginas tiene al menos una referencia, y en promedio cada una tiene entre 5 y 15 referencias. La mayoría de estas referencias son a páginas en el mismo servidor. De hecho, la conectividad entre sitios distintos no es muy buena. En particular, la mayoría de las páginas no son referenciadas por nadie y las que sí son referenciadas, lo son por páginas en el mismo servidor. Considerando sólo referencias externas (entre sitios distintos), más del 80% de las páginas tienen menos de 10 referencias a ella. Otros sitios son muy populares, teniendo decenas de miles de referencias a ellos. Si contamos sitios que referencian a sitios, aparecen ODP (www.dmoz.org), el directorio abierto, y el directorio de Yahoo! en los dos primeros lugares. Estos sitios son los que conectan la Web. Por otro lado, hay algunos sitios que no son

referenciados por nadie (están porque fueron incluidos mediante el envío directo de una dirección Web a Yahoo! u otros buscadores, pero que realmente son islas dentro de la Web). En este mismo sentido, las páginas personales también se pueden considerar como entes aislados en muchos casos. Asimismo, la mayoría de los sitios (80%) no tiene ninguna referencia hacia páginas en otros servidores. Esto significa que una minoría de los servidores mantiene toda la carga navegacional de la red. Estadísticas recientes indican que el 1% de los servidores contienen aproximadamente el 50% del volumen de datos de la Web, que se estimaba mayor a 20,000 millones de páginas durante 2006.

2.3.5 Estructura

Para analizar la estructura de la Web se buscan las partes del grafo que están conectadas entre sí. El estudio ya mencionado, y el único realizado a nivel global, muestran que el núcleo o centro de la Web lo constituían más de 56 millones de páginas, existiendo un camino para ir de cualquier página a otra, con un largo máximo (diámetro) de al menos 28. En otras palabras, el camino más corto entre dos páginas en el peor caso implicaba visitar 28 de ellas. Esto contrasta con el modelo del mundo pequeño mencionado al comienzo que predecía un diámetro máximo de 20 páginas para toda la Web. En la práctica se encontraron caminos hasta de largo 900, lo que indica que el diámetro de la Web es mucho mayor. De todos modos, este número no es tan grande considerando que son cientos de millones de páginas. La Figura 2.2 muestra la estructura de la Web de acuerdo al estudio mencionado. A la izquierda había 43 millones de páginas desde las cuales se podía llegar al centro, pero no viceversa. Del mismo modo, a la derecha había otras 43 millones que podían ser accedidas desde el centro, pero que no enlazaban páginas del núcleo. Alrededor de estos dos grupos hay tentáculos que contienen 44 millones de páginas y que son caminos sin salida, con la excepción de algunos tubos, que conectan el grupo de la izquierda con el de la derecha. Finalmente, tenemos 17 millones de páginas que están agrupadas en islas que no están conectadas al centro de la Web. Es muy simple: estos son sitios Web que fueron directamente enviados al buscador y por lo tanto están en su índice aunque el resto del mundo no las conozca. Los autores del estudio no hacen ninguna interpretación sobre esta estructura. En las investigaciones que

hemos realizado en Chile, que muestran una estructura similar, el grupo de la izquierda son páginas más nuevas que aún no son demasiado conocidas y que si tienen éxito pasarán al centro de la Web, donde están las páginas consolidadas. En cambio, en el grupo de la derecha son páginas antiguas que no enlazan al centro de la Web porque en su época esas páginas no existían, pero sí fueron enlazadas por nuevas páginas. También incluyen muchos sitios Web que no tienen enlaces externos pero que se han preocupado de tener un enlace desde un buen sitio, por ejemplo vía enlaces publicitarios. (Boccaletti, 2006)

Hemos encontrado que la proporción de sitios que son islas es muy alta, mucho mayor que en el estudio original, gracias a que conocemos todos los dominios.

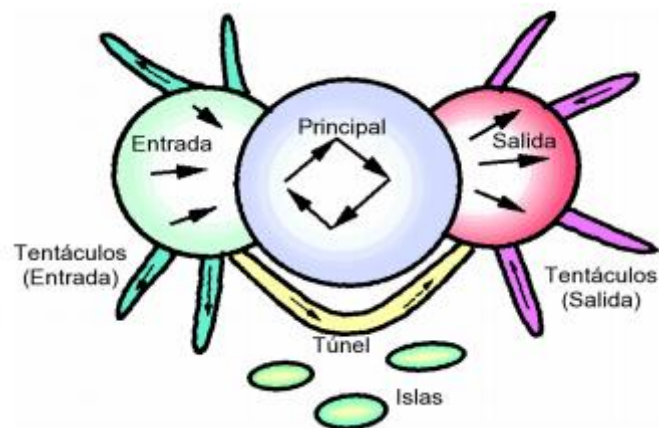


Figura 2.2 Estructura grafo de la Web.

2.4 INGENIERÍA DE REQUERIMIENTOS

La ingeniería de requisitos es una disciplina de la Ingeniería de software. Esta disciplina considera diferentes líneas de trabajo, pero una de las más importantes es la gestión de requisitos, la cual se encarga de proveer la dirección y alcance del proyecto. Los requisitos deben ser la base de cualquier desarrollo de software. La obtención de una especificación de requisitos de alta calidad es fundamental para asegurar que el software se corresponde con las necesidades

del cliente. En el análisis de requisitos se investiga la parte del mundo real (también llamado universo de discurso o mini mundo) que se va a modelar para tener en cuenta todas las necesidades de los usuarios finales y así dejarlas documentadas de la forma más completa posible. (Anaya, 2002)

2.4.1 Requerimiento/requisito

Normalmente, un tema de la Ingeniería de Software tiene diferentes significados. De las muchas definiciones que existen para requerimiento, a continuación se presenta la definición que aparece en el glosario de la IEEE. (1) Una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo. (2) Una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal. (3) Una representación documentada de una condición o capacidad como en (1) o (2). (Richard, 1997)

2.4.2 Características de los requerimientos

Los requerimientos deben especificarse antes de intentar comenzar la construcción del producto, sin ellos no podrá ser posible llevar a cabo las etapas de diseño y construcción correctamente. Los mismos pueden verse como una declaración abstracta de alto nivel de un servicio que el sistema debe proporcionar, como una definición matemática detallada y formal. Los requisitos cumplen una doble función ya que son la base para una oferta de contrato, por lo tanto deben estar abiertos a la interpretación. Además son la base para redactar el contrato en sí mismo. Los requisitos una vez establecidos y documentados, sufren cambios continuos, en este sentido, no se trata la obtención ni el análisis de los mismos, se trata de su gestión, es decir, el seguimiento respecto a los cambios que se generan durante el ciclo de vida del proyecto y las herramientas de gestión de requisitos que auxilian y/o automatizan estas tareas. El uso de herramientas para auxiliar la gestión de requisitos se ha convertido en un aspecto importante de la Ingeniería de Sistemas y el diseño. Considerando el tamaño y la complejidad del desarrollo, las herramientas vienen siendo algo esencial. Las herramientas que los gestores de requisitos utilizan

para automatizar los procesos de Ingeniería de Requisitos han disminuido el trabajo duro en el mantenimiento de requisitos, añadiendo beneficios significativos al reducir errores. (Loucopoulos, 1995). Las características de un requerimiento son sus propiedades principales. Un conjunto de requerimientos en estado de madurez, deben presentar una serie de características tanto individualmente como en grupo. A continuación se presentan las más importantes.

- **Necesario:** Un requerimiento es necesario si su omisión provoca una deficiencia en el sistema a construir, y además su capacidad, características físicas o factor de calidad no pueden ser reemplazados por otras capacidades del producto o del proceso.
- **Conciso:** Un requerimiento es conciso si es fácil de leer y entender. Su redacción debe ser simple y clara para aquellos que vayan a consultarlo en un futuro.
- **Completo:** Un requerimiento está completo si no necesita ampliar detalles en su redacción, es decir, si se proporciona la información suficiente para su comprensión.
- **Consistente:** Un requerimiento es consistente si no es contradictorio con otro requerimiento.
- **No ambiguo:** Un requerimiento no es ambiguo cuando tiene una sola interpretación. El lenguaje usado en su definición, no debe causar confusiones al lector.
- **Verificable:** Un requerimiento es verificable cuando puede ser cuantificado de manera que permita hacer uso de los siguientes métodos de verificación, inspección, demostración o pruebas. (Booch, 2000)

2.4.3 Tipos de requisitos

Los requerimientos pueden dividirse en varios tipos dentro de ellos, se hará referencia a los siguientes:

- Requisitos de usuario
- Requisitos del sistema
- Requisitos funcionales
- Requisitos no funcionales

- **Requisitos de usuario**

Declaraciones en lenguaje natural y en diversos diagramas de los servicios del sistema y de las restricciones bajo las que debe operar.

- 1.- El sistema debe permitir representar y acceder a archivos externos creados por otras herramientas.
2. Sentencias muy generales sobre lo que el sistema debería hacer.

- **Requisitos del sistema**

Un documento estructurado que determina las descripciones detalladas de los servicios de sistema. Escrito como contrato entre el cliente y el contratista.

- 1.- El usuario deberá poder definir el tipo de un nuevo archivo externo.
- 2.- Cada tipo de archivo tendrá una herramienta asociada, que se le aplicará.
- 3.- Cada tipo de archivo se representará con un icono específico.
- 4.- El usuario deberá poder definir el icono que representa un tipo de archivo externo.

5.- Cuando el usuario selecciona un icono que representa un archivo externo, el efecto es aplicar la herramienta asociada con este tipo de archivo al archivo representado por el icono seleccionado.

- **Requisitos funcionales**

Declaración de los servicios que el sistema debe proporcionar, cómo debe reaccionar a una entrada particular y cómo se debe comportar ante situaciones particulares. Describen la funcionalidad del sistema, y dependen del tipo de software, del sistema a desarrollar y de los usuarios del mismo.

Por lo general se describen mejor a través del modelo de Casos de uso y los Casos de uso como tal. Por lo tanto los requerimientos funcionales especifican el comportamiento de entrada y salida del sistema y surgen de la razón fundamental de la existencia del producto.

- **Requisitos no funcionales**

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Restricciones que afectan a los servicios o funciones del sistema, tales como restricciones de tiempo, sobre el proceso de desarrollo, estándares, etc. Los requerimientos no funcionales tienen que ver con características que de una u otra forma puedan limitar el sistema, como por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, etc. Algunas propiedades de los requerimientos no funcionales que hacen al producto atractivo, usable, rápido o confiable, son las siguientes:

Clasificación de los requisitos no funcionales

- **Requisitos del producto:** Especifican el comportamiento del producto obtenido, velocidad de ejecución, memoria requerida, y porcentaje de fallos aceptables.
- **Requisitos organizacionales:** Son una consecuencia de las políticas y procedimientos existentes en la organización, procesos estándar utilizados, de fechas de entrega, y documentación a entregar.
- **Requisitos externos:** Presentan factores externos al sistema y a su proceso de desarrollo, interoperabilidad del sistema con otros, requisitos, legales, y éticos.
- **Requerimientos de apariencia o interfaz externa** Ejemplo: Muy legible, Simple de usar, Profesional o tipo ejecutivo.
- **Requerimientos de Usabilidad** Ejemplo: Facilidad de uso por personas que hablen otros idiomas distintos al del país donde el producto fue creado, Accesibilidad para personas discapacitadas, Consistencia en la interfaz de usuario, Documentación de usuario.
- **Requerimientos de Rendimiento** Ejemplo: Velocidad de procesamiento o cálculo, Eficiencia, Disponibilidad, Tiempo de respuesta.
- **Requerimientos de Soporte** Ejemplo: Adaptabilidad, Mantenimiento. **Requerimientos de Portabilidad** Ejemplo: El producto podrá ser usado bajo el sistema operativo Linux.
- **Requerimientos de Seguridad Confidencialidad:** La información manejada por el sistema está protegida de acceso no autorizado y divulgación.
- **Integridad:** la información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes.

- **Disponibilidad:** Significa que los usuarios autorizados se les garantizará el acceso a la información y que los dispositivos o mecanismos utilizados para lograr la seguridad no ocultarán o retrasarán a los usuarios para obtener los datos deseados en un momento dado.
- **Requerimientos de confiabilidad:** Frecuencia y severidad de los fallos, Protección contra fallos, Recuperación, Predicción de fallos, Tiempo medio entre fallos.
- **Requerimientos de Software:** Ejemplo: Sistema Operativo Windows 95 o Superior; Máquina Virtual de Java versión 1.3 o Superior; etc.
- **Requerimientos de Hardware:** Ejemplo: se requiere disponer de un MODEM estándar o una tarjeta digitalizadora de video, etc.

A pesar de las diferentes características que nos brindan los requerimientos, existen dificultades para recolectar los requisitos, las cuales no nos permiten elegir los requerimientos con la calidad necesaria; ya que estos pueden relacionarse unos con otros y a su vez con otras partes del proceso. Pero aun así, se plantea que sin el levantamiento de requisitos no se podrían desarrollar procesos que son de vital importancia para el desarrollo del software. Los requisitos constituyen el enlace entre las necesidades reales de los clientes, usuarios y otros participantes vinculados al sistema. (Pressman, 2009)

2.4.4 Actividades para la Ingeniería de Requerimientos

Las actividades de la Ingeniería de Requisitos más comunes son:

- ✓ Estudio de Viabilidad
- ✓ Determinación de Requisitos
- ✓ Análisis de Requisitos

- ✓ Especificación de Requisitos (ERS)
- ✓ Validación de Requisitos
- ✓ Gestión de Requisitos

- **Estudio de viabilidad**

El estudio de viabilidad permite decidir si el sistema propuesto es conveniente. Es un estudio rápido y orientado a conocer. Además tiene en cuenta si el sistema contribuye a los objetivos de la organización, si el sistema se puede realizar con la tecnología actual y con el tiempo y el coste previsto, y si el sistema puede integrarse con otros existentes.

- **Determinación de requisitos**

Determinación (o extracción o Elicitación) de requisitos, es el proceso mediante el cual los usuarios descubren, revelan, articulan y comprenden los requisitos que desean. En esta etapa, se trata de descubrir los requisitos y personal técnico trabaja con los clientes y usuarios para descubrir el dominio de la aplicación, los servicios que se deben proporcionar y las restricciones. Puede implicar a usuarios finales, encargados, ingenieros implicados en el mantenimiento, expertos del dominio, etc. Son los llamados participantes (stakeholders).

- **Análisis de requisitos**

El proceso de razonamiento sobre los requisitos obtenidos en la etapa anterior, detectando y resolviendo posibles inconsistencias o conflictos, coordinando los requisitos relacionados entre sí, etc.

- **Especificación de Requisitos (ERS)**

La especificación de requisitos de software es la actividad en la cual se genera el documento, con el mismo nombre, que contiene una descripción completa de las necesidades y funcionalidades del sistema que será desarrollado; describe el alcance del sistema y la forma en como hará sus

funciones, definiendo los requerimientos funcionales y los no funcionales. En la SRS se definen todos los requerimientos de hardware y software, diagramas, modelos de sistemas y cualquier otra información que sirva de soporte y guía para fases posteriores.

- **Validación de requisitos**

El proceso de confirmación, por parte de los usuarios, de que los requisitos especificados son válidos, consistentes, y completos.

- **Gestión de Requisitos**

Es el proceso de manejar los requisitos que cambian durante el desarrollo del sistema. El proceso de Ingeniería de Requisitos se adapta a los diferentes modelos de procesos de Ingeniería de Software como pueden ser, de cascada, espiral, prototipazo, transformacional, etc.

2.4.5 Personal involucrado en la Ingeniería de Requerimientos

Realmente, son muchas las personas involucradas en el desarrollo de los requerimientos de un sistema. Es importante saber que cada una de esas personas tienen diversos intereses y juegan roles específicos dentro de la planificación del proyecto; el conocimiento de cada papel desempeñado, asegura que se involucren a las personas correctas en las diferentes fases del ciclo de vida, y en las diferentes actividades de la IR. No conocer estos intereses puede ocasionar una comunicación poco efectiva entre clientes y desarrolladores, que a la vez traería impactos negativos tanto en tiempo como en presupuesto. Los roles más importantes pueden clasificarse como sigue:

- **Usuario final:** Son las personas que usarán el sistema desarrollado. Ellos están relacionados con la usabilidad, la disponibilidad y la fiabilidad del sistema; están familiarizados con los procesos específicos que debe

realizar el software, dentro de los parámetros de su ambiente laboral. Serán quienes utilicen las interfaces y los manuales de usuario.

- **Usuario Líder:** Son los individuos que comprenden el ambiente del sistema o el dominio del problema en donde será empleado el software desarrollado. Ellos proporcionan al equipo técnico los detalles y requerimientos de las interfaces del sistema.
- **Personal de Mantenimiento:** Para proyectos que requieran un mantenimiento eventual, estas personas son las responsables de la administración de cambios, de la implementación y resolución de anomalías. Su trabajo consiste en revisar y mejorar los procesos del producto ya finalizado.
- **Analistas y programadores:** Son los responsables del desarrollo del producto en sí; ellos interactúan directamente con el cliente.
- **Personal de pruebas:** Se encargan de elaborar y ejecutar el plan de pruebas para asegurar que las condiciones presentadas por el sistema son las adecuadas. Son quienes van a validar si los requerimientos satisfacen las necesidades del cliente.
- **Otras personas** que pueden estar involucradas, dependiendo de la magnitud del proyecto, pueden ser: administradores de proyecto, documentadores, diseñadores de base de datos, entre otros.

2.4.6 Análisis comparativo de las técnicas de Ingeniería de Requerimientos.

En la Ingeniería de Requisitos se describen técnicas que permiten la captura requisitos de software, la recopilación de la información y en qué casos es adecuada usar cada cual. A continuación se hace un análisis de estas técnicas. (Sommerville, 1997)

- **Técnica Entrevistas**

Características.

Forma de conversación, no de interrogación.

Ocupan un lugar preponderante de acuerdo al tiempo que ocupan y el objetivo que tienen.

Mayor fuente de información del analista

Basadas en un cuestionario rígido o una guía que las orienta hacia puntos bien definidos.

Entre sus ventajas se presenta necesidades de forma directa y se verifica si las preguntas fueron interpretadas correctamente. Oportunidad para conocer el grado de aceptación o no entre los usuarios hacia el sistema que se desea diseñar. Mediante ellas se obtiene una gran cantidad de información correcta a través del usuario. Pueden ser usadas para obtener un pantallazo del dominio del problema. Son flexibles y permiten combinarse con otras técnicas (Sommerville, 1997).

Realización de las Entrevistas

Los pasos:

Preparación.

Ejecución.

Recapitulación.

Acordar una cita por anticipado con las personas que se entrevistarán.

Avisar a los entrevistados sobre la naturaleza de la entrevista.

Planear una entrevista común por no más de una hora.

Prepararla conociendo de antemano a los individuos que se van a entrevistar.

Familiarizarse con el tema y preparar un conjunto apropiado de preguntas.

- **Técnica Cuestionarios**

Características

Permiten obtener información de un gran número de personas en corto tiempo, sin que estas deban estar presentes.

Son recomendables cuando:

Se requiere una pequeña cantidad de información de un gran número de personas en un corto periodo de tiempo. La información se desea consolidar en tablas estadísticas. Usuarios geográficamente dispersos.

Antes de aplicar un Cuestionario: Probarlo en un grupo pequeño para detectar otros problemas. Analizar las respuestas de prueba para asegurar que el análisis se pueda llevar a cabo con los datos recopilados.

- **Técnica: Lluvia de Ideas**

Sus ventajas son los diferentes puntos de vista y las confusiones en cuanto a terminología, son aclarados por expertos.

Ayuda a desarrollar ideas unificadas basadas en la experiencia de un experto.

- **Técnica: Casos de Uso**

Ventajas: Representan los requerimientos desde el punto de vista del usuario.

Identifica requerimientos estancados, dentro de un conjunto de requerimientos.

Desventaja: En sistemas grandes, toma mucho tiempo definir todos los casos de uso. El análisis de calidad depende de la calidad con que se haya hecho la descripción inicial.

2.4.7 Importancia de la Ingeniería de Requerimientos

Los principales beneficios que se obtienen de la Ingeniería de Requerimientos son:

- **Permite gestionar las necesidades del proyecto en forma estructurada:** Cada actividad de la IR consiste de una serie de pasos organizados y bien definidos.
- **Mejora la capacidad de predecir cronogramas de proyectos, así como sus resultados:** La IR proporciona un punto de partida para controles subsecuentes y actividades de mantenimiento, tales como estimación de costos, tiempo y recursos necesarios.
- **Disminuye los costos y retrasos del proyecto:** Muchos estudios han demostrado que reparar errores por un mal desarrollo no descubierto a tiempo, es sumamente caro.
- **Mejora la comunicación entre equipos:** La especificación de requerimientos representa una forma de consenso entre clientes y desarrolladores. Si este consenso no ocurre, el proyecto no será exitoso.
- **Mejora la calidad del software:** La calidad en el software tiene que ver con cumplir un conjunto de requerimientos (funcionalidad, facilidad de uso, confiabilidad, desempeño, etc.).
- **Evita rechazos de usuarios finales:** La ingeniería de requerimientos obliga al cliente a considerar sus requerimientos cuidadosamente y revisarlos dentro del marco del problema, por lo que se le involucra durante todo el desarrollo del proyecto.

2.4.8 Gestión de Requisitos. Principales características

La Gestión de Requisitos es un componente vital en el desarrollo de un proyecto software ya que es una de las actividades de la Ingeniería de Requisitos más importantes. Los requisitos se inician cuando empieza un proyecto en las etapas de análisis y especificación de requisitos, posteriormente, dichos requisitos en el ciclo de vida de un proyecto pueden ser modificados por lo que

se establece el concepto de Gestión de Requisitos, que es el tratamiento y control de las actualizaciones y cambios a los mismos. Debido a que un proyecto informático es susceptible de cambios, habría que proceder a su actualización o a la incorporación de nuevas funcionalidades o eliminar otras, esto obliga a mantener controlado y documentado el producto. Los cambios de requisitos deben ser gestionados para asegurar que la calidad de los mismos se mantenga, los problemas suscitados por los cambios de requisitos podrían incurrir en altos costos, siendo el requisito factor crítico de riesgo. Más formalmente el Manejo de Requisitos es una forma sistemática de descubrir, organizar y documentar los requisitos del sistema. Además es el proceso que establece y mantiene un consenso entre el cliente y el grupo del proyecto en el cambio de los requisitos del sistema.

El término Gestión de Requisitos incluye:

- Técnicas para Descubrimiento/Recogida de Requisitos
- Recoger las peticiones del usuario y determinar las verdaderas necesidades de éste.
- Técnicas de Análisis
- Especificación y verificación
- Manejo de Requisitos

Tareas principales de la Gestión de Requisitos.

Durante el proceso de la gestión de requisitos, hay que planear algunas actividades, dentro de las que se pueden mencionar las siguientes: la identificación de los requisitos, en proceso de gestión de los cambios, las políticas de trazabilidad, la cantidad de información sobre las relaciones entre los requisitos que se mantiene, entre otras.

Actividades y su descripción:

- **Recolección:** Recolección y documentación de requisitos es una actividad de comunicación iterativa entre clientes, gerentes y practicantes,

para descubrir, definir, refinar y registrar una representación precisa de los requisitos del producto. Varios métodos son utilizados para la recolección de requisitos. Algunos análisis iniciales como es la agrupación, categorización, priorización son desarrollados durante esta actividad.

- **Documentación:** Después que los requisitos han sido recolectados, hay que analizarlos a detalle y documentarlos en una especificación de requisitos. El resultado de la especificación de requisitos y de cualquier especificación de requisitos de componentes hardware/software derivado sirve como registro de convenio con el cliente y compromiso con el proveedor. Estas especificaciones son rastreadas utilizando una matriz de trazabilidad de requerimientos y son sujetos a verificación y gestión de cambio a través del ciclo de vida del producto.
- **Verificación:** Una vez que la especificación de requisitos ha sido desarrollada, los requisitos son verificados. La verificación de requisitos es un proceso para asegurar que la especificación de requisito del producto es una representación exacta de las necesidades del cliente. Este proceso también asegura que los requisitos sean trazados y verificados a través de varias fases del ciclo de vida; particularmente en el diseño, implementación y pruebas. Además, todos estos requerimientos deben ser trazados al diseño, implementación y pruebas para asegurarse que los requerimientos han sido satisfechos.
- **Gestión de Cambios:** Gestión de cambios es un proceso formal para identificar, evaluar, trazar y reportar cambios propuestos y aprobados a la especificación del producto. Como el proyecto va evolucionando, los requerimientos pueden cambiar o expandirse para ajustar algunas modificaciones en el alcance o diseño del proyecto. Un proceso de gestión de cambios proporciona un rastreo completo y preciso de todos los cambios que son pertinentes al proyecto. El proceso del ciclo de vida de la Gestión de Requisitos, debe ser flexible y adaptable para reunir las necesidades del proyecto. Las características del alcance e

implementación del proceso del ciclo de vida de la Gestión de Requisitos en un proyecto, variará dependiendo de algunos factores claves. (Pressman, 2009)

2.5 PROCESO UNIFICADO ÁGIL (AUP)

2.5.1 Agile Unified Process (AUP)

El Proceso Unificado Agil de Scott Ambler o Agile Unified Process (AUP) en inglés es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP. El AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas (test driven development - TDD), 14 Modelado Agil, Gestión de Cambios Agil, y Refactorización de Base de Datos para mejorar la productividad.

El proceso unificado (Unified Process o UP) es un marco de desarrollo software iterativo e incremental. A menudo es considerado como un proceso altamente ceremonioso porque especifica muchas actividades y artefactos involucrados en el desarrollo de un proyecto software. Dado que es un marco de procesos, puede ser adaptado y la más conocida es RUP (Rational Unified Process) de IBM.

AUP se preocupa especialmente de la gestión de riesgos. Propone que aquellos elementos con alto riesgo obtengan prioridad en el proceso de desarrollo y sean abordados en etapas tempranas del mismo. Para ello, se crean y mantienen listas identificando los riesgos desde etapas iniciales del proyecto. Especialmente relevante en este sentido es el desarrollo de prototipos ejecutables durante la base de elaboración del producto, donde se demuestre la validez de la arquitectura para los requisitos clave del producto y que determinan los riesgos técnicos.

El proceso AUP establece un Modelo más simple que el que aparece en RUP por lo que reúne en una única disciplina las disciplinas de Modelado de Negocio, Requisitos y Análisis y Diseño. El resto de disciplinas (Implementación, Pruebas, Despliegue, Gestión de Configuración, Gestión y Entorno) coinciden con las restantes de RUP.

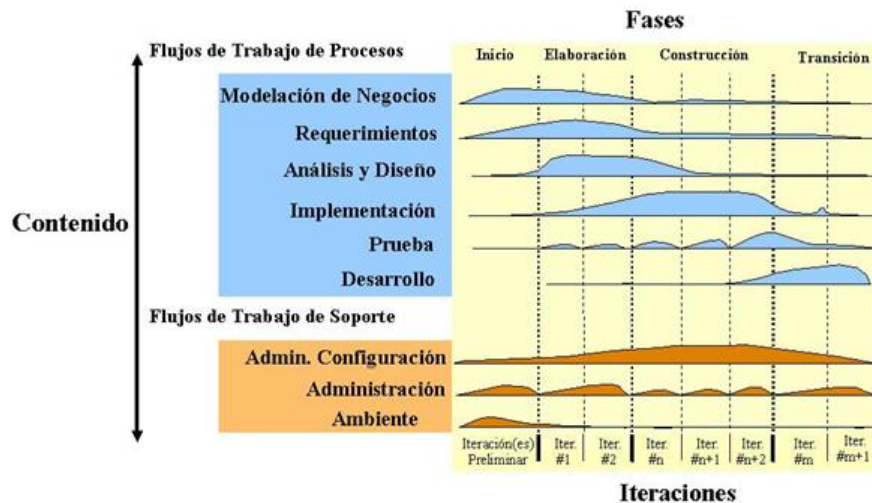


Figura 2.3 Fases de Iteraciones de AUP (ciclo de vida).

2.5.2 Disciplinas de AUP

Las disciplinas son ejecutadas de una forma iterativa, definiendo las actividades, las cuales el equipo de desarrollo ejecuta para construir, validar y liberar software funcional, el cual cumple con las necesidades de los involucrados. Las disciplinas son:

a. Modelado. El objetivo de esta disciplina es entender del negocio de la organización, el problema de dominio que se abordan en el proyecto, y determinar una solución viable para resolver el problema de dominio.

b. Implementación. El objetivo de esta disciplina es transformar su modelo (s) en código ejecutable y llevar a cabo un nivel básico de las pruebas, en particular, la unidad de prueba.

c. Pruebas. El objetivo de esta disciplina es ejecutar una objetiva evaluación para asegurar la calidad. Esto incluye la detección de defectos, validaciones de

que el sistema funciona como fue diseñado, y verificar que se cumplan los requerimientos.

d. Despliegue. El objetivo de esta disciplina es planificar la entrega del proyecto de desarrollo y ejecutar el plan, para dejar disponible el sistema al usuario final.

e. Administración de la Configuración. La meta de esta disciplina es manejar el acceso a sus productos de trabajo de proyecto. Esta no solo incluye el retraso de versiones del trabajo del producto en el tiempo, sino que también el control y administración de los cambios estos productos.

f. Administración de Proyecto. El objetivo de esta disciplina es manejar al acceso a sus productos de trabajo de proyecto. Esta incluye la administración del riesgo, administración del personal (asignación de tareas, rastreo del progreso, etc.), y coordinación con personas y sistemas fuera del alcance del proyecto para asegurar su liberación a tiempo y dentro del presupuesto.

g. Ambiente. El objetivo de esta disciplina es soportar el resto del esfuerzo asegurando que el proceso apropiado, las guías (normas y directrices), y herramientas (hardware y software) estén disponibles para cuando el equipo las necesite.

2.5.3 Fases de AUP

Las cuatro fases en las que divide el ciclo de vida del proyecto son:

- Fase de iniciación
- Fase de elaboración
- Fase de construcción

- Fase de transición

a. Inicio

El objetivo principal de la fase de iniciación es archivar el consenso de los interesados del proyecto en relación a los objetivos del proyecto para obtener el financiamiento. Las principales actividades de esta fase incluyen:

1. Definir el alcance del proyecto. Esto incluye la definición, aun alto nivel, de que es lo que hará el sistema. Es igualmente importante también definir qué es lo que el sistema no va hacer. Aquí se establecen los límites desde donde el equipo 17 operara. Esto suele tomar la forma de una lista de características de alto nivel y/o el punto de casos de uso.

2. Estimación de costos y calendario. En un nivel alto, el calendario y el costo del proyecto son estimados. Estimaciones generales son realizadas en iteraciones de fases posteriores de fases posteriores, más específicamente es implementado en las fases tempranas de la Elaboración. Esto no debe interpretarse en el sentido de que todo el proyecto es planeado en este punto. Como en todas las planificaciones, estas tareas que van a ser completadas en un futuro cercano y son detalladas con más precisión y con una gran confianza mientras que las tareas bajo la línea son entidades para ser estimadas con uno que no es posible programar todo un proyecto, en su pistoletazo de salida con cualquier grado aceptable de confianza con un margen de error más grande. Esto ha sido (finalmente) reconocido por la mayoría de las industrias de que no es posible programar un proyecto completo de un solo con algún grado de aceptable desacuerdo. Lo mejor que se puede hacer es planificar para el corto plazo y precisar a largo plazo lo mejor que se pueda.

3. Definición de riesgos. Los riesgos del proyecto son primeramente definidos. La administración del riesgo es importante en proyectos de AUP. La lista de riesgos es una complicación en vivo que cambiara en el tiempo cuando los riesgos serán identificados, mitigados, evitados y/o materializados o exterminados. El control de riesgos del proyecto, como los riesgos de más alta prioridad, maneja la programación de las iteraciones. Los riesgos más altos, por ejemplo, son dirigidos en iteraciones más tempranas que los riesgos de menor prioridad.

4. Determinar la factibilidad del proyecto. Su proyecto debe tener sentido desde la perspectiva técnica, operacional y del negocio. En otras palabras, se debe ser capaz de crearlo, una vez desplegado debe ser capaz de correrlo, y debe tener un sentido económico para hacer estos aspectos. Si su proyecto no es viable, este debe ser cancelado.

5. Preparar el entorno del proyecto. Esto incluye la reserva de las áreas de trabajo para el equipo. Solicitar del personal que se necesitara, obtenido hardware y software que será necesitado después. Además, deberá ajustar AUP para completar las necesidades de su equipo.

Para salir de la etapa de Iniciación su equipo debe terminar el hito de Objetivos del Ciclo de Vida (LCO). El principal aspecto es hacer que el equipo entienda el alcance del proyecto y el esfuerzo requerido y como los usuarios patrocinaran el proyecto. Si el equipo pasa es hito, el proyecto sigue a las fase de Elaboración, en otra forma el proyecto deberá ser redirigido o cancelado.

b. Elaboración

El principal objetivo de la fase de elaboración es probar la arquitectura del sistema a desarrollar. El punto es asegurar que el equipo puede desarrollar un sistema que satisfaga los requisitos, y la menor manera de hacerlo que es la construcción de extremo o extremo del esqueleto de trabajo del sistema conocido como “prototipo de la arquitectura”. Esto es

en realidad un concepto pobre porque mucha gente piensa es deshacerse de los prototipos. En cambio, su significado es software funcional de alto nivel, el cual incluye varios casos de uso de alto de riesgos (a partir de un punto de vista técnico) para demostrar que el sistema es técnicamente factible.

c. Construcción

El objetivo de la fase de Construcción consiste en desarrollar el sistema hasta el punto en que está listo para la pre-producción de pruebas. En las etapas anteriores, la mayoría de los requisitos han sido identificados y la arquitectura del sistema se ha establecido. El énfasis es priorizar y comprender los requerimientos, modelado que ataca una solución y, a luego, la codificación y las pruebas del software. Si es necesario, las primeras versiones del sistema se desarrollan, ya sea interna o externamente, para obtener los comentarios de los usuarios. Para salir de la fase de Construcción su equipo debe pasar el hito de la Capacidad Operativa Inicial (IOC). Si el equipo pasa esta etapa el proyecto pasa de la fase de Transición, de lo contrario puede ser re-dirigido o cancelado.

d. Transición

La fase de Transición se enfoca en liberar el sistema a producción. Deben hacerse pruebas extensivas a los largo de esta fase, incluyendo las pruebas beta. Para finalizar la fase de Transición su equipo debe pasar el hito de Liberación del Producto (PR). El principal problema aquí es el sistema puede ser desplegado segura y eficientemente en producción. Si el equipo pasa ese hito el proyecto se mueve a producción. Si el proyecto fracasa en alguna de las áreas de arriba, el proyecto podría ser redirigido o cancelado (algunos proyectos son tan desastrosos que no querrá ni siquiera instalarlos).

2.6 INGENIERÍA WEB BASADO EN UML (UWE)

El área de Ingeniería Web es relativamente una nueva dirección de la Ingeniería de Software para el desarrollo de Aplicaciones Web. La Ingeniería Web trata varios aspectos, metodologías, herramientas y técnicas que hacen único del desarrollo y construcción de aplicaciones que se ejecutan en la World Wide Web.

Este artículo se enfoca en el aspecto de diseño en Ingeniería Web. Para el desarrollo de modelos conceptuales de aplicaciones Web existen varios métodos de diseño en Ingeniería Web, por ejemplo: OOHDM (Object-Oriented Hypermedia Design Model), WebML (Web Modeling Language) , OO-H (Object Oriented approach) , UWE (UML Web Engineering), entre otros. UWE fue uno de los primeros proyectos usado especialmente para aplicaciones Web.

Desde hace unos años, la World Wide Web se ha convertido en una plataforma para la ejecución de toda clase de aplicaciones que cumplen un sinnúmero de funciones. Partiendo de páginas estáticas, la Web ha evolucionado incorporando elementos de seguridad, optimización, concurrencia y demás requerimientos que son necesarios para crear soluciones sólidas. Sin embargo, el desarrollo de una aplicación Web incluye elementos que no son comunes a una aplicación de escritorio. Esto requiere cambios importantes en la forma de realizar y controlar el proceso de desarrollo. Es decir, pasar de una Ingeniería de Software a una Ingeniería Web. Una de las primeras metodologías desarrolladas fue la Ingeniería Web basada en UML (UWE).

UWE es una metodología que permite especificar de mejor manera una aplicación Web en su proceso de creación mantiene una notación estándar basada en el uso de UML (Unified Modeling Language) para sus modelos y sus métodos, lo que facilita la transición. La metodología define claramente la construcción de cada uno de los elementos del modelo. (Sons, 2006)

En su implementación se deben contemplar las siguientes etapas y modelos:

- **Análisis de requisitos.** Plasma los requisitos funcionales de la aplicación Web mediante un modelo de casos de uso.
- **Modelo de contenido.** Define, mediante un diagrama de clases, los conceptos a detalle involucrados en la aplicación.
- **Modelo de navegación.** Representa la navegación de los objetos dentro de la aplicación y un conjunto de estructuras como son índices, menús y consultas.
- **Modelo de presentación.** Representa las interfaces de usuario por medio de vistas abstractas.
- **Modelo de proceso.** Representa el aspecto que tienen las actividades que se conectan con cada clase de proceso.

Como se hace notar, UWE provee diferentes modelos que permite describir una aplicación Web desde varios puntos de vista abstractos, dichos modelos están relacionados tal como se ilustra en la figura 2.4. Cada uno de estos modelos se representa como paquetes UML, dichos paquetes son procesos relacionados que pueden ser refinados en iteraciones sucesivas durante el desarrollo del UWE.

El análisis de requisitos en UWE se modela con casos de uso. Está conformado por los elementos actor y caso de uso. En este sentido, los actores se utilizan para modelar los usuarios de la aplicación Web.

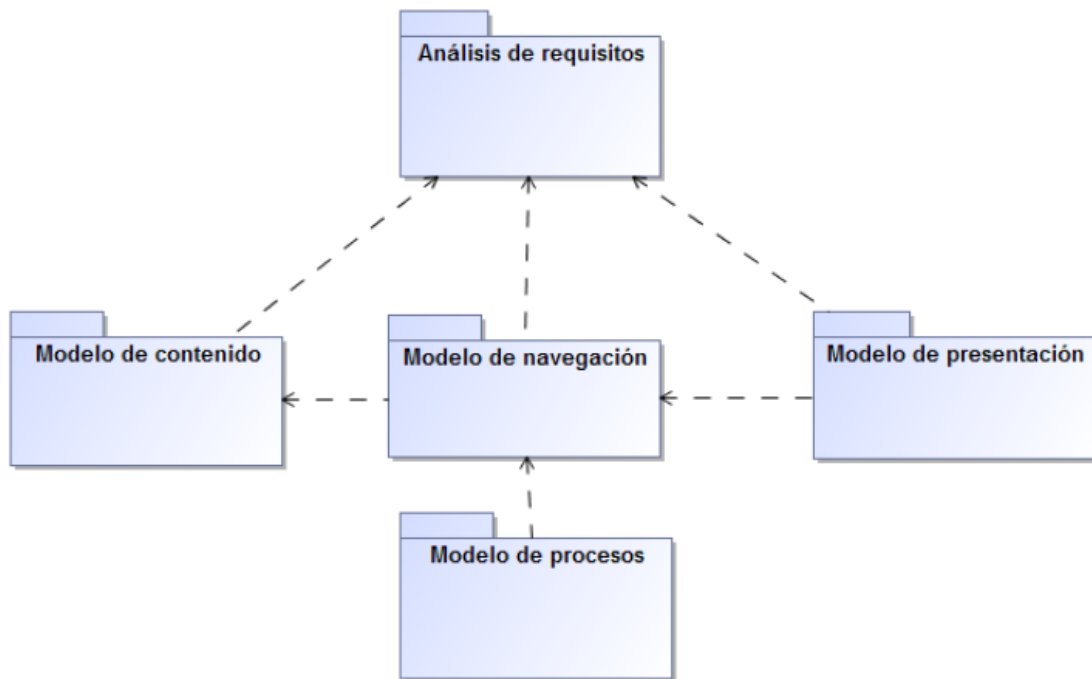


Figura 2.4 Modelos de UWE.

Fuente: Citlali G. Nieves-Guerrero, Juan P. Ucán-Pech (2014)

El modelo de contenido es el modelo conceptual del dominio de aplicación tomando en cuenta los requerimientos especificados en los casos de uso y se representa con un diagrama de clases. Basado en el análisis de requisitos y el modelo de contenido se obtiene el modelo de navegación. Éste se representa con clases de navegación que serán explicados en el caso de estudio de este artículo. Basado en el modelo de navegación y en los aspectos de la interfaz usuario (requisitos), se obtiene el modelo de presentación. Dicho modelo describe la estructura de la interacción del usuario con la aplicación Web. El modelo de navegación puede ser extendido mediante clases de procesos. El modelo del proceso representa el aspecto que tienen las acciones de las clases de proceso. (Springer London, 2008)

2.6.1 Especificando los requisitos

Una de las primeras actividades en la construcción de aplicaciones Web es la identificación de los requisitos, y en UWE se especifican mediante el modelo de requerimientos, que involucra el modelado de casos de uso con UML. El

diagrama de casos de uso está conformado por los elementos actor y caso de uso. Los actores se utilizan para modelar los usuarios de la aplicación Web que para este caso de estudio son los diferentes tipos de usuarios (anónimo, consultor, tutor, alumno) que pueden interactuar con el mismo. Los casos de uso se utilizan para visualizar las diferentes funcionalidades que la aplicación tiene que proporcionar, como son: crear a un nuevo usuario, identificar al usuario, realizar una búsqueda, realizar la composición de un nuevo objeto y guardar el objeto compuesto En la figura 2.5 se ilustra el diagrama de casos de usos para la aplicación web. Es de mencionar que para cada etapa del modelado, UWE provee diferentes estereotipos. La lista de todos los estereotipos que pueden utilizarse en esta etapa se encuentra el Perfil UWE (Profile UWE) del sitio oficial de UWE.

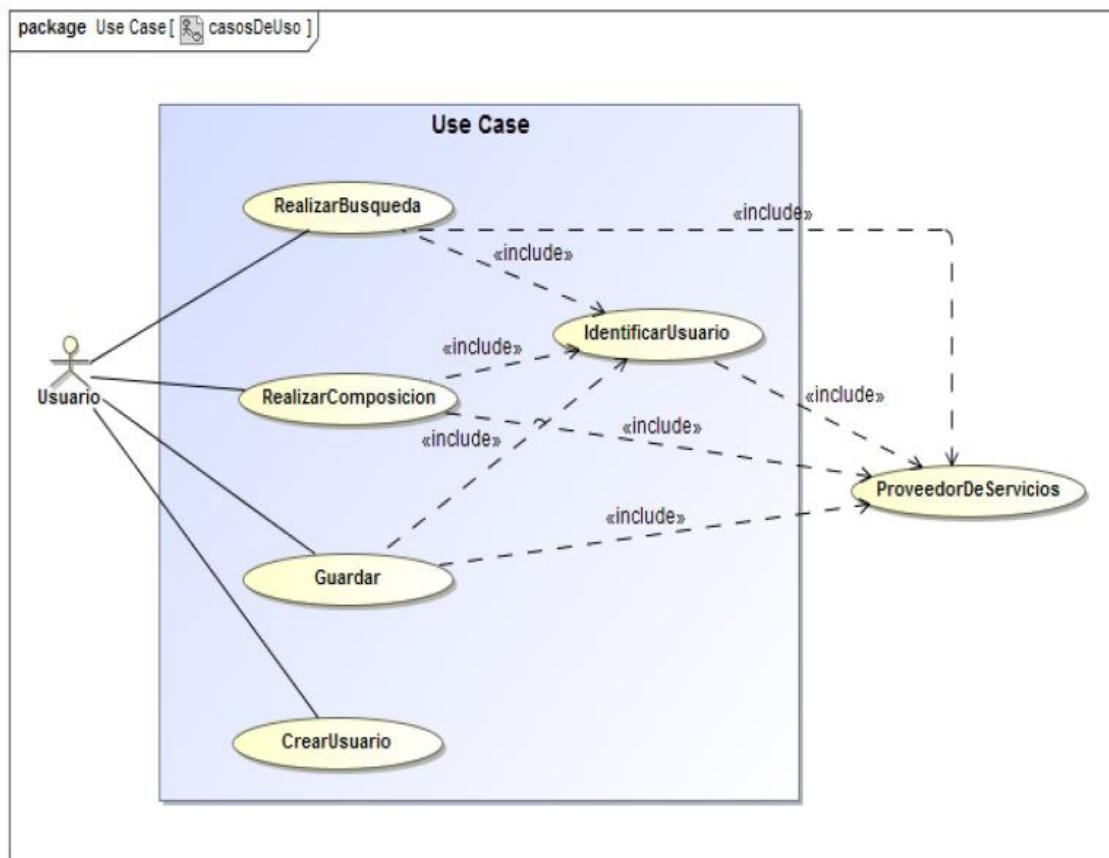


Figura 2.5 Casos de uso.

Fuente: Citlali G. Nieves-Guerrero, Juan P. Ucán-Pech (2014)

El caso de uso "Realizar Búsqueda" es del estereotipo explorar («browsing»). Modelar la búsqueda de los objetos de aprendizaje por medio de las

características de los objetos y de los usuarios para que el sistema pueda proporcionar una recomendación personalizada. El caso de uso "Realizar Composición" es del estereotipo procesar («processing»). Según la lista final seleccionada por el usuario, ejecuta el proceso de composición para conformar un nuevo objeto de mayor nivel de instrucción añadiendo cambios a los metadatos si el usuario así lo decide.

El caso de uso "IdentificarUsuario" es del estereotipo explorar («browsing»). Ejecuta el proceso de inicio de sesión el cual verifica si el usuario proporcionado existe en el sistema. El caso de uso "Guardar" es del estereotipo procesar («processing»). Ejecuta la conversión del objeto compuesto al estándar IEEE-LOM y almacena el objeto compuesto en la computadora o en el repositorio para su posterior uso. El caso de uso "CrearUsuario" es el estereotipo procesar («processing»). Registra los datos de un nuevo usuario que se agrega al sistema, lo que facilita información de su perfil y mejora la personalización de los resultados. El nivel de detalle y la formalidad de la especificación de requerimientos dependen de los riesgos del proyecto y de la complejidad de la aplicación Web a construir. A menudo una especificación basada solamente en casos de uso no es suficiente. Siguiendo el principio de usar UML para la especificación hasta donde sea posible, es factible emplear diagramas de actividades en esta fase. Para cada caso de uso descrito para actividades no triviales se puede construir al menos un diagrama de actividad por cada flujo principal de tareas realizadas en orden. Esto con el fin de describir la funcionalidad indicada por el caso de uso correspondiente.

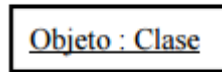
En este caso viendo que UWE está basada en UML se opta por resaltar los siguientes diagramas de secuencia y actividades así detallar su construcción.

Diagrama de Secuencias

Los diagramas de clases y los de objetos representan información estática. No obstante, en un sistema funcional, los objetos interactúan entre sí, y tales interacciones suceden con el tiempo. El diagrama de secuencias UML muestra la mecánica de la interacción con base en tiempos.

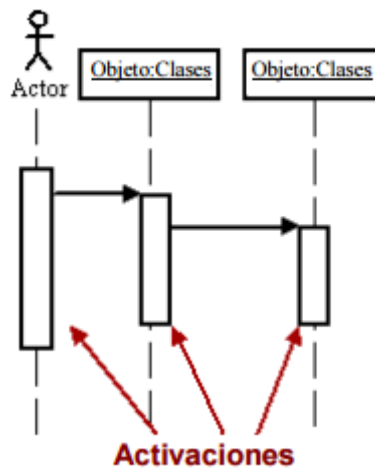
Rol de la Clase

El rol de la clase describe la manera en que un objeto se va a comportar en el contexto. No se listan los atributos del objeto.



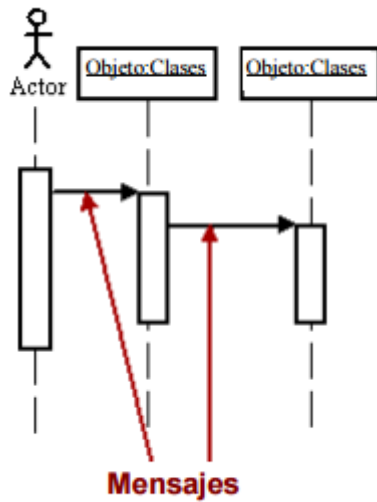
Activación

Los cuadros de activación representan el tiempo que un objeto necesita para completar una tarea.



Mensajes

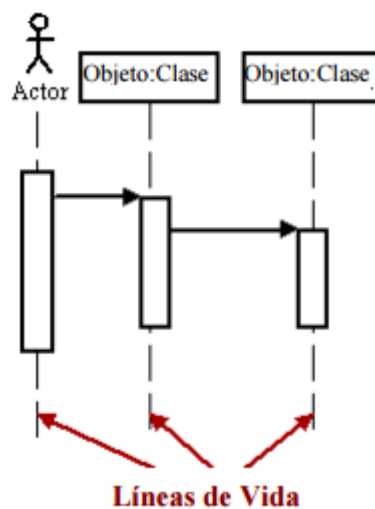
Los mensajes son flechas que representan comunicaciones entre objetos. Las medias flechas representan mensajes asincrónicos. Los mensajes asincrónicos son enviados desde un objeto que no va a esperar una respuesta del receptor para continuar con sus tareas.



Flecha	Tipo de mensaje
	Simple
	Sincrónico
	Asincrónico
	Rechazado
	Time out

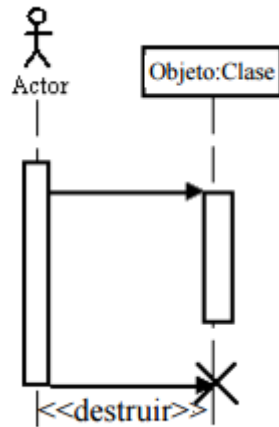
Líneas de Vida

Las líneas de vida son verticales y en línea de puntos, ellas indican la presencia del objeto durante el tiempo.



Destrucción de Objetos

Los objetos pueden ser eliminados tempranamente usando una flecha etiquetada "<>" que apunta a una X.



Loops

Una repetición o loop en un diagrama de secuencias, es representado como un rectángulo. La condición para abandonar el loop se coloca en la parte inferior entre corchetes [].

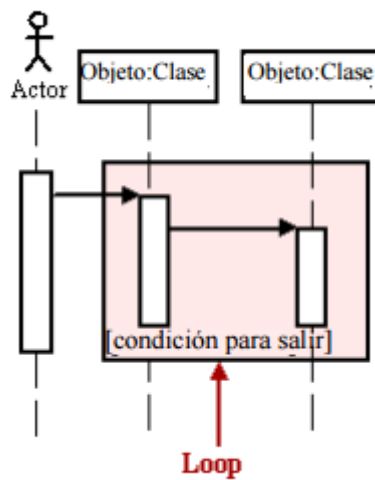
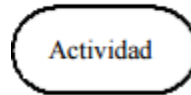


Diagrama de Actividades

Un diagrama de actividades ilustra la naturaleza dinámica de un sistema mediante el modelado del flujo ocurrente de actividad en actividad. Una actividad representa una operación en alguna clase del sistema y que resulta en un cambio en el estado del sistema. Típicamente, los diagramas de actividad son utilizados para modelar el flujo de trabajo interno de una operación.

Estados de Acción

Los estados de acción representan las acciones no interrumpidas de los objetos.



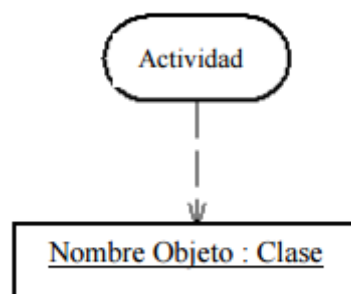
Flujo de la Acción

Los flujos de acción, representados con flechas, ilustran las relaciones entre los estados de acción.



Flujo de Objetos

El flujo de objetos se refiere a la creación y modificación de objetos por parte de actividades. Una flecha de flujo de objeto, desde una acción a un objeto, significa que la acción está creando o influyendo sobre dicho objeto. Una flecha de flujo de objeto, desde un objeto a una acción, indica que el estado de acción utiliza dicho objeto.



Estado Inicial

Estado inicial de un estado de acción.



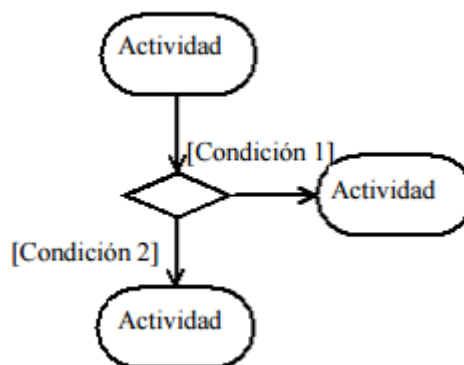
Estado final

Estado final de un estado de acción.



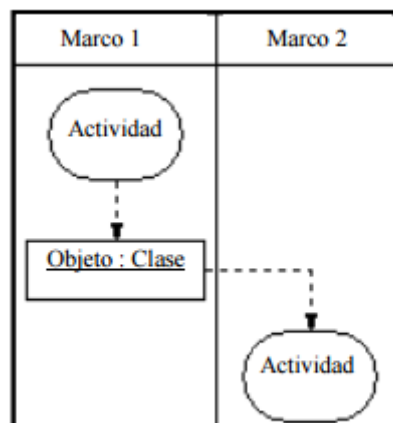
Ramificación

Un rombo representa una decisión con caminos alternativos. Las salidas alternativas deben estar etiquetadas con una condición.



Marcos de Responsabilidad

Los marcos de responsabilidad agrupan a las actividades relacionadas en una misma columna.



2.6.2 Definiendo el contenido

El objetivo del modelo de contenido es proporcionar una especificación visual de la información en el dominio relevante para la aplicación Web. Este es un diagrama UML normal de clases, por ello se debe pensar en las clases que son necesarias para el caso de estudio presentado. En la figura 2.6 se presenta el diagrama de clases para el modelo de contenido. En particular, la información de los usuarios es modelada por la clase "PerfilUsuario" donde se almacenan las propiedades que describen a los diferentes tipos de usuarios.

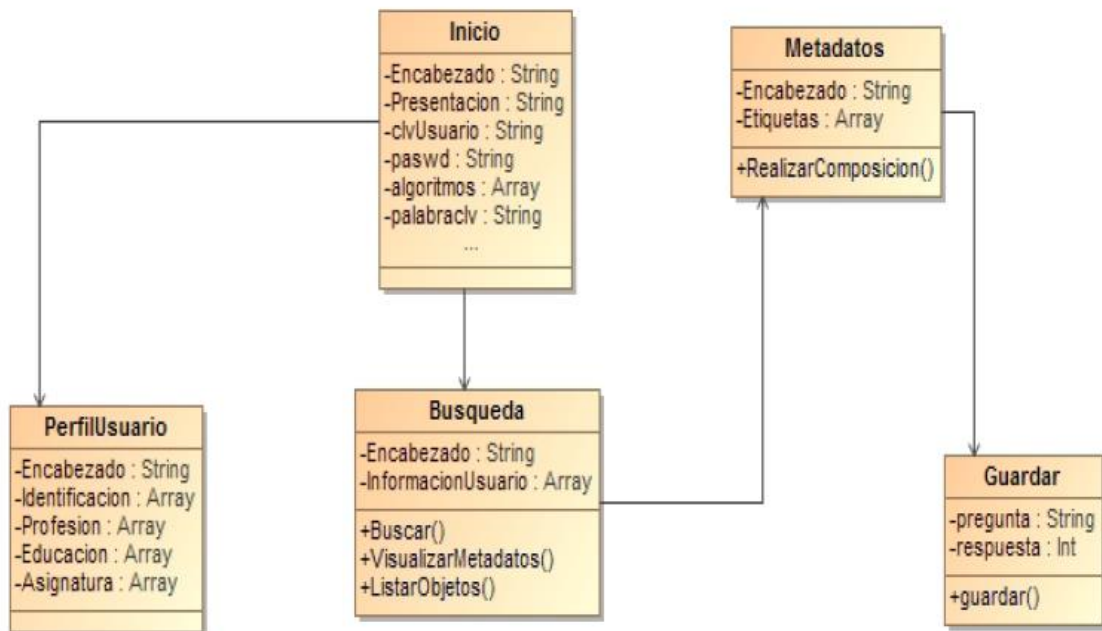


Figura 2.6 Modelo de Contenido.

Fuente: Citlali G. Nieves-Guerrero, Juan P. Ucán-Pech (2014)

En la clase "Inicio" se modela el inicio de la aplicación web, se almacenan las credenciales y propiedades que sirven para identificar al usuario que quiere iniciar sesión. La clase "Búsqueda" modela la información que el usuario proporciona para realizar una consulta y los métodos que se ejecutan para generar la lista de recomendación, la selección de los objetos y la recuperación de los mismos con sus metadatos. La clase "metadatos" modela las características devueltas por los objetos de aprendizaje que el usuario ha seleccionado y el método de realizar la composición con la selección y los

metadatos proporcionados. La clase "guardar" modela las características de almacenamiento del nuevo objeto compuesto.

2.6.3 Estructura y Modelo de Navegación

En una aplicación para la Web es útil saber cómo están enlazadas las páginas. Ello significa que se requiere un diagrama de navegación con nodos y enlaces. Este diagrama se modela con base en el análisis de los requisitos y el modelo de contenido. UWE provee diferentes estereotipos para el modelado de navegación, en la figura 2.7 se presentan los usados en este caso de estudio y seguidamente se da una descripción de cada uno de ellos.

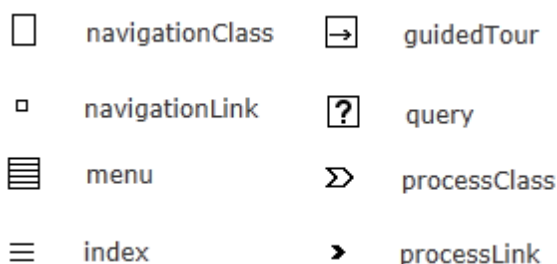


Figura 2.7 Estereotipos de estructura de navegación.
Citlali G. Nieves-Guerrero, Juan P. Ucán-Pech (2014)

Las clases de navegación («navigationClass») representan nodos navegables de la estructura de hipertexto; los enlaces de navegación («navigationLink») muestran vínculos directos entre las clases de navegación; las rutas alternativas de navegación son manejadas por menú («menu»). Los accesos se utilizan para llegar a múltiples instancias de una clase de navegación («index» o «guidedTour») o para seleccionar los elementos («query»). Las clases de procesos («processClass») forman los puntos de entrada y salida de los procesos de negocio en este modelado y la vinculación entre sí y a las clases de navegación se modela por enlaces de procesos («processLink»). En la figura 2.8, las clases de navegación "Inicio y PerfilUsuario" representan nodos navegables de la estructura de hipertexto y se consideran relevantes para la navegación. Los enlaces de navegación "navigationLink" y "processLink" muestran vínculos directos entre las clases de navegación y representan

posibles pasos a seguir por el usuario y, por lo tanto, estos vínculos tienen que ser dirigidos.

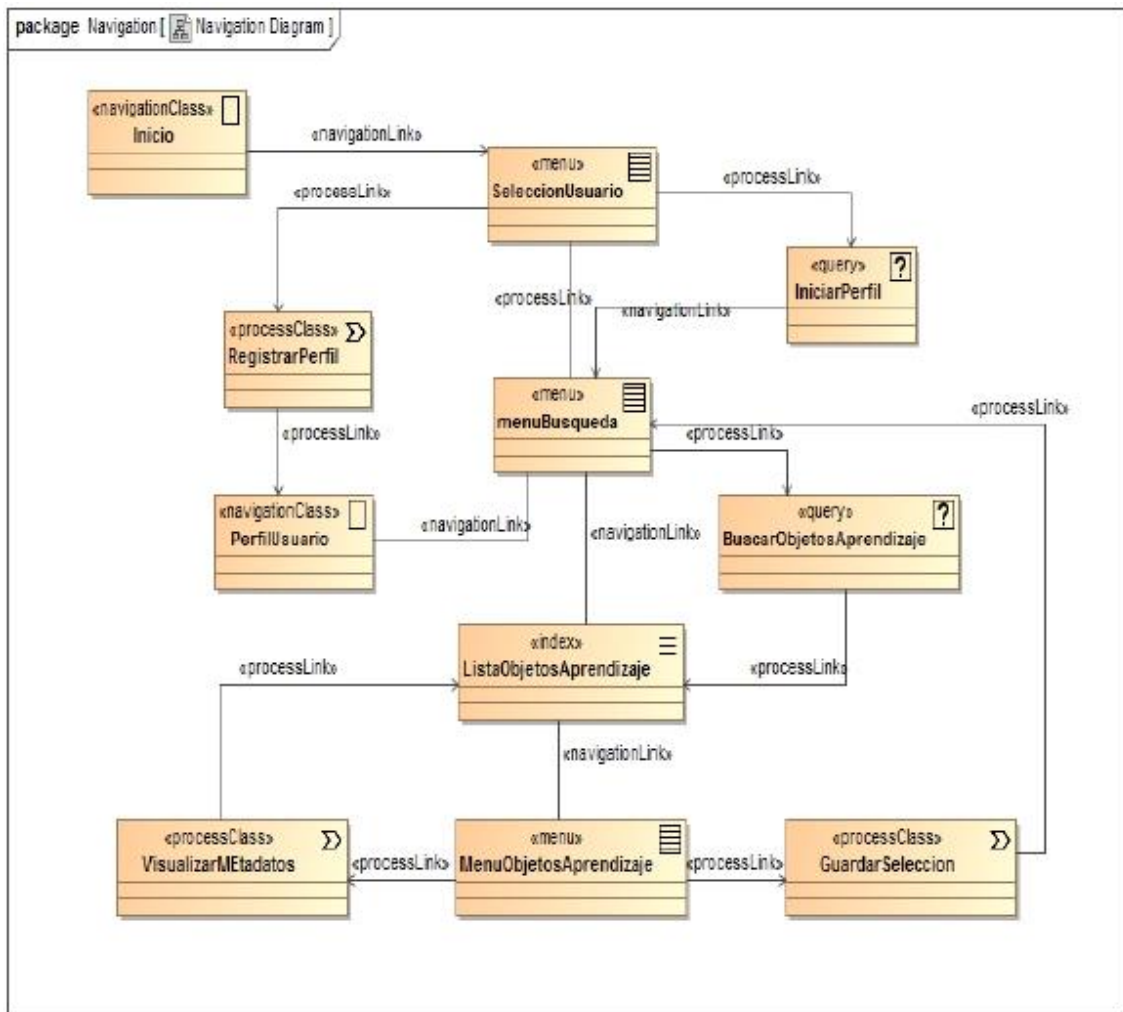


Figura 2.8. Clases de navegación

Fuente: Citlali G. Nieves-Guerrero, Juan P. Ucán-Pech (2014)

La navegación por diferentes alternativas es representada por las clases «menu» ("SeleccionUsuario, MenuBusqueda y MenuObjetosAprendizaje") que se añaden a cada clase de navegación que tiene más de una asociación saliente. Las primitivas de acceso «index» como es "ListaObjetosAprendizaje" se utilizan para llegar a múltiples instancias de una clase de navegación o para seleccionar los elementos con los tipos «query» como "IniciarPerfil y BuscarObjetosAprendizaje", este tipo de clase se debe agregar entre dos clases de navegación cada vez que la multiplicidad de la meta final de su asociación de enlace sea mayor que 1. Las entradas y salidas de las clases "RegistrarPerfil, VisualizarMetadatos y GuardarSeleccion" son modeladas por las clases «process». Es así que desde la página de Inicio un usuario puede, por medio de

"SeleccionUsuario", tener una representación personalizada según sea su tipo de usuario con el que accede al sistema. Puede optar por usar "IniciarPerfil" para consultar si existe su clave de usuario proporcionada, o por "registrarPerfil" que inicia el proceso de registro del nuevo usuario. El usuario que ingresa a la aplicación proporciona palabras clave para "BuscarObjetosAprendizaje" que arroja una "ListaObjetosAprendizaje" para la selección por parte del usuario. De los objetos que son seleccionados en un "MenuObjetosAprendizaje", el usuario puede "VisualizarMetadatos" de los objetos que son candidatos a conformar un nuevo Objeto de Aprendizaje de nivel superior de complejidad para "GuardarSelección".

2.6.4 Modelo de presentación

El modelo de presentación ofrece una visión abstracta de la interfaz de usuario de una aplicación Web. Se basa en el modelo de navegación y en los aspectos concretos de la interfaz de usuario (IU). Describe la estructura básica de la IU, es decir, qué elementos de interfaz de usuario (por ejemplo, texto, imágenes, enlaces, formularios) se utilizan para presentar los nodos de navegación. Su ventaja es que es independiente de las técnicas actuales que se utilizan para implementar un sitio Web, lo que permite a las partes interesadas discutir la conveniencia de la presentación antes de que realmente se aplique. Una clase de presentación está compuesta de elementos de IU como texto («text»), enlaces («anchor»), botones («button»), imágenes («image»), formularios («form») y colecciones de enlaces («anchored collection»). La figura 2.9 muestra un ejemplo de la clase de presentación para la clase de navegación Inicio, se modela la página de presentación "PaginaInicio". Existe una representación de texto para el encabezado y un mensaje de presentación. Modela también un formulario de entrada para que el usuario introduzca clave y contraseña, así como los botones de "iniciarperfil" y "registrarPerfil". Usualmente la información de varios nodos de aplicación es presentada en una página Web, la cual es modelada por páginas en UWE, por ejemplo, en la figura 2.9 se tiene una («presentationPage»). Las páginas de presentación también pueden contener grupos de presentación («presentationGroup»), grupos de presentación iterativos («iteratedPresentationGroup»), y presentaciones alternativas

(«presentationAlternative»), por ejemplo ajustar la interfaz al dispositivo utilizado para ejecutar la aplicación. Un grupo de presentación puede contener a si mismo grupos de presentación y clases de presentación.

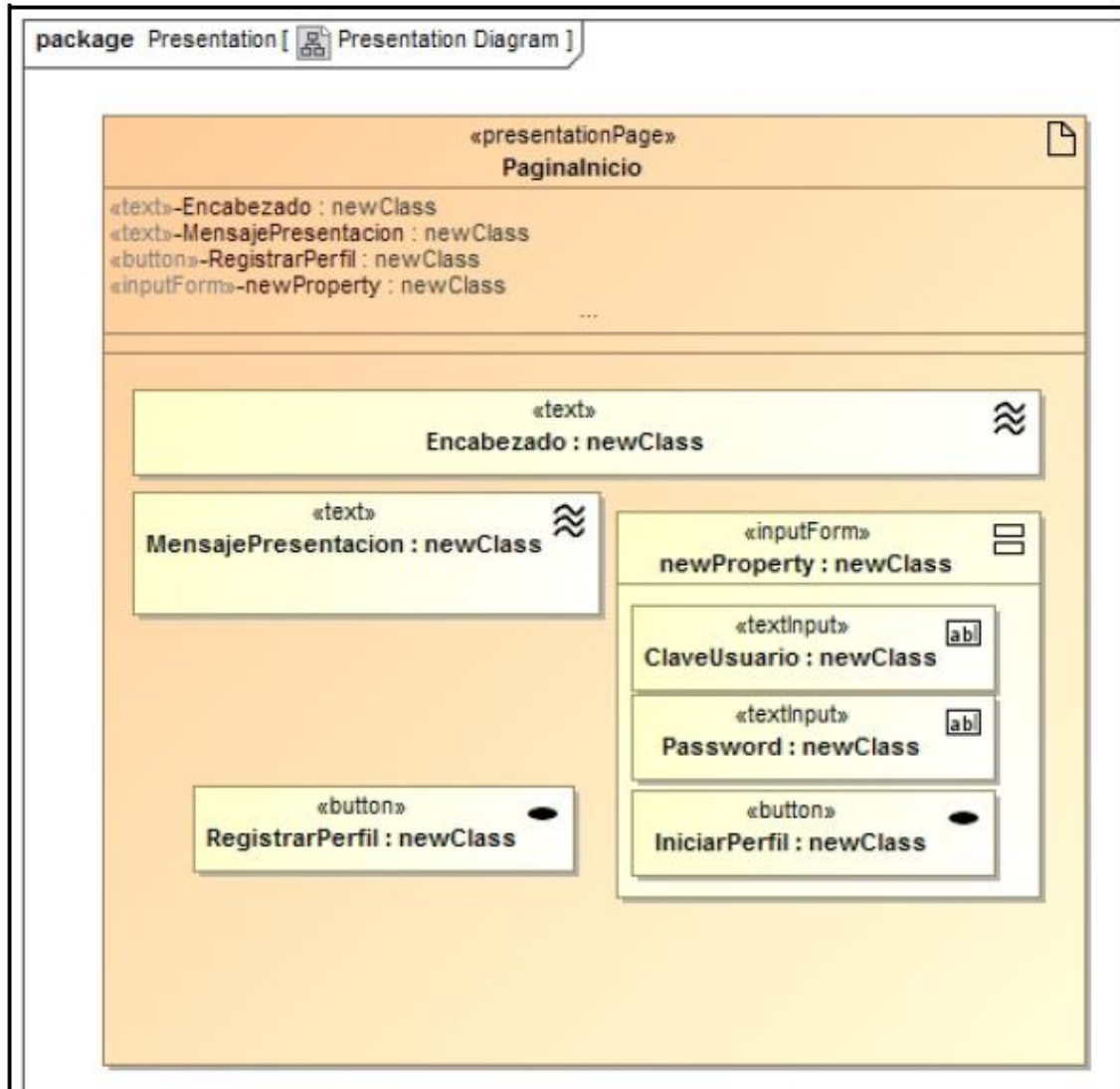


Figura 2.9. Página de presentación: Inicio

Fuente: Citlali G. Nieves-Guerrero, Juan P. Ucán-Pech (2014)

En la figura 2.10 se modela la página de presentación "paginaBusqueda" donde se representa como texto un encabezado y el nombre del usuario. Existe un formulario donde se puede introducir las palabras clave de búsqueda así como seleccionar los algoritmos que se pueden aplicar. Esta página de presentación contiene un grupo de presentación para modelar las listas de objetos candidatos a la composición y los botones de buscar y ver metadatos.

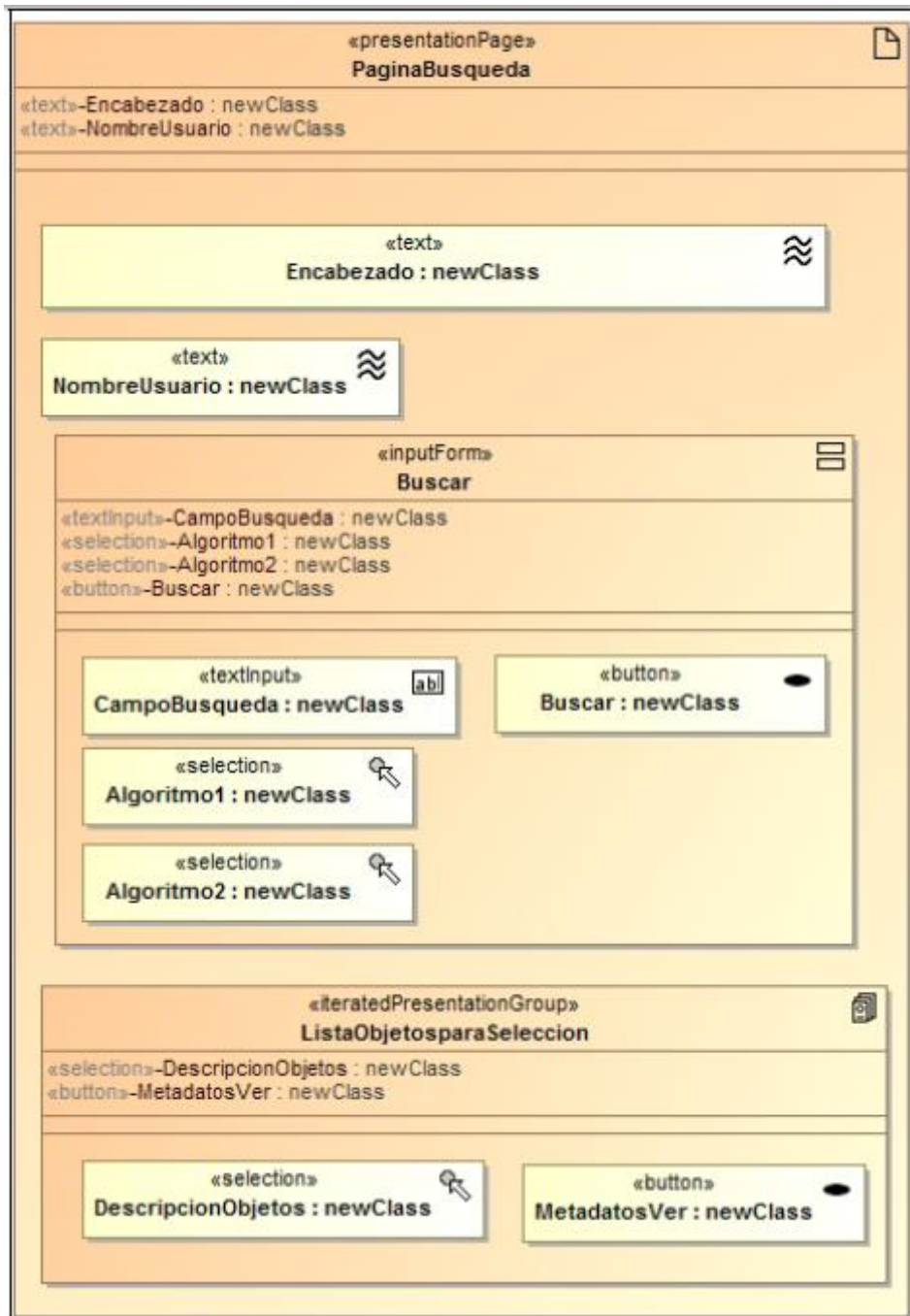


Figura 2.10 Página de presentación: Búsqueda.

Fuente: Citlali G. Nieves-Guerrero, Juan P. Ucán-Pech (2014)

2.6.5 Modelo de proceso

La estructura de navegación puede ser extendida mediante clases de procesos que representan la entrada y la salida de procesos de negocio. El

modelo del proceso representa el aspecto que tienen las acciones de las clases de proceso. En este modelo se tienen dos tipos de modelos:

- Modelo de estructura del proceso, que describe las relaciones entre las diferentes clases de proceso, y
- Modelo de flujo del proceso, que especifica las actividades conectadas con cada « processClass».

A continuación se describen cada uno de ellos:

Modelo de estructura del proceso: Es representado por un diagrama de clases donde se describen las relaciones entre las diferentes clases de proceso. La figura 2.11 presenta la aplicación del modelo para el caso de estudio analizado.

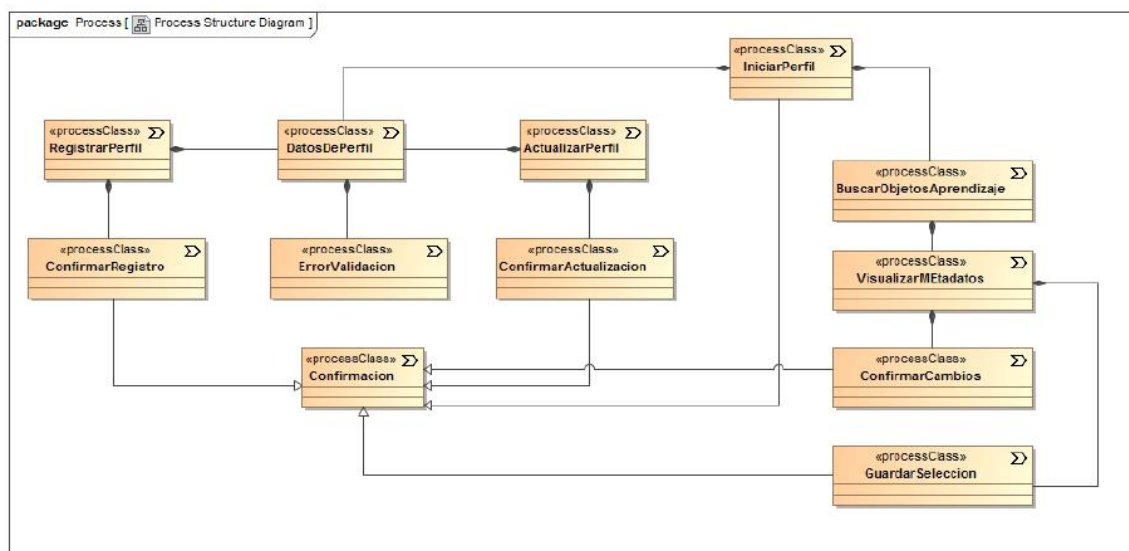


Figura 2.11 Estructura del proceso.

Fuente: Citlali G. Nieves-Guerrero, Juan P. Ucán-Pech (2014)

Modelo del flujo del proceso: Siguiendo el principio de la utilización de UML se han refinado los requisitos con los diagramas de actividad UML. Los diagramas de actividades incluyen actividades, actores responsables de estas actividades (opcional) y elementos de flujo de control. Ellos pueden ser enriquecidos con flujos de objetos que muestran objetos relevantes para la entrada o salida de esas actividades. Estos diagramas representan el flujo del proceso, describiendo el comportamiento de una clase de proceso.

Con el propósito de ejemplificar su utilización se ha desarrollado un caso de estudio para un Sistema de Recomendación de Objetos de Aprendizaje que se ejecuta en la Web.

Se ha ilustrado cada uno de los elementos del modelo a partir de las funcionalidades, características y elementos que conforman las especificaciones de la aplicación Web. Uno de los beneficios de la metodología es reutilizar el conocimiento previo que se cuenta con respecto al empleo de UML. Además la conjugación de todos los modelos permite una visión integral de los requerimientos de la aplicación Web, facilitando su descripción y en consecuencia su comprensión. (N. Koch. 2006)

2.7 MODELO DE CALIDAD ESTABLECIDO POR EL ESTÁNDAR ISO 9126

El estándar ISO/IEC 9126 proviene desde el modelo establecido en 1977 por McCall y sus colegas, los cuales propusieron un modelo para especificar la calidad del software. El modelo de calidad McCall está organizado sobre tres tipos de Características de Calidad:

Factores (especificar): Describen la visión externa del software, como es visto por los usuarios.

Criterios (construir): Describen la visión interna del software, como es visto por el desarrollador.

Métricas (controlar): Se definen y se usan para proveer una escala y método para la medida.

ISO/IEC 9126 es un estándar internacional para la evaluación del Software. Está supervisado por el proyecto SQuaRE, ISO 25000:2005, el cual sigue los mismos conceptos.

2.7.1 ISO/IEC 9126: Tecnologías de la Información

ISO/IEC 9126 es un estándar de calidad de productos compuesto por 4 partes.

ISO/IEC 9126-1 describe un modelo de dos partes para calidad de productos de software:

- a) *Calidad interna y externa, y*
- b) *Calidad en uso.*

El objetivo es abarcar todos los aspectos que pueden afectar a la calidad de los productos de software. Existe un equivalente chileno - NCh 2812-1.c2002 - que es una homologación idéntica de la norma original hecha por el INN.

El estándar internacional posee otras tres partes con carácter de reportes técnicos (aún no son estándares propiamente tales):

- ISO/IEC TR 9126-2 Software engineering - Product quality - Part 2: External metrics
- ISO/IEC TR 9126-3 Software engineering - Product quality - Part 3: Internal metrics
- ISO/IEC TR 9126-4 Software engineering - Product quality - Part 4: Quality in use metrics

Un producto software está definido en un sentido amplio como: los ejecutables, código fuente, descripciones de arquitectura, y así. Como resultado, la noción de usuario se amplía tanto a operadores como a programadores, los cuales son usuarios de componentes como son bibliotecas software. ISO/IEC 9126 distingue entre fallo y no conformidad. Un fallo es el incumplimiento de los requisitos previos, mientras que la no conformidad es el incumplimiento de los requisitos especificados. Una distinción similar es la que se establece entre validación y verificación. (McCall 1977).

2.7.3 Ámbitos de uso de ISO/IEC 9126

- Validar la integridad de una definición de requisitos;
- Identificar los requisitos del software;
- Identificar los objetivos del diseño del software;

- Identificar los objetivos de la prueba de software;
- Identificar el criterio de aseguramiento de calidad;
- Identificar el criterio de aceptación para un producto de software completo.
- Priorizar los recursos en los aspectos más importantes en términos de calidad.
- Etc.

2.7.4 Atributos de la Norma para Calidad Externa e Interna

a) FUNCIONALIDAD

Adecuación: Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.

Exactitud: Capacidad del producto software para proporcionar los resultados o efectos correctos o acordados, con el grado necesario de precisión.

Interoperabilidad: Capacidad del producto software para interactuar con uno o más sistemas especificados.

Seguridad de acceso: Capacidad del producto software para proteger información y datos de manera que las personas o sistemas no autorizados no puedan leerlos o modificarlos, al tiempo que no se deniega el acceso a las personas o sistemas autorizados.

Cumplimiento funcional: Capacidad del producto software para adherirse a normas, convenciones o regulaciones en leyes y prescripciones similares relacionadas con funcionalidad.

b) CONFIABILIDAD (FIABILIDAD)

Madurez: Capacidad del producto software para evitar fallar como resultado de fallos en el software.

Tolerancia a fallos: Capacidad del software para mantener un nivel especificado de prestaciones en caso de fallos software o de infringir sus interfaces especificados.

Capacidad de recuperación: Capacidad del producto software para reestablecer un nivel de prestaciones especificado y de recuperar los datos directamente afectados en caso de fallo.

Cumplimiento de la fiabilidad: Capacidad del producto software para adherirse a normas, convenciones o regulaciones relacionadas con al fiabilidad.

c) FACTIBILIDAD DE USO (USABILIDAD)

Capacidad para ser entendido: Capacidad del producto software que permite al usuario entender si el software es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares.

Capacidad para ser aprendido: Capacidad del producto software que permite al usuario aprender sobre su aplicación.

Capacidad para ser operado: Capacidad del producto software que permite al usuario operarlo y controlarlo.

Capacidad de atracción: Capacidad del producto software para ser atractivo al usuario.

Cumplimiento de la usabilidad: Capacidad del producto software para adherirse a normas, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad.

d) EFICIENCIA

Comportamiento temporal: Capacidad del producto software para proporcionar tiempos de respuesta, tiempos de proceso y potencia apropiados, bajo condiciones determinadas.

Utilización de recursos: Capacidad del producto software para usar las cantidades y tipos de recursos adecuados cuando el software lleva a cabo su función bajo condiciones determinadas.

Cumplimiento de la eficiencia: Capacidad del producto software para adherirse a normas o convenciones relacionadas con la eficiencia.

e) MANTENIBILIDAD

Capacidad para ser analizado: Es la capacidad del producto software para serle diagnosticadas deficiencias o causas de los fallos en el software, o para identificar las partes que han de ser modificadas.

Capacidad para ser cambiado: Capacidad del producto software que permite que una determinada modificación sea implementada.

Estabilidad: Capacidad del producto software para evitar efectos inesperados debidos a modificaciones del software.

Capacidad para ser probado: Capacidad del producto software que permite que el software modificado sea validado.

Cumplimiento de la mantenibilidad: Capacidad del producto software para adherirse a normas o convenciones relacionadas con la mantenibilidad.

f) PORTABILIDAD

Adaptabilidad: Capacidad del producto software para ser adaptado a diferentes entornos especificados, sin aplicar acciones o mecanismos

distintos de aquellos proporcionados para este propósito por el propio software considerado.

Instalabilidad: Capacidad del producto software para ser instalado en un entorno especificado.

Coexistencia: Capacidad del producto software para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes.

Capacidad para reemplazar: Capacidad del producto software para ser usado en lugar de otro producto software, para el mismo propósito, en el mismo entorno.

Cumplimiento de la portabilidad: Capacidad del producto software para adherirse a normas o convenciones relacionadas con la portabilidad.

2.7.5 Atributos Para Calidad en Uso

1. Efectividad: Capacidad del producto software para permitir a los usuarios alcanzar objetivos especificados con exactitud y completitud, en un contexto de uso especificado.

2. Productividad: Capacidad del producto software para permitir a los usuarios gastar una cantidad adecuada de recursos con relación a la efectividad alcanzada, en un contexto de uso especificado.

3. Seguridad física: Capacidad del producto software para alcanzar niveles aceptables del riesgo de hacer daño a personas, al negocio, al software, a las propiedades o al medio ambiente en un contexto de uso especificado.

4. Satisfacción: Capacidad del producto software para satisfacer a los usuarios en un contexto de uso especificado.

2.7.6 Perfil de Calidad Usando ISO/IEC 9126

Un perfil de calidad permite focalizar la definición o evaluación de calidad de un producto de software en los criterios de calidad más importantes según el contexto requerido.

En un perfil están definidos:

Los atributos y subcaracterísticas relevantes para el producto de software.

Las métricas que se usarán en la medición.

Los rangos de aceptación de esas métricas.

El estándar provee un entorno para que las organizaciones definan un modelo de calidad para el producto software. Haciendo esto así, sin embargo, se lleva a cada organización la tarea de especificar precisamente su propio modelo. Esto podría ser hecho, por ejemplo, especificando los objetivos para las métricas de calidad las cuales evalúan el grado de presencia de los atributos de calidad.

Métricas internas son aquellas que no dependen de la ejecución del software (medidas estáticas).

Métricas externas son aquellas aplicables al software en ejecución.

La calidad en las métricas de uso están sólo disponibles cuando el producto final es usado en condiciones reales. Idealmente, la calidad interna determina la calidad externa y esta a su vez la calidad en el uso. (McCall 1977)

2.8 MODELO CONSTRUCTIVO DE COSTOS (COCOMO)

El Modelo Constructivo de Costos (o COCOMO, por su acrónimo del inglés CONstructive COst MOdel) es un modelo matemático de base empírica utilizado para estimación de costos de software. Incluye tres sub modelos, cada uno ofrece un nivel de detalle y aproximación, cada vez mayor, a medida que avanza el proceso de desarrollo del software: básico, intermedio y detallado.

Este modelo fue desarrollado por Barry W. Boehm a finales de los años 70 y comienzos de los 80, exponiéndolo detalladamente en su libro "Software Engineering Economics" (Prentice-Hall, 1981).

2.8.1 Características Generales

Pertenece a la categoría de modelos de subestimaciones basados en estimaciones matemáticas. Está orientado a la magnitud del producto final, midiendo el "tamaño" del proyecto, en líneas de código principalmente.

Inconvenientes

- Los resultados no son proporcionales a las tareas de gestión ya que no tiene en cuenta los recursos necesarios para realizarlas.
- Se puede desviar de la realidad si se indica mal el porcentaje de líneas de comentarios en el código fuente.
- Es un tanto subjetivo, puesto que está basado en estimaciones y parámetros que pueden ser "vistos" de distinta manera por distintos analistas que usen el método.
- Se miden los costes del producto, de acuerdo a su tamaño y otras características, pero no la productividad.
- La medición por líneas de código no es válida para orientación a objetos.
- Utilizar este modelo puede resultar un poco complicado, en comparación con otros métodos (que también sólo estiman).

2.8.2 Modelos de estimación

Las ecuaciones que se utilizan en los tres modelos son:

- $E = a(Kl)^b * m(X)$, en persona-mes
- $Tdev = c(E)^d$, en meses
- $P = E/Tdev$, en personas

Donde:

- E : Es el esfuerzo requerido por el proyecto, en persona-mes
- $Tdev$: Es el tiempo requerido por el proyecto, en meses

- P : Es el número de personas requerido por el proyecto
- a, b, c y d : Son constantes con valores definidos en una tabla, según cada submodelo
- Kl : Es la cantidad de líneas de código, en miles.
- $m(X)$: Es un multiplicador que depende de 15 atributos.

A la vez, cada submodelo también se divide en **modos** que representan el tipo de proyecto, y puede ser:

- **modo orgánico**: un pequeño grupo de programadores experimentados desarrollan software en un entorno familiar. El tamaño del software varía desde unos pocos miles de líneas (tamaño pequeño) a unas decenas de miles (medio).
- **modo semilibre** o **semiencajado**: corresponde a un esquema intermedio entre el orgánico y el rígido; el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.
- **modo rígido** o **empotrado**: el proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único y es difícil basarse en la experiencia, puesto que puede no haberla.

2.8.3 Modelo Básico

Se utiliza para obtener una primera aproximación rápida del esfuerzo, y hace uso de la siguiente tabla de constantes para calcular distintos aspectos de costes:

MODO	a	b	c	D
-------------	----------	----------	----------	----------

Orgánico	2.40	1.05	2.50	0.38
Semi - Orgánico	3.00	1.12	2.50	0.35
Empotrado	3.60	1.20	2.50	0.33

Estos valores son para las fórmulas:

- Personas necesarias por mes para llevar adelante el proyecto (**MM**) = $a \cdot (Kl^b)$
- Tiempo de desarrollo del proyecto (**TDEV**) = $c \cdot (MM^d)$
- Personas necesarias para realizar el proyecto (**CosteH**) = $MM/TDEV$
- Costo total del proyecto (**CosteM**) = $CosteH \cdot \text{Salario medio entre los programadores y analistas.}$

Se puede observar que a medida que aumenta la complejidad del proyecto (modo), las constantes aumentan de 2.4 a 3.6, que corresponde a un incremento del esfuerzo del personal. Hay que utilizar con mucho cuidado el modelo básico puesto que se obvian muchas características del entorno.

2.8.4 Modelo intermedio

Este añade al modelo básico quince modificadores opcionales para tener en cuenta en el entorno de trabajo, incrementando así la precisión de la estimación.

Para este ajuste, al resultado de la fórmula general se lo multiplica por el coeficiente surgido de aplicar los atributos que se decidan utilizar.

Los valores de las constantes a reemplazar en la fórmula son:

MODO	a	b
-------------	----------	----------

Orgánico	3.20	1.05
Semi - Orgánico	3.00	1.12
Empotrado	2.80	1.20

Se puede observar que los exponentes son los mismos que los del modelo básico, confirmando el papel que representa el tamaño; mientras que los coeficientes de los modos orgánico y rígido han cambiado, para mantener el equilibrio alrededor del semilibre con respecto al efecto multiplicador de los atributos de coste.

Atributos

Cada atributo se cuantifica para un entorno de proyecto. La escala es ***muy bajo - bajo - nominal - alto - muy alto - extremadamente alto***. Dependiendo de la calificación de cada atributo, se asigna un valor para usar de multiplicador en la fórmula (por ejemplo, si para un proyecto el atributo *DATA* es calificado como *muy alto*, el resultado de la fórmula debe ser multiplicado por 1000).

El significado de los atributos es el siguiente, según su tipo:

- De software
 - **RELY**: garantía de funcionamiento requerida al software. Indica las posibles consecuencias para el usuario en el caso que existan defectos en el producto. Va desde la sola inconveniencia de corregir un fallo (*muy bajo*) hasta la posible pérdida de vidas humanas (*extremadamente alto*, software de alta criticidad).
 - **DATA**: tamaño de la base de datos en relación con el tamaño del programa. El valor del modificador se define por la relación: D/K , donde D corresponde al tamaño de la base de datos en bytes y K es el tamaño del programa en cantidad de líneas de código.
 - **CPLX**: representa la complejidad del producto.
- De hardware

- **TIME:** limitaciones en el porcentaje del uso de la CPU.
 - **STOR:** limitaciones en el porcentaje del uso de la memoria.
 - **VIRT:** volatilidad de la máquina virtual.
 - **TURN:** tiempo de respuesta requerido.
- De personal
 - **ACAP:** calificación de los analistas.
 - **AEXP:** experiencia del personal en aplicaciones similares.
 - **PCAP:** calificación de los programadores.
 - **VEXP:** experiencia del personal en la máquina virtual.
 - **LEXP:** experiencia en el lenguaje de programación a usar.
 - De proyecto
 - **MODP:** uso de prácticas modernas de programación.
 - **TOOL:** uso de herramientas de desarrollo de software.
 - **SCED:** limitaciones en el cumplimiento de la planificación.

El valor de cada atributo, de acuerdo a su calificación, se muestra en la siguiente tabla:

Atributos	Valor					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos de software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
Atributos de hardware						

Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributos de personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		
Atributos del proyecto						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	

Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10	
---------------------------------------	------	------	------	------	------	--

2.8.5 Modelo detallado

Presenta principalmente dos mejoras respecto al anterior:

- Los factores correspondientes a los atributos son sensibles o dependientes de la fase sobre la que se realizan las estimaciones. Aspectos tales como la experiencia en la aplicación, utilización de herramientas de software, etc., tienen mayor influencia en unas fases que en otras, y además van variando de una etapa a otra.
- Establece una jerarquía de tres niveles de productos, de forma que los aspectos que representan gran variación a bajo nivel, se consideran a nivel módulo, los que representan pocas variaciones, a nivel de subsistema; y los restantes son considerados a nivel sistema. (Prentice-Hall, 1981).

CAPITULO III MARCO APLICATIVO

i. FASE DE INICIO

En el presente capítulo se desarrollaran las fases correspondientes a la conceptualización, análisis y diseño del sistema siguiendo el proceso de desarrollo AUP detallando en el capítulo de marco teórico, además de la aplicación de ingeniería de requerimientos siguiendo las diversas fases del modelado y desarrollo de la metodología UWE.

En esta fase del presente proyecto introduce en su etapa del diseño, el comportamiento actual del sistema, se identifican los actores que interactúan y principalmente los requerimientos del sistema. Primero se centra en el modelo del negocio y posteriormente se establecen los casos de uso del mismo.

3.1 DIAGNÓSTICO DEL SISTEMA ACTUAL

Antes de realizar las fases de modelo se debe ver la situación actual del Área de Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto, en cuanto a las actividades concernientes al seguimiento de correspondencia que sostiene.

El Área de Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto, compuesta con las siguientes carreras Contaduría Pública, Administración de Empresas, Economía, Comercio Internacional y Gestión Turística y Hotelera, todas estas contribuyen a la formación de profesionales innovadores, competentes y éticos, para coadyuvar al desarrollo integral y sostenible en ámbitos de gestión, económico-financiero, de servicio y comercio internacional, por tal razón que dentro del Área albergan y se realizan distintas correspondencias internas así como externas.

Para el proceso del desarrollo se realizó en una primera etapa la recolección de información a través de entrevistas, consultas y documentos existentes (Libros de Acta, características de los tipos de correspondencia del Área de Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto, etc.)

3.1.1 Descripción de funciones

Dada la recolección de información mediante entrevistas realizadas se describe, a continuación, con base en el manual de funciones la descripción de cada uno de los puestos.

Nivel Área:

- **Decano de Área**

Representar al área con todos sus actos, teniendo a cargo la dirección, promoción y supervisión del desarrollo académico – administrativo en base al Estatuto Orgánico de la U.P.E.A. y disposiciones legales, reglamentos y resoluciones u otras normativas que regulen las relaciones académicas – administrativas y estudiantiles.

- **Secretaria Ejecutiva**

Apoyar al señor decano en las siguientes funciones que desempeña, algunas de sus funciones es recibir, clasificar, registrar, distribuir y archivar la documentación de decanatura, redactar oficios, informes, circulares, cartas, memorándums, para comunicación interna como externa según indicaciones u orientaciones generales recibidas.

- **Secretario ejecutivo de Área**

Es unidad académica por excelencia y como tal ejecuta íntegramente la planificación ejecución y administración de la docencia y sus procesos asociados, así como la de gestión de conocimiento que se genera en las facultades. Autoridad unipersonal, subordinada al decano que tiene como función organizar la enseñanza y la investigación de su departamento. Es quien representa toda la parte académica del departamento con los estudiantes.

- **Profesional en Planillas, Información Estadística y Contrataciones**

Crear desarrollar mantener sistemas estadísticos e informáticos que permitan obtener, procesar y suministrar los datos de información requeridas por las autoridades superiores para la toma de decisiones y fijación de las políticas institucionales entre sus funciones esta, coadyuvar con la conformación de un sistema único de información, promover actualización permanente de la información para el logro de los objetivos.

- **Técnico Académico del Área**

Responsable de emitir informes en la parte académica también de los trabajos asignados por mediato superior, apoyar a la planificación, organización y dirigir, controlar todas las actividades académicas del Área. Supervisar las actividades académicas de investigación, extensión y proyección social del Área.

Auxiliar de Oficina

Apoyo en coordinación, realización y supervisión de actividades variadas, entre sus funciones están, colaborar y elaborar anualmente los planes y programas académicos del Área.

Nivel Carrera:

- **Director de Carrera**

Administrar y desarrollar la carrera, coordinando y articulando la actividad académica y administrativa, relacionando con la comunidad universitaria, carreras de la UPEA, instituciones públicas y privadas y la sociedad en su conjunto de acuerdo a la visión y la misión de la carrera.

- **Secretaria**

Llevar el archivo sistemático a diario en toda la documentación académica, redacción, transcripción, recepción y despacho de documentos de la Carrera. Así mismo, coadyuvar a la dirección de Carrera en el desarrollo de actividades académicas y administrativas.

- **Unidad de Kardex**

Ejecutar las actividades técnicas en registros académicos relacionados con el sistema seguimiento académico entre sus funciones esta, recibir, clasificar, registrar y archivar los documentos académicos de los estudiantes, verifica y realizar la actualización de registros, fichas, actas de notas de estudiantes.

- **Laboratorio de Información**

Velar por el funcionamiento de la sala de computación, así como apoyar el uso de los recursos informáticos, entre sus funciones están, presentar informe mensual al director de la carrera acerca novedades presentadas durante el mes.

- **Instituto de Investigación**

Proponer, organizar y dirigir los programas de investigación dentro de los ámbitos de competencia académica de la carrera entre sus funciones están,





formular políticas de investigación para la carrera, apoyar actividades de investigación para la carrera, organizar actividades de investigación de acuerdo con la política y planes, reglamentos de la carrera.

- **Auxiliar I y II**

Apoyar en las actividades de mantenimiento y actualización del Kardex individual, del personal docente y administrativo de la universidad entre sus funciones esta, apoyar al encargado de kardex en la elaboración de reportes y otros a requerimientos, apoyar en el mantenimiento y actualización.

3.1.2 Identificación de actores del negocio

La identificación de actores nos permite conocer a las personas involucradas en el proceso de correspondencia que tiene el Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, a objeto de formar los casos de uso.

Actor	Descripción
 Solicitante	Se encarga de solicitar su trámite ya sea de correspondencia interna o externa.
 Auxiliar	Se encarga de recepción, revisar, derivar el respectivo trámite ya sea de correspondencia interna como externa.
 Secretaria	Se encarga de elaborar, registrar, revisar, derivar el respectivo trámite ya sea de correspondencia interna como externa.
 Técnico kardex	Se encarga de elaborar informes de trámite, revisar, derivar el respectivo trámite ya sea de correspondencia interna como externa.




 Director	Persona da curso o el visto bueno del trámite ingresado, puede ser correspondencia interna o externa.
 Técnico Académico	Se encarga de elaborar informes de trámite, revisar, derivar el respectivo trámite ya sea de correspondencia interna como externa
 Decano	Persona da curso o el visto bueno del trámite ingresado, puede ser correspondencia interna o externa

Tabla .3.1 Descripción General de Actores

Fuente: (Elaboración Propia)

3.1.3 Modelo del Negocio

El modelo de negocio es una técnica para comprender los procesos de organización del entorno, describiendo los procesos en términos de casos de uso y actores que intervienen en el contexto.

El modelo de negocio explica gráficamente un conjunto de casos de uso de un sistema, en la figura 3.1 observamos el diagrama de caso de uso del sistema actual.

3.1.4 Diagrama de Casos de uso del negocio

Diagrama de Casos de uso del negocio: Solicitante-Auxiliar o Secretaria

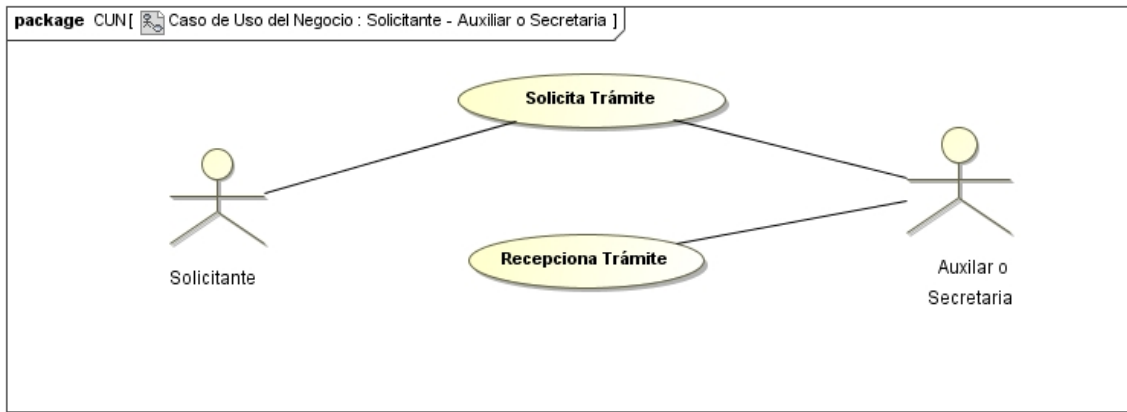


Figura 3.1 Diagrama de casos de uso del Negocio: Solicitante-Auxiliar o Secretaria

Fuente: (Elaboración Propia)

Diagrama de Casos de uso del negocio: Auxiliar- Técnico kardex - Director

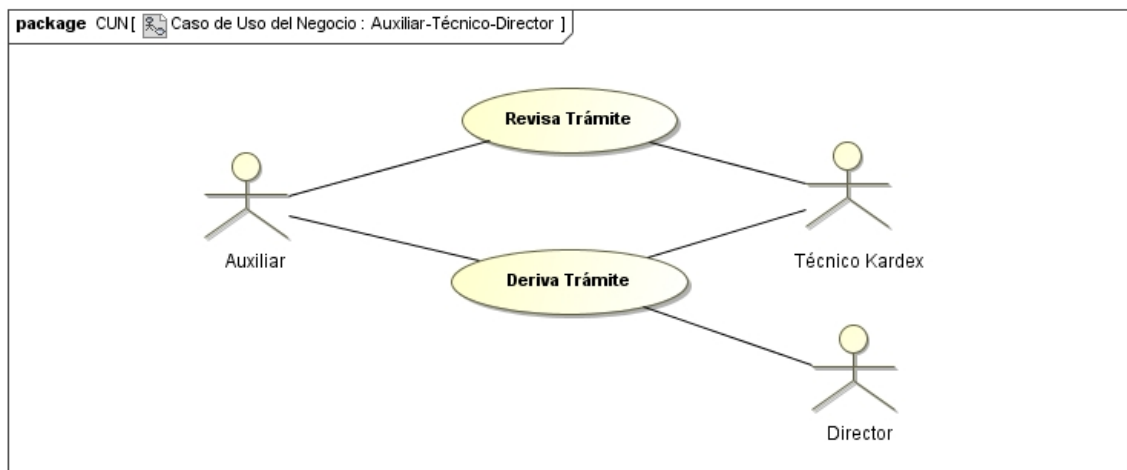


Figura 3.2 Diagrama de casos de uso del Negocio: Auxiliar-Técnico Kardex - Director

Fuente: (Elaboración Propia)

Diagrama de Casos de uso del negocio: Técnico Kardex- Director

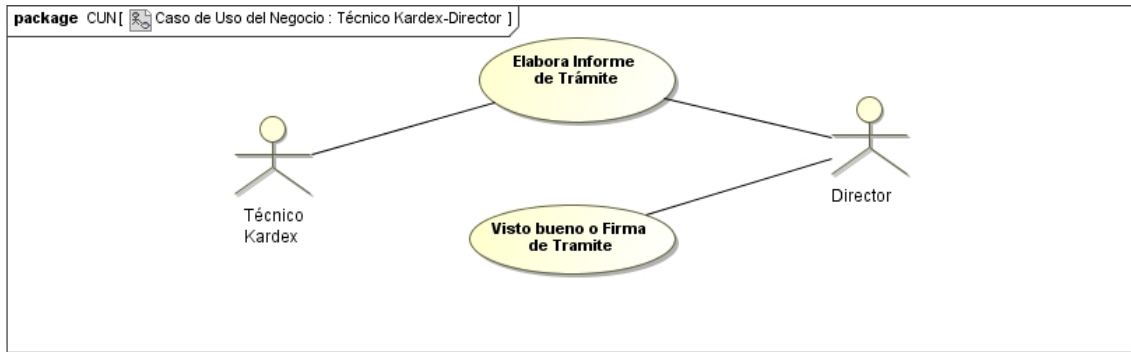


Figura 3.3 Diagrama de casos de uso del Negocio: Técnico Kardex- Director

Fuente: (Elaboración Propia)

Diagrama de Casos de Uso del Negocio General:

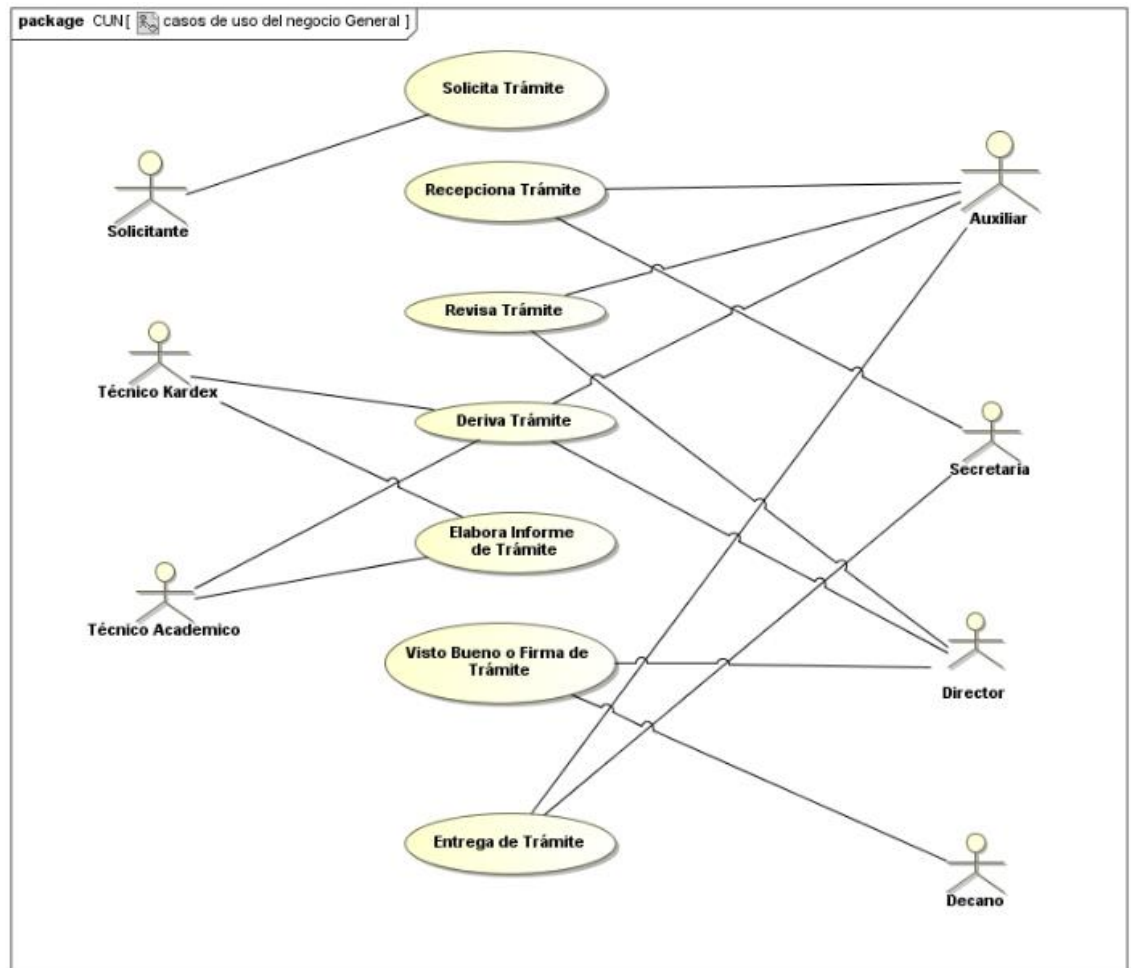


Figura 3.4 Modelo de Casos de Uso del Negocio General

Fuente: (Elaboración Propia)

3.1.5 Diagrama de Actividades del Negocio

Los diagramas de actividades nos muestran cómo se ejecutan el trabajo, proporcionando una descripción de los actores que intervienen en el Área de Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto.

Diagrama de actividades del Negocio: Proceso del Área Ciencias Económicas, Financieras y Administrativas

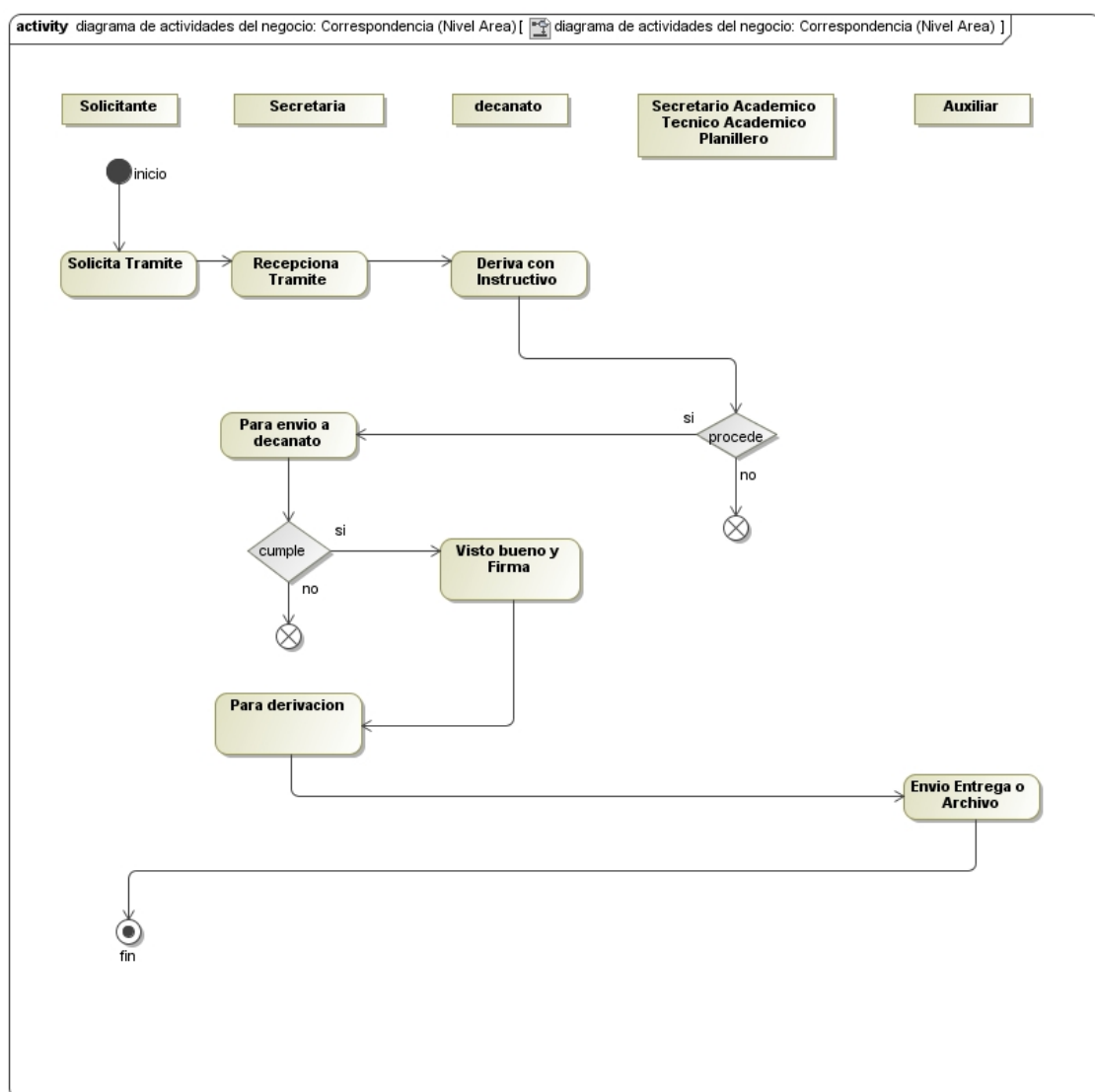


Figura 3.5 Diagrama de actividades del Negocio: Proceso Área Ciencias Económicas, Financieras y Administrativas

Fuente:(elaboración Propia)

En la figura 3.5 se hace el diagnóstico de la situación actual, reflejando los pasos que siguen, empezando desde el solicitante que necesita hacer un trámite hasta la entrega de dicha correspondencia.

Diagrama de actividades del Negocio: Proceso de Correspondencia a nivel Carrera

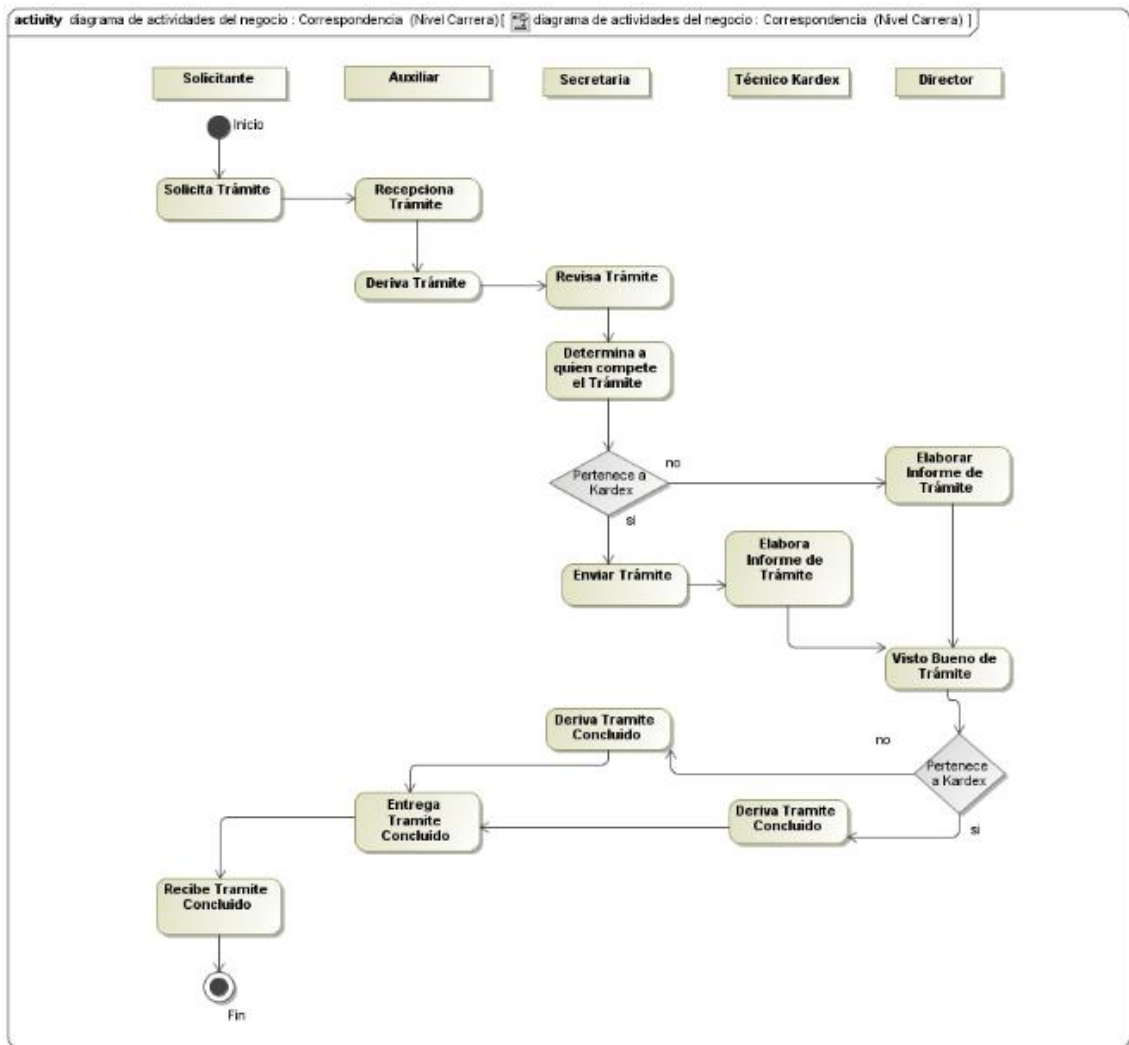


Figura 3.6 Diagrama de actividades del Negocio: Proceso de correspondencia a nivel Carrera

Fuente:(elaboración Propia)

En la figura 3.6 también se hace el diagnóstico de la situación actual, reflejando los pasos que siguen desde el solicitante del trámite hasta la entrega o conclusión del trámite.

3.1.6 Diagramas de secuencias del negocio

Diagramas de secuencias del negocio: Trámite unidad kardex

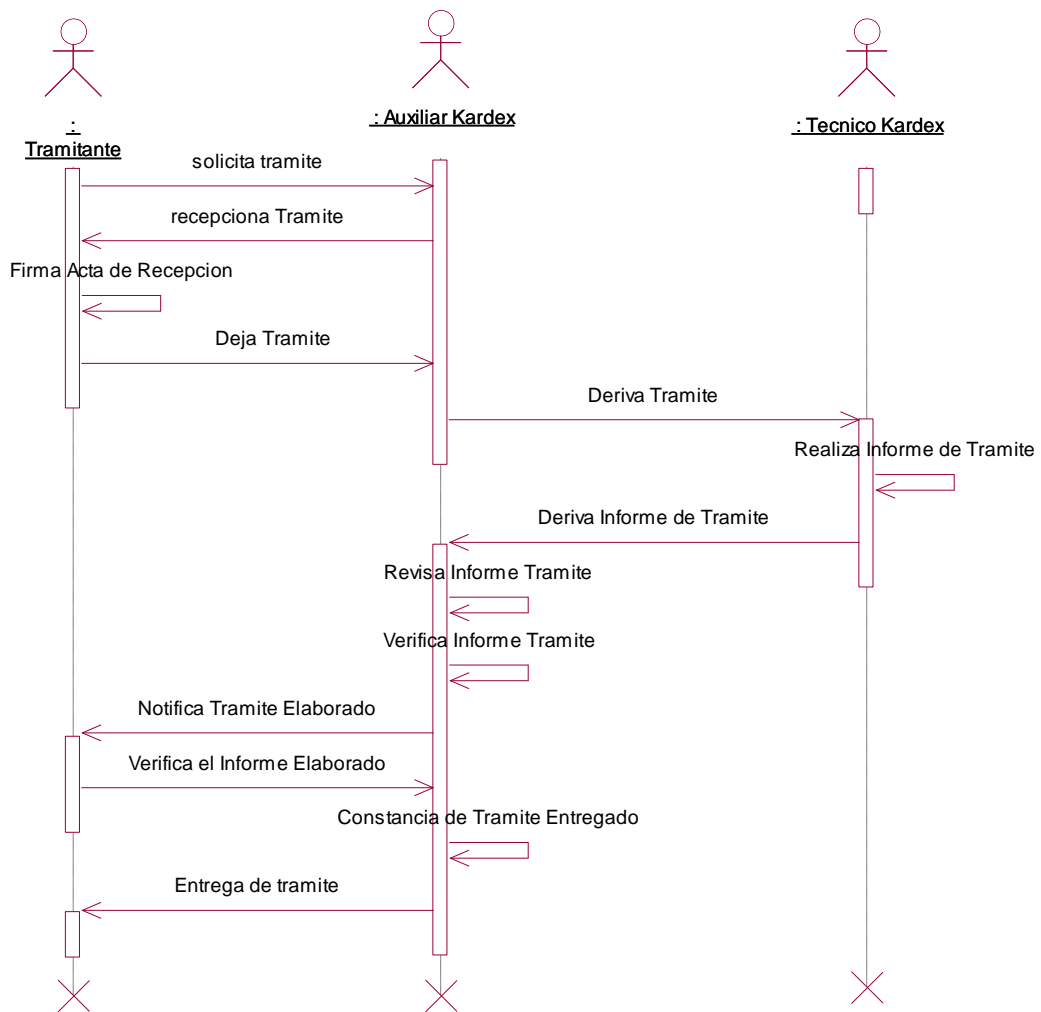


Figura 3.7 Diagramas de secuencias del negocio: Trámite Unidad Kardex

Fuente:(elaboración Propia)

Diagramas de secuencias del negocio: Tramite Dirección y Decanatura

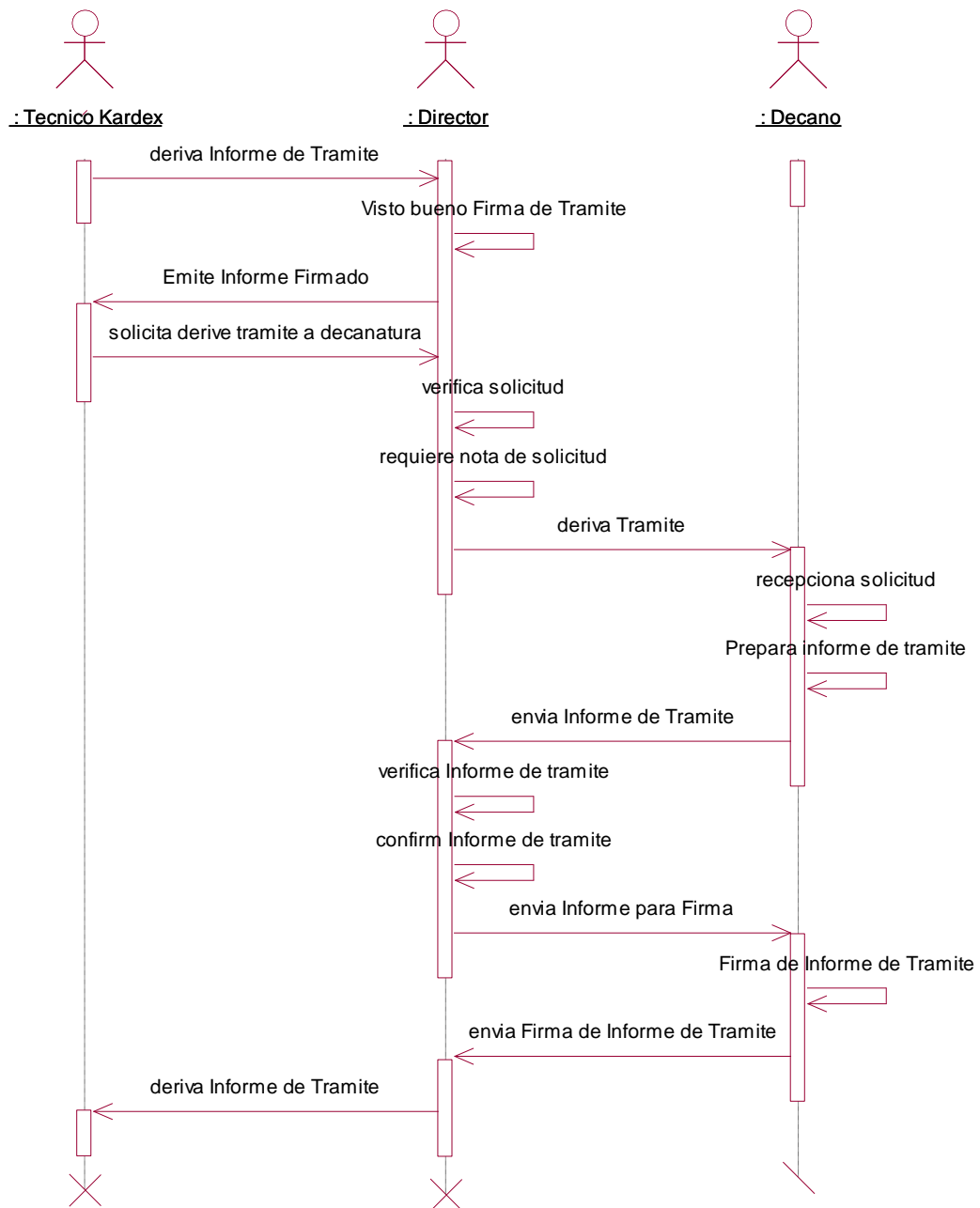


Figura 3.8 Diagramas de secuencias del negocio: Tramite Dirección y Decanatura

Fuente:(elaboración Propia)

3.1.7 Ingeniería de requerimientos

En el punto de desarrollo se identifican dos tipos de requerimientos que son los requerimientos funcionales y los no funcionales los cuales se describen a continuación.

Requisitos funcionales

- El sistema debe administrar la información del flujo de correspondencia del Área Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto.
- El sistema debe realizar el control y seguimiento de la correspondencia interna como externa.
- El sistema debe gestionar usuarios y acorde a sus privilegios limitar el acceso a recursos funcionales.
- El sistema debe tener seguridad en el acceso a la información del sistema.
- El sistema debe generar reportes.
- El sistema debe realizar consultas y búsquedas, de las correspondencias registradas.
- El sistema debe registrar la correspondencia según la categoría.
- El sistema debe registrar usuarios.
- El sistema debe tener Facilidad de uso.

- El sistema debe trabajar con el Mínimo tiempo de respuestas.

Requisitos no funcionales

- Framework CodeIgniter
- MySql
- Apache
- Plataforma Windows Linux Android
- Navegador (internet explore, Mozilla Firefox y Chrome)

ii. FASE DE ELABORACION

3.2 ANÁLISIS DEL SISTEMA

Una vez estudiado la situación actual del negocio se pasa a esta etapa del análisis del sistema donde se plantea un nuevo flujo de trabajo, además se insertan nuevos actores a fin de una mejora en cuanto a la nueva administración de correspondencia del Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto.

3.2.1 Modelo de casos de uso del sistema

Los Casos de uso representan la interacción entre los usuarios y el sistema. Este modelado de casos de uso se desarrolla a lo largo de varias iteraciones añadiendo nuevos casos de usos y mejorando la descripción de los casos de uso que ya se crearon anteriormente.

3.2.1.1 Diagrama de Casos de uso del Director

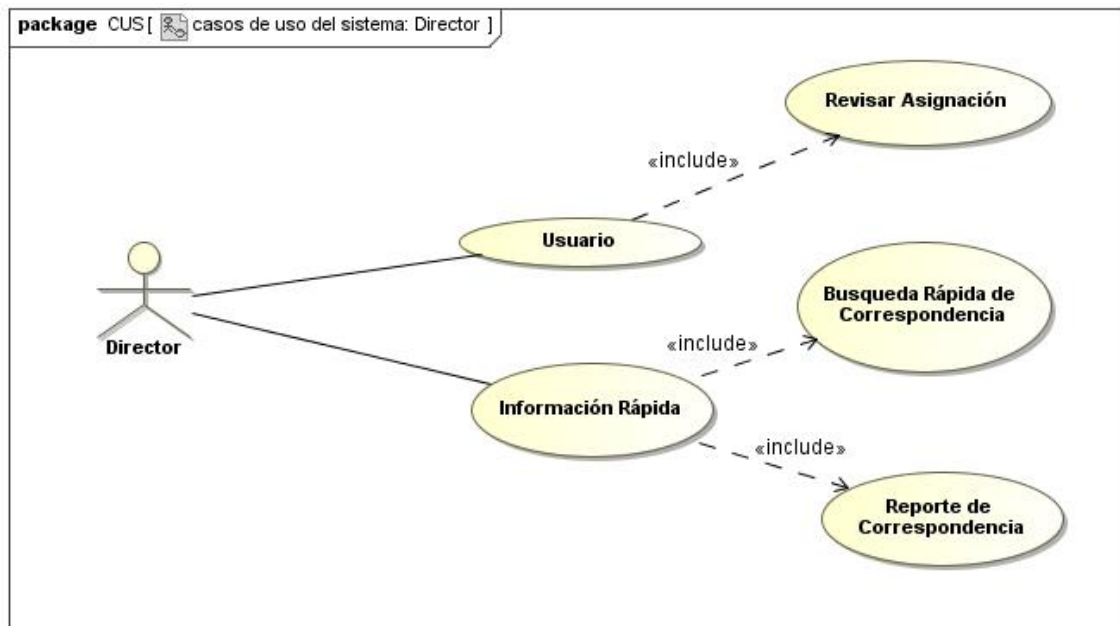


Figura 3.9 Diagrama de Casos de uso del Director
Fuente:(elaboración Propia)

Descripciones de casos de uso: Director

Descripción de caso de uso: Usuario		
Nombre	Usuario	
Actores	Director	
Propósito	Revisar su bandeja de entrada y ver su designación.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Escoge en el menú y panel la opción de Usuarios.	Despliega la lista de tareas pendientes según su derivación.

Tabla 3.2 Descripción de caso de

uso: Usuario

Fuente: (Elaboración propia)

Descripción de caso de uso: Revisar Asignación		
Nombre	Revisar Asignación	
Actores	Director	
Propósito	Revisar la asignación de correspondencia.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije una de las opciones usuario, luego Bandeja de entrada.	Se desplegara la bandeja de entrada donde encontrara si tiene una designación pendiente y dar prioridad de urgente o normal a dicha correspondencia.

Tabla 3.3 Descripción de caso de uso: Revisar Asignación

Fuente: (Elaboración propia)

Descripción de caso de uso: Información Rápida	
Nombre	Información rápida
Actores	Director

Propósito	Proporciona Información ágil de la Correspondencias Registradas.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije una de las opciones de información rápida.	Despliega la lista de correspondencias realiza búsquedas según el campo que vea usted pertinente.

Tabla 3.4 Descripción de caso de uso: Información Rápida

Fuente: (Elaboración propia)

Descripción de caso de uso: Búsqueda Rápida de Correspondencia		
Nombre	Búsqueda Rápida de Correspondencia	
Actores	Director	
Propósito	Proporciona Información ágil de la Correspondencias Registradas.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.

	Elije una de las opciones de información rápida.	Despliega la lista de correspondencias realiza búsquedas según el campo que vea usted pertinente.
--	--------------------------------------------------	---------------------------------------------------------------------------------------------------

Tabla 3.5 Descripción de caso de uso: Búsqueda Rápida de Correspondencia

Fuente: (Elaboración propia)

Descripción de caso de uso: Reporte de Correspondencia		
Nombre	Reporte de Correspondencia	
Actores	Director	
Propósito	Facilita rápidamente información de la correspondencia.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresar al sistema	Verifica la Sesión
	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije una de las opciones de información rápida.	Despliega la lista de correspondencias realiza búsquedas según el campo que vea usted pertinente, como acción debe elegir Historial de la correspondencia, y pulsar reporte.

Tabla 3.6 Descripción de caso de uso: Reporte de Correspondencia

Fuente: (Elaboración propia)

3.2.1.2 Diagrama de Casos de uso de Técnico académico de kardex, Auxiliar de kardex, Secretaria y Auxiliar de Oficina

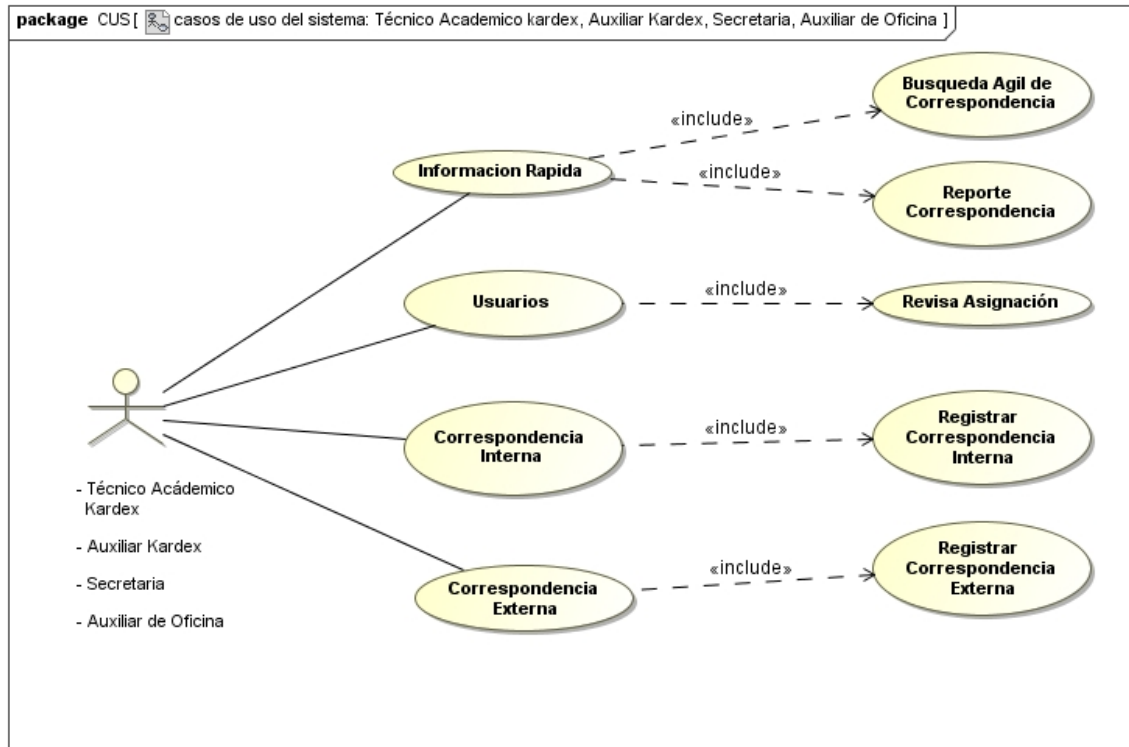


Figura 3.10 Diagrama de Casos de uso de Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina

Fuente:(Elaboración Propia)

Descripciones de casos de uso: Técnico Académico Kardex, Auxiliar de kardex, Secretaria y Auxiliar de Oficina

Descripción de caso de uso: Información Rápida	
Nombre	Información rápida
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina

Propósito	Proporciona Información ágil de la Correspondencias Registradas.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresar al sistema	Verifica la Sesión
	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elige una de las opciones de información rápida.	Despliega la lista de correspondencias realiza búsquedas según el campo que vea usted pertinente.

Tabla 3.7 Descripción de caso de uso: Información Rápida

Fuente: (Elaboración propia)

Descripción de caso de uso: Búsqueda Ágil de Correspondencia		
Nombre	Búsqueda Ágil de Correspondencia	
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina	
Propósito	Proporciona Información ágil de la Correspondencias Registradas.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresar al sistema	Verifica la Sesión

	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije una de las opciones de información rápida.	Despliega la lista de correspondencias realiza búsquedas según el campo que vea usted pertinente.

Tabla 3.8 Descripción de caso de uso: Búsqueda Ágil de Correspondencia

Fuente: (Elaboración propia)

Descripción de caso de uso: Reporte Correspondencia		
Nombre	Reporte Correspondencia	
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina	
Propósito	Facilita rápidamente información de la correspondencia.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresar al sistema	Verifica la Sesión
	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije una de las opciones de información rápida.	Despliega la lista de correspondencias realiza búsquedas según el campo que vea usted pertinente, como acción debe elegir Historial de la correspondencia, y pulsar reporte.

Tabla 3.9 Descripción de caso de uso: Reporte de Correspondencia

Fuente: (Elaboración propia)

Descripción de caso de uso: Usuario		
Nombre	Usuario	
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina	
Propósito	Revisar su bandeja de entrada y ver su designación.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Escoge en el menú y panel la opción de Usuarios.	Despliega la lista de tareas pendientes según su derivación.

Tabla 3.10 Descripción de caso

de uso: Usuario

Fuente: (Elaboración propia)

Descripción de caso de uso: Revisa Asignación	
Nombre	Revisa Asignación
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina
Propósito	Revisar la asignación de correspondencia.

	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije una de las opciones usuario, luego Bandeja de entrada.	Se desplegara la bandeja de entrada donde encontrara si tiene una designación pendiente y dar prioridad de urgente o normal a dicha correspondencia.

Tabla 3.11 Descripción de caso de uso: Revisar Asignación

Fuente: (Elaboración propia)

Descripción de caso de uso: Correspondencia Interna		
Nombre	Correspondencia Interna	
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina	
Propósito	Información de correspondencias Internas.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Ingresa al menú de Correspondencia Interna	Despliega la lista y detalles de los Correspondencia Interna.

	Elije la opción adicionar nueva correspondencia.	Despliega un formulario de registro donde debe ingresar datos requeridos.
	Ingresa a la información de la correspondencia registrada lista para su designación y guardado.	Valida los formularios desplegando mensajes de alerta de la acción que realizo.

Tabla 3.12 Descripción de caso de uso: Correspondencia Interna

Fuente:(Elaboración Propia)

Descripción de caso de uso: Registrar Correspondencia Interna		
Nombre	Registrar Correspondencia Interna	
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina	
Propósito	Registrar Correspondencias Internas	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije en el menú la opción Correspondencia Interna, luego adicionar nueva Correspondencia.	El sistema comprueba la validez de los datos, si los datos no son correctos se avisa al actor de ello, permitiéndole que los corrija.

Tabla 3.13 Descripción de caso de uso: Registrar Correspondencia Interna

Fuente:(Elaboración Propia)

Descripción de caso de uso: Correspondencia Externa		
Nombre	Correspondencia Externa	
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina	
Propósito	Información de correspondencias Externas.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Ingresar al menú de Correspondencia Externa	Despliega la lista y detalles de los Correspondencia Externa.
	Elige la opción adicionar nueva correspondencia.	Despliega un formulario de registro donde debe ingresar datos requeridos.
	Ingresar a la información de la correspondencia registrada lista para su designación y guardado.	Valida los formularios desplegando mensajes de alerta de la acción que realizo.

Tabla 3.14 Descripción de caso de uso: Correspondencia Externa

Fuente:(Elaboración Propia)

Descripción de caso de uso: Registrar Correspondencia Externa		
Nombre	Registrar Correspondencia Externa	
Actores	Técnico Académico Kardex, Auxiliar de Kardex, Secretaria y Auxiliar de Oficina	
Propósito	Registrar Correspondencias Externa	
	Evento del Actor	Evento del Sistema

Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elige en el menú la opción Correspondencia Externa, luego adicionar nueva Correspondencia.	El sistema comprueba la validez de los datos, si los datos no son correctos se avisa al actor de ello, permitiéndole que los corrija.

Tabla 3.15 Descripción de caso de uso: Registrar Correspondencia Externa

Fuente:(Elaboración Propia)

3.2.1.3 Diagrama de Casos de uso del Administrador

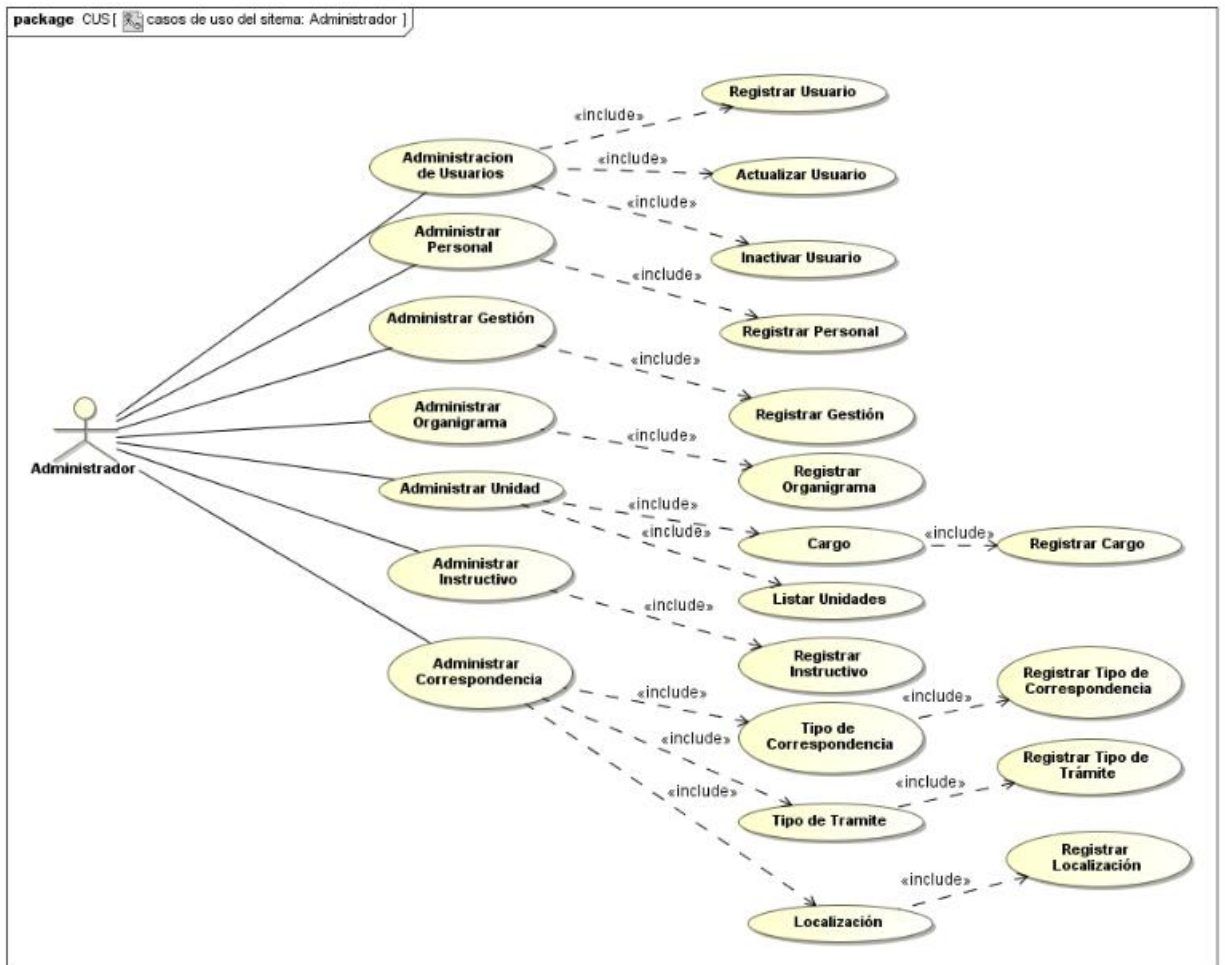


Figura 3.11 Diagrama de Casos de uso del Administrador

Fuente:(Elaboración Propia)

Descripciones de casos de uso: Administrador

Descripción de caso de uso: Administrar Usuarios		
Nombre	Administrar Usuarios	
Actores	Administrador	
Propósito	Gestionar datos de usuarios como activar, registrar, modificar, eliminar.	
	Evento del Actor	Evento del Sistema

Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Elegir la opción de Administrador Usuarios.	Visualiza el panel o interfaz seleccionado para la administración de usuarios.

Tabla 3.16 Descripción de caso de uso: Administrar Usuarios

Fuente:(Elaboración Propia)

Descripción de caso de uso: Registrar Usuario		
Nombre	Registrar Usuario	
Actores	Administrador	
Propósito	Registrar los datos de los Usuarios	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije en el menú la opción Administrador Usuarios, luego crear nuevo Usuario.	El sistema comprueba la validez de los datos, si los datos no son correctos se avisa al actor de ello permitiéndole que los corrija.

Tabla 3.17 Descripción de caso de uso: Registrar Usuario

Fuente:(Elaboración Propia)

Descripción de caso de uso: Actualizar Usuario		
Nombre	Actualizar Usuario	
Actores	Administrador	
Propósito	Actualizar los datos de los Usuarios	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije en el menú la opción Administrar Usuarios, luego Editar Usuario.	El sistema comprueba la validez de los datos, si los datos no son correctos se avisa al actor de ello permitiéndole que los corrija.

Tabla 3.18 Descripción de caso de uso: Actualizar Usuario

Fuente:(Elaboración Propia)

Descripción de caso de uso: Inactivar Usuario		
Nombre	Inactivar Usuario	
Actores	Administrador	
Propósito	Eliminar y dar Inactivo los datos de los Usuarios	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elije en el menú la opción Administrar Usuarios, luego Dar Inactivar Usuario.	El sistema comprueba la validez de los datos, si los datos no son correctos se

		avisa al actor de ello permitiéndole que los corrija.
--	--	-------------------------------------------------------

Tabla 3.19 Descripción de caso de uso: Actualizar Usuario

Fuente:(Elaboración Propia)

Descripción de caso de uso: Administrar Personal		
Nombre	Administrar Personal	
Actores	Administrador	
Propósito	Gestionar datos del Personal como registrar, modificar, eliminar.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Elegir la opción del Módulo Personal.	Visualiza el panel o interfaz seleccionado para la administración del Personal.

Tabla 3.20 Descripción de caso de uso: Administrar Personal

Fuente:(Elaboración Propia)

Descripción de caso de uso: Registrar Personal		
Nombre	Registrar Personal	
Actores	Administrador	
Propósito	Registrar los datos del Personal	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresa al sistema	Verifica la Sesión

	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al módulo.
	Elige en el menú la opción Modulo Personal, luego Anadir personal.	El sistema comprueba la validez de los datos, si los datos no son correctos se avisa al actor de ello permitiéndole que los corrija.

Tabla 3.21 Descripción de caso de uso: Registrar Personal

Fuente:(Elaboración Propia)

Descripción de caso de uso: Administrar Gestión		
Nombre	Administrar Gestión	
Actores	Administrador	
Propósito	Registrar, Eliminar, editar la Gestión.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresar al sistema	Verifica la Sesión
	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Escoge en el menú Principal el Modulo de Gestión.	Despliega la lista de las gestiones registradas, tomando como opciones también editar o eliminar.

Tabla 3.22 Descripción de caso de uso: Administrar Gestión

Fuente:(Elaboración Propia)

Descripción de caso de uso: Administrar Organigrama		
Nombre	Administrar Organigrama	
Actores	Administrador	
Propósito	Registrar, Eliminar, editar el Organigrama.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresar al sistema	Verifica la Sesión
	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Escoge en el menú principal Organigrama.	Despliega la lista del organigrama planteado y registrado, tomando como opciones también editar o eliminar.

Tabla 3.23 Descripción de caso de uso: Administrar Organigrama

Fuente:(Elaboración Propia)

Descripción de caso de uso: Administrar Unidad		
Nombre	Administrar Unidad	
Actores	Administrador	
Propósito	Registrar, Eliminar, editar la Unidad.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresar al sistema	Verifica la Sesión
	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Escoge en el menú Principal el Modulo de Unidad.	Despliega la lista de las unidades registradas,

		tomando como opciones también editar o eliminar.
--	--	--------------------------------------------------

Tabla 3.24 Descripción de caso de uso: Administrar Unidad

Fuente:(Elaboración Propia)

Descripción de caso de uso: Cargo		
Nombre	Cargo	
Actores	Administrador	
Propósito	Registrar, Eliminar, editar el Cargo.	
	Evento del Actor	Evento del Sistema
Flujo Principal	Ingresar al sistema	Verifica la Sesión
	Ingresar usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Escoge en el menú Principal el Modulo de Unidad, se desplegara e ingresar a la pestaña Cargo.	Despliega la lista de los Cargos registrados, tomando como opciones también editar o eliminar.

Tabla 3.25 Descripción de caso de uso: Cargo

Fuente:(Elaboración Propia)

Descripción de caso de uso: Administrar Instructivo		
Nombre	Administrar Instructivo	
Actores	Administrador	
Propósito	Registrar, Eliminar, editar el Instructivo.	
	Evento del Actor	Evento del Sistema

Flujo Principal	Ingresa al sistema	Verifica la Sesión
	Ingresa usuario contraseña	Inicia sesión y despliega menú correspondiente al modulo
	Escoge en el menú de Principal el Modulo de Instructivo.	Despliega la lista de los Instructivos registradas, tomando como opciones también editar o eliminar.

Tabla 3.26 Descripción de caso de uso: Administrar Unidad

Fuente:(Elaboración Propia)

3.2.2 Diagrama de secuencia del sistema

Los Diagramas de Secuencias muestran la forma en que un grupo de objetos se comunican, entre sí a lo largo del tiempo, que consta de objetos, mensajes entre estos objetos y una línea de vida del objeto representado por una línea vertical

3.2.2.1 Diagrama de secuencia del sistema: Registrar Usuario

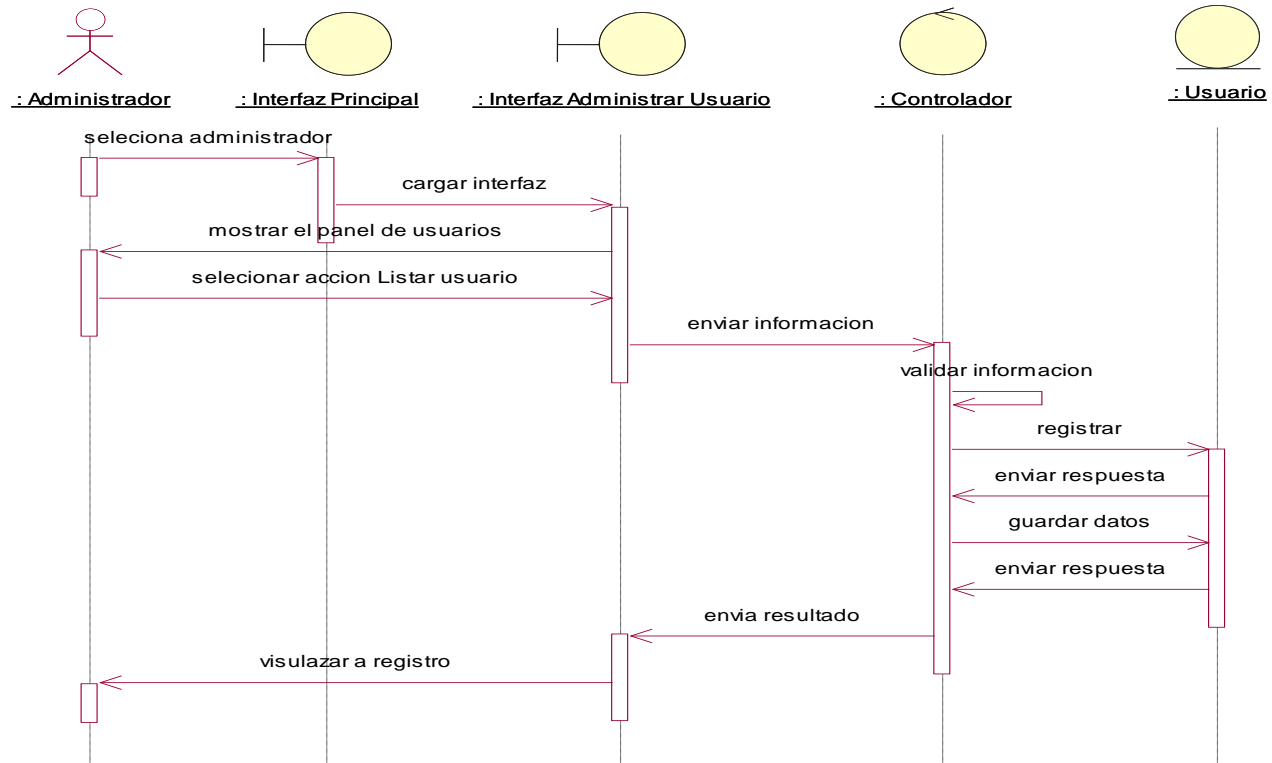


Figura 3.15 Diagrama de secuencia del sistema: Registrar Usuario

Fuente:(Elaboración Propia)

3.2.2.2 Diagrama de secuencia del sistema: Registrar Organigrama

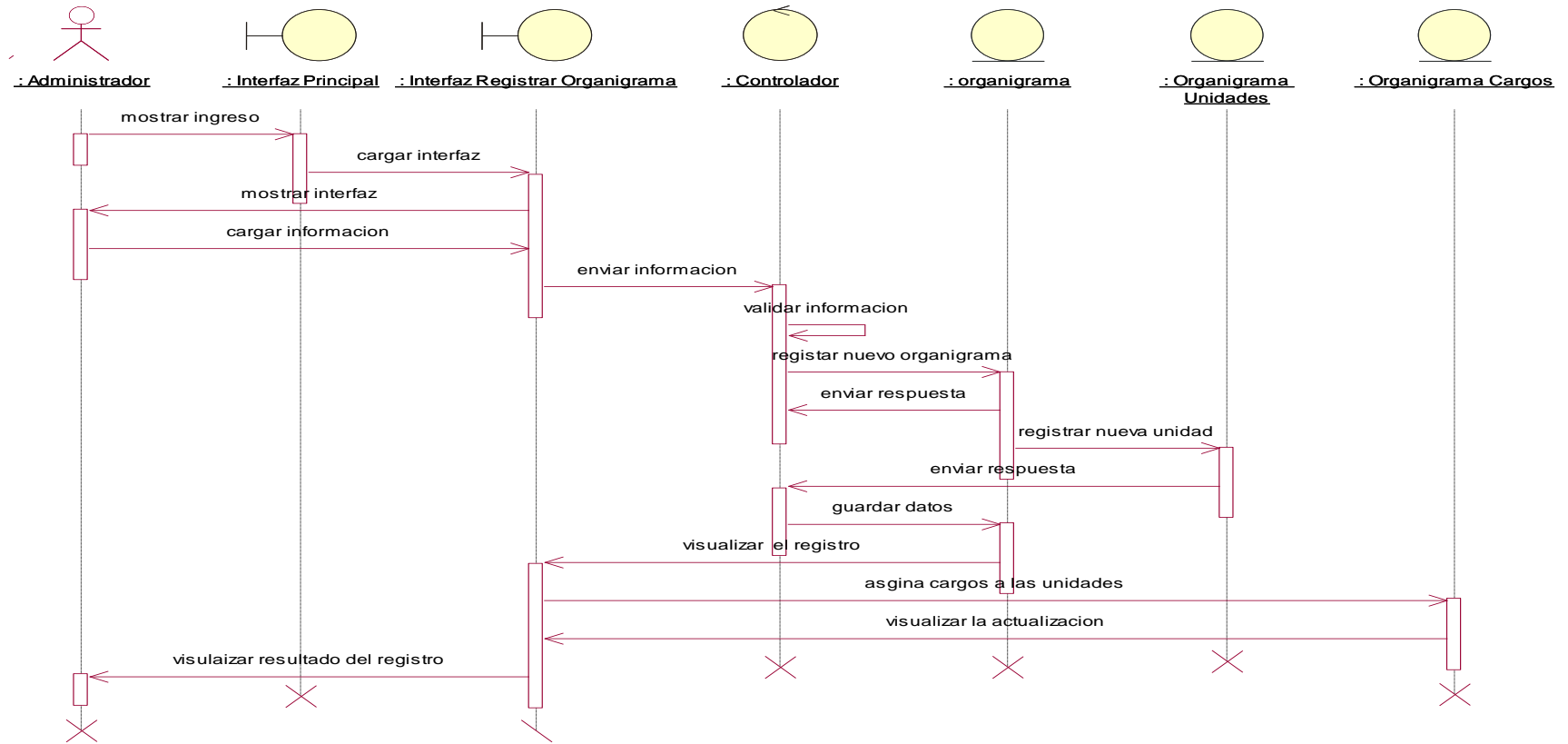


Figura 3.16 Diagrama de secuencia del sistema: Registrar Organigrama

Fuente:(Elaboración Propia)

3.2.2.3 Diagrama de secuencia del sistema: Registrar Tipo de Correspondencia

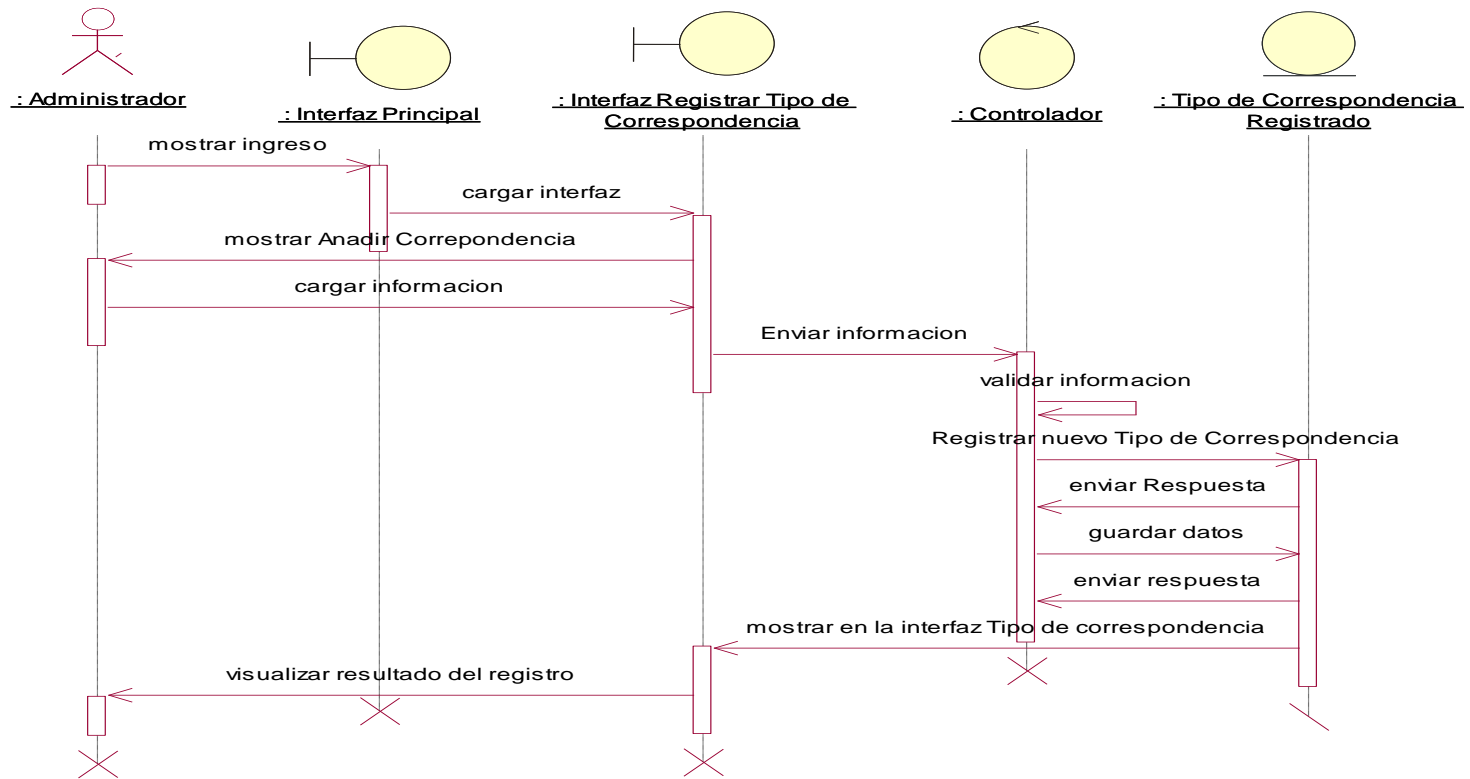


Figura 3.17 Diagrama de secuencia del sistema: Registrar Tipo de Correspondencia

Fuente:(Elaboración Propia)

3.2.2.4 Diagrama de secuencia del sistema: Registrar Correspondencia Interna

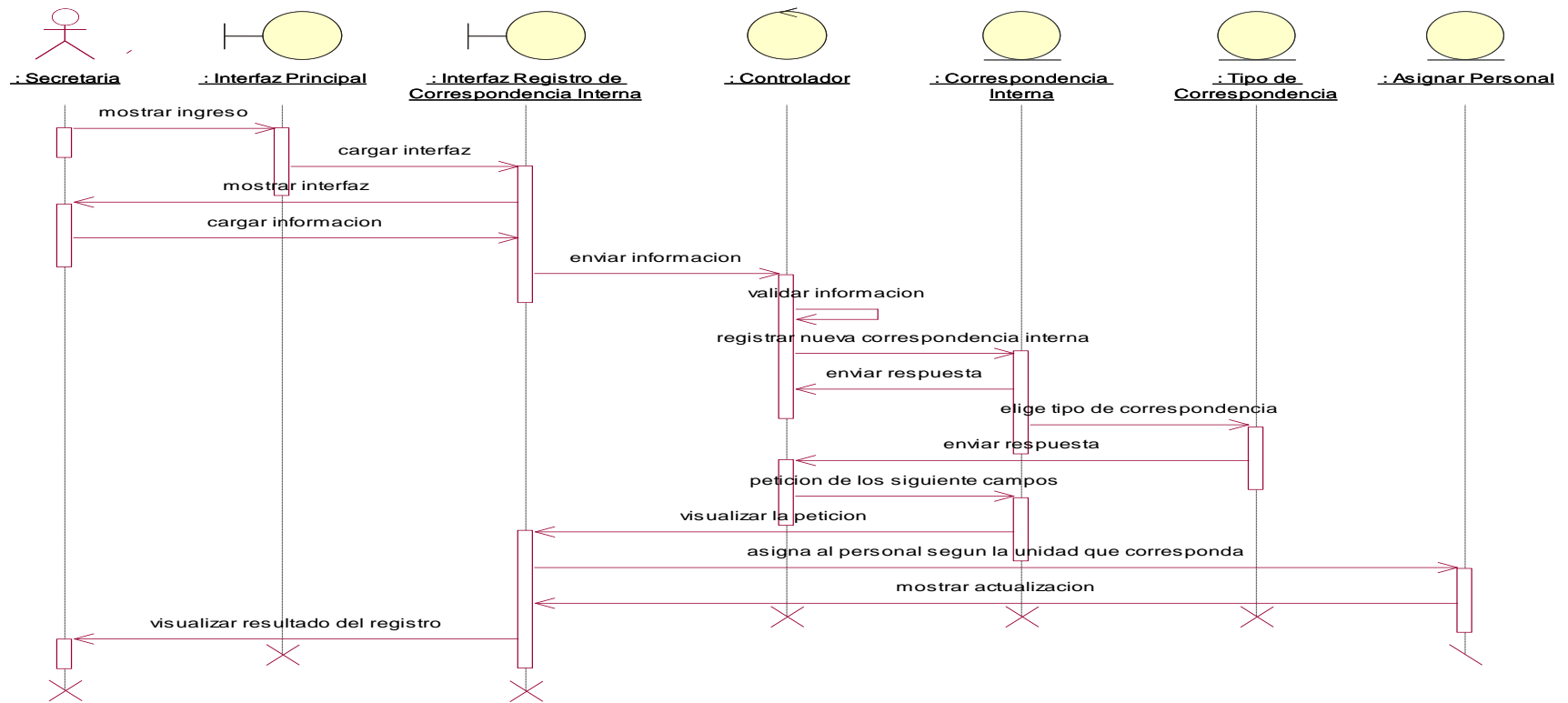


Figura 3.18 Diagrama de secuencia del sistema: Registrar Correspondencia Interna

Fuente:(Elaboración Propia)

3.2.2.5 Diagrama de secuencia del sistema: Registrar Instrucción

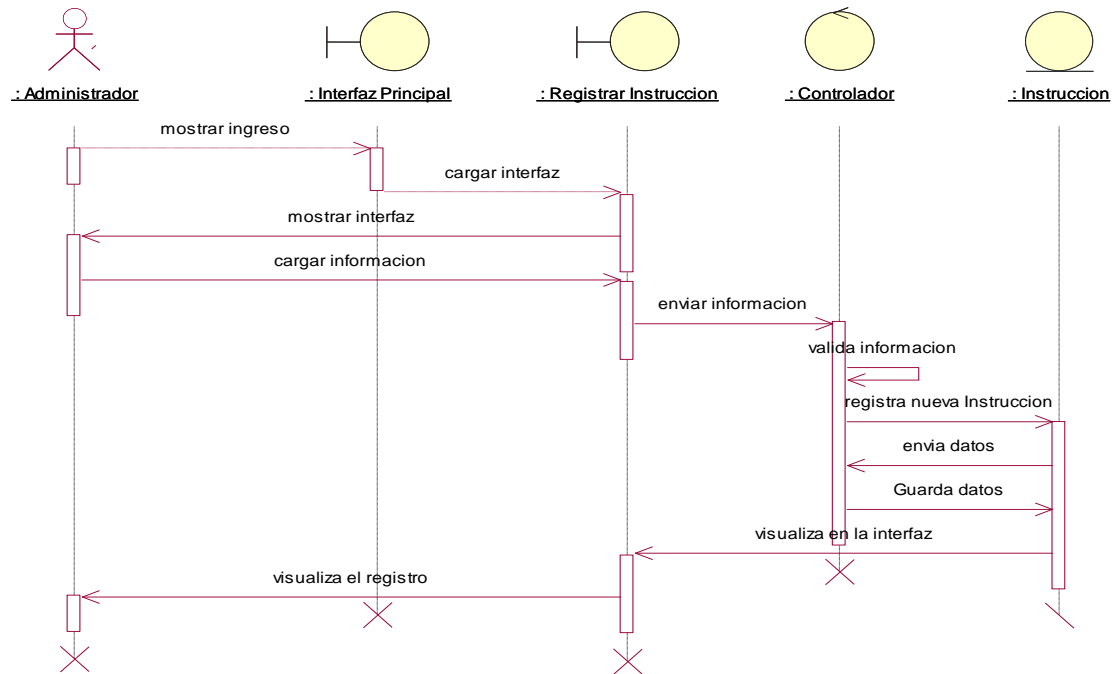


Figura 3.20 Diagrama de secuencia del sistema: Registrar Instrucción

Fuente:(Elaboración Propia)

3.2.3 Diagrama de colaboración del sistema

3.2.3.1 Diagrama de colaboración: Registrar Usuario

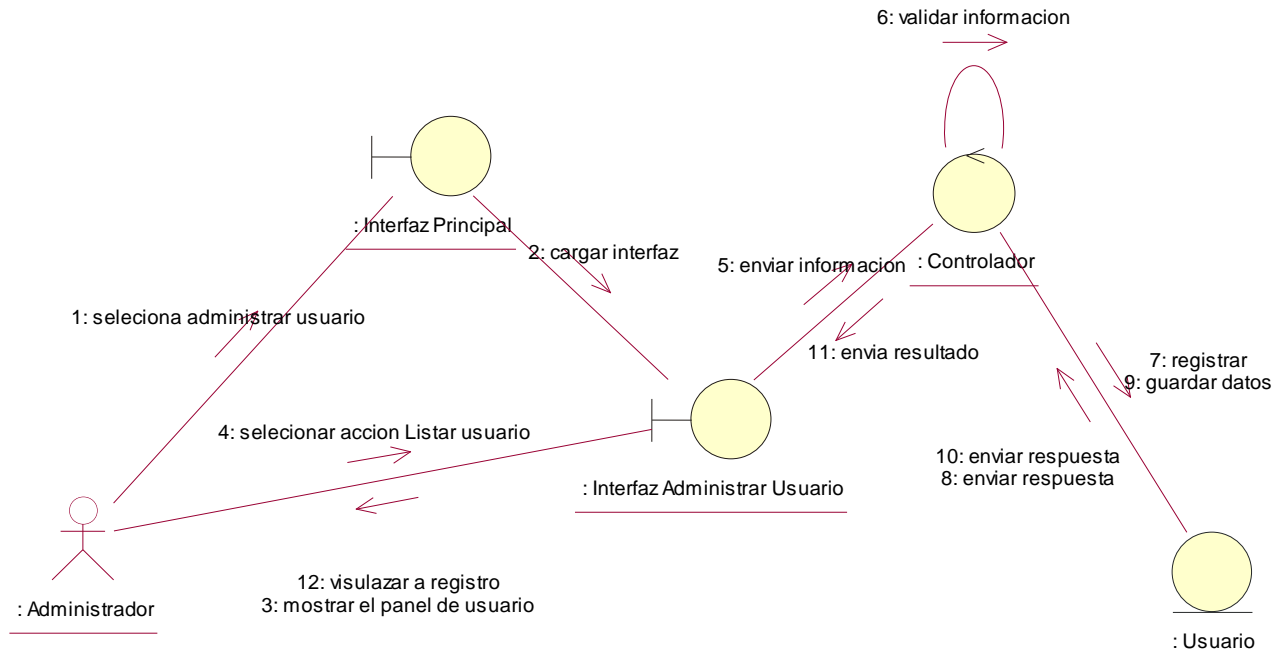


Figura 3.21 Diagrama de colaboración: Registrar Usuario

Fuente:(Elaboración Propia)

3.2.3.2 Diagrama de colaboración: Registrar Organigrama

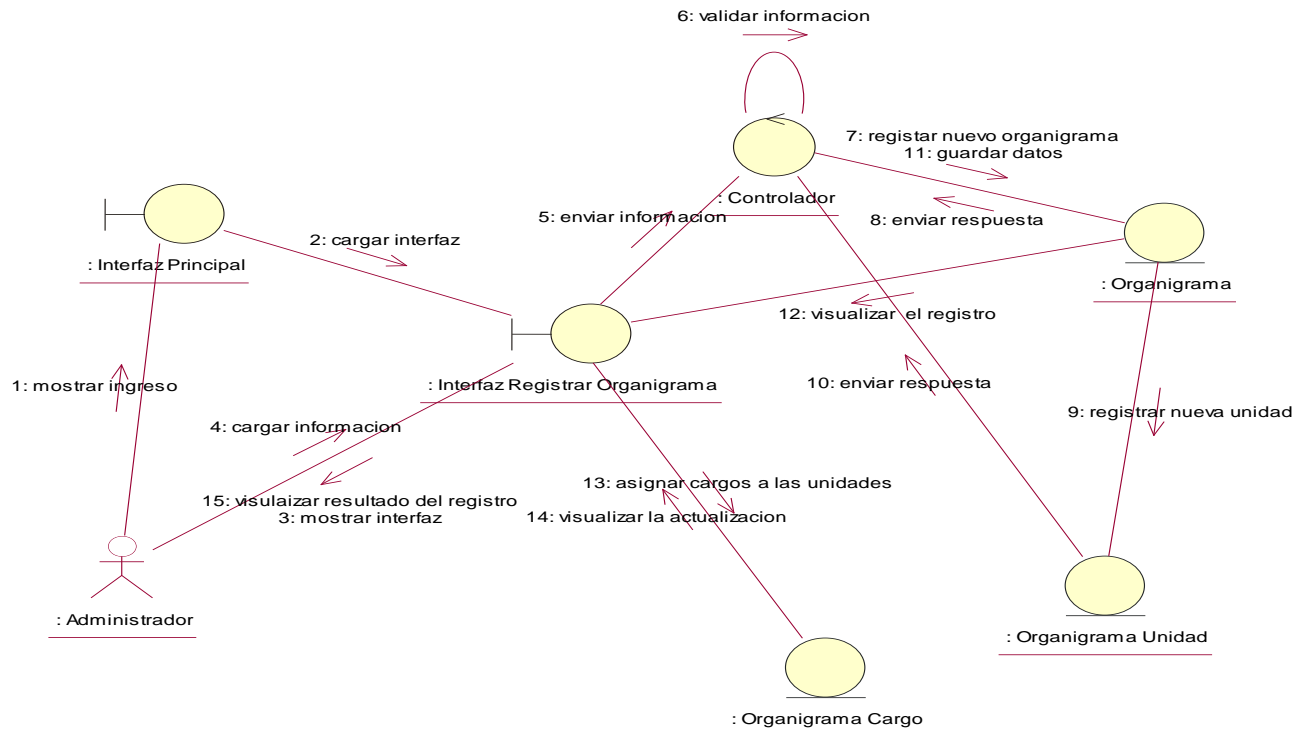


Figura 3.22 Diagrama de colaboración: Registrar Organigrama

Fuente:(Elaboración Propia)

3.2.3.3 Diagrama de colaboración: Registrar Tipo de Correspondencia

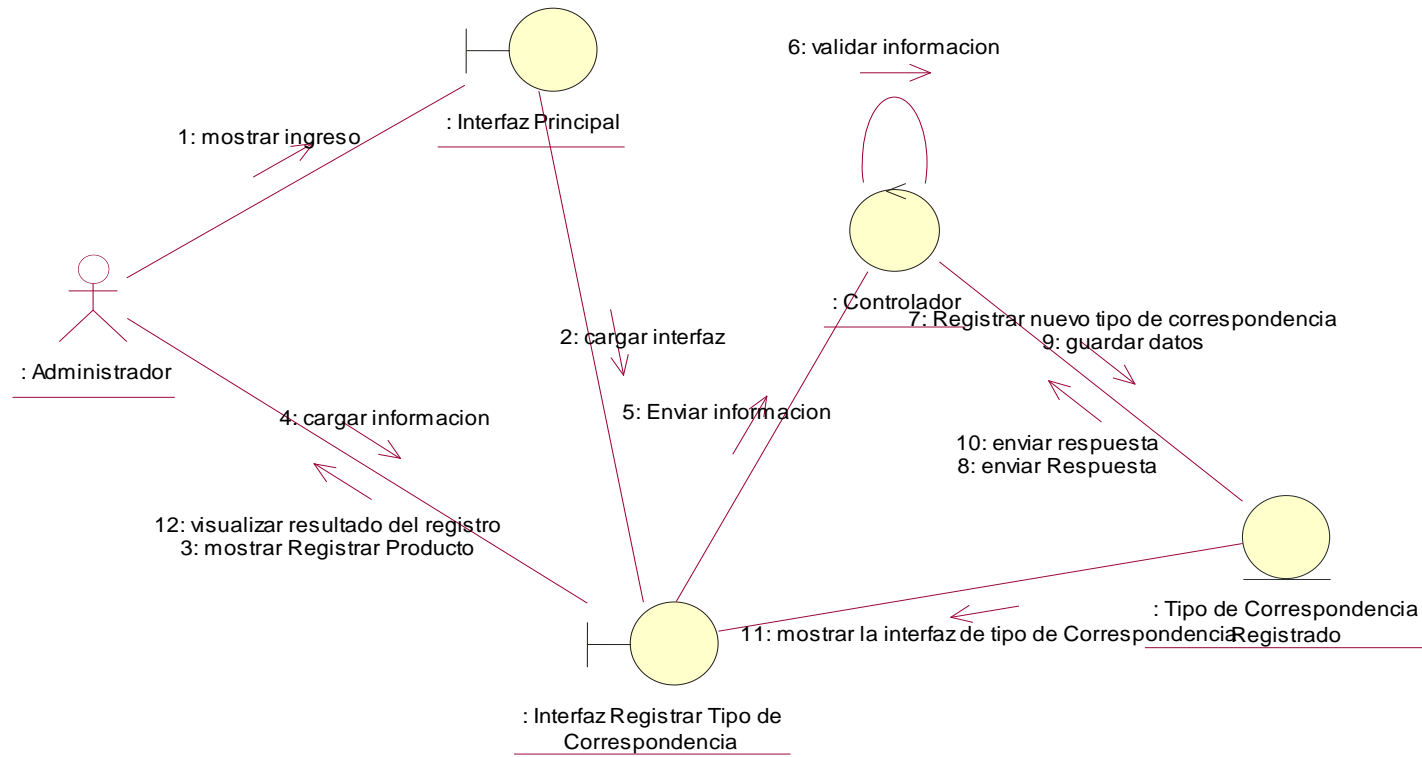


Figura 3.23 Diagrama de colaboración: Registrar Tipo de Correspondencia

Fuente:(Elaboración Propia)

3.2.3.4 Diagrama de colaboración: Registrar Correspondencia Interna

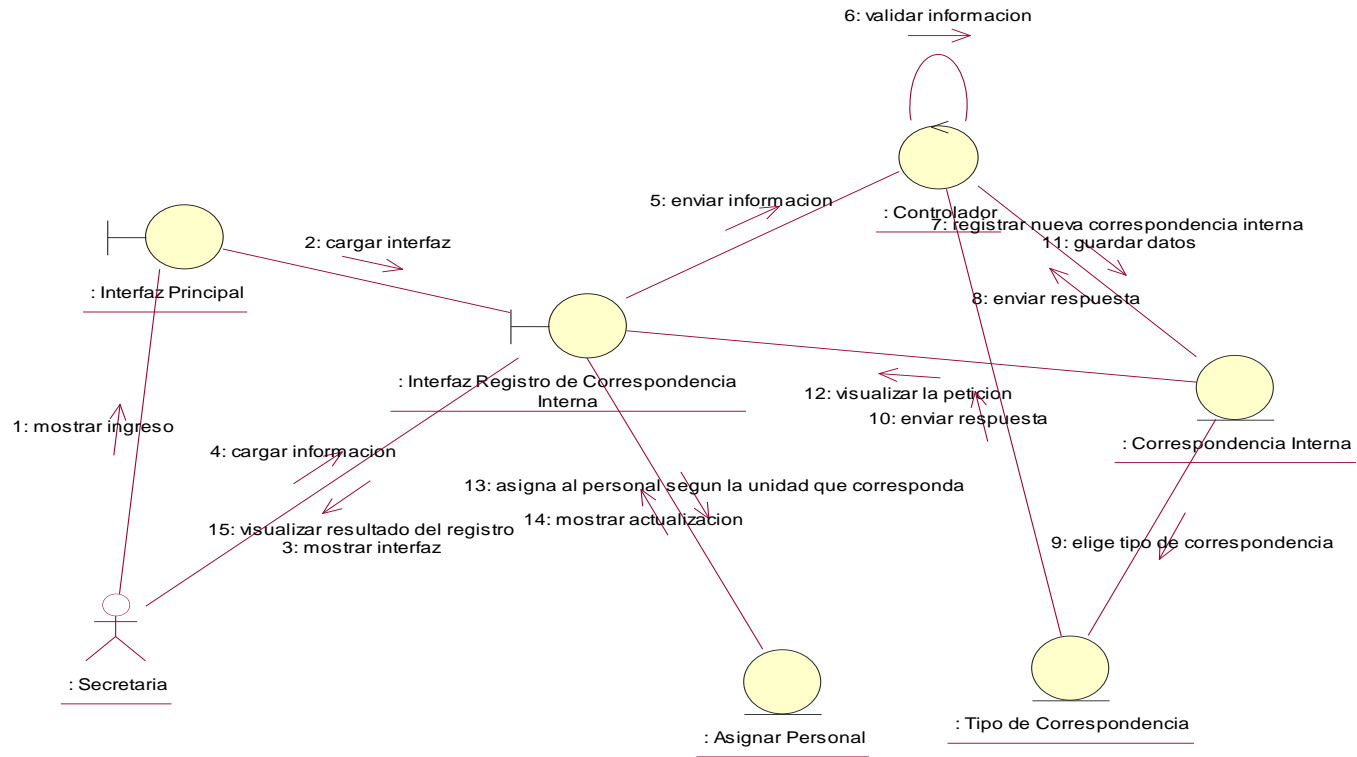


Figura 3.24 Diagrama de colaboración: Registrar Correspondencia Interna

Fuente:(Elaboración Propia)

3.2.3.5 Diagrama de colaboración: Registrar Instrucción

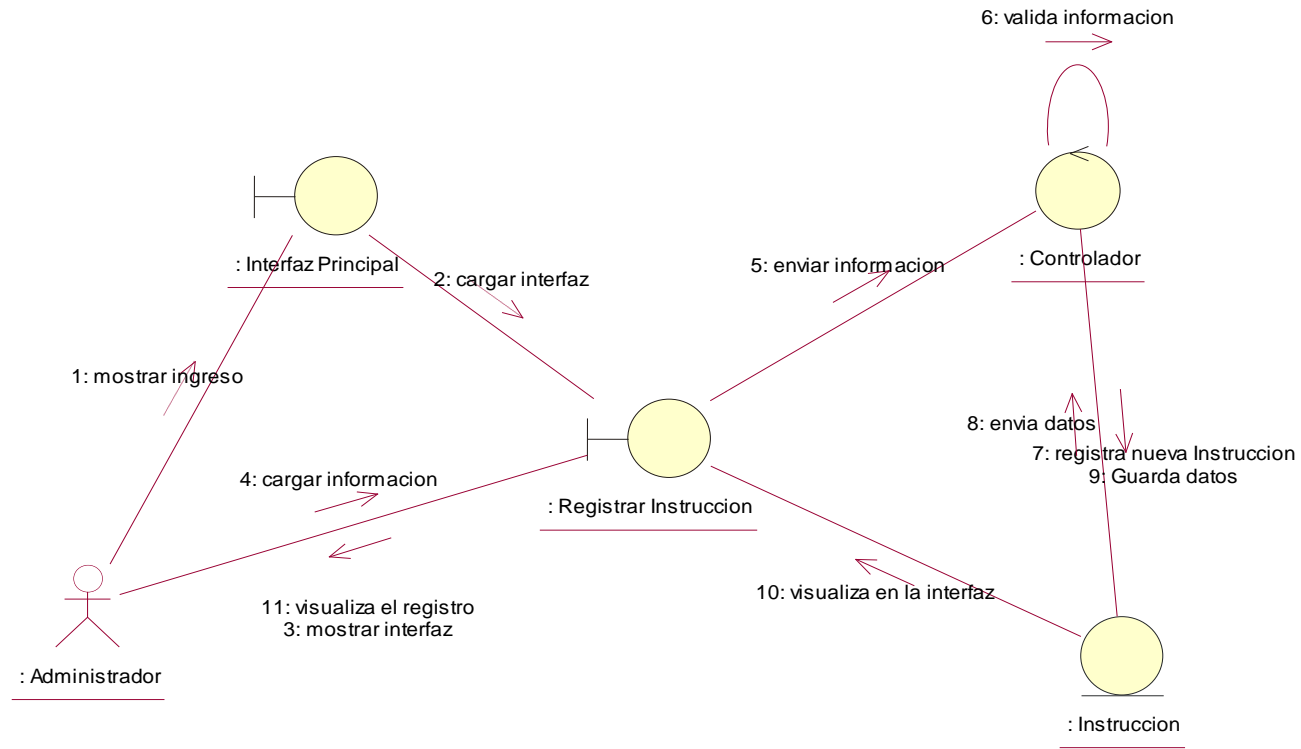


Figura 3.24 Diagrama de colaboración: Registrar Instrucción

Fuente:(Elaboración Propia)

iii. FASE DE CONSTRUCCIÓN

3.3 DISEÑO DEL SISTEMA

3.3.1 Modelo conceptual del dominio del sistema

Se materializado considerando los requisitos reflejados en los casos de uso.

“La identificación de clases del dominio, se obtiene principalmente de algún documento textual. Aunque se pudiera tomar como punto de partida los documentos desarrollados para el modelo de casos de uso, a menudo la descripción original del problema es suficiente” (Weitzenfeld, 2005)

3.3.2 Modelo de navegación

Estos tienen por cometido ilustrar los vínculos lógicos de navegación entre clases. Por tratarse de una aplicación orientada a la web es necesario modelar la navegación de la aplicación.

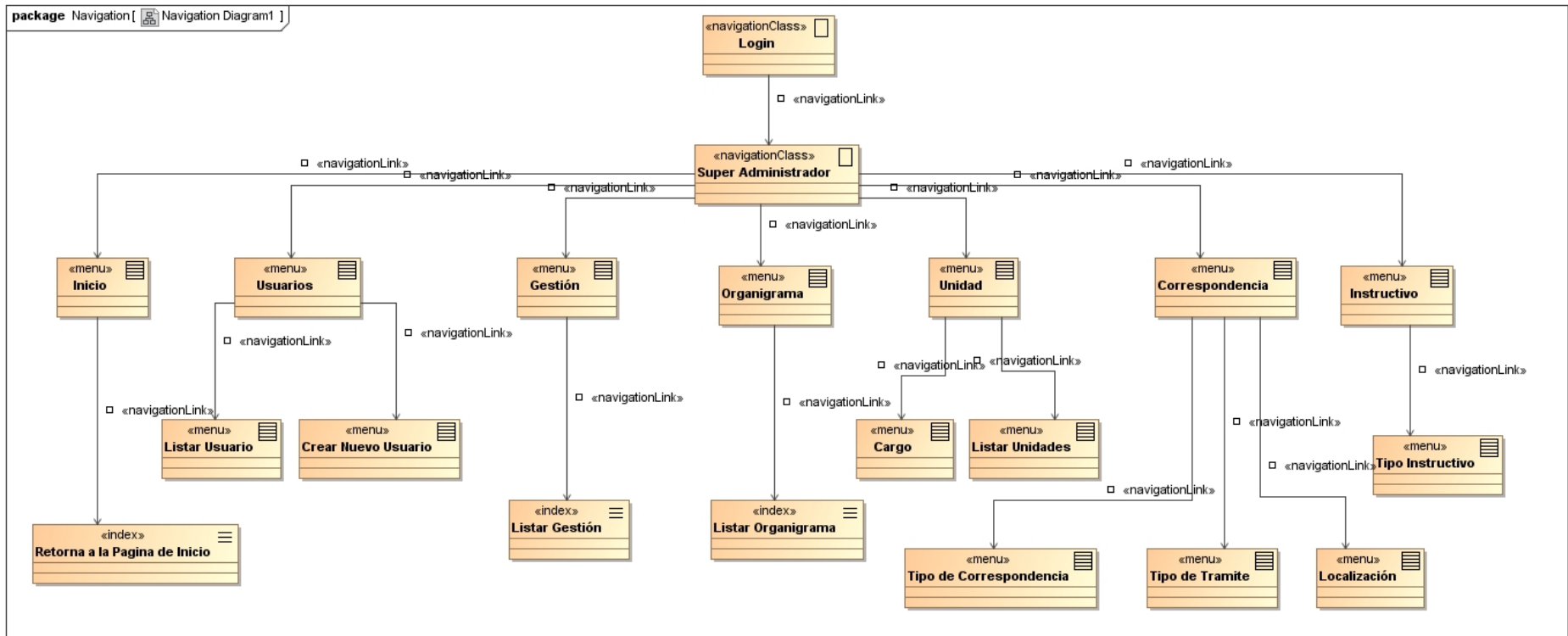


Figura 3.27 Modelo de Navegación de Menús (Administrador)

Fuente: (Elaboración Propia)

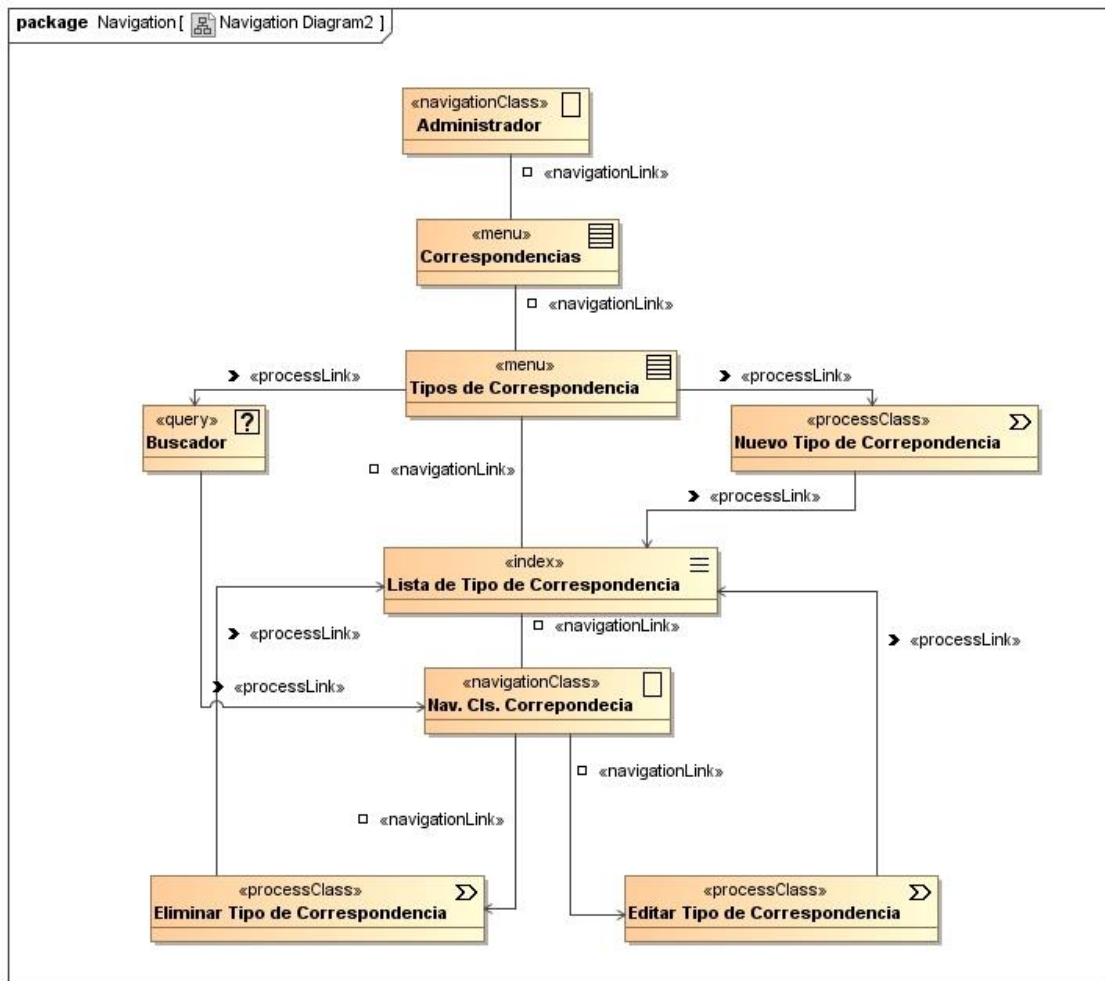


Figura 3.29 Modelo de Navegación del Administrador (Menú Correspondencia)

Fuente: (Elaboración propia)

3.3.3 Modelo de Presentación

A continuación se muestran los modelos de presentación, según UWE propone para la construcción de las páginas en forma de bosquejos.

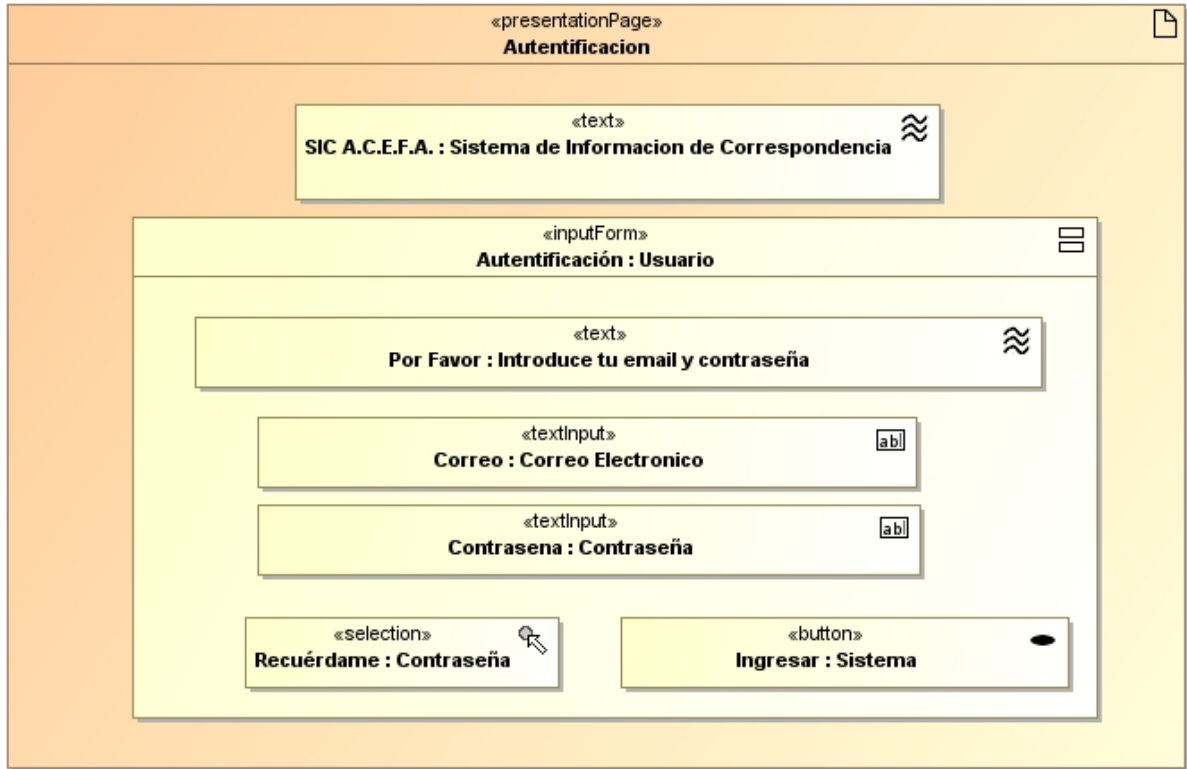


Figura 3.30 Modelo de Presentación Autenticación (Iniciar Sesión)

Fuente: (Elaboración propia)

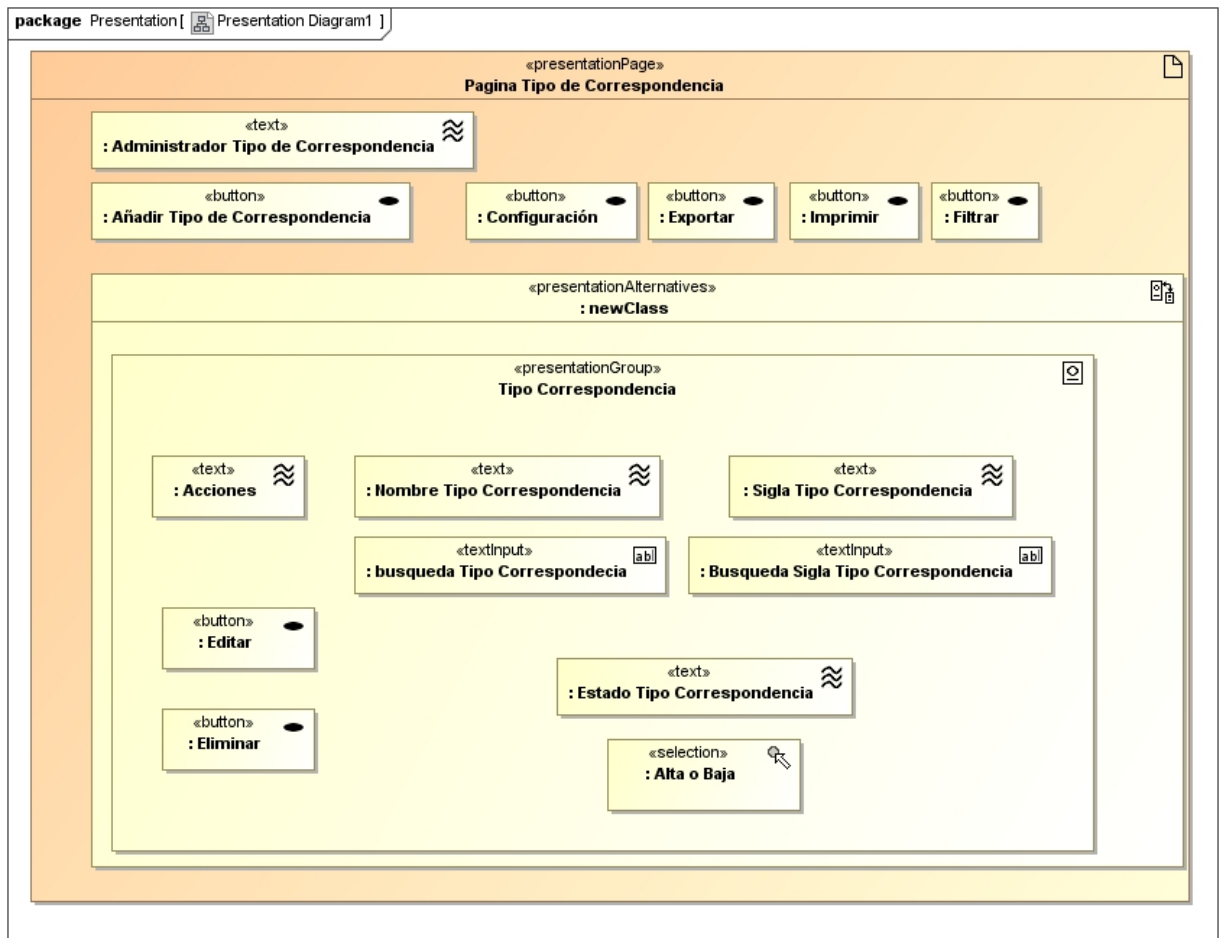


Figura 3.31 Modelo de Presentación de Registro Tipo de Correspondencia

Fuente: (Elaboración propia)

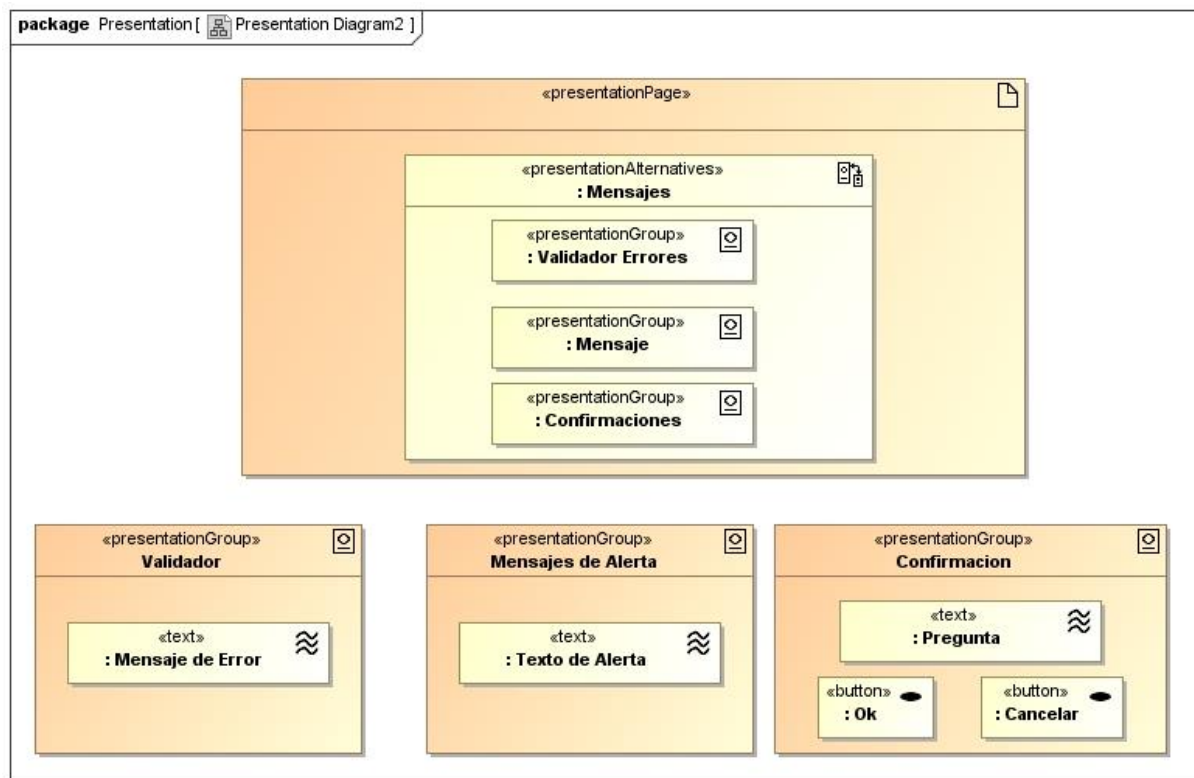


Figura 3.32 Modelo de Presentación de Mensajes, Confirmaciones, Validador de Errores.

Fuente: (Elaboración propia)

3.3.4 Modelo de Implementación

En esta fase de implementación consiste en mostrar el desarrollo de la presentación de las interfaces del sistema y sus elementos construidos a partir del diseño del modelo de presentación UWE, para el sistema tenemos siguiente secuencia de pantallas básicas.

En la figura 3.33 se muestra el fondo de Área de ciencias económicas financieras, en el centro el formulario de Inicio de sesión del sistema, como campos principales el correo electrónico del usuario y su respectiva contraseña.

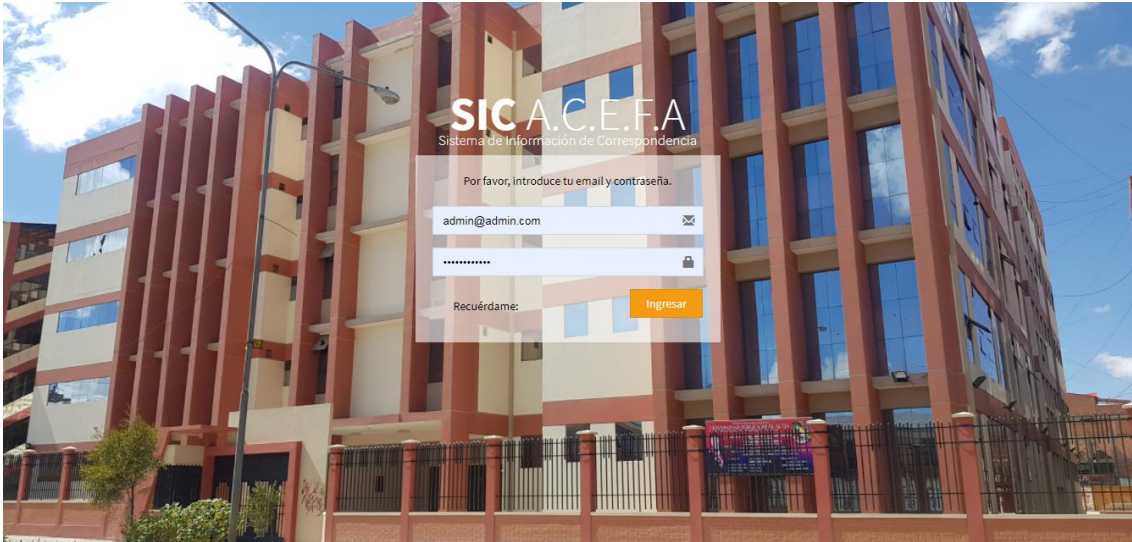


Figura 3.33 Formulario de Inicio de Sesión

Fuente: (Elaboración Propia)

En la figura 3.34 se inicia sesión como usuario súper Administrador mostrando el menú de opciones generales a la cual puede acceder el administrador, se muestran todos los módulos, correspondiente de acuerdo al rol que se le asigno.

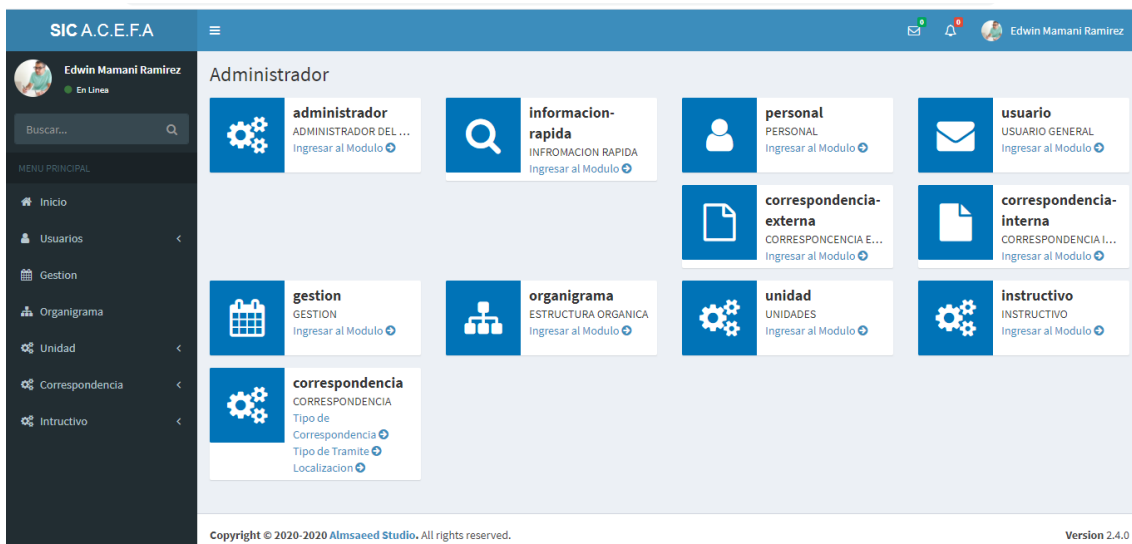


Figura 3.34 Pantalla Principal del Sistema

Fuente: (Elaboración Propia)

En la figura 3.35 Tomamos como ejemplo al módulo **Organigrama** donde muestra la interfaz y las acciones que podemos optar en este módulo, como Anadir Organigrama, ver el listado de los organigramas creados entre otras opciones.

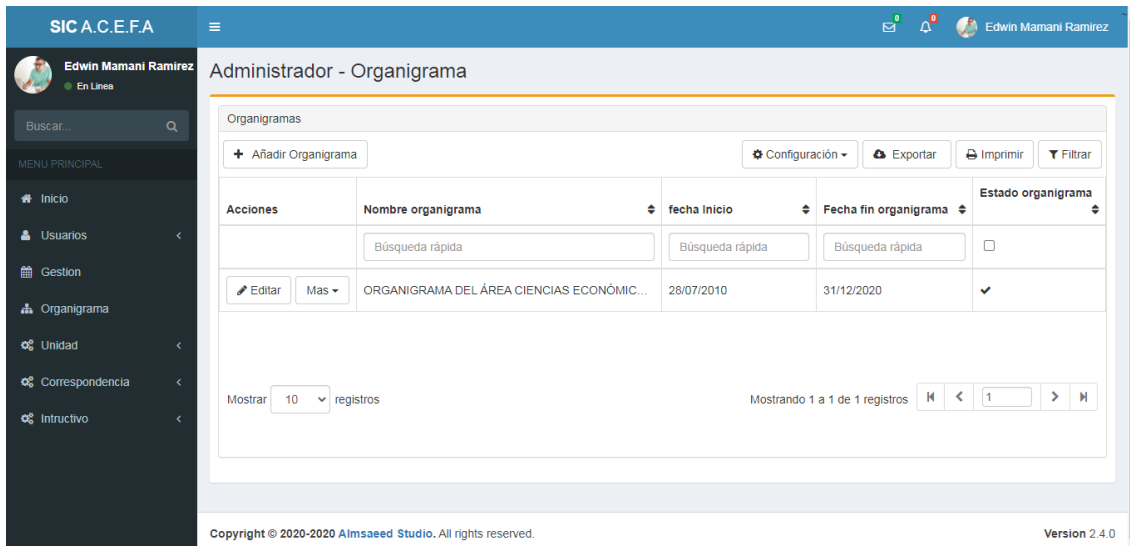


Figura 3.35 Interfaz del Módulo Organigrama

Fuente: (Elaboración Propia)

En la figura 3.36 podemos observar ya el Organigrama con sus respectivas unidades creadas y también sub unidades según la dependencia del caso.



Figura 3.36 Interfaz de las Unidades Creadas

Fuente: (Elaboración Propia)

En la figura 3.37 podemos observar el formulario de creación de Nueva Unidad como los campos de nombre de la unidad, sigla, unidad dependiente y por ultimo nivel, dando como opción a los botones cancelar o guardar el registro.

The image shows a web form titled "Adicionar Nueva Unidad" with a close button (x) in the top right corner. The form contains the following fields and controls:

- Nombre Unidad:** A text input field with the placeholder text "Nombre Unidad".
- Sigla:** A text input field with the placeholder text "Sigla".
- Unidad Dependencia:** A dropdown menu.
- Nivel:** A text input field with the placeholder text "nivel".
- Estado:** A checked checkbox.
- Buttons:** "Cancelar" (light gray) and "Guardar" (blue).

Figura 3.37 Interfaz de Formulario de Añadir Nueva Unidad

Fuente: (Elaboración Propia)

En la figura 3.38 se puede apreciar ya una unidad agregado los cargos que le corresponden, así mismo al personal asignado a su respectivo cargo.



Figura 3.38 Interfaz Unidad, Cargo y Personal

Fuente: (Elaboración Propia)

En la figura 3.39 se observa el listado de los cargos que hay en dicha Unidad anteriormente creada, así mismo podemos observar la opción de asignar el personal.



Figura 3.39 Interfaz del listado de Cargos de una Unidad

Fuente: (Elaboración Propia)

En la figura 3.40 se observa la unidad y el cargo la cual es Auxiliar de Oficina con el personal ya asignado, también se puede registrar otro personal y dar de alta al nuevo personal.



Figura 3.40 Interfaz de Persona Asignado a un Cargo

Fuente: (Elaboración Propia)

En la figura 3.41 se observa el formulario de Añadir Asignación, donde el campo de persona se hace la búsqueda de la persona que asumirá el cargo, luego así mismo se completa los campos de fecha inicio asignación, fecha fin de asignación, tipo de asignación y por ultimo fecha de registro de asignación.

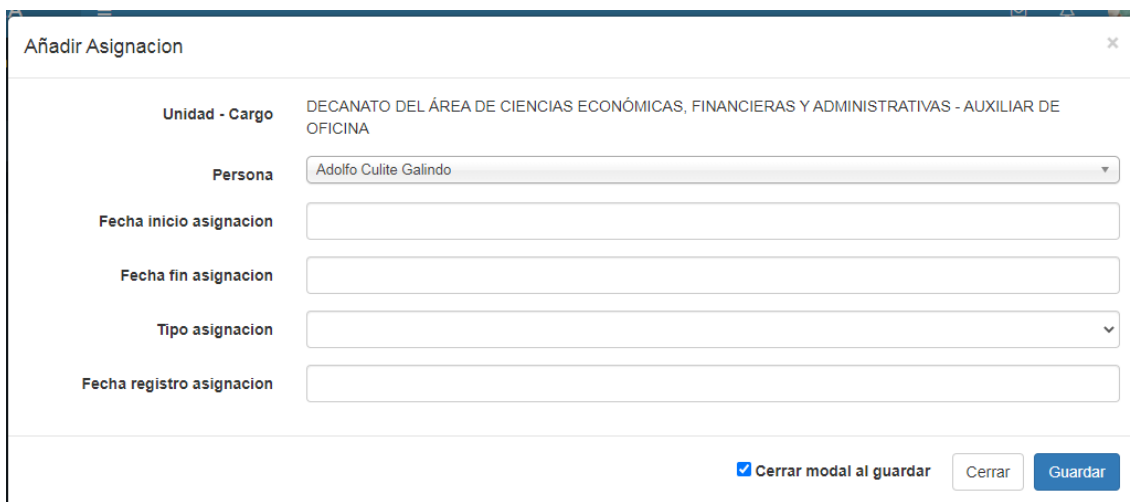


Figura 3.41 Interfaz Formulario Añadir Personal a Cargo

Fuente: (Elaboración Propia)

En la figura 3.42 se observa el módulo de **correspondencia Interna**, en el cual se encuentra un listado de las correspondencias registradas, también se muestra un botón verde donde dice historial el cual muestra un detalle del seguimiento de la correspondencia.

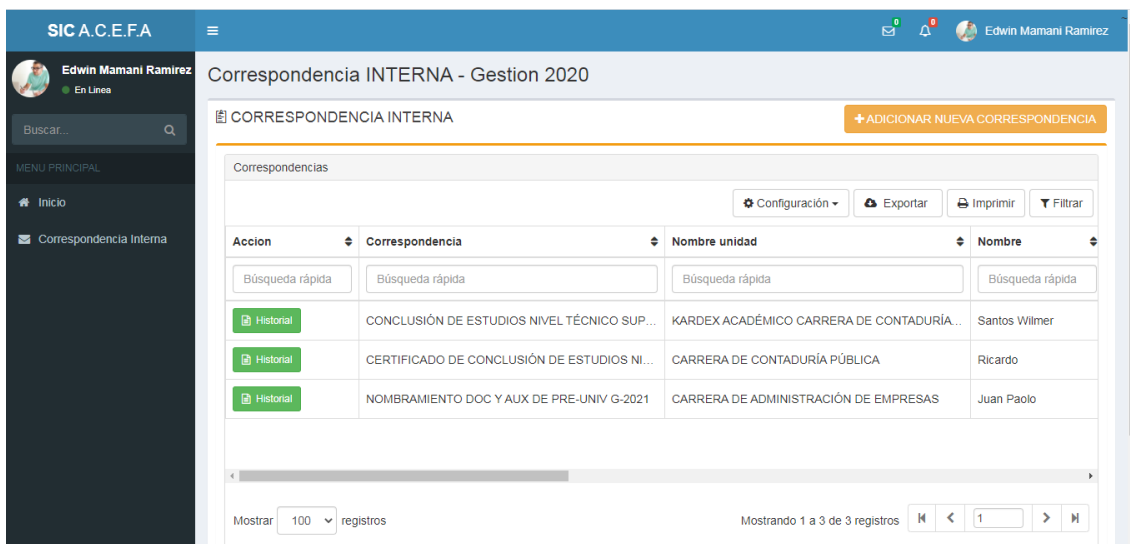


Figura 3.42 Interfaz principal de modulo Correspondencia Interna

Fuente: (Elaboración Propia)

En la figura 3.43 se muestra el formulario de Registro de Adicionar nueva Correspondencia, donde nos piden los siguientes campos, Tipo de Correspondencia, Numeración interna, Nombre del Remitente, Nr Antecedente de Tramite, Referencia, Nro Cite Remitente, Archivo Adjunto si es Necesario, por otra parte también Hay Datos Anexos que no son obligatoriamente requeridos.

Figura 3.43 Interfaz Formulario de Registro de Nueva Correspondencia Interna

Fuente: (Elaboración Propia)

En la figura 3.44 se observa la plantilla ya registrada del módulo de correspondencia interna, dando como opciones también a adicionar nueva copia e imprimir reporte.

Nº SUIT 1-2020		+ ADICIONAR NUEVA COPIA		Imprimir Reporte	
SOLICITUD DE EMISIÓN DE NOMBRAMIENTO DE DOCENTE Y AUXILIAR DE PRE-UNIVERSITARIO GESTIÓN 2021					
Nº SUIT	1-2020	Remitente	Juan Paolo Campero Lazarte		
Tipo Correspondencia	NOMBRAMIENTO DOC Y AUX DE PRE-UNIV G-2021	Localizacion			
Numeracion Interna	NDAPG21-153-2020	Cargo	DIRECTOR DE CARRERA		
Tipo Tramite	NOMBRAMIENTO	CI	4932228 LP		
Institucion	CARRERA DE ADMINISTRACIÓN DE EMPRESAS	Telefono	60106045		
Nº Cite	0142	Fecha Recepcion	2020-11-10 15:35:31		
Nº Antecedente de Tramite	Tramite sin antecedente	Archivo Adjunto	No tiene archivos adjuntos		

Figura 3.44 Interfaz de una Correspondencia Registrada

Fuente: (Elaboración Propia)

En la figura 3.45 se muestra la línea de tiempo que sigue dicho trámite, al mismo tiempo se observa la derivación y designación al personal responsable.



Figura 3.45 Interfaz de la Línea de Tiempo de Correspondencia

Fuente: (Elaboración Propia)

En la figura 3.46 se muestra el reporte del registro de la Correspondencia Interna, así también se observa la línea de tiempo que lleva dicho trámite, derivación y asignación del personal.

1-2020 1 / 1

HISTORIAL DE CORRESPONDENCIA AREA A.C.E.F.A.

Referencia: SOLICITUD DE EMISION DE NOMBRAMIENTO DE DOCENTE Y AUXILIAR DE PRE-UNIVERSITARIO GESTION 2021

No SUII: 1-2020 Remitante: Juan Pablo Campero Lazarte

Tipo Correspondencia: NOMBRAMIENTO DOCY AUX DE PRE-UNIV G-2021 Localizacion:

Tipo Trámite: NOMBRAMIENTO Cargo: DIRECTOR DE CARRERA

Numeracion Interna: NDAPQ21-153-2020 CI: 493228 LP

Institucion: CARRERA DE ADMINISTRACION DE EMPRESAS Telefono: 60106045

Nro Cita: 0142 Fecha Recepcion: 10/Nov/2020

Nro Antecedente Trámite:

Documentos Adjuntos:

ORIGINAL

RESPONSABLE	INSTRUCCION	No INTERNO	FECHA DERIVADO	FECHA PLAZO	ESTADO	OBSERVACION
Juan Pablo Campero Lazarte	ARCHIVAR		10/Nov/2020	10/Nov/2020	DERIVADO	
Mery Beatriz Hueda Bernal	PROCESAR DE ACUERDO A NOMBR	NDAPQ21-153-2020	10/Nov/2020	10/Nov/2020	DERIVADO	
José Giovanni Cordero Bernal	PROCESAR DE ACUERDO A NOMBR	NDAPQ21-153-2020	10/Nov/2020	10/Nov/2020	DERIVADO	
Marta Alicia Peralta Hernández	TRAB. SU ATENCION E INFORME EN PLAZO PRESIDENTE	NDAPQ21-153-2020	10/Nov/2020	17/Nov/2020	DERIVADO	
Juan Pablo Campero Lazarte	PARA SU ATENCION	NDAPQ21-153-2020	10/Nov/2020	09/Nov/2020	DERIVADO	
Mery Beatriz Hueda Bernal	ARCHIVAR		10/Nov/2020	10/Nov/2020	ASIGNADO	

Abrir en Acrobat

Figura 3.46 Reporte de Correspondencia Interna

Fuente: (Elaboración Propia)

3.4 PRUEBAS DE CALIDAD

Se hará la medición de calidad de software mediante la métrica ISO-9126, que establece que cualquier componente de calidad de software puede ser descrito en términos de uno o más de seis características básicas las cuales son: funcionalidad, confiabilidad, usabilidad, eficiencia, facilidad de mantenimiento y portabilidad.

3.4.1 Factores de calidad ISO 9126

El estándar ISO 9126 ha sido desarrollado en un intento de identificar atributos clave de calidad para el software. El estándar identifica seis atributos clave de calidad.

3.4.1.1 Funcionalidad

La funcionalidad no se mide directamente, por lo tanto es necesario evaluar un conjunto de características y capacidades del sistema. Nuestro sistema debe ser

capaz de proveer las funciones que cumplan con las necesidades explícitas e implícitas, cuando es utilizado en las condiciones específicas por el cliente.

Para el cual se utilizara la métrica punto de fusión, para esto se debe determinar cinco características de dominio de información y se proporcionan las cuentas en la posición apropiada de la tabla. Los valores de los dominios de información se definen de la siguiente forma.

Tipo / Complejidad	Baja	Media	Alta
(EI) Entrada externa	3 PF	4 PF	6 PF
(EO) Salida externa	4 PF	5 PF	7 PF
(EQ) Consulta externa	3 PF	4 PF	6 PF
(ILF) Archivo lógico interno	7 PF	10 PF	15 PF
(EIF) Archivo de interfaz externo	5 PF	7 PF	10 PF

Tabla 3.27 Valores estándar (IFPUG) International Function Point Users Group

Fuente: Elaboración en base a criterios (Pressman 2007).

En la tabla 3.27 se muestra la tabla de Tipo de Complejidad donde muestran valores determinados por la (IFPUG), dados por Baja, Media y Alta.

Para la hallar el punto de fusión del nuestro sistema, se toma en cuenta de la tabla 3.27 el Tipo de Complejidad o Parámetro de Medición Media, como se observa en la siguiente tabla.

Parámetros de Medición	Factor de Ponderación
------------------------	-----------------------

	Cuenta	Media	TOTAL
(EI) Entrada externa	35	4 PF	140
(EO) Salida externa	22	5 PF	110
(EQ) Consulta externa	61	4 PF	244
(ILF) Archivo lógico interno	28	10 PF	280
(EIF) Archivo de interfaz externo	0	7 PF	0
		PFSA	774

Tabla 3.28 Calculo de Punto de Función No Ajustado

Fuente: (Elaboración Propia).

Para calcular el punto de función (PF) se utiliza la siguiente relación.

$$\text{Punto de Función} = \text{Cuenta Total} * (0,65 + 0.01 * \sum fi)$$

Donde:

Cuenta total = Es la suma de todos los parámetros de complejidad, N° de entradas, N° de salidas, N° de peticiones, N° de archivos y N° de interfaces externas.

0,01: Error de Confiabilidad del Sistemas

$\sum fi$: Son los valores de ajuste de complejidad según las respuestas a las preguntas destacadas en la siguiente tabla.

Factor de Ajuste:

- 0 Sin Influencia
- 1 Incidental

- 2 Moderado
- 3 Medio
- 4 Significativo
- 5 Esencial

Son los valores que se tomara en cuenta a las 14 preguntas que se refiere a la siguiente tabla.

N°	Factores de Complejidad	Valor
1	¿Requiere el sistema copias de seguridad y de recuperación fiables?	5
2	¿Requiere comunicación de datos?	3
3	¿Existen funciones de procesamiento distribuido?	5
4	¿Es crítico el rendimiento?	5
5	¿Sera ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?	5
6	¿Requiere el sistema entrada de datos interactiva?	5
7	¿Requiere la entrada de datos interactiva que las transiciones de entrada se lleven a cabo sobre múltiples pantallas u operaciones?	4
8	¿Se actualizan los archivos maestros de forma interactiva?	5
9	¿Son complejas las entradas, las salidas, archivos o las peticiones?	3
10	¿Es complejo el procesamiento interno?	3
11	¿Se ha diseñado el código para ser reutilizable?	4
12	¿Están incluidas en el diseño la conversión y la instalación?	2
13	¿Se ha diseñado el sistema para múltiples instalaciones en diferentes organizaciones?	5
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	4

Tabla 3.29 Calculo de Punto de Función Ajustado**Fuente:** (Elaboración Propia).

Una vez obtenido el valor de ajuste y remplazando a la ecuación se tiene:

$$\mathbf{Punto\ de\ Funci3n} = Cuenta\ Total * (0,65 + 0.01 * \sum fi)$$

$$\mathbf{Punto\ de\ Funci3n} = 774 * (0,65 + 0.01 * 58)$$

$$\mathbf{Punto\ de\ Funci3n} = 952.02$$

Ajustando a la curva Normal:

$$Funcionalidad = PF/PFMaxima$$

Donde:

$PF\ Maximo = Cuenta\ Total * (0,65 + 0.01 * \sum fi)$, se toma el valor máximo de $\sum fi$, $[i = 1\ a\ 14]$, $[0 \leq \sum fi \leq 70]$.

Remplazando se tiene:

$$\mathbf{PF\ Maxima} = Cuenta\ Total * (0,65 + 0.01 * \sum fi)$$

$$\mathbf{PF\ Maxima} = 774 * (0,65 + 0.01 * 70)$$

$$PF \text{ Maxima} = 1044.9$$

Por tanto:

$$Funcionalidad = (PF / PFMaximo) * 100\%$$

$$Funcionalidad = (952.02/1044.9) * 100\%$$

$$Funcionalidad = 0.91 * 100\%$$

$$Funcionalidad = 91\%$$

La funcionalidad del sistema es de 91%.

3.4.1.2 Confiabilidad

Para determinar la confiabilidad se toma en cuenta las fallas que se producen en el sistema en un tiempo determinado, también es el grado en el que el sistema responde bajo las condiciones definidas durante un intervalo de tiempo dado.

$$F(t) = f * e^{(-\frac{\lambda}{10} * t)}$$

Donde:

f : Es la funcionalidad del sistema ya calculada: 0,91

λ : Es la probabilidad de error que puede tener el sistema: 0,03 (3%)

t : Tiempo que dura una gestión en el sistema: 12 meses.

A continuación se realiza el cálculo de la probabilidad de que el sistema tenga fallas.

$$F(t) = 0.91 * e^{(-\frac{3}{10} * 12)}$$

$$F(t) = 0.02486458743 * 100\%$$

$$F(t) = 2.5\%$$

La probabilidad de que ocurran fallas en el sistema es de 2.5% entonces la probabilidad de que el sistema esté libre de fallos es:

$$F(T \geq t) = 1 - F(t)$$

$$F(T \geq t) = (1 - 0.025) * 100\%$$

$$F(T \geq t) = 98\%$$

Por lo tanto se puede decir que el sistema tiene un grado de confiabilidad del $F(12) = 98\%$, y seguirá funcionando en un año (12 meses).

3.4.1.3 Usabilidad

Para poder determinar la usabilidad del sistema se utiliza la siguiente ecuación.

$$FU = [(\sum Xi/n) * 100]/N$$

Donde:

xi: sumatoria de valores.

n: número de preguntas.

N: cantidad de personas.

Para calcular la usabilidad del sistema se realiza una encuesta a todos los usuarios de la siguiente manera:

Escala	Valor
Muy buena	5
Buena	4
Regular	3
Malo	2
Pésimo	1

Tabla 3.30 Escala de valores para la Usabilidad

Fuente: (Elaboración Propia).

N°	Preguntas	Valor
1	¿El sistema es fácil de utilizar?	12
2	¿Puedes para el programa y salir de el en cualquier momento?	12
3	¿El sistema facilita el trabajo que usted realiza?	12
4	¿Se ha satisfecho todos los requerimientos establecidos?	12
5	¿Cómo considera los formularios que elabora el sistema?	11
6	¿El sistema tiene la seguridad necesaria?	12
7	¿Cómo considera el ingreso de datos al sistema?	12
8	¿La generación de resultados de ayuda al progreso de toma de decisiones?	12
TOTAL		95

Tabla 3.31 Preguntas para hallar la Usabilidad

Fuente: (Elaboración Propia).

Calculando la facilidad de uso, tomando en cuenta el Nro. de preguntas son 8, y la encuesta se realizó a 12 personas.

$$FU = [(95/8) * 100]/12$$

$$FU = 98.9$$

Por lo tanto la facilidad de usabilidad es de 98.9%.

3.4.1.4 Facilidad de Mantenimiento

Conjunto de atributos relacionados con la facilidad de extender, modificar o corregir un sistema de software, en este sentido se usara el índice de madurez.

$$IMS = \frac{[M_T - (F_a + F_C + F_d)]}{M_T}$$

Donde:

Tipo	Descripción	Valor
M_T	Numero de módulos de la versión actual.	16
F_C	Numero de módulos de la versión actual que se ha cambiado.	0
F_a	Numero de módulos de la versión actual que se ha añadido.	1
F_d	Numero de módulos de la versión anterior que se han borrado en la versión actual.	0

Tabla 3.32 Valores de Facilidad de Mantenimiento IMS

Fuente: (Elaboración Propia).

Reemplazando los datos de la ecuación actual tenemos:

$$IMS = \frac{[16 - (1 + 0 + 0)]}{16}$$

$$IMS = 0.94 * 100\%$$

$$IMS = 94\%$$

Se concluye que el sistema tiene un índice de Madurez del 94% al momento de realizar el mantenimiento.

3.4.1.5 Portabilidad

De acuerdo a los factores de calidad “la probabilidad es el esfuerzo necesario para transferir una aplicación de un entorno sistema Hardware y/o Software a otro”. [Pressman, 2001]

El Sistema de Gestión de Correspondencia Vía Web caso: Área Ciencias Económicas, Financieras y Administrativas - Universidad Pública de El Alto, por estar diseñado en un entorno de acceso vía web, mide la portabilidad en: lado del servidor y lado del cliente.

La portabilidad del Software se enfoca en tres aspectos:

- Hardware del Servidor.
- Sistema operativo del Servidor.
- Software del Servidor.

Por lo mencionado anteriormente el Sistema de Gestión de Correspondencia Vía Web caso: Área Ciencias Económicas, Financieras y Administrativas - Universidad

Pública de El Alto, es portable en sus diferentes entornos tanto en hardware y software por lo que puede considerarse una portabilidad de un 100%.

3.5 ANÁLISIS DE COSTOS

Como se conoce es una de las tareas de mayor importancia en la planificación de proyectos de software es la estimación, la cual consiste en determinar con cierto grado de certeza, los recursos software y hardware, costo, tiempo de esfuerzo necesario para el desarrollo de los mismos.

3.5.1 COCOMO II

Para este punto se utiliza el modelo Post-Arquitectura, la cual se desarrolla a continuación.

a) Estimar el tamaño del software

$$PFA = PF * (0,65 + 0.01 * \sum fi)$$

Para obtener el Punto de Función Sin Ajustar y el Ajuste de Complejidad aemos referencia a la Tabla 3.28 y tabla 3.29 respectivamente, donde los valores son:

$$Cuenta Total = 774 \quad y \sum fi = 58.$$

Por lo tanto el Multiplicador será:

$$\mathbf{Multiplicador} = 0,65 + 0.01 * \sum fi$$

$$\mathbf{Multiplicador} = 0,65 + 0.01 * 58$$

$$\mathbf{Multiplicador} = 1.23$$

Reemplazando en la ecuación:

$$PFA = Cuenta\ total * Multiplicador$$

$$PFA = 774 * 1.23$$

$$PFA = 952.02$$

b) Conversión de los puntos de fusión a KLDC

Ahora veremos los PFA a miles de líneas de código. Para ello veremos la tabla xx.

Lenguaje	Nivel	Factor LCD / PF
C	2.5	128
Anci Basic	5	64
Java	6	53
Anci Cobol	3	107
Visual Basic	7	46
ASP	9	36
PHP	11	29
Visual C++	9.5	34

Tabla 3.33 Conversión de Puntos de Función a KLCD

Fuente: (Elaboración Propia).

$$KLCD = \frac{Factor\ LCD/PF * PFA}{1000}$$

$$KLCD = (29 * 952.02)/1000$$

$$KLCD = 27.60$$

Donde:

KLDC = Número estimado de líneas de códigos distribuidas (en miles).

c) Valor del factor exponencial de escala B

$$B = 0.91 + .01 * \sum Sfi$$

Factores de Escala	Abreviatura	Valor	Significado
Precedentes	PREC	1.24	Muy parecido
Flexibilidad de Desarrollo	FLEX	5.07	Riguroso
Resolución de Arquitectura/Riesgo	RESL	7.07	Poco 20%
Cohesión del equipo de trabajo	TEAM	0	Alto nivel de integración
Madurez del Proceso	PMAT	4.68	Nivel 2
Sumatoria <i>Sfi</i>		18.06	

Tabla 3.34 Factores de Escala Cinco

Fuente: (Elaboración Propia).

Una vez obtenido la sumatoria reemplazamos en la ecuación para hallar el factor exponencial B.

$$B = 0.91 + .01 * 18.06$$

$$B = 1.0906$$

Para calcular los factores de escala se aplica la siguiente ecuación:

$$E = A * (\text{Tamaño})^B * \prod_{i=1}^{17} EMi$$

Donde:

E: Esfuerzo Nominal medida (hombres/mes).

A: Es una constante que captura los efectos lineales sobre el esfuerzo de acuerdo a la variación del tamaño, (A = 2.94).

Tamaño: Es el tamaño del software a desarrollar expresando en miles de línea de códigos de fuentes (KLCD = 27.60).

B: Es el factor exponencial de escala, ya calculada anteriormente (1.0906)

EMi: Factores de esfuerzo compuesto, corresponden a los factores de costo que tienen un efecto multiplicativo sobre el esfuerzo.

d) Determinar el Factor de Esfuerzo Compuesto (EMi)

Factores de Escala	Abreviatura	Valor	Significado
Seguridad Requerida	RELY	0.75	Efecto de falta sin ninguna consecuencia
Tamaño de Base de Datos	DATA	1.19	>=1000 Byts
Complejidad	CPLX	1	Nominal
Reutilización Requerida	RUSE	1	Atreves de programas

Documentación adaptada al ciclo de vida	DOCU	1.13	Muy alto
-----------------------------------------	------	------	----------

Tabla 3.35 Factores de Esfuerzo Compuesto (**Indicadores de Producto**)

Fuente: (Elaboración Propia).

Factores de Escala	Abreviatura	Valor	Significado
Restricción de tiempo de ejecución	TIME	1	Muy bajo
Restricción de almacenamiento principal	STOR	1.21	Muy alto 85%
Volatilidad de la Plataforma	PVOL	1	Nominal

Tabla 3.36 Factores de Esfuerzo Compuesto (**Indicadores de Plataforma**)

Fuente: (Elaboración Propia).

Factores de Escala	Abreviatura	Valor	Significado
Capacidad de Analista	ACAP	0.67	Muy Alto 90%
Capacidad del Programador	PCAP	0.74	Muy alto 90%
Continuidad del Personal	PCON	1	Ex Alto 100%
Experiencia en la aplicación	AEXP	0.92	36 meses
Experiencia en la plataforma	PEXP	1	12 meses
Experiencia en el lenguaje y las herramientas	LEXP	0.91	36 meses

Tabla 3.37 Factores de Esfuerzo Compuesto (**Indicadores de Personal**)

Fuente: (Elaboración Propia).

Factores de Escala	Abreviatura	Valor	Significado
Uso de Herramientas de Software	TOOL	1	Integración Moderada
Desarrollo Multisitio	SITE	0.78	Múltiples formas, Interactivo
Cronograma Requerido para Desarrollo	SCED	1.29	75% Nominal

Tabla 3.38 Factores de Esfuerzo Compuesto (**Indicadores del Proyecto**)

Fuente: (Elaboración Propia).

$$\prod_{i=1}^{17} EMi = RELY * DATA * CPLX * RUSE * DOCU * TIME * STOR * VPOL * ACAP * PCAP * PCON * AEXP * PEXP * LTEX * TOOL * SITE * SCED$$

$$\prod_{i=1}^{17} EMi = 0.75 * 1.19 * 1 * 1 * 1.13 * 1 * 1.21 * 1 * 0.67 * 0.74 * 1 * 0.92 * 1 * 0.91 * 1 * 0.78 * 1.29$$

$$\prod_{i=1}^{17} EMi = 0.5096$$

Remplazando en la ecuación:

$$E = A * (Tamaño)^B * \prod_{i=1}^{17} EMi$$

$$E = 2.94 * (27.60)^{1.0906} * 0.5096$$

$$E = 55.85$$

e) Determinar el tiempo de Desarrollo (TDES)

$$TDES = 3.67 * (E)^{(0.28+0.002*\sum Sfi)}$$

$$TDES = 3.67 * (55.85)^{(0.28+0.002*18.06)}$$

$$TDES = 13.08(meses)$$

f) Determinar la Cantidad de Personas (CH)

$$CH = E/TDES$$

$$CH = 55.85/13.08$$

$$CH = 4.26 (personas)$$

g) Costo del Software

El salario en un programador es de y su equivalente en dólares es

$$Costo Estimado = Salario Minimo * CH$$

$$Costo Estimado = 500 * 4 = 2000$$

$$Costo Estimado = 2000 * 13$$

$$Costo Estimado = 26000 \$us.$$

De acuerdo al análisis de costos de COCOMO II, el software tiene un costo total de 26000 \$us.

3.6 SISTEMA DE GESTION DE SEGURIDAD DE LA INFORMACION ISO-27002

La ISO 27002 sirve como un punto de información de la serie de normas 27000.

Evalúa y rectifica su implementación mediante la aplicación de objetivos de control. Dicho objetivos han de ser cumplidos para garantizar la correcta implantación de las normas, así como el funcionamiento de la empresa en cuanto la seguridad de la información, para este aspecto se ha tomado los siguientes tipos de seguridad.

3.6.1 Seguridad lógica

Los respaldos (back-up) de la base de datos del sistema se deberán realizar de acuerdo al movimiento del Área Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto, en este caso se recomienda una vez por semana.

El personal interviniente en el proceso de administración del Sistema del Área Ciencias Económicas, Financieras y Administrativas de la Universidad Pública de El Alto deberá cambiar el password del sistema periódicamente una vez al mes.

3.6.2 Seguridad física

Se recomienda guardar los back-up 3 o más copias que sean almacenadas en distintos lugares.

Los back-up de la base de datos deberán ser protegidas en áreas seguras, que solo se permita acceso al personal autorizado.

3.6.2 Seguridad Organizativa

La información con respecto al sistema debe recibir un nivel de protección apropiada como ser:

- Gestión de activos

- Etiquetar y manejar los Back-up de acuerdo con la fecha que se realizó el mismo.

➤ Recursos humanos

- Una vez que el personal administrativo interviniente del proceso de administración, haya concluido el contrato o haya sido revelado de su cargo deberá ser bloqueado o eliminado del sistema.

3.6.3 Buenas Prácticas para la Seguridad

1. En la institución se recomienda comprar licencia del software tanto como office y antivirus.
2. Se recomienda que los programas que están utilizando en el ordenador estén actualizados para el buen funcionamiento del mismo.
3. Utilizar contraseñas fuertes y cambiarlas con frecuencia.
4. Realizar copias de resguardo.
5. Educar a los usuarios en cuanto a los riesgos de seguridad que se debe tener al momento de utilizar el sistema.
6. Evitar el ingreso al sistema en cafés internet, ya que estos pueden tener software instalados para obtener el nombre de usuario y contraseña.

Se debe tomar en cuenta que el cumplimiento de esas recomendaciones descritas dará un buen funcionamiento del sistema.

i. FASE DE TRANSICIÓN

En esta fase todo el trabajo ha sido desarrollado y completado de las fases anteriores y se tiene un sistema estable, integrado y probado. La fase de transición es cuando el producto de software ha sido completado y se encuentra listo para ser entregado e implementado para los usuarios.

El sistema es suficientemente maduro para ser utilizado en el ambiente de aplicación. Se han realizado las pruebas y las correcciones necesarias con datos e información real de la empresa y así lograr el correcto funcionamiento y cubrir las expectativas y los requisitos establecidos.

Cabe recalcar que el sistema no necesita ser instalado, se ingresa mediante internet usando un navegador ya sea (Internet Explore, Mozilla Firefox, Opera y Crome) teniendo previamente la dirección de la página web donde está alojada.

4 CAPITULO IV CONCLUSIONES Y RECOMENDACIONES

En este capítulo se dará las conclusiones y recomendaciones pertinentes del Proyecto de Grado.

4.1 CONCLUSIONES

Se concluyen los objetivos planteados en el presente proyecto, desarrollando un Sistema de Gestión de Correspondencia - Vía Web Caso: Área Ciencias Económicas, Financieras y Administrativas en la Universidad Pública de El Alto.

Por lo que se concluyó los siguientes módulos con éxito.

- Módulo de Administración de Usuarios.
- Módulo de Información Rápida.
- Módulo de Organigrama.
- Módulo del Personal.
- Módulo de Correspondencia.
- Módulo de Gestión.
- Módulo de Derivación.
- Módulo de Correspondencia Interna.
- Módulo de Correspondencia Externa.
- Módulo de reportes.

Aplicando con éxitos las normas de calidad y las herramientas de programación para que tengan alta usabilidad y funcionalidad.

Una vez concluido el presente proyecto, se llegaron a las siguientes conclusiones:

- El manejo de información de las correspondencias se realiza de forma segura, rápida, precisa y confiable.

- El registro del trámite ya es una tarea automática previa selección de correspondencia interna o externa, lo cual reduce el tiempo de espera para el remitente.
- La correspondencia tiene una línea de tiempo donde indica la fecha de registro, personal designado, la unidad donde se encuentra y todos los por menores y detalles de la correspondencia interna o externa.
- Para la Correspondencia interna o externa también cuenta con una generación de reportes, donde se descargara en documento pdf, así también ver el seguimiento en un documento digital, lista para su impresión.

Finalmente en forma general se concluye que todos los objetivos fueron alcanzados, desarrollando un Sistema de Gestión de Correspondencia - Vía Web Caso: Área Ciencias Económicas, Financieras y Administrativas en la Universidad Pública de El Alto. Lo cual permite disminuir los problemas frecuentes que se presentan en el registro de Correspondencias, información descentralizada, la búsqueda de información se realizaba de forma manual, no existía ningún tipo de reporte del seguimiento del trámite, por lo mismo el sistema desarrollado resolvió estos problemas mencionados.

4.2 RECOMENDACIONES

En base a las observaciones realizadas en la implementación durante la prueba se elaboraron las siguientes recomendaciones.

- ✓ Se recomienda cambiar al usuario cambiar continuamente su contraseña o clave para la seguridad del sistema y proteger el acceso a personas ajenas y malintencionadas al sistema.
- ✓ Mantener un control acerca del equipo que hace de servidor físico en el ambiente.

- ✓ Para resguardar la información, el administrador del sistema debe realizar copias de seguridad de la base de datos.
- ✓ La seguridad lógica debe estar a cargo del administrador del servidor web debe incluir el uso de contraseña, la admiración de puertos y servicios.

5 BIBLIOGRAFÍA

- ALÒS-MONER, A. D. (2006). *“LA GESTIÓN DOCUMENTAL: ASPECTOS PREVIOS A SU IMPLEMENTACIÓN”*. EN: *EL PROFESIONAL DE LA INFORMACIÓN*.
- ANDINO, M. E. (2001). *CORRESPONDENCIA Y DOCUMENTACION COMERCIAL*. GUAYMURAS.
- BEAZA, R. (2003). *EVOLUCIÓN DE LA WEB CHILENA* . CHILE.
- BOCCALETTI. (2006). *COMPLEX NETWORKS: STRUCTURE & DYNAMICS*. ELSEVIER.
- BUSTELO RUESTA, C. (2007). *"GESTIÓN DE DOCUMENTOS: ENFOQUE EN LAS ORGANIZACIONES."* EN: *ANUARIO THINKERI 1*, 141-145.
- CONTRERAS. (2005). *DISEÑO DE UN SISTEMA DOCUMENTAL PARA EL PROGRAMA DE INGENIERÍA DE SISTEMAS DE LA UNIVERSIDAD SAN BUENAVENTURA – SEDE BOGOTA*.
- GUINCHAT, C., & MENOUE, M. (1983). *INTRODUCCIÓN GENERAL A LAS CIENCIAS Y TÉCNICAS DE LA INFORMACIÓN Y DE LA DOCUMENTACIÓN* .
- J., F. S. (S.F.). *INTRODUCCIÓN A LA INVESTIGACIÓN DE OPERACIONES VOLUMEN I TERCERA EDICIÓN MCGRAW, MÉXICO*.
- JAMES, A. (1994). *ANÁLISIS Y DISEÑO DE SISTEMAS DE INFORMACIÓN, VOLUMEN I SEGUNDA EDICIÓN MCGRAW, MÉXICO*.

- LAMAS, E. (1998). *GESTIÓN DE LA RADIO COMUNITARIA Y CUIDADANA*. QUITO.
- MOYA. (2014). *SISTEMA DE ADMINISTRACIÓN DE ARCHIVOS ESTUDIANTILES KARDEX-INFORMÁTICA-UMSA*.
- PACO. (2015). *SISTEMAS DE INFORMACIÓN ACADÉMICA (SIA)-UPEA*.
- PAULO, F. (1973). MEXICO.
- PRESSMAN, R. S. (2009). *INGENIERÍA DE SOFTWARE, EDICIÓN MCGRAW*.
- SAMPIERI, H. (S.F.). *“METODOLOGÍA DE LA INVESTIGACIÓN” INTERAMERICANA DE MÉXICO*.
- URANGA. (2001). *GESTIONAR LA COMUNICACIÓN EN LAS PRACTICAS SOCIALES, ORGANIZACIONES Y COMUNIDADES*. BUENOS AIRES.
- WEITZENFELD, A. (2005). *INGENIERIA DE SOFTWARE ORIENTADO A OBJETOS CON UML, JAVA INTERNET*. MEXICO: THOMSON.

ANEXOS

Anexo A: Árbol de problemas

Anexo B: árbol de objetivos

Anexo C manual de usuarios

Anexo D: documentos de aval