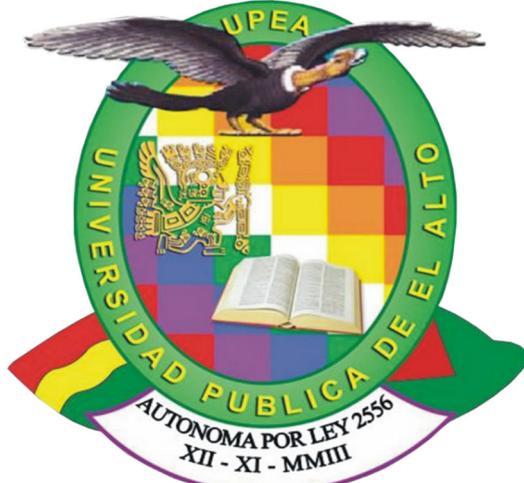


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

**SISTEMA DE GESTIÓN PARA LA GACETA UNIVERSITARIA
ON-LINE UTILIZANDO HERRAMIENTAS OPEN SOURCE
CASO: SECRETARIA GENERAL DE LA UNIVERSIDAD PÚBLICA DE EL ALTO**

Para optar al Título de Licenciatura en Ingeniería de Sistemas

Mención: GESTIÓN Y PRODUCCIÓN

Postulante: Hernán Catari Yujra

Tutor Metodológico: Ing. Enrique Flores Baltazar

Tutor Especialista: Ing. Isaac Magno Quispe Mamani

Tutor Revisor: Lic. Freddy Alanoca Coarite

EL ALTO – BOLIVIA

2020

DEDICATORIA

A DIOS por, sobre todas las cosas, porque sin su amor y misericordia, por estar siempre a mi lado brindarme salud, alegrías, tristezas que me tocó vivir y mucha fortaleza para seguir adelante con mis metas.

A mi Mamita Antonia Yujra Parapo que ya está en la presencia de DIOS y a mi Papá Juan Francisco Catarí Mamani, gracias por sus enseñanzas, cariño y comprensión que fueron algunas de las razones para poder concluir el presente proyecto.

A mi esposa Melvy Isabel Carvajal Cumara y mis hermanos y hermanas, que con su ayuda constante y palabras de apoyo me motivan a seguir siempre hacia adelante.

AGRADECIMIENTO

Quiero expresar mis sinceros agradecimientos:

Agradecer a DIOS todopoderoso en el nombre de su hijo JESÚS, quien murió en la cruz por mi para limpiar mis pecados, gracias por darme un día más de vida y salud.

Es necesario agradecer a las personas que influyeron para que termine este trabajo, mis padres que fueron el mejor ejemplo de superación y muestra de que en esta vida todo se puede con esfuerzo y trabajo, a mi familia que es el apoyo diario y hermanos, hermanas y la familia Carvajal Cumara, todos ellos apoyaron a mi superación intelectual y creyeron en mí.

Un especial agradecimiento, a mi Tutor Especialista Ing. Isaac Magno Quispe Mamani, por todo su tiempo invertido en la revisión y por haber creído en mí y por haberme apoyado para culminar con éxito este presente trabajo.

Un especial agradecimiento, a mi tutor metodológico Ing. Enrique Flores Baltazar, por su orientación y paciencia quien acompaña para la elaboración del presente proyecto.

Un especial agradecimiento, a mi tutor revisor Lic. Freddy Alanoca Coarite, por el seguimiento de este trabajo, orientación profesional, correcciones y observaciones para culminar el presente proyecto.

Finalmente, un agradecimiento a la Universidad Pública de El Alto, por haberme dado los conocimientos necesarios para poder desempeñarme con éxito en el futuro de toda mi vida, no sólo profesional sino también personal.

¡A TODOS GRACIAS!

RESUMEN

El presente trabajo tiene la finalidad de la implementación de un sistema de gestión de la Gaceta Universitaria On-Line utilizando herramientas Open Source, el cual permita realizar el ingreso de la información correspondiente a los convenios, resoluciones y su debido control, para poder mantener los documentos e información siempre disponible de forma digitalizada y de esta manera agilizar los procesos respectivos en el área de archivos quien es la encargada de custodiar la información institucional, generada desde la creación de la Universidad, los cuales serán registrados en el servidor local de la institución de donde se podrá consultar.

El presente documento está compuesto de cinco capítulos, los mismos que contienen información de cómo se desarrolló el proyecto, se utilizó tecnologías actualizadas para asegurar la seguridad y el rendimiento de este, entre esas tecnologías esta Laravel, como gestor de base de datos se tiene MySQL ,el cual se basó en las fases propuestas por la Metodología de proceso de desarrollo Ágil XP (Programación Extrema) que consta de cuatro fases (Planificación, Diseño, Codificación y Pruebas) y se complementó la fase de diseño con la ayuda de UWE (UML-Web Engineering), un método de ingeniería basada en diagramas UML (Lenguaje de modelado unificado). La calidad del software fue medida bajo la metodología WebQEM (Quality Evaluation Method – Método de Evaluación de Calidad) este evalúa aspectos como la Usabilidad, Funcionalidad, Confiabilidad, Eficiencia y finalmente para la estimación del costo, esfuerzo y tiempo de desarrollo se usó COCOMO (Modelo Constructivo de Costos).

ÍNDICE GENERAL

CAPITULO I – MARCO PRELIMINAR	1
1.1 INTRODUCCIÓN	1
1.2 ANTECEDENTES	2
1.2.1 ANTECEDENTES DE TRABAJOS SIMILARES.....	2
1.3 PLANTEAMIENTO DEL PROBLEMA	3
1.3.1 Problema Principal.....	4
1.3.2 Problemas específicos.....	4
1.4 OBJETIVOS	4
1.4.1 Objetivo General.....	4
1.4.2 Objetivos Específicos	4
1.5 JUSTIFICACIÓN	5
1.5.1 Justificación Técnica.....	5
1.5.2 Justificación Económica.....	6
1.5.3 Justificación Social	6
1.6 METODOLOGÍA	6
1.6.1 Metodología de Ingeniería	6
1.6.2 Métricas de calidad WEBQEM.....	7
1.6.3 Costo COCOMO.....	7
1.7 HERRAMIENTAS.....	8
1.8 LÍMITES Y ALCANCES	9
1.8.1 Límites	9
1.8.2 Alcances	9
1.9 APORTES	10
II – MARCO TEÓRICO	11
2.1 INTRODUCCIÓN	11
2.2 DESCRIPCIÓN DE LA UNIDAD DE SECRETARIA GENERAL	11
2.2.1Reseña Histórica.....	12
2.2.2Misión y Visión	14
2.2.3Organigrama	14
2.3 SISTEMA	15
2.4 GESTIÓN	15

2.5	GACETA	16
2.6	GACETA UNIVERSITARIA	16
2.6.1	Resolución del Honorable Consejo Universitario	17
2.6.2	Convenio	17
2.6.2.1	Convenios Interinstitucionales.....	17
2.6.2.2	Convenio Marco	18
2.6.2.3	Convenio Especifico.....	18
2.7	REPOSITORIO DIGITAL	18
2.7.1	Tipos de repositorios.....	19
2.8	INGENIERÍA DE SOFTWARE	20
2.8.1	Capas de ingeniería de software	21
2.9	METODOLOGÍAS DE DESARROLLO DE SOFTWARE.....	22
2.10	METODOLOGÍAS AGILES	23
2.11	PROGRAMACIÓN EXTREMA (XP).....	25
2.11.1	Valores XP	26
2.11.2	Roles XP	27
2.11.3	Modelo XP.....	27
2.11.4	Proceso XP	28
2.11.4.1	Planeación	29
2.11.4.2	Diseño.....	30
2.11.4.3	Codificación	31
2.11.4.4	Pruebas	32
2.11.5	Ciclo de vida XP	32
2.12	METODOLOGÍA UWE	33
2.12.1	Modelo de Caso de Uso.....	34
2.12.2	Modelo de Contenido	34
2.12.3	Modelo de Usuario	35
2.12.4	Modelo de Navegación.....	36
2.12.5	Modelo de Proceso	37
2.13	HERRAMIENTAS PARA EL DESARROLLO DEL SOFTWARE	39
2.13.1	Php (Hypertext Pre-Processor)	39
2.13.2	Mysql.....	40

2.13.3 Servidor Apache.....	41
2.13.4 Html.....	41
2.13.5 Javascript.....	41
2.13.7 Bootrap.....	42
2.13.8 CSS.....	42
2.13.9 Framework	42
2.13.10 Laravel	43
2.13.11 Visual Studio Code.....	45
2.13.12 MagicDraw	46
2.14 MÉTRICAS DE CALIDAD	46
2.14.1 METODOLOGÍA DE WEBQEM	46
2.14.1.1 Características de WebQem.....	47
2.15 MÉTODO DE ESTIMACIÓN DE COSTOS DE SOFTWARE	50
2.15.1 Costo y Beneficio	50
2.15.2 Modelo constructivo de Costo (COCOMO)	51
2.15.2.1 Modelo de COCOMO: Básico e intermedio.....	52
2.15.2.2 Modelo de COCOMO detallado	54
2.16 SEGURIDAD	54
CAPITULO III – MARCO APlicATIVO.....	56
3.1 INTRODUCCIÓN	56
3.2 ANÁLISIS DE REQUERIMIENTOS.....	56
3.2.1Situación actual.....	57
3.2.2Ingeniería de Requerimientos	57
3.2.2.1 Requerimientos Funcionales.....	58
3.2.2.2 Requerimientos No funcionales	59
3.3 APLICACIÓN DE LA METODOLOGÍA XP	59
3.3.1Fase I: Planificación	60
3.3.1.1 Historia de usuarios	60
A. Iteración 1 (Portal web).....	60
B. Iteración 2 (Gestión de Menús).....	61
C. Iteración 3 (Gestión de Usuarios)	62
D. Iteración 4 (Gestión Resoluciones).....	64

E. Iteración 5 (Gestión de Convenios)	66
3.3.1.2 Tareas de las historias de usuario	69
3.3.1.3 Planificación de Iteraciones.....	70
3.3.1.4 Plan de Entregas.....	71
3.3.2Fase II: Diseño.....	72
3.3.3Fase III: Codificación.....	81
3.3.4Fase IV: Pruebas	87
CAPÍTULO IV – CALIDAD, COSTO Y SEGURIDAD DEL SOFTWARE	92
4.1 METODOLOGÍA WEBQEM	92
4.1.1 Usabilidad	93
4.1.2 Funcionalidad.....	95
4.1.3 Confiabilidad	97
4.1.4 Eficiencia.....	97
4.2 ESTIMACIÓN DE COSTOS.....	99
4.2.1 Modelo de desarrollo intermedio.....	99
4.2.2 Estimación de costos	100
4.3 SEGURIDAD	103
4.3.1. Seguridad a nivel de Base de datos.....	104
4.3.2. Seguridad a nivel de aplicación.....	104
5.1 CONCLUSIONES	106
5.2 RECOMENDACIONES.....	107
BIBLIOGRAFÍA	108
ANEXOS	113

ÍNDICE DE FIGURAS

CAPÍTULO II

Figura 2.1: Organigrama de la Universidad Pública de El Alto.....	13
Figura 2.2: Organigrama Secretaría General – UPEA.	14
Figura 2.3: Capas de la Ingeniería de Software	21
Figura 2.4: Proceso de la Programación Extrema.....	28
Figura 2.5: Planificación en Programación Extrema.....	30
Figura 2.6: <i>Diseño en Programación Extrema</i>	30
Figura 2.7: Codificación en Programación Extrema	31
Figura 2.8: Pruebas en Programación Extrema	32
Figura 2.9: Modelo de casos de uso	34
Figura 2.10: Modelo de contenido	35
Figura 2.11: Nombre e iconos utilizados en el Modelo de navegación.....	36
Figura 2.12: Modelo de navegación	37
Figura 2.13: Modelo de estructura de procesos	37
Figura 2.14: Elementos del modelo de presentación	38
Figura 2.15: Modelo de presentación	38
Figura 2.16: Funcionamiento básico de Laravel.....	45

CAPÍTULO III

Figura 3.1: Fases de la programación extrema.....	56
Figura 3.2: Diagrama de contenido	73
Figura 3.3: Diagrama de navegación cliente.....	74
Figura 3.4: Diagrama de navegación Administrador Principal.....	75
Figura 3.5: Diagrama de navegación Encargado Archivo	75
Figura 3.6: Diagrama de presentación: interfaz web página de inicio	76
Figura 3.7: Diagrama de presentación: autenticación al sistema	76
Figura 3.8: Diagrama de presentación: página principal administrador	77
Figura 3.9: Diagrama de presentación: Agregar Usuario	78
Figura 3.10: Diagrama de presentación: página principal encargado archivo.....	79
Figura 3.11: Diagrama de presentación: Agregar Resolución.....	80
Figura 3.12: Página de inicio del sistema.....	81
Figura 3.13: Menú de inicio de sesión donde el usuario ingresa	81
Figura 3.14: Página principal del administrador del sistema	82
Figura 3.15: Pagina de registro de un usuario	82
Figura 3.16: Lista de usuarios del sistema.....	83
Figura 3.17: Modificar menú categoría.....	83
Figura 3.18: Modificar menú opción	84
Figura 3.19: Permisos Administrador principal.....	84
Figura 3.20: Registrar Resolución.....	85

Figura 3.21: Registrar Archivo PDF de la Resolución	85
Figura 3.22: Lista de Resoluciones registradas	86
Figura 3.23: Ver resolución PDF registrada	86

ÍNDICE DE TABLAS

CAPÍTULO II

Tabla 2.2: Valores constantes por modo de desarrollo	52
Tabla 2.3 Ecuaciones por tipo de modelo COCOMO: Básico e Intermedio	52
Tabla 2.4: Multiplicadores de esfuerzo (ME) del COCOMO Intermedio	53

CAPÍTULO III

Tabla 3.1: Categoría de Funciones.....	58
Tabla 3.2: Requerimientos Funcionales	58
Tabla 3.3: Requerimientos no funcionales.....	59
Tabla 3.4: Historia de Usuario, Gestión Portal Web	60
Tabla 3.5: Historia de Usuario, Gestión Categoría Menú	61
Tabla 3.6: Historia de Usuario, Gestión Menú	62
Tabla 3.7: Historia de Usuario, Gestión Tipo Usuario.....	63
Tabla 3.8: Historia de Usuario, Gestión Usuario.....	63
Tabla 3.9: Historia de Usuario, Gestión Tipo Resolución	64
Tabla 3.10: Historia de Usuario, Gestión Resolución	65
Tabla 3.11: Historia de Usuario, Gestión Representante.....	66
Tabla 3.12: Historia de Usuario, Gestión Institución	67
Tabla 3.13: Historia de Usuario, Gestión Tipo Convenio	68
Tabla 3.14: Historia de Usuario, Gestión Convenios	68
Tabla 3.15: Detalle de las tareas por iteración.....	69
Tabla 3.16: Planificación de Iteraciones	71
Tabla 3.17: Planificación de Iteraciones	71
Tabla 3.18: Planificación de Iteraciones	72
Tabla 3.19: Ejecución de las pruebas de aceptación.....	87

CAPÍTULO IV

Tabla 4.1: Resultados de Preferencia elemental – Usabilidad	94
Tabla 4.2: Evaluación total de Usabilidad.....	95
Tabla 4.3: Resultados de Preferencia elemental – Funcionabilidad	95
Tabla 4.4: Evaluación total de funcionalidad	96
Tabla 4.5: Resultados de preferencia elemental – Confiabilidad	97
Tabla 4.6: Resultados de preferencia elemental – Eficiencia	97
Tabla 4.7: Evaluación total de Eficiencia	98
Tabla 4.8: Resultados de la calidad total	98
Tabla 4.9: Escala de medición de aceptabilidad.....	99
Tabla 4.10: Modo de desarrollo, COCOMO Intermedio.....	101
Tabla 4.11: Cálculo de los atributos Multiplicador de Esfuerzo.....	101

CAPÍTULO I

MARCO PRELIMINAR

1.1 INTRODUCCIÓN

En la actualidad existen diversas herramientas tecnológicas que el hombre ha creado para la solución de variadas necesidades para mejorar su calidad de vida, actividades, procedimientos o procesos del que hacer vivir y/o cotidiano.

La automatización de procesos ha aparecido con la finalidad de reducir costos y tiempos de servicio. Como resultado de la reducción de los tiempos de servicio, se ha aumentado la satisfacción del cliente y con ello los beneficios de la institución.

La principal ventaja que tienen los sistemas web es su independencia de ser instaladas en una computadora, ya que solo se necesita tener acceso a Internet, un buen navegador web y un dispositivo electrónico (teléfono, pc) para poder acceder al mismo. Esta es la ventaja más significativa en cuanto a los sistemas web se refiere.

La Unidad de Secretaría General, dependiente de la Universidad Pública de El Alto (UPEA) se encuentra ubicada en la Ciudad de El Alto - La Paz, en la actualidad la Unidad de Secretaría General cuenta con una dependencia de Archivo y Resguardo de Documentación que alberga una gran cantidad de documentación generada desde hace más de 15 años.

Para el desarrollo del proyecto es conveniente el análisis de la gran cantidad de información que llega a la Unidad de Secretaría General, como ser el tipo de documentación que alberga. La implementación del Sistema de Registro y Administración Documental, busca obtener información clara, rápida y sólida que permita a los usuarios tener acceso a los documentos de forma rápida y segura, para así mejorar la atención de los funcionarios de la unidad y que reflejen el fenómeno de la difusión y socialización de la información generada de manera oportuna cuando lo soliciten.

Por tanto, el presente proyecto tiene como objetivo presentar un sistema de Gestión para la Gaceta Universitaria on-line, para automatizar los procesos de la Unidad de Secretaría General, encargada de difundir los programas (resoluciones, convenios) y resguardar la documentación generada y aprobada en asambleas de co-gobierno, congresos, Honorable Consejo Universitario y otros de índole institucional propias de la Universidad.

En ese sentido, el presente Proyecto de Grado está estructurado en cinco capítulos, los cuales se desarrollan del siguiente modo. Así, en el primer capítulo, se describe y fundamenta todos los elementos concernientes al marco preliminar. En el segundo se analizan las principales variables en torno a la problemática planteada. En el tercero, se presenta, describe y analiza los componentes del marco aplicativo. Mientras que en el cuarto capítulo se describe sistemáticamente la calidad, los costos y seguridad del Software. A su vez, en el quinto capítulo se presentan las conclusiones y recomendaciones conforme a los objetivos predeterminados.

Y finalmente, con el propósito de sustentar mucho más el presente trabajo académico, se muestran las fuentes de información empleadas, y en la parte de anexos se grafican el árbol de problemas, árbol de objetivos, requerimiento de hardware y software, manual de usuario, entre otros aspectos que complementan el trabajo.

1.2 ANTECEDENTES

1.2.1 ANTECEDENTES DE TRABAJOS SIMILARES

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

- **Sistema de Información Académica Carrera de Lingüística e Idiomas Universidad Pública de El Alto**, Para el desarrollo del sistema, se ha utilizado la metodología para aplicación interactiva, PhpExcel y Zpdf para la

emisión de reportes. Procesos Unificado Ágil (AUP), y sus respectivas herramientas de Desarrollo que propone. En la implementación, se utilizó la plataforma Windows, Php para la programación, Apache como servidor, MySql para la base de datos y Jquery para aplicación interactiva, PhpExcel y Zpdf para la emisión de reportes (ANGEL HERNÁN TICONA VILLA-UMSA).

- **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, Walter, Tito, Jesus y Hernan – UPEA).
- **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 Loza Moya – UMSA).

1.3 PLANTEAMIENTO DEL PROBLEMA

Debido que a la unidad de Secretaría General, genera y llega documentación, como ser: Antecedentes Anti-Autonomistas, Resoluciones (Rectorales, Administrativas, Honorable Consejo Universitario, Congreso, Académicas de Títulos Profesionales, Diplomas Académicos, Bachiller), Convenios y otros, cada uno con diferentes características, la cual no se encuentra actualizada ni centralizada en una base de datos, y al mismo tiempo realizan un proceso manual, para realizar el préstamo de la documentación. En tal sentido esta forma de registro genera demoras, a consecuencia de lo ya mencionado anteriormente existen retrasos al momento de buscar algún documento y tener conocimiento si este ha llegado o existe en la unidad de Secretaría General.

Secretaría General también elabora publicaciones de gacetas universitarias,

impresas que tienen un costo y existe demora en su publicación.

1.3.1 Problema Principal

El problema entonces radica en la falta de un sistema informático y procedimientos adecuados que coadyuven en el proceso de registro, préstamo y búsqueda de documentos relacionados a resoluciones, convenios y otros, por el cual se produce demora al momento de requerir algún documento, por la falta de conocimiento de la ubicación del documento.

1.3.2 Problemas específicos

- Falta un sistema informático de búsqueda de información documentaria.
- Falta Información oportuna y ágil con respecto a documentación generada.
- Existe una enorme cantidad de documentación que no se encuentra registrada en una base de datos.
- Falta la socialización de documentos de índole público como ser las normas y resoluciones aprobadas en co-gobierno y convenios a la comunidad universitaria en su debido tiempo.

1.4 OBJETIVOS

1.4.1 Objetivo General

Desarrollar un Sistema para la gestión de la Gaceta Universitaria On-Line utilizando herramientas Open Source, el cual permita realizar el ingreso de la información correspondiente a los convenios, resoluciones y su debido control, para poder mantener los documentos e información siempre disponible de forma digitalizada para la Unidad de Secretaría General de la Universidad Pública de El Alto.

1.4.2 Objetivos Específicos

- Analizar, identificar y realizar un diagnóstico sobre los principales procesos de la Unidad de Secretaría General.

- Recopilar información básica relacionada con los convenios, resoluciones y entre otros para responder a los requerimientos del usuario.
- Desarrollar y diseñar un sistema en línea utilizando herramientas open source (o código abierto), tomando como base la información recopilada para demostrar su funcionamiento e integración que agilice el proceso de la gestión de la gaceta Universitaria.
- Desarrollar un sistema de gestión e información con interfaz atractivo y amigable con módulos de usuarios acuerdo a roles, privilegios y niveles de acceso al sistema.
- Realizar pruebas con el usuario y ajustes del sistema para lograr un correcto funcionamiento.

1.5 JUSTIFICACIÓN

El sistema a desarrollar permitirá al personal de Secretaría General llevar un registro y control de la documentación concerniente a resoluciones, convenios que elabora y llega a esta unidad, donde dicha información servirá para obtener un acceso rápido a la misma, y un mejor control de la documentación y así facilitar el uso de la documentación, a toda la comunidad universitaria que lo requieran y público en general.

1.5.1 Justificación Técnica

La información es un recurso valioso en toda institución, más aún si es manejada en grandes volúmenes de datos, en la medida, en que las mismas sean automatizadas implica ahorro en tiempo, reducción en material de escritorio, reduce el trabajo manual.

El rápido crecimiento de la tecnología permite proporcionar técnicas y herramientas que facilitan el desarrollo de software. Se utilizarán las herramientas, lenguajes de programación y base de datos más acordes que respondan a las necesidades del proyecto, para un manejo sencillo, rápido y eficaz de la información.

1.5.2 Justificación Económica

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.

También la UPEA cuenta con un data center que soportan la inserción o instalación de los sistemas informáticos, que estará supervisada por el personal de la Unidad de Sistema de Información y Estadística (SIE).

Al tratarse de un proyecto a desarrollarse para una institución Pública, las herramientas de desarrollo serán libres, según la ley N° 164, del 08 de agosto de 2011 (Ley General de Telecomunicaciones, Tecnologías de Información y Comunicación), del estado Plurinacional de Bolivia, promoviendo el software libre.

1.5.3 Justificación Social

El presente trabajo contribuye a mejorar la atención, comunicación y disposición de la información al personal de la Unidad de Secretaría General y a la comunidad universitaria, sobre todo a sedes académicas universitarias, agilizando el servicio para que este no realice esperas de respuesta de la posible existencia o no de los documentos que se requiera o deseé obtener de la unidad, por lo que se tendrá la información oportuna y segura de forma rápida y en tiempo real.

1.6 METODOLOGÍA

1.6.1 Metodología de Ingeniería

La ingeniería de Software es una disciplina de la ingeniería que comprende todos los aspectos de la producción de software desde las etapas iniciales de la especificación del sistema, hasta el mantenimiento de este después de que se utiliza.

En el desarrollo del sistema se basará en las fases propuestas por la metodología

de desarrollo ágil XP es un conjunto de técnicas que dan agilidad y flexibilidad en la gestión de proyectos. También es conocida como Programación Extrema (Extreme Programming) y se centra crear un producto según los requisitos exactos del cliente.

Para el diseño y modelado se hará el uso del método UWE (Ingeniería Web basada en UML), pertenece a los métodos de apoyo a la ingeniería de software y si bien está basado en el modelado UML, tiene como objetivo y ventaja la adaptación de este modelado a sistemas o aplicaciones web, además de utilizar múltiples herramientas o componentes que pertenecen a UML.

1.6.2 Métricas de calidad WEBQEM

Modelo WEBQEM (1998). Es el modelo de prueba de calidad que utilizan los sitios Web, el nombre de WEBQEM proviene de las siglas en inglés Web-site Quality Evaluation method). Este modelo está diseñado en seis fases de pruebas que incluyen algunas innovaciones como la necesidad de definiciones, evaluaciones en distintos grados y termina en una validación de métricas, que significa que las métricas también son sometidas a evaluación antes de que estas puedan evaluar el programa.

1.6.3 Costo COCOMO

Como se conoce, 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 de hardware y software, costo, tiempo y esfuerzo necesarios para el desarrollo de los mismos. Este trabajo describe un modelo de estimación, propuesto por Barry Boehm, llamado COCOMO II. Este modelo permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de costo y de escala. Los factores de costo describen aspectos relacionados con la naturaleza del producto, hardware utilizado, personal involucrado, y características propias del proyecto. El conjunto de factores de escala explica las economías y deseconomías de escala producidas a medida que un proyecto de software incrementa su tamaño. COCOMO II posee tres modelos

denominados Composición de Aplicación, Diseño Temprano y Post-Arquitectura. Cada uno de ellos orientados a sectores específicos del mercado de desarrollo de software y a las distintas etapas del desarrollo de software.

1.7 HERRAMIENTAS

En diseño de software el Frontend es la parte del software que interactúa con los usuarios y el Backend es la parte que procesa la entrada desde el frontend.

La separación del sistema en frontends y backends es un tipo de abstracción que ayuda a mantener las diferentes partes del sistema separadas.

- **El Frontend** se enfoca en el usuario, en todo con lo que podemos interactuar y lo que vemos mientras navegamos. Así como en una primera cita, nuestra web busca causar una buena impresión y agradar al usuario, para lo cual utiliza: **Html, Css, Javascript, Jquery, Bootstrap** y entre otros.

Buena experiencia de usuario, inmersión y usabilidad, son algunos de los objetivos que busca un buen frontend y hoy en día existen una gran variedad de frameworks, preprocesadores y librerías que nos ayudarán en esta tarea.

Para un frontend la creatividad es el recurso más valioso, ya que tendrá que tomar fuentes, colores, imágenes y todos los recursos de los cuales disponga para crear sitios agradables que se vean bien en todos los dispositivos y resoluciones.

- Por otro lado, tenemos el **Backend** enfocado en hacer que todo lo que está detrás de un sitio web funcione correctamente. Toma los datos, los procesa y los envía al usuario, además de encargarse de las consultas o peticiones a la Base de Datos, la conexión con el servidor, entre otras tareas que debe realizar en su día a día.

Para ser programador del lado Servidor, son numerosos los lenguajes y frameworks entre los que elegir, los más comunes son: **Php, Java, Ruby, Python, Asp.Net, Javascript, Laravel etc.**

Sin embargo, no es suficiente con dominar un lenguaje y un framework. Toda aplicación web debe almacenar datos de alguna manera. Por lo tanto, un desarrollador back-end también debe estar familiarizado con las bases de datos. Entre las más comunes se destacan: **Sql Server, Mysql, Oracle, Postgresql y Mongodb.**

- **MagicDraw UML:** es una herramienta CASE desarrollada por No Magic. La herramienta es compatible con el estándar UML 2.3, desarrollo de código para diversos lenguajes de programación (Java, C++ y C#, entre otros) así como para modelar datos. La herramienta cuenta con capacidad para trabajar en equipo y es compatible con varios entornos de desarrollo (IDEs).

1.8 LÍMITES Y ALCANCES

1.8.1 Límites

El proyecto se desenvolverá dentro de la Unidad de Secretaría General de la Universidad Pública de El Alto. El software no considera la catalogación y digitalización, tampoco el seguimiento administrativo ni financiero.

1.8.2 Alcances

Asimismo, el alcance del presente proyecto de grado será la implementación del sistema de registro y administración documental de la gaceta universitaria on-line en la Unidad de Secretaría General que se realizará a partir del análisis, diseño, codificación, pruebas para finalmente lograr la implementación del sistema propuesto. El sistema permitirá:

- Generación del prototipo inicial.
- Subida de software a Data Center, entrega de software y ajustes extras.

- Registro del personal administrativos de acuerdo a roles de usuario.
- Asignación de tareas de acuerdo a los roles de usuario.
- Registro y búsqueda de la documentación según el tipo de documentación
- Otros procesos de la Unidad de Secretaría General no contemplados.

1.9 APORTES

El Sistema será de mucha utilidad en la Unidad Secretaría General y Comunidad Universitaria de la Universidad Pública de El Alto, aportando con la información precisa, segura y rápida en la Gestión de la Gaceta Universitaria permitiendo generar y obtener toda las normas y publicaciones aprobadas en co-gobierno, aportando con una Repositorio de datos virtuales para mayor accesibilidad, logrando un prestigio a la Universidad.

Como Proyecto de Grado. Se empleará la metodología de desarrollo de Software XP, juntamente con el Modelado de ingeniería Web UWE. Se manejará con mucha cautela sobre la seguridad y calidad del software aplicando las normas correspondientes en cuanto al desarrollo.

CAPÍTULO II

MARCO TEÓRICO

2.1 INTRODUCCIÓN

En este capítulo se conocerá las definiciones y conceptos fundamentales de apoyo para una mejor comprensión de los procesos involucrados para el desarrollo del presente Proyecto de Grado.

2.2 DESCRIPCIÓN DE LA UNIDAD DE SECRETARIA GENERAL

La Unidad de Secretaría General - UPEA, es una unidad Administrativa dependiente de Rectorado y es responsable de transcribir, resguardar y administrar la documentación Institucional, generados en el Congreso Universitario, Asamblea General Docente Estudiantil, Honorable Consejo Universitario, Honorable Consejo de Área, Rectorado, Vicerrectorado y las Unidades Académicas, conforme al Estatuto Orgánico de la Universidad Pública de El Alto.

Es responsable de administrar y supervisar la Unidad de Títulos y Diplomas; y la Unidad del Archivo Central de manera transparente, eficiente y eficaz de la Universidad Pública de El Alto.

Objetivo Unidad de Secretaría General:

- Coordinar y apoyar en las actividades del Congreso, de la Asamblea General Docente Estudiantil, del Honorable Consejo Universitario, del Honorable Consejo de Área, del Rectorado, del Vicerrectorado y las Unidades Académicas.

Objetivos secundarios:

- Cumplir y hacer cumplir las disposiciones del Estatuto de la Universidad Boliviana, Estatuto Orgánico de la Universidad Pública de El Alto, las resoluciones de la Asamblea General Docente Estudiantil, del Honorable Consejo Universitario.
- Contribuir en la toma oportuna y eficiente de las decisiones del Cogobierno

Universitario.

- Elaborar las Resoluciones de HCU, Rectorales fidedignos y apegados a las normas universitarias.
- Proponer, elaborar, gestionar Convenios Interinstitucionales para beneficiar a la comunidad estudiantil.
- Publicar y Difundir la documentación para mejorar las oportunidades de la comunidad universitaria.
- Atención esmerada a la comunidad universitaria y la población en todos los servicios de Secretaría General.

Servicios que Brinda:

- Coordina y apoya en el trabajo del Congreso, AGDE y HCU.
- Colabora al Rector en los asuntos de la Universidad.
- Administra, resguarda y facilita cuando corresponda la información pertinente de las resoluciones del Congreso, Asamblea General Docente Estudiantil, Honorable Consejo Universitario.
- Gestiona, publica y difunde los convenios interinstitucionales.
- Otorga copias legalizadas de resoluciones cuando legalmente corresponda.
- Es responsable de los trámites de Diplomas, Títulos Universitarios.
- Es responsable de la administración del Archivo Central.

2.2.1 Reseña Histórica

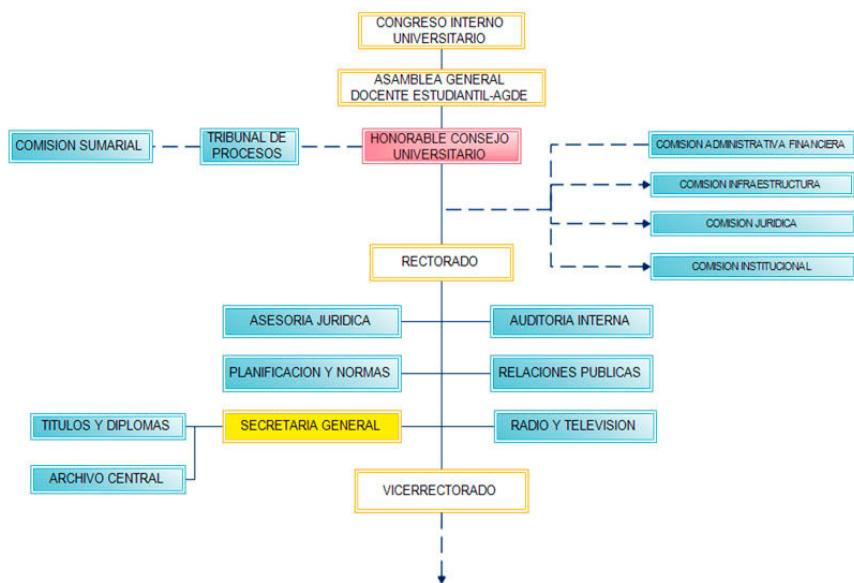
Un primero de mayo del año 2000 se toma los predios de Villa Esperanza del entonces Universidad Técnica Laboral de El Alto, dependiente de la Universidad Mayor de San Andrés, esto por un convenio firmado con organizaciones sociales de la Ciudad de El Alto y la UMSA en el año 1989. A pesar de existir una Universidad la población de la ciudad Alteña no estaba conforme con solo carreras técnicas, es entonces que las organizaciones sociales deciden movilizarse y tomar los predios de Villa Esperanza para crear una Universidad conforme a las necesidades de la

población Alteña, un cinco de septiembre del año 2000, se crea por ley 2115 sin autonomía la UPEA y 12 de noviembre del año 2003 se consigue la autonomía plena por ley 2556 , y por resolución Nº 2/2009, del XI Congreso Nacional de Universidades se acepta al seno del Sistema Universitario Boliviano, esto gracias a la lucha constante y merecida de la población de la ciudad de El Alto y universitarios de entonces por crear una Universidad. Hoy por hoy lo que conocemos como la Universidad Pública El Alto (UPEA).

La UPEA como su misión principal es formar profesionales integrales altamente calificados en todas las disciplinas del conocimiento científico-tecnológico, con conciencia crítica y reflexiva; capaz de crear, adaptar y transformar la realidad en que vive; desarrollar la investigación productiva para fomentar el desarrollo local, regional y nacional para que responda al encargo social y las necesidades de las nacionalidades de manera eficiente y oportuna hacia la trasformación revolucionaria de la sociedad. Fuente: Estatuto Orgánico UPEA.

En la figura 2.1 se muestra la estructura organizacional y organigrama de la Universidad Pública de El Alto.

Figura 2.1: Organigrama de la Universidad Pública de El Alto



Fuente: (Manual de Organización y Funciones – UPEA, 2019)

2.2.2 Misión y Visión

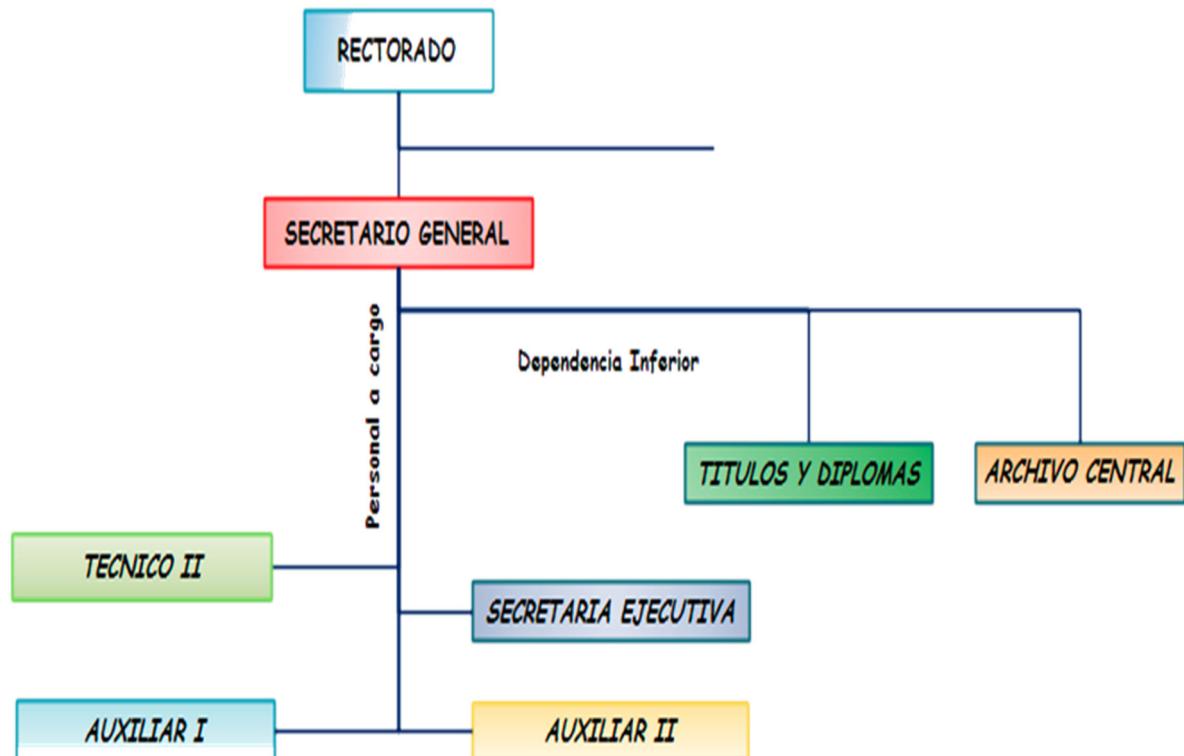
Misión: “Administra, Resguarda y facilita la documentación Institucional, Títulos y Diplomas para brindar información pertinente, para la toma de decisiones oportunas y eficientes”.

Visión: “Líder en el apoyo a las unidades tomadoras de decisión y difusión de la información Institucional de manera transparente y en tiempo real”.

2.2.3 Organigrama

En la figura 2.2 se muestra la estructura organizacional y organigrama de la Unidad de Secretaría General.

Figura 2.2: Organigrama Secretaria General – UPEA.



Fuente: (Manual de Organización y Funciones – UPEA, 2019)

2.3 SISTEMA

Para la página <http://concepto.de> el sistema se entiende como “un conjunto ordenado de componentes relacionados entre sí, ya se trate de elementos materiales o conceptuales, dotado de una estructura, una composición y un entorno particulares. Se trata de un término que aplica a diversas áreas del saber, como la física, la biología y la informática o computación”¹.

Mientras que para Real Academia Española el sistema se define como “un conjunto de reglas o principios sobre una materia racionalmente enlazados entre si” (Real Academia Española, 2001).

Por tanto, el sistema es un conjunto de partes o elementos organizados y relacionados que interactúan entre si para lograr un objetivo. Los sistemas reciben datos(entrada) y proveen información (salida).

2.4 GESTIÓN

La palabra gestión proviene del latín gestio, y hace la referencia a la administración de recursos, sea dentro de una institución estatal o privada, para alcanzar los objetivos propuestos por la misma. Para ello, uno o más individuos dirigen los proyectos laborales de otras personas para poder mejorar los resultados, que de otra manera no podrían ser obtenidos².

El término gestión es utilizado para referirse al conjunto de acciones, o diligencias que permiten la realización de cualquier actividad o deseo. Dicho de otra manera, una gestión se refiere a todos aquellos trámites que se realizan con la finalidad de resolver una situación o materializar un proyecto. En el entorno empresarial o comercial, la gestión es asociada con la administración de un negocio.

¹ Disponible en <https://concepto.de/sistema/#ixzz6Pwii2G93> _____ (consultado el 20-05-2020)

² Disponible en <https://concepto.de/gestion/#ixzz6Pwt3U1vJ> _____ (consultado el 20-05-2020)

2.5 GACETA

El término gaceta³ hace alusión a la publicación o divulgación que se difunde cada cierto tiempo con noticias sobre alguna materia o asunto. Es una palabra que llega del italiano “gazzetta”, referente a una moneda italiana, específicamente que circulaba en Venecia alrededor del siglo XVII, con cuya moneda se podía comprar un periódico para aquel entonces, y dicho periódico adquirió el sobrenombre de gazzetta gracias a esta. Dicha voz es el diminutivo de “gaza” originaria del indo-persa que significa “tesoro”, y que llegó al italiano por medio del latín. El diccionario de la real academia española describe el vocablo gaceta como aquella publicación periódica en la que se imparten noticias ya sean administrativas, literarias, comerciales o de otro tipo.

2.6 GACETA UNIVERSITARIA

Para Wikipedia la “Gaceta Universitaria fue un periódico estudiantil de la ciudad argentina de Córdoba fundado en 1918 como órgano oficial de la Federación Universitaria de Córdoba, en el momento que se iniciaba la rebelión estudiantil que dio origen al movimiento de la Reforma Universitaria de 1918. Su director fue Emilio Biagosch. Publicó veintidós números entre 1918 y 1919. En sus páginas se publicaron los primeros documentos del movimiento latinoamericano de la Reforma Universitaria, iniciado en Córdoba en 1918. En la edición del 21 de junio de 1918 publicó por primera vez el célebre Manifiesto liminar, documento máximo de la Reforma Universitaria”⁴.

La gaceta Universitaria para la Universidad Pública de El Alto es la publicación de normas aprobadas en co-gobierno como resoluciones del Honorable Consejo Universitario, congresos, resoluciones administrativas y entre ellos también se incluye los convenios firmados con instituciones para el conocimiento de la

³ Disponible en <https://conceptodefinicion.de/gaceta> _____ (consultado el 20-05-2020)

⁴ Disponible en https://es.wikipedia.org/wiki/La_Gaceta_Universitaria _____ (consultado el 20-05-2020)

comunidad universitaria y público en general.

2.6.1 Resolución del Honorable Consejo Universitario

El Honorable Consejo Universitario (HCU) es un órgano de gobierno paritario que tiene los siguientes objetivos:

- Analizar y definir la política interna y externa de la universidad, en concordancia con las resoluciones de la Asamblea General Docente Estudiantil (AGDE).
- Planificar y definir los aspectos institucionales, académicos, económicos y administrativos de la UPEA.

También las resoluciones aprobadas serán publicadas mensualmente en la Gaceta de la Universidad, para conocimiento de la Comunidad Universitaria y público en general (Estatuto Organico,2008).

2.6.2 Convenio

Un convenio es un acuerdo de voluntades entre dos o más personas o instituciones.

Los convenios son una de las herramientas principales para formalizar la vinculación entre la Universidad Pública de El Alto y las diferentes empresas u otras entidades que pretendan participar activamente en este proceso a través de la firma de convenios de colaboración, contribuyendo y creando espacios para la realización de programas y áreas de interés mutuo, a través de la Dirección de Vinculación con la Colectividad, se establecen, redactan y gestionan diferentes tipos de convenios.

2.6.2.1 Convenios Interinstitucionales

Los convenios interinstitucionales ayudan a fortalecer las relaciones entre las diferentes instituciones, para establecer mecanismos de cooperación y que estos sean utilizados por la comunidad universitaria, ya sea para realizar pasantías, prácticas pre-profesionales, intercambio, entre otros, esto permite a los estudiantes

aprender y poder formarse profesionalmente para el futuro.

Los convenios interinstitucionales pueden ser firmados con instituciones locales, nacionales e internacionales de acuerdo a la problemática económica, política, sociedad, cultural y educativa de la universidad y la sociedad.

2.6.2.2 Convenio Marco

Se define como un acuerdo de voluntades suscrito entre dos o más instituciones, en el que se establecen compromisos e intenciones para desarrollar en forma planificada actividades de cooperación mutua, se puede decir que son convenios marco, los que definen políticas de cooperación sin comprometer específicamente recursos económicos desembolsables de la universidad.

2.6.2.3 Convenio Específico

El convenio específico se suscribe para desarrollar programas o proyectos en los que las partes establecen compromisos precisos de cooperación; estos acuerdos generalmente se derivan de convenios marco previamente existentes, es decir que pueden llegar a ser el desarrollo o la operación de un convenio marco, sin perjuicio de que puedan celebrarse en ausencia de este, cuando haya la necesidad de desarrollar una actividad específica en un tiempo determinado.

2.7 REPOSITORIO DIGITAL

En la actualidad el uso de la tecnología ha logrado convertirse en tendencia para los seres humanos, esto se debe a que simplifica tareas y brinda la información de una forma más rápida y eficiente, en las empresas, que actualmente, aún manejan su información de forma manual, vienen presentando problemas cuando contienen excesiva documentación, para lo cual una alternativa es el desarrollo de repositorios digitales que simplifique la búsqueda y mantenga de mejor forma la documentación, un repositorio digital es un proveedor de datos disponible en medios electrónicos de

internet, y de forma genérica se puede definir como un lugar central donde se registran datos para su almacenamiento y conservación con propósitos diversos de seguridad o consulta posterior. Un repositorio digital, además de tener la misión de preservar la información para consulta posterior, tiene la capacidad de facilitar y potenciar la distribución de los objetos de aprendizaje u otros recursos (Ramírez, 2012). Por lo tanto, un repositorio digital en una institución agiliza los trámites y mantiene la conservación documental.

Se define también un repositorio como sistema informático donde distintas bases de datos archivos se encuentran para su distribución en internet. Es un proveedor de datos que integra un conjunto de servicios que permiten incorporar, reunir, preservar, consultar y dar soporte a la gestión y difusión de los recursos digitales creados por la propia universidad o los miembros de la comunidad, a través de una interfaz o portal web, mediante una adecuada clasificación de sus recursos a través de metadatos (Morera & Ramírez, 2004).

2.7.1 Tipos de repositorios

Repositorio de software: la variedad del servicio que ofertan depende del tipo de licencia usada:

- **Licencia privativa:** el administrador limita o restringe las propiedades del software. Ejemplo: Windows Update.
- **Licencia de uso libre:** ofrecen una plataforma de trabajo colaborativo y compartida de conocimiento libre sobre cualquier temática, sin ningún tipo de restricciones. Ejemplo: repositorios de software libre, paquetes para el sistema operativo GNU/Linux, desde plataformas como SourceForge o Forja de Guadalinex.

Repositorios institucionales: desarrollado por organismos políticos, sociales y educativos como universidades e institutos o asociaciones, para depositar, usar y preservar la producción científica y académica que generan en formato digital y

haciéndola accesible al público. De esta manera la institución ofrece un servicio acorde al movimiento de acceso abierto.

- **Repositorios temáticos:** creados por un grupo de investigadores, una institución, etc. que reúnen documentos relacionados con un área temática particular. La temática suele ser social, de educación ciudadana o académica.
- **Repositorios de datos:** repositorios que almacenan, conservan y comparten los datos de las investigaciones.

2.8 INGENIERÍA DE SOFTWARE

Haciendo una recopilación de todos los conceptos que se han dado sobre la Ingeniería de software, la podemos definir como la disciplina o área de la informática, que hace uso razonable de los principios de ingeniería con el objetivo de obtener soluciones informáticas económicamente factible y que se adapte a las necesidades de las empresas reales, tomando en cuenta los procesos de producción y mantenimiento de software que son desarrollados y modificados en el tiempo y con los costos estimados.

Esta ingeniería trata con áreas muy diversas de la informática y de las Ciencias de la Computación, tales como construcción de compiladores, Sistemas Operativos, o desarrollos Intranet/Internet, abordando todas las fases del ciclo de vida del desarrollo de cualquier tipo de Sistema de Información y aplicables a infinidad de áreas (negocios, investigación científica, medicina, producción, logística, banca, etc.).

Algunas definiciones, dadas a través del tiempo son:

- “Ingeniería de Software es el estudio de los principios y metodologías para el desarrollo y mantenimiento de sistemas de software” (Zelkovitz, 1978)
- “Ingeniería de software es la aplicación práctica del conocimiento científico al diseño y construcción de programas de computadora y a la documentación

asociada requerida para desarrollar, operar y mantenerlos. Se conoce también como Desarrollo de Software o Producción de Software" (Bohem, 1976).

- “Ingeniería de Software trata del establecimiento de los principios y métodos de la ingeniería a fin de obtener software de modo rentable, que sea fiable y trabaje en máquinas reales” (Bauer, 1972).
- “Es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento del software; es decir, la aplicación de la ingeniería al software” (IEEE, 1993).

En conclusión, podemos decir que los cuatro autores anteriores, de manera diferente describen en si el principal objetivo de la ingeniería de software, la cual es el establecimiento y puesta en práctica de los principios y metodologías que nos lleven a un desarrollo eficiente de software en todas las etapas desde sus inicios hasta su implementación y mantenimiento.

2.8.1 Capas de ingeniería de software

El enfoque de la ingeniería del software cuenta con un compromiso organizacional con la calidad porque es posible incorporar la ingeniería del software en una organización que no está centrada en conseguir calidad. La ingeniería de software es una tecnología estratificada, es decir que cualquier enfoque de la ingeniería debe estar sustentado en un compromiso con la calidad, como se puede representar a continuación.

Figura 2.3: Capas de la Ingeniería de Software



Fuente: (Pressman, 2008)

Dichas capas se describen como:

- **Enfoque de Calidad:** Es donde se realiza la gestión para obtener un software de calidad y hace revisiones para hacer mejoras al producto.
- **Proceso:** El cimiento de la ingeniería del software es la capa de proceso. El proceso de la ingeniería del software consiste en la cohesión que conserva las capas de tecnología unidas y que da pie a un incremento lógico y conveniente de la ingeniería del software. El proceso determina un entorno de trabajo para un cúmulo de Áreas clave de proceso (ACPs) que se deben constituir para la adjudicación efectiva de la tecnología de la ingeniería del software.
- **Métodos:** Los métodos de la ingeniería del software orientan “cómo” elaborar técnicamente el software. Los métodos comprenden una diversidad de tareas tales como el análisis de requisitos, diseño, elaboración de programas, verificación y mantenimiento. Los métodos de la ingeniería del software obedecen a un grupo de principios básicos que rigen cada área de la tecnología e incorpora funciones de modelado y otras técnicas descriptivas.
- **Herramientas:** Las herramientas de la Ingeniería del software aportan un punto de vista automático o semi-automático para el proceso y para los métodos. Un sistema de sustentación para el buen desempeño del software, se forma por la integración de herramientas para que la información elaborada por una herramienta la pueda utilizar otra, a esto es lo que llamamos ingeniería del software asistida por computadora (CASE).

Dado lo anterior, el objetivo de la ingeniería de software es lograr productos de software de calidad (tanto en su forma final como durante su elaboración), mediante un proceso apoyado por métodos y herramientas. (Pressman, 2008).

2.9 METODOLOGÍAS DE DESARROLLO DE SOFTWARE

Una parte importante de la ingeniería de software es el desarrollo de métodos y

modelos. En la actualidad ha habido muchos esfuerzos que se han encaminado al estudio de los métodos y técnicas para lograr una aplicación más eficiente de los métodos y lograr sistemas más eficientes y de mayor calidad con la documentación necesaria en perfecto orden y en el tiempo requerido.

Gacitúa, plantea que una metodología impone un proceso de forma disciplinada sobre el desarrollo de software con el objetivo de hacerlo más predecible y eficiente. Una metodología define una representación que permite facilitar la manipulación de modelos, y la comunicación e intercambio de información entre todas las partes involucradas en la construcción de un sistema (Gacitúa ,2003).

2.10 METODOLOGÍAS AGILES

Para el proyecto se usa una metodología ágil. Existen muchas formas de desarrollo tradicional pero no son efectivas en algunos proyectos, de esta manera, las metodologías agiles son las que pueden acoplarse a los requerimientos del cliente, esto también se debe a que los requisitos del cliente pueden ir cambiando constantemente y respecto al tiempo de desarrollo se pueden elaborar proyecto con una muy alta calidad. (Lelier & Penades, 2006).

Manifiesto Ágil

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:

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

Los métodos ágiles también enfatizan que el software funcional es la primera medida del progreso. Combinado con la preferencia por las comunicaciones cara a cara.

Los valores anteriores inspiran los doce principios del manifiesto. Son características que diferencian un proceso ágil de uno tradicional. Los dos primeros principios son generales y resumen gran parte del espíritu ágil. El resto tienen que ver con el proceso a seguir y con el equipo de desarrollo, en cuanto metas a seguir y organización del mismo. Los principios son:

- La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.
- Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.
- Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.
- Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- El software que funciona es la medida principal de progreso.
- Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante.
- La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
- La simplicidad es esencial.
- Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.

- En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento. (Lelier & Penades, 2006).

Aunque los creadores e impulsores de los métodos ágiles más populares han suscrito el manifiesto ágil y coinciden con los principios enunciados anteriormente, cada método tiene características propias y hace hincapié en algunos aspectos más específicos. Entre los métodos conocidos como ágiles se pueden mencionar:

- Programación Extrema (XP)
- Scrum
- Modelo de desarrollo de sistemas dinámicos (MDSD).
- Cristal
- Desarrollo impulsado por las características (DIC)
- Proceso Unificado Ágil (PUA)
- Desarrollo esbelto de software (DES)

En este trabajo no enfocaremos en las metodologías agiles y se profundizará el proceso XP, la cual pertenece a este tipo de metodología.

2.11 PROGRAMACIÓN EXTREMA (XP)

La Programación Extrema o Extreme Programming, es un enfoque de la ingeniería de software formulado por Kent Beck, se considera el más destacado de los procesos ágiles de desarrollo de software. Al igual que estos, la programación extrema se diferencia de los métodos tradicionales principalmente en que presenta más énfasis en la adaptabilidad que en la previsibilidad. (Bautista Q, 2012)

La programación extrema o XP es una metodología de desarrollo que se englobaría dentro de las denominadas metodologías Ágiles en la que se da máxima prioridad a la obtención de resultados y reduce la burocracia que utiliza las metodologías tradicionales.

¿Qué es XP?: XP es una metodología ágil que sigue reglas estrictas centradas en el cliente para lograr un producto de buena calidad en poco tiempo. Promueve el trabajo en equipo se preocupa por el buen clima laboral y el aprendizaje de los desarrolladores. Está orientada para aplicaciones que no necesitan un grupo grande donde la comunicación sea constante y fácil.

2.11.1 Valores XP

- **Comunicación:** debe ser fluida entre todos los participantes en el proyecto. El entorno tiene que favorecer la comunicación espontánea, ubicando a todos los miembros en un mismo lugar. La comunicación directa nos da mucho más valor que la escrita, podemos observar los gestos del cliente, o la expresión de cansancio de nuestro compañero.
- **Simplicidad:** cuanto más sencilla sea la solución, más fácilmente podremos adaptarla a los cambios. Las complejidades aumentan el coste del cambio y disminuyen la calidad del software. Sólo se utiliza lo que en ese momento nos da valor, y lo haremos de la forma más sencilla posible. Esto podría dar a pensar que va en contra de toda la filosofía de diseño y utilización de patrones. Nada más alejado de la realidad. En un proyecto XP, el uso de patrones nos va a ayudar a reducir el tiempo de implantación, pero lo que no vamos a hacer es dedicar tiempo a la implementación de patrones que no vayamos a utilizar en este proyecto, sólo haremos los que sean necesarios para éste, no utilizaremos tiempo del proyecto para beneficiar a otro proyecto futuro que quizás no llegue nunca. Por otro lado, nada nos impide desarrollar un proyecto que únicamente se dedique a desarrollar patrones que más tarde se utilicen en proyecto XP.
- **Realimentación:** el usuario debe utilizar y probar el software desarrollado desde la primera entrega, dándonos sus impresiones y sus necesidades no satisfechas, de manera que estas cosas encontradas vuelvan a formar parte de los requisitos del sistema y así poder atacarlas con tiempo.

- **Coraje:** con XP debemos tocar continuamente cosas que ya funcionan, para mejorarlas, optimizarlas o agregar funcionalidad. Es por eso que el coraje es parte del proyecto también, ya que el miedo a tocar o modificar cosas que ya funcionan perfectamente, a veces puede ser difícil.
- **Respeto:** Cada integrante del equipo es pieza integral en el proyecto, ninguno toma decisiones repentinamente así todos pueden trabajar como uno.

2.11.2 Roles XP

Las historias de usuario en toda metodología proporcionan información entendible para el cliente y el desarrollador, ya que especifica los requisitos del software, se realiza un documento rápido y preciso, con características de los requisitos funcionales o no funcionales que el sistema debe contener, son de carácter variable ya que se puede añadir, modificar o reducir, va en conjunto con los roles XP (Borja Lopez, 2013) que se asignan a los miembros del equipo, entre los que se encuentran; programador, cliente, probador, encargado de seguimiento, coach, consultor, gestor, cada uno con su función clave donde:

- **Programador:** Desarrolla el código del sistema.
- **Cliente:** Especifica requisitos del sistema
- **Tester:** Realiza pruebas regulares
- **Tracker:** Verifica los cumplimientos por tiempos y objetivos
- **Coach:** Controla la aplicación de las prácticas XP
- **Consultor:** Resuelve problemas externos relacionados con el proyecto
- **Gestor:** Intermediario entre clientes y programadores, actúa de coordinador

2.11.3 Modelo XP

La metodología XP define cuatro variables para cualquier proyecto de software: costo, tiempo, calidad y alcance.

Se especifica que, de estas, solo tres pueden ser fijadas por el cliente o jefes de

proyecto y el valor de la variable restante podrá ser definido por el equipo de desarrollo.

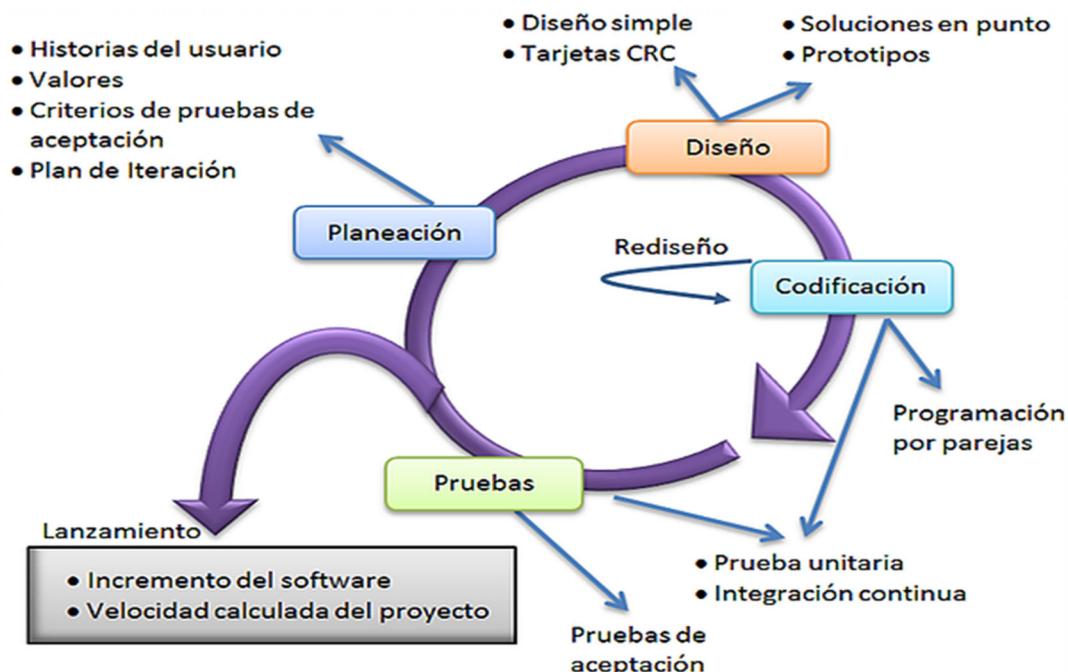
Este mecanismo indica que, por ejemplo, si el cliente establece el alcance y la calidad, y el jefe de proyecto el precio, el grupo de desarrollo tendrá libertad para determinar el tiempo que durará el proyecto.

Para esto se realiza iteraciones con un entregable al final de cada una, estas tienen también un análisis, diseño, desarrollo y pruebas, siempre utilizando las reglas y prácticas de XP.

2.11.4 Proceso XP

La forma en la que se establecen las reglas de desarrollo XP se dan en base a la programación orientada a objetos, en este tipo de desarrollo se establecen 4 actividades estructurales que son expuestas en el siguiente gráfico.

Figura 2.4: Proceso de la Programación Extrema



Fuente: (Pressman, 2010)

Un proyecto XP tiene éxito cuando el cliente conoce el valor de negocio basado en

la habilidad del equipo para medir la funcionalidad que puede otorgar a través del tiempo. El ciclo de desarrollo tiene estos pasos:

- El cliente define el valor de negocio a implementar.
- El programador estima el esfuerzo necesario para su implementación.
- El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
- El programador construye ese valor de negocio.
- Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden.

No se presiona al programador para realizar más trabajo del que se necesite, esto talvez arruinaría la calidad o podrían incumplirse los plazos.

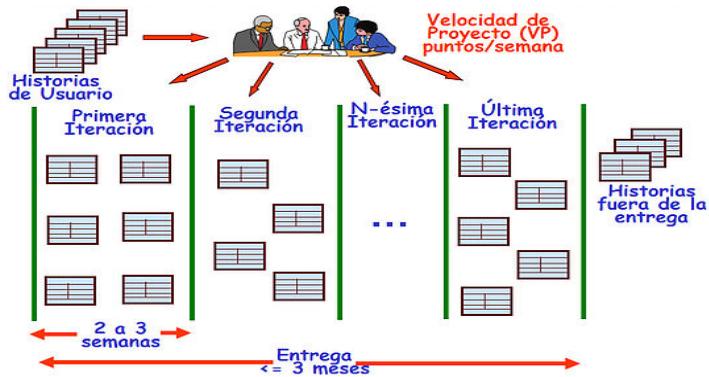
De esta manera el cliente tiene la obligación de manejar el plazo de entrega del producto así poder saber que el sistema tiene el mayor valor posible.

2.11.4.1 Planeación

La planificación se realiza por etapas, es decir, que es iterativa, sin embargo, no puede haber una planificación sin que antes haya habido comunicación con el cliente, en esta reunión con el cliente, el establece sus requerimientos lo cual hace que el equipo de software entienda cuáles son las características y funcionalidades que se necesitan.

Cuando el desarrollador entiende lo que escucha del cliente, este puede ser capaz de hacer los casos de uso, en los cuales el cliente puede asignar prioridades entre todas las funciones del software. (Tecno Tips. 2012).

Figura 2.5: Planificación en Programación Extrema



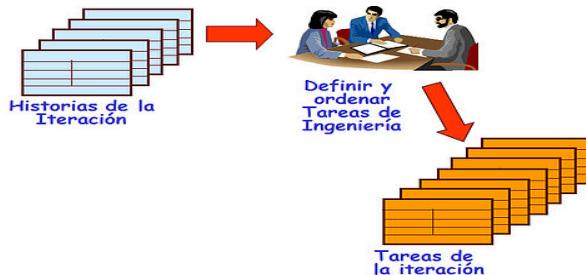
Fuente: (Programación Extrema⁵)

Velocidad en la Planeación: Se puede apreciar la velocidad de un software, contando los casos de uso que fueron realizados e implementados en la primera iteración o entrega, esta medición de la velocidad, ayuda a estimar fechas y programar actividades y compromiso con las funcionalidades del software.

2.11.4.2 Diseño

El diseño es la guía para la implementación del sistema, por lo tanto, debe ser claro, y para poder ser claro necesita de simplicidad, ya que no sólo será entendido por el programador, sino que también en muchas ocasiones, por el usuario. En el diseño se pueden también asignar las responsabilidades y módulos de cada persona en el equipo.

Figura 2.6: Diseño en Programación Extrema



Fuente: (Programación Extrema⁶)

⁵ Disponible en <https://jraquelm2.wixsite.com/ingenieriadessoftware> (consultado el 20-05-2020)

⁶ Disponible en <https://jraquelm2.wixsite.com/ingenieriadessoftware> (consultado el 20-05-2020)

Rediseño: Significa que un sistema puede cambiar, y si este lo hace, no debe afectar su comportamiento externo, es decir, cuando existe un problema en el que se quiere rediseñar o rehacer el software se hace sin que la estructura cambie para el usuario, es como si se hiciera una limpieza de código la cual es muy importante para su posterior documentación.

El diseño es una tarea que siempre se está realizando, antes y después de que el software es codificado y en su transcurso, y cuando el software ya está codificado en ocasiones se vuelve a diseñar una funcionalidad y es ahí donde entra el rediseño.

2.11.4.3 Codificación

Para poder empezar en la codificación, antes se debieron hacer pruebas unitarias de avances en diseño a los clientes, para así, poder establecer los requerimientos primordiales.

Uno de los mejores mecanismos para hacer que la codificación funcione de manera correcta es la unión de dos personas del equipo, es decir, la programación en parejas, cada una de estas personas, con características distintas y especializadas en distintas áreas puede encargarse de tareas distintas dentro de un mismo código.

Figura 2.7: Codificación en Programación Extrema



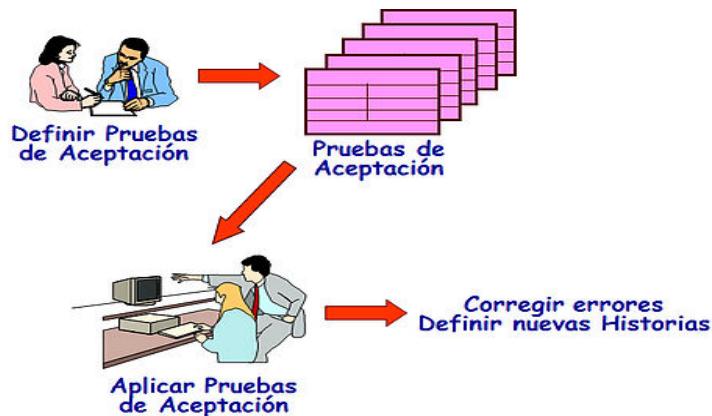
Fuente: (Programación Extrema⁷)

⁷ Disponible en <https://jraquelm2.wixsite.com/ingenieriadessoftware> (consultado el 20-05-2020)

2.11.4.4 Pruebas

Las pruebas unitarias son la medida de comprobación de la funcionalidad de cada uno de los módulos o componentes del sistema, estas pueden ser ejecutadas a diario y brindan una información del avance que tiene el proyecto.

Figura 2.8: Pruebas en Programación Extrema



Fuente: (Programación Extrema⁸)

Hacer pruebas de funcionamiento es primordial si se requiere un software sin errores, y realizar pruebas a diario y por separado ayuda a que lo que se entregue al cliente sea un producto final confiable.

Las pruebas no solamente las realiza el desarrollador con su equipo, sino que también interviene el cliente, él tiene la última palabra, y si tiene nuevos requerimientos puede darlos en el momento de las pruebas, a esto se le llama pruebas del cliente.

2.11.5 Ciclo de vida XP

El ciclo de vida de XP (Fig. 2.4) permite verificar la finalización del proyecto en los tiempos determinados, de acuerdo con las habilidades de los miembros del equipo

⁸ Disponible en <https://jraquelm2.wixsite.com/ingenieriadessoftware> (consultado el 20-05-2020)

desarrollador, sin la necesidad de presionar al equipo a realizar más trabajo del especificado. Según los autores (Canós, Letelier, & Penadés, 2016) el proceso XP consta de 4 fases principales:

- **Exploración:** En esta fase se plantean los requisitos en las historias de usuario, se analiza las herramientas tecnológicas a aplicarse y de ser el caso, se diseña inicialmente la arquitectura del software y un prototipo.
- **Planificación:** Se establece la prioridad sobre cada una de las historias de usuario, el esfuerzo a aplicar y tiempos de entrega en base a la velocidad de desarrollo del equipo.
- **Iteraciones:** Se escoge las historias de usuario a ser aplicadas en cada iteración, es decir se organizan subtareas en base a una historia de usuario, asignando responsables, todo en una planificación de entrega.
- **Producción:** Fase en la que se realizan pruebas y revisiones de rendimiento antes de ser ingresado en un servidor principal (el del cliente).

2.12 METODOLOGÍA 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.

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

- **Modelo de casos de Uso:** Modelo para capturar los requisitos del sistema.
- **Modelo de contenido:** Es un modelo conceptual para el desarrollo del contenido.
- **Modelo de Usuario:** Es un modelo de navegación, en el cual se incluyen

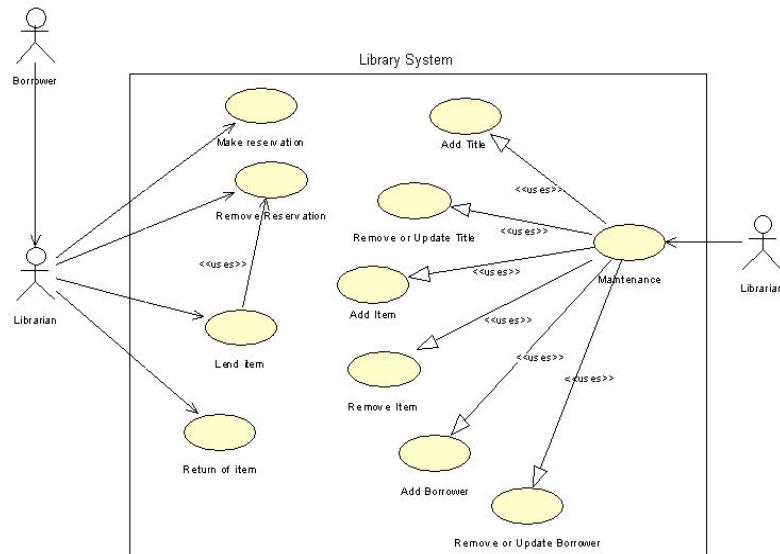
modelos estáticos y modelos dinámicos.

- **Modelo de estructura o presentación:** En el cual se encuentra la presentación del sistema y el modelo de flujo.
- **Modelo Abstracto:** Incluye el modelo de interfaz de usuario modelo de ciclo de vida del objeto.
- **Modelo de adaptación.**

2.12.1 Modelo de Caso de Uso

Plasma los requisitos funcionales de la aplicación Web mediante un modelo de casos de uso. Este modelo describe un trozo de comportamiento de la aplicación sin relevar su estructura interna, como se observa en la figura 2.8. Siguiendo el Proceso de Desarrollo de Software Unificado a la ingeniería Web (Galiano, 2012).

Figura 2.9: Modelo de casos de uso



Fuente: (Galiano, 2012)

2.12.2 Modelo de Contenido

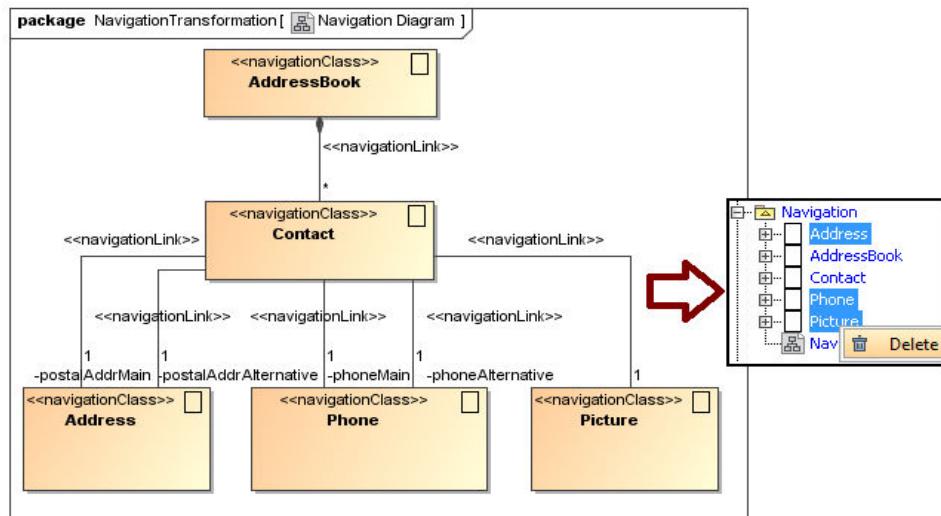
Define, mediante un diagrama de clases, los conceptos a detalle involucrados en la aplicación. El diseño conceptual se basa en el sistema en el análisis de

requerimientos del paso anterior. Esto incluye los objetos involucrados entre los usuarios y la aplicación (Galiano, 2012).

Este modelo especifica cómo se encuentra relacionado los contenidos del sistema, es decir define la estructura de los datos que se encuentran alojados en el sitio web.

A continuación, se muestra un ejemplo de este modelo contenido en la página web de UWE.

Figura 2.10: Modelo de contenido



Fuente: (Galiano, 2012)

En este ejemplo se puede ver representado que el contenido web está formado por una agenda básica de contactos, esta agenda representada por la clase AddressBook contiene un conjunto de uno o más contactos (clase Contact), cada uno de ellos tiene un nombre, un email, dirección y teléfono. De los cuales los dos primeros son de tipo String y los dos últimos son estructura de otros atributos, representadas por las clases Address y Phone, cada contacto puede tener una dirección y un teléfono principal y otros secundarios (Galiano, 2012).

2.12.3 Modelo de Usuario

Representa la navegación de los objetos dentro de la aplicación y un conjunto de

estructuras como son índices, menús y consultas.

El modelo usuario tiene dos objetivos diferenciados.

- Contiene las clases que define qué información debe ser almacenada en el contexto de una sesión. En este caso práctico una sesión está formada por el usuario actual y sus discos.
- Estas clases contenidas proveen de operaciones que puede ser usados en el proceso de negociación de procesos.

2.12.4 Modelo de Navegación

El modelo de navegación de una aplicación Web comprende la especificación de qué objetos pueden ser visitados mediante la navegación a través del sistema y las asociaciones entre ellos.

Mediante estos diagramas se representó el diseño y la estructura de las rutas de navegación al usuario para evitar la desorientación en el proceso de navegación. Este modelo se destaca en el marco de UWE como el más importante, pues con él se pueden representar elementos estáticos, a la vez que se pueden incorporar lineamientos semánticos de referencia para las funcionalidades dinámicas de una aplicación Web.

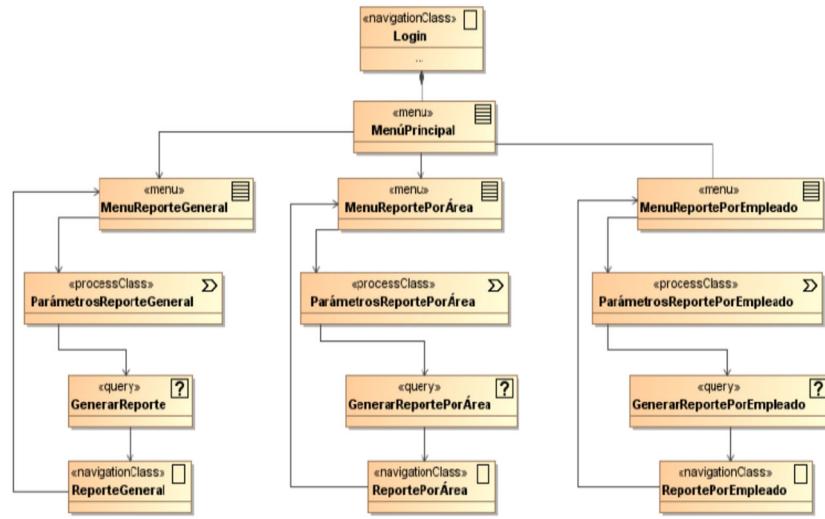
Figura 2.11: Nombre e iconos utilizados en el Modelo de navegación

clase de navegación	menú
índice	pregunta
visita guiada	clase de proceso
nodo externo	

Fuente: (Galiano, 2012)

Aquí en la figura siguiente se muestra un ejemplo de navegación del sitio web:

Figura 2.12: Modelo de navegación



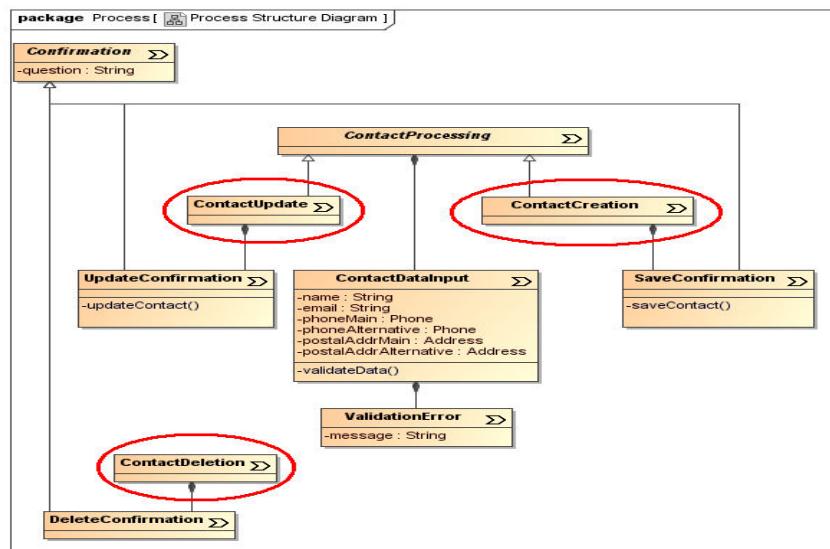
Fuente: (Galiano, 2012)

2.12.5 Modelo de Proceso

Esta modelo especifica las acciones que realiza cada clase de procesos, en este modelo se incluye:

- **Modelo de Estructura de Procesos:** Que define las relaciones entre las diferentes clases de procesos.

Figura 2.13: Modelo de estructura de procesos



Fuente: (Galiano, 2012)

2.12.6 Modelo de Presentación

El Modelo de Navegación detalla cuáles son las clases de navegación y de procesos que pertenecen a una página Web. Estos son los elementos que introduce la metodología UWE en este modelo:

Figura 2.14: Elementos del modelo de presentación

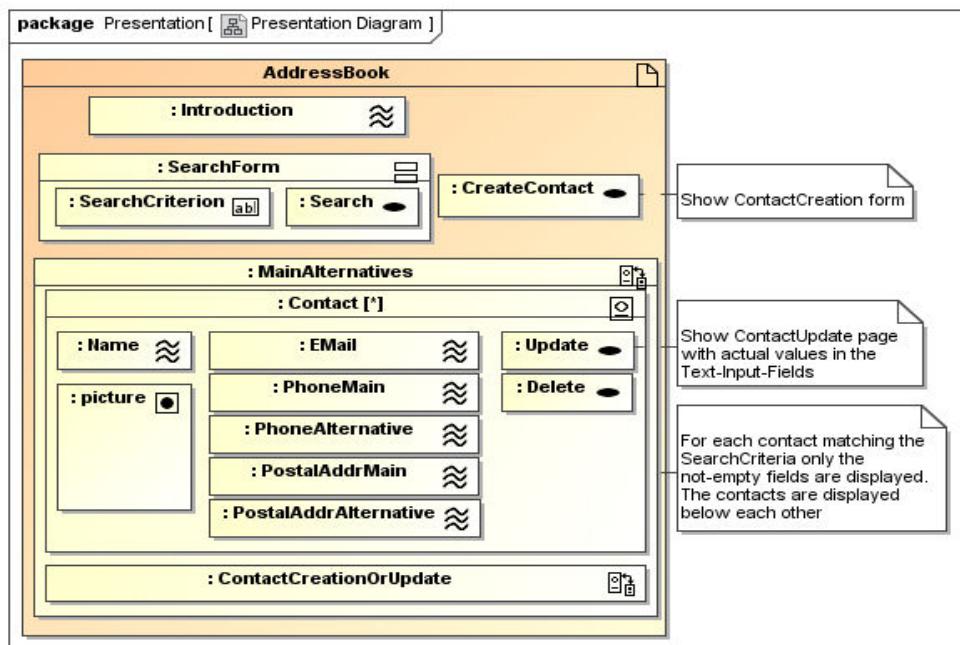
nombres de estereotípos y sus iconos

□ grupo de presentación	□ página de presentación
≈ texto	abl entrada de texto
— ancla	↑ fileUpload
● botón	○ imagen
■ formulario	□ componente de cliente
□ alternativas de presentación	→ selección

Fuente: (Galiano, 2012)

Estos diagramas permitieron especificar dónde y cómo los objetos de navegación serán presentados al usuario, es decir, una representación esquemática de los objetos visibles al usuario.

Figura 2.15: Modelo de presentación



Fuente: (Galiano, 2012)

2.13 HERRAMIENTAS PARA EL DESARROLLO DEL SOFTWARE

2.13.1 Php (Hypertext Pre-Processor)

PHP es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Es usado principalmente en interpretación del lado del servidor (server-side scripting) pero actualmente puede ser utilizado desde una interfaz de línea de comandos o en la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre (Beati, 2011).

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird, SQLite, entre otras. La mayoría del código de PHP ha sido rescatado de C, Java y Perl, con unas cuantas características específicas propias. La meta de este lenguaje es permitirte escribir páginas generadas dinámicamente de forma rápida.

Características de PHP

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Biblioteca nativa de funciones sumamente amplia e incluida y permite las técnicas de programación Orientada a Objetos.
- No requiere definición de tipos de variables (Esta característica también

podría considerarse una desventaja del lenguaje).

- Tiene manejo de excepciones (desde PHP5).

Cómo funciona PHP

- Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica (por ejemplo obteniendo información de una base de datos).
- El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente.

2.13.2 Mysql

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual, MySQL es propietario y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código, además esta base de datos cuenta con una gran cantidad de información muy buena sobre cómo manejarla.

Características de Mysql

Algunas de las características son las siguientes.

- Agrupación de transacciones, reuniendo múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo.
- APIs disponibles para C, C++, Eiffel, Java, Perl, PHP, Python, Ruby, y Tcl.
- El servidor está disponible como un programa separado para usar en un entorno de red cliente/servidor. También está disponible como biblioteca y puede ser incrustado en aplicaciones autónomas.

- Soporte completo para operadores y funciones en las diversas cláusulas tanto de consultas como de agrupación.
- Soporte para alias en tablas y columnas como lo requiere el estándar SQL.

2.13.3 Servidor Apache

Apache es un poderoso servidor web de código abierto, un programa diseñado para la transferencia de datos de hipertexto, utiliza el protocolo de transferencia de hipertexto denotado como HTTP el cual se encarga de la transferencia de datos en la World Wide Web.

Apache presenta ventajas como ser un software de código abierto, costo gratuito, alta aceptación en la red y popularidad para la contribución de su evolución, multiplataforma, capacidad de manejar más de un millón de visitas.

Así como presenta ventajas, presenta también inconvenientes como la falta de integración, formatos de configuración no estándar y carece de un panel de configuración (Fumás, 2014).

2.13.4 Html

Hypertext Markup Language denotado como HTML, es un lenguaje de programación que ayudan a diseñar la interfaz y crear los textos de una página web (González, 2009).

2.13.5 Javascript

Es un lenguaje de scripting basado en objetos, utilizando para acceder a objetos en aplicaciones. Principalmente, se utiliza integrado en un navegador Web permitiendo el desarrollo de interfaces de usuario mejoradas y páginas web dinámicas (Librosweb, 2016).

2.13.6 JQuery

JQuery es una librería de JavaScript (JavaScript es un lenguaje de programación muy usado en desarrollo web). Esta librería de código abierto, simplifica la tarea de programar en JavaScript y permite agregar interactividad a un sitio web sin tener conocimientos del lenguaje.

2.13.7 Bootstrap

Bootstrap, es un framework originalmente creado por Twitter, que permite crear interfaces web con CSS y JavaScript, cuya particularidad es la de adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, el sitio web se adapta automáticamente al tamaño de una PC, una Tablet u otro dispositivo. Esta técnica de diseño y desarrollo se conoce como “responsive design” o diseño adaptativo (Armweb, 2014).

2.13.8 CSS

Son las siglas de Cascading Style Sheets o también Hojas de Estilo en Cascada, es un leguaje que describe la presentación de los documentos estructurados en hojas de estilo para diferentes métodos de interpretación en html y otros tipos de contenidos web (Masadelante, 2016).

2.13.9 Framework

¿Qué es un framework?: Framework es una estructura adaptable conformada por un conjunto de elementos que permiten ejecutar proyectos de diversa índole de una forma más organizada y eficiente, especialmente en el área de la programación.

Framework es un término del idioma inglés que se traduce como “marco de trabajo” o “entorno de trabajo”. En el desarrollo de software, un framework está formado por

conceptos, artefactos, o prácticas que ayudan a organizar los recursos que requiere el programador para hacer su trabajo.

Dicho de otro modo, un marco de trabajo es como una especie de esquema, plantilla o guía que utiliza el programador para aquellas partes del desarrollo que pueden ser automatizadas. Esto genera una mejor gestión del tiempo, al evitar tener que hacer tareas repetitivas de forma manual.

¿Para qué sirve un framework?: Un framework sirve tanto para escribir código fuente como para desarrollar toda una aplicación o software. Permite almacenar y organizar recursos esenciales para el desarrollo como el código fuente, los ficheros de configuración o las librerías de la aplicación.

De esta forma, el programador tiene todos los recursos a la vista y en un solo lugar y el entorno de trabajo se vuelve una estructura que hay que completar, en vez de partir de cero.

2.13.10 Laravel

Laravel es un framework de código abierto para el desarrollo de aplicaciones web en PHP 5 o PHP 7, que posee una sintaxis simple, expresiva y elegante. Fue creado en 2011 por Taylor Otwell, inspirándose en Ruby on Rails y Symfony. Laravel facilita el desarrollo, simplificando el trabajo con tareas comunes como la autenticación, el enrutamiento, gestión sesiones, el almacenamiento en caché, etc. Algunas de las principales características y ventajas de Laravel son” (Gallego, 2017):

Ventajas: El autor (Gallego, 2017) menciona las siguientes ventajas:

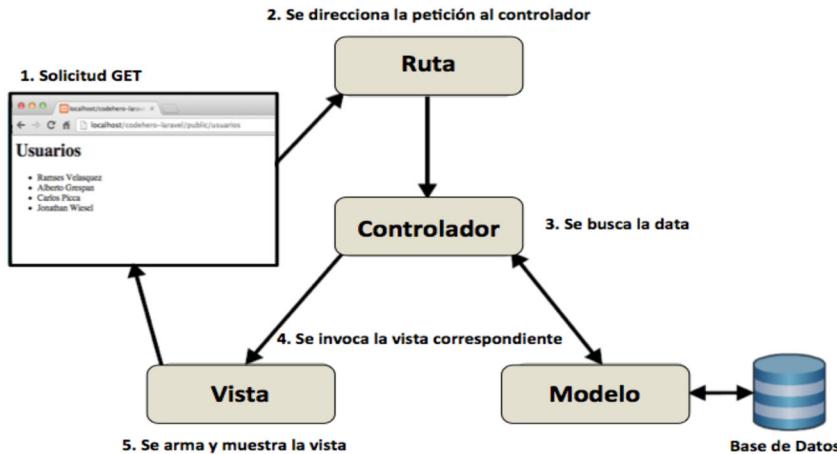
- Está diseñado para desarrollar bajo el patrón MVC (modelo - vista - controlador), centrándose en la correcta separación y modularización del código. Lo que facilita el trabajo en equipo, así como la claridad, el

mantenimiento y la reutilización del código.

- Integra un sistema ORM de mapeado de datos relacional llamado Eloquent aunque también permite la construcción de consultas directas a base de datos mediante su Query Builder.
- Permite la gestión de bases de datos y la manipulación de tablas desde código, manteniendo un control de versiones de las mismas mediante su sistema de Migraciones.
- Utiliza un sistema de plantillas para las vistas llamado Blade, el cual hace uso de la cache para darle mayor velocidad. Blade facilita la creación de vistas mediante el uso de layouts, herencia y secciones.
- Facilita la extensión de funcionalidad mediante paquetes o librerías externas. De esta forma es muy sencillo añadir paquetes que nos faciliten el desarrollo de una aplicación y nos ahorren mucho tiempo de programación.
- Incorpora un intérprete de línea de comandos llamado Artisan que nos ayudará con un montón de tareas rutinarias como la creación de distintos componentes de código, trabajo con la base de datos y migraciones, gestión de rutas, cachés, colas, tareas programadas, etc.

Modelo Vista Controlador MVC: La modelo vista controlador (MVC) es un patrón de arquitectura de software que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones. Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado, define componentes para la representación de la información, y por otro lado para la interacción del usuario. Este patrón de arquitectura de software se basa en las ideas de reutilización de código y la separación de conceptos, características que buscan facilitar la tarea de desarrollo de aplicaciones y su posterior mantenimiento (Gallego, 2017).

Figura 2.16: Funcionamiento básico de Laravel



Fuente: (<https://proyectolaravelmizraim.wordpress.com>)

Los componentes de MVC se podrían definir como sigue:

- **El Modelo:** Es la representación de la información con la cual el sistema opera, por lo tanto, gestiona todos los accesos a dicha información, tantas consultas como actualizaciones. Las peticiones de acceso o manipulación de información llegan al modelo a través del controlador.
- **El Controlador:** Responde a eventos usualmente acciones del usuario e invoca peticiones al modelo cuando se hace alguna solicitud de información, por ejemplo, editar un documento o un registro en una base de datos. Por tanto, se podría decir que el controlador hace de intermediario entre la vista y el modelo.
- **La Vista:** Presenta el modelo y los datos preparados por el controlador al usuario de forma visual. El usuario podrá interactuar con la vista y realizar otras peticiones que se enviarán al controlador.

2.13.11 Visual Studio Code

Visual Studio Code está basado en Electron, una pieza de software que se basa en Chromium, utilizada para el desarrollo de aplicaciones de escritorio con io.js. Fue desarrollado por Microsoft, es totalmente gratuito y se encuentra disponible para utilizar en sistemas Windows, Linux y OS X.

Visual Studio Code incluye soporte para depuración, control de versiones Git, autocompletado de código (también conocido como IntelliSense), y otras características. También es personalizable, así que puede cambiar el tema del editor, los atajos del teclado, preferencias, entre otros.

2.13.12 MagicDraw

MagicDraw, es una herramienta CASE desarrollada por No Magic. Es compatible con el estándar UML 2.3, desarrollo de código para diversos lenguajes de programación (Java, C++ y C#, entre otros) así como para modelar datos. Cuenta con capacidad para trabajar en equipo y es compatible con varios entornos de desarrollo (IDEs).

2.14 MÉTRICAS DE CALIDAD

2.14.1 METODOLOGÍA DE WEBQEM

Es el modelo de prueba de calidad que utilizan los sitios Web, el nombre de WEBQEM proviene de las siglas en inglés Web-site Quality Evaluation method).

Esta metodología de evaluación de calidad de sitios web, propone un enfoque sistemático, disciplinado y cuantitativo que se adecue a la evaluación, comparación y análisis de calidad de sistemas web. WebQEM toma las características del modelo de calidad ISO 9126, las cuales son: usabilidad, funcionalidad, confiabilidad, eficiencia, portabilidad y mantenibilidad (Olsina, 1999).

- **Usabilidad:** Se define como un conjunto de atributos que otorgan el esfuerzo necesario para su uso, y en la evaluación individual de dicho uso, mediante un conjunto de usuarios declarados implícitos.
- **Funcionalidad:** Se define como un conjunto de atributos que otorgan la existencia de un conjunto de funciones y sus propiedades específicas. Las

funciones son aquellas que satisfacen conjuntos de usuarios declarados implícitos.

- **Confiabilidad:** Se define como un conjunto de atributos de habilidad del software para mantener condiciones de establecer su propio nivel de desempeño por un periodo determinado.
- **Eficiencia:** Se define como un conjunto de atributos que otorgan la relación entre el nivel de rendimiento del software y la cantidad de recursos usados por el usuario, bajo las condiciones establecidas.
- **Mantenibilidad:** Se define como un conjunto de atributos que otorgan el esfuerzo necesario para hacer modificaciones específicas.
- **Portabilidad:** Se define como un conjunto de atributos que otorgan la habilidad de software para ser transferido de un entorno a otro.

2.14.1.1 Características de WebQem

Para la medida de calidad se especificarán a continuación las características de usabilidad, funcionalidad, confiabilidad, eficiencia y mantenibilidad.

a) Usabilidad

Es una característica de calidad de producto de alto nivel, que se la puede medir mediante cálculo a partir de métricas directas e indirectas, representa la capacidad o potencialidad del producto para ser utilizado, comprendido y operado por los usuarios, además de ser atractivo para cualquiera.

El criterio de evaluación es un criterio binario, discreto y absoluto. Solo se pregunta si está disponible representado por 1 y si no está disponible con un 0.

Según Olsina de 1999 para evaluar la usabilidad se debe considerar las siguientes características:

- **Comprendibilidad global del sitio:** Es una característica que representa a todas aquellas facilidades que permiten la audiencia, tener una rápida comprensión tanto de la estructura organizativa, como el contenido del sitio web como un todo, facilitando el rápido acceso y recorrido del mismo con sus componentes. Por tal razón, los atributos y sub-características se hallan principalmente en la página principal o en los primeros niveles del sitio.
- **Mecanismo de ayuda y retroalimentación en línea:** Este atributo representa a un conjunto de preguntas (agrupadas y enlazadas) que se realizan con mayor frecuencia, y que están ya publicadas en el sitio con sus respectivas respuestas. A su vez, las respuestas pueden estar enlazadas a otros contenidos. Esto favorece al mecanismo de aprendizaje y/o ayuda evitando potencialmente la demora cognitiva de los visitantes.
- **Aspectos de interfaces y estéticos:** Son factores y elementos relativos a la interacción del usuario, enfocados a un entorno o dispositivos concretos cuyo resultado es la generación de una percepción positiva o negativa de dicho servicio, producto o dispositivo. El diseño de los elementos de la interfaz debe facilitar la interacción del usuario con la funcionalidad, debe generar y formalizar documentos hipermediales comprensibles, interactivos, navegables y facilitar su visualización.
- **Misceláneas:** Este atributo modela el número de lenguajes extranjeros soportados por un sitio (sitios de dominios de aplicación de índole académica, museos, comercio electrónico y otros). Además, especifica el nivel de soporte de cada lenguaje: Total (todas las páginas del sitio), parcial (algunos sub-sitios del sitio), o mínimo (algunas páginas o documentos de algunos sub-sitios). No se computa obviamente el lenguaje nativo del sitio web.

b) Funcionalidad

Para determinar la calidad de la funcionalidad de la aplicación se debe analizar la búsqueda y exploración de contenidos. El criterio de evaluación es un criterio

binario, discreto y absoluto. Solo se pregunta si está disponible representado por 1 y si no está disponible con un 0. Según Olsina de 1999 para evaluar la funcionalidad se debe considerar las siguientes características:

- **Aspectos de búsqueda y recuperación:** Es una característica que modela el mecanismo que permite tener un modo directo de encontrar información.
- **Aspectos de navegación y exploración:** Son los aspectos que tiene el usuario de interacción con las páginas del sistema.
- **Aspectos de dominio orientados al usuario:** Es la información que el usuario puede tener y la información de aprobación de operaciones realizadas por este.

c) Confiabilidad

La medición de esta característica está definida por el complemento de los casos de deficiencia encontrados en la aplicación. El criterio elemental es uno de variable normalizada, continuo y absoluto; en donde si $BL =$ Número de enlaces rotos encontrados. $TL =$ Número total de enlaces del sitio. La fórmula para computar la variable es: $X=100 - (BL * 100/TL) * 10;$ donde, si $X<0$ entonces $X= 0.$

- No deficiencia: Este atributo representa básicamente a los enlaces encontrados que conducen a nodos destinos ausentes (también llamados enlaces ausentes o pendientes).

d) Eficiencia

Es una característica de calidad de producto de alto nivel que se la puede medir mediante calculo a partir de métricas directas e indirectas y principalmente representa a la relación entre el grado de performance del artefacto y la cantidad de recursos (tiempo, espacio, etc.) usados bajo ciertas condiciones.

El criterio de evaluación es un criterio binario, discreto y absoluto. Solo se pregunta si está disponible representado por 1 y si no está disponible con un 0.

Según Olsina de 1999 para evaluar la eficiencia se debe considerar las siguientes características:

- **Desempeño:** Se mide el tamaño de todas las páginas (estáticas) del sitio web considerando todos sus componentes gráficos, tabulares y contextuales. El tamaño de cada página se especifica como una función del tiempo de espera y de la velocidad mínima establecida para una línea de comunicación dada.
- **Accesibilidad:** Este atributo representa la accesibilidad de la información que se encuentran en las páginas del sitio.

e) Mantenibilidad

Esta característica es la interfaz a modificar, incluyendo correcciones o mejoras del software, a cambios de entorno, y especificaciones de requerimientos funcionales, para poder medir la calidad del mantenimiento del sistema se utilizará el índice de madurez del software (IMS), que indica la estabilidad de la aplicación. Esto se calcula mediante el número total de módulos del sistema menos la suma de los módulos añadidos, modificados y eliminados, dividido por el total de módulos del sistema.

2.15 MÉTODO DE ESTIMACIÓN DE COSTOS DE SOFTWARE

2.15.1 Costo y Beneficio

La planificación es una actividad de gran importancia, en la que se establecen objetivos y metas de un proyecto, además de las estrategias, políticas y procedimientos para alcanzarlos.

Una de las variables a calcular durante la planificación del proyecto es el esfuerzo, es decir, la fuerza de trabajo requerida para el desarrollo, medida en meses hombre,

días-hombre, y en general, unidad de tiempo hombre.

La estimación de lo que costará el desarrollo de un software es una de las actividades de planeación que reviste especial importancia, ya que una de las características que debe tener un producto de software es que su costo sea adecuado, de lo contrario el proyecto puede fracasar.

2.15.2 Modelo constructivo de Costo (COCOMO)

Método de estimación de costos COCOMO⁹ para estimar el costo de software. Es un modelo matemático de base empírica utilizada para valoración de costo de software. Este modelo al ser un modelo basado en modelos matemático usa formulas y puede ser calculado a través de un software para tener el mismo error. Además, es orientado al modelo final no a las fases intermedias (Pressman,2010).

El modelo COCOMO está orientado a la magnitud del producto final, está basado en estimaciones matemáticas, mide el “tamaño” del proyecto y utiliza las líneas de código como unidad de medida. Dos de los aspectos fundamentales del modelo COCOMO son los submodelos y los modos de desarrollo.

Los submodelos de COCOMO son tres y cada uno ofrece un nivel de detalle y aproximación, cada vez mayor, a medida que avanza el proceso de desarrollo de software: básico, intermedio y detallado.

- **Básico:** Líneas de código (LDC), se basa exclusivamente en el tamaño expresado en LDC para obtener una primera aproximación rápida de esfuerzo.
- **Intermedio:** Líneas de código (LDC) + multiplicador de esfuerzo (ME). Toma en cuenta las características del básico, pero incluye factores que tienen que ver con el aseguramiento de calidad.

⁹ Por su acrónimo del idioma inglés COnstructive COst MOdel

- **Avanzado o detallado:** Líneas de código (LDC) + multiplicador de esfuerzo (ME) + calculo por fases. Incluye todo lo del modelo intermedio además del impacto de cada conductor de costos en las distintas fases del desarrollo.

2.15.2.1 Modelo de COCOMO: Básico e intermedio

Estos modos de desarrollo permiten utilizar cuatro valores constantes. En la Tabla 2.2 se muestran los modos de desarrollo y los valores constantes respectivos. Estos valores constantes, codificados aquí como “a”, “b”, “c” y “d”, son propuestos por el modelo COCOMO para complementar las ecuaciones de cálculo usadas en el modelo.

Tabla 2.2: Valores constantes por modo de desarrollo

<i>Modo de desarrollo</i>	COCOMO Básico <i>a</i>	COCOMO Intermedio <i>A</i>	<i>b</i>	<i>c</i>	<i>d</i>
Orgánico	2.4	3.2	1.05	2.50	0.38
Semiacoplado		3.0	1.12		0.35
Empotrado	3.6	2.8	1.20		0.32

Fuente: (Boehm,1983)

También en la tabla 2.3 se muestra las ecuaciones se utilizan para calcular el esfuerzo nominal en personas/mes (E), tiempo estimado en meses (T) y personal requerido (P) así como los multiplicadores de esfuerzo (ME).

Tabla: 2.3 Ecuaciones por tipo de modelo COCOMO: Básico e Intermedio

Ecuación	Submodelo básico	Submodelo intermedio
Esfuerzo (E)	$(E) = a * (KLDC)b$	$(E) = a * (KLDC)b * ME$
Tiempo (T)	$(T) = c * (E)d$	$(T) = c * (E)d$
Personal (P)	$(P) = E/T$	$(P) = E/T$

Fuente: (Boehm,1983)

Donde:

E: Es el esfuerzo requerido por el proyecto, en personas/mes.

T: Es el tiempo requerido por el proyecto, en mes.

P: Es el número de personas requeridas por el proyecto.

a,b,c,d: Son constantes con valores definidos descritas en la tabla 2.2 según cada modelo de desarrollo.

KLDC: Es la cantidad de líneas de código:

ME: Es el multiplicador de esfuerzo que depende de 15 atributos.

Los multiplicadores de esfuerzo, utilizados en la ecuación de esfuerzo del submodelo intermedio, son quince agrupados en cuatro grandes categorías: atributos de producto, atributos de computador, atributos personales y atributos del proyecto. Cada uno de estos multiplicadores de esfuerzo, tiene una valoración que se clasifica en una escala de 6 valores desde “muy bajo”, “bajo”, “nominal”, “alto”, “muy alto” y “extraordinariamente alto”.

Tabla 2.4: Multiplicadores de esfuerzo (ME) del COCOMO Intermedio

Multiplicadores de esfuerzo (ME)			Valoración					
			Muy bajo	Bajo	Nominal	Alto	Muy alto	Extr. alto
Atributos del producto								
1.	RELY	Fiabilidad requerida del software	0,75	0,88	1,00	1,15	1,40	
2.	DATA	Tamaño de la base de datos		0,94	1,00	1,08	1,16	
3.	CPLX	Complejidad del producto	0,70	0,85	1,00	1,15	1,30	1,65
Atributos de la computadora								
4.	TIME	Restricciones del tiempo de ejecución			1,00	1,11	1,30	1,66
5.	STOR	Restricciones del almacenamiento princ.			1,00	1,06	1,21	1,56
6.	VIRT	Inestabilidad de la máquina virtual	0,87		1,00	1,15	1,30	
7.	TURN	Tiempo de respuesta del computador	0,87		1,00	1,07	1,15	
Atributos del personal								
8.	ACAP	Capacidad del analista	1,46	1,19	1,00	0,86	0,71	
9.	AEXP	Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
10.	PCAP	Capacidad de los programadores	1,42	1,17	1,00	0,86	0,70	
11.	VEXP	Experiencia en S.O. utilizado	1,21	1,10	1,00	0,90		
12.	LEXP	Experiencia en el lenguaje de progr.	1,14	1,07	1,00	0,95		
Atributos del proyecto								
13.	MODP	Uso de prácticas de programación modernas	1,24	1,10	1,00	0,91	0,82	
14.	TOOL	Uso de herramientas software	1,24	1,10	1,00	0,91	0,83	
15.	SCED	Restricciones en la duración del proy.	1,23	1,08	1,00	1,04	1,10	

Fuente: (Boehm,1983)

2.15.2.2 Modelo de COCOMO detallado

Presenta principalmente dos mejoras respecto a los anteriores modelos:

- 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.

Para el presente proyecto se utilizará el modelo COCOMO intermedio, dado que para nuestro caso este tipo de modelo realiza la estimación de costo con bastante precisión.

2.16 SEGURIDAD

El crecimiento que ha tenido internet, trae consigo amenazas a la privacidad de la información. Cuando se habla de seguridad comúnmente es la protección de la información contra el acceso desautorizado o la modificación de la información del sistema.

a) Seguridad a nivel de base de datos:

- El acceso a la base de datos

b) Seguridad a nivel de aplicación:

- Control de acceso
- Validación de datos de entrada
- Pruebas de código

c) Copias de seguridad:

- Backup de la base de datos

Las pruebas de seguridad están diseñadas para probar la vulnerabilidad en el

ambiente del lado del cliente, las comunicaciones de red que ocurren mientras los datos pasan del cliente al servidor. Cada uno de estos dominios puede recibir ataques y es labor de quien prueba la seguridad descubrir las debilidades que puede explorar quienes tengan la intención de hacerlo (Pressman, 2010).

A continuación, se muestra las vulnerabilidades del lado del cliente.

- Vulnerabilidad en los navegadores
- Vulnerabilidad en los correos electrónicos
- Acceso no autorizado a cookies

Por lo tanto, se puede aplicar las siguientes medidas de seguridad:

- **Cortafuegos:** Mecanismo de filtrado que combina hardware y software para examinar cada paquete de información entrante para garantizar que no sea un dato sospechoso.
- **Uso de autenticación:** Una contraseña o clave es una forma de autenticación que utiliza información secreta para controlar el acceso hacia algún recurso. A aquellos que desean acceder a la información se les solicita un clave; si conoce o desconoce la contraseña, se concede o se niega a la información según el caso.
- **Modelo de criptografía:** La encriptación está asociada con la transformación de un mensaje inteligente a una forma o inteligente con la ayuda de una clave secreta antes de que sea colocada en un medio inseguro.
- **Algoritmo básico del Hash:** El nombre de hash se usa para identificar una función criptográfica muy importante en el mundo informático. Estas funciones tienen como objetivo primordial codificar datos para formar una cadena de caracteres única. Todo ello sin importar la cantidad de datos introducidos inicialmente en la función. Estas funciones sirven para asegurar la autenticidad de datos, almacenar de forma segura contraseñas, y la firma de documento electrónicos.

CAPÍTULO III

MARCO APlicativo

3.1 INTRODUCCIÓN

El presente capítulo, tiene como finalidad describir el análisis y diseño del sistema web de registro y control de gaceta Universitaria propuesto para la Unidad de Secretaría General dependiente de la Universidad Pública de El Alto. Para el desarrollo del sistema se aplicará la metodología de desarrollo ágil Programación Extrema (XP), cuyas fases y otras características de la metodología fueron descritas en el capítulo anterior que corresponde al marco teórico, para recordar en figura 3.1; así mismo la aplicación de otros conceptos descrito en el mismo capítulo.

Para trabajar exitosamente con XP y UWE se necesita entender el proceso de desarrollo con XP. Se optó por UWE por ser una versión liviana de UML que no genera una gran cantidad de documentación, lo que respecta uno de los puntos que valora más el desarrollo ágil de software.

Figura 3.1: Fases de la programación extrema



Fuente: (ciclo de vida XP¹⁰)

3.2 ANÁLISIS DE REQUERIMIENTOS

¹⁰ Disponible en <http://managementplaza.es/blog/sabes-como-funciona-xp> (consultado el 20-05-2020)

3.2.1 Situación actual

De acuerdo a los procedimientos de publicación de la gaceta universitaria y demora en su ejecución según reglamento del Honorable Consejo Universitario artículos 23 y 29, se tuvo algunas observaciones, por lo cual dio el siguiente análisis de la situación actual en la que se encuentra la unidad de Secretaría General.

La Unidad de Secretaría General de la UPEA es la responsable de transcribir, resguardar y administrar la documentación Institucional, generados en el Congreso Universitario, Asamblea General Docente Estudiantil, Honorable Consejo Universitario, Honorable Consejo de Área, Rectorado, Vice rectorado y las Unidades Académicas. Al momento la gestión es un poco lenta y presenta dificultad al brindar información a los usuarios, sobre los contenidos mencionados anteriormente, actualmente se almacena toda la información en carpetas físicas, al momento de necesitar algún tipo de información se procede a realizar la búsqueda de forma manual, por tal razón no se puede acceder de una manera rápida y precisa a la información que se requieran, según las necesidades de las autoridades, personal administrativo o por parte de los estudiantes.

Proceso de gaceta universitaria

- Por la cantidad de documentación que se genera en la unidad de secretaría general no se tiene un registro sistematizado y la búsqueda es manual.
- Las publicaciones de la gaceta universitaria se elaboran de forma impresa una vez al año, por lo que se demora en la publicación de normas que fueron aprobadas en el día, semana o mes.

3.2.2 Ingeniería de Requerimientos

Ingeniería de requerimientos es la descripción de una condición o capacidad que debe cumplir un sistema, ya sea derivada de una necesidad de usuario identificada,

o bien, estipulada en un contrato, especificación u otro documento formalmente impuesto al inicio del proceso.

En la tabla siguiente se describen las categorías de funciones que se utilizan para poder clasificar los requerimientos que se realizan en las tablas 3.1 y 3.2.

Tabla 3.1: Categoría de Funciones

Categoría de la función	Significado
Evidente	Debe realizarse y los usuarios están consistentes de que se ha realizado.
Oculto	Debe realizarse, aunque no es visible para los usuarios. Esto se aplica a muchos servicios técnicos subyacentes, por ejemplo, guardar información en un mecanismo persistente de almacenamiento.
Superflua	Opcionales, su inclusión no repercute de forma significativa en costo ni otras funciones.

Fuente: (Larman, 1999)

3.2.2.1 Requerimientos Funcionales

Especifica el “que” tiene que hacer el sistema, es decir, el propósito del sistema.

Tabla 3.2: Requerimientos Funcionales

Referencia	Función	Categoría
R.1.1	Buscar registro de publicaciones (resoluciones y convenios).	Evidente
R.1.2	Visualización y muestra de la información publicada	Evidente
R.1.3	El sistema debe tener seguridad al acceso del sistema.	Oculto
R.1.4	Autenticación y validación de usuarios de acuerdo a al tipo de usuario de asignación.	Oculto
R.1.5	El Usuario Administrador Principal, tiene la facultad de registrar, adicionar, eliminar y listar (Gestión de Usuarios y Gestión de Menús del sistema).	Evidente
R.1.6	El Usuario Encargado de Archivo, tiene la facultad de registrar, adicionar, eliminar y listar (Gestión de Resoluciones y Gestión de Convenios).	Evidente
R.1.7	Asignación de usuarios a la gestión de la gaceta universitaria a de tareas específicas de acuerdo al tipo de usuario.	Oculto
R.1.8	Consulta dentro de la base de datos.	Oculto

R.1.9	Validación de entrada de datos.	Oculto
R.1.10	Generación de reportes de Resoluciones y Convenios.	Evidente
R.1.11	Inicio y cierre del sistema	Oculto

Fuente: Elaboración Propia

3.2.2.2 Requerimientos No funcionales

Especifican el “como” tiene que hacerse y funcionar el sistema.

Tabla 3.3: Requerimientos no funcionales

Referencia	Función	Categoría
R.1.1	El sistema debe visualizarse y funcionar correctamente en cualquier navegador como Mozilla Firefox, Google Chrome y Opera.	Evidente
R.1.2	El sistema debe visualizarse de forma responsive en dispositivos móviles.	Evidente
R.1.3	Soporte y mantenimiento periódico, para asegurar el buen rendimiento y funcionamiento del sistema.	Evidente

Fuente: Elaboración Propia

3.3 APLICACIÓN DE LA METODOLOGÍA XP

Una de las razones más importantes por lo que se ha optado por utilizar la metodología XP es que se necesita el empleo del menor tiempo para la pronta culminación del proyecto a realizarse.

El proceso de desarrollo de software XP estará basado en entregas e iteraciones. Se dividió el proyecto en 5 iteraciones, con un tiempo aproximado de uno a tres semanas por interacción.

Esta metodología se centra en la comunicación interpersonal del equipo de trabajo, siendo esta la clave para el éxito en el desarrollo de los sistemas que se van desarrollando en periodos cortos. En tal sentido este proyecto se realizará por una sola persona, no será posible la aplicación de la práctica de programación en pareja.

3.3.1 Fase I: Planificación

Según lo planteado anteriormente desarrollaremos el sistema haciendo uso de la metodología XP, es decir mediante **historias de usuarios**, además de los requerimientos, se definirán todas las tareas que serán necesarias para poder desarrollar el sistema.

3.3.1.1 Historia de usuarios

Una historia de usuario es una descripción de la función o el trabajo deseado. Describe lo que hay que hacer y no como se debe hacer. Las historias de usuario son los requisitos del cliente para el desarrollo de un proyecto de software, estas se clasifican en iteraciones para su implementación y son escritas por los clientes.

A. Iteración 1 (Portal web)

Acceso al portal web: Este módulo realiza la función de mostrar la información al público en general sobre las resoluciones, convenios y publicaciones registradas en el sistema no puede manipular ninguna operación de altas, bajas y modificaciones correspondientes.

Tabla 3.4: Historia de Usuario, Gestión Portal Web

Historia de Usuario	
Numero: 01	Usuario: Cliente
Nombre Historia: Acceso al portal web	
Prioridad en negocio: Media	Riesgo en Desarrollo: Media
Puntos estimados: 1	Iteración Asignada: 3°
Programador responsable: Hernan Catari Yujra	
Descripción: El usuario podrá ingresar fácilmente al portal web, para informarse de los servicios y publicaciones de la gaceta universitaria con	

los que cuenta la Unidad de Secretaría General, también podrá hacer búsquedas de registros con filtros de búsqueda.

Observaciones: Cualquier usuario, desde cualquier parte del mundo podrá visitar el portal web mediante la URL de la página web se Secretaría General.

Fuente: Elaboración Propia

B. Iteración 2 (Gestión de Menús)

A continuación, se dan a conocer las diferentes iteraciones del menú del sistema, que consta de 02 historias de usuario: Categoría Menú y Menú.

- ✓ **Gestión Categoría Menú:** Este módulo realiza la función de asignar una categoría de menú a un menú que operara el sistema, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.5: Historia de Usuario, Gestión Categoría Menú

Historia de Usuario	
Numero: 01	Usuario: Administrador Principal
Nombre Historia: Registrar Categoría Menú	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Alto
Puntos estimados: 1	Iteración Asignada: 2°
Programador responsable: Hernan Catari Yujra	
Descripción: Se procede al registro de nueva categoría de menú con nombre, orden, posición e ícono se validan los datos para evitar duplicidad con el nombre, se asigna la categoría menú esto hará la diferencia para acceder al 100% del sistema o parcialmente, una vez colocado los datos necesarios registra y se brinda las opciones de edición y eliminación.	
Observaciones: La manipulación de la opción de categoría menú se asigna al administrador principal que recae al encargado de la unidad de sistemas de información y estadística SIE.	

También el usuario podrá realizar más módulos de categoría de menú de acuerdo a requerimiento de la unidad solicitante.

Fuente: Elaboración Propia

- ✓ **Gestión Menú:** Este componente realiza la función de asignar un enlace de menú al sistema de acuerdo a la categoría de menú, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.6: Historia de Usuario, Gestión Menú

Historia de Usuario	
Numero: 02	Usuario: Administrador Principal
Nombre Historia: Registrar Menú	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Alto
Puntos estimados: 2	Iteración Asignada: 2°
Programador responsable: Hernan Catari Yujra	
Descripción: Se procede al registro de nuevo enlace de menú con tipo de categoría, nombre, link, orden e ícono se validan los datos para evitar duplicidad con el nombre y tipo de categoría, se asigna el enlace de menú esto hará la diferencia para acceder al 100% del sistema o parcialmente, una vez colocado los datos necesarios registra y se brinda las opciones de edición y eliminación.	
Observaciones: La manipulación de la opción de gestión de menú se asigna al administrador principal que recae al encargado de la unidad se sistemas de información y estadística SIE. También el usuario podrá realizar más módulos de menú de acuerdo a requerimiento de la unidad solicitante.	

Fuente: Elaboración Propia

C. Iteración 3 (Gestión de Usuarios)

A continuación, se dan a conocer las diferentes iteraciones del Usuario del sistema, que consta de 02 historias de usuario: Tipo Usuario y Usuario.

- ✓ **Gestión Tipo Usuario:** Este módulo realiza la función de asignar un tipo de usuario para un usuario que operara el sistema, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.7: Historia de Usuario, Gestión Tipo Usuario

Historia de Usuario	
Numero: 01	Usuario: Administrador Principal
Nombre Historia: Registrar Tipo Usuario	
Prioridad en negocio: Alta	Riesgo en Desarrollo: Alto
Puntos estimados: 1	Iteración Asignada: 2°
Programador responsable: Hernan Catari Yujra	
Descripción: El usuario podrá administrar la creación de tipos de usuarios. También se procederá al registro de nuevo tipo de usuario con nombre, y descripción de usuario se validan los datos para evitar duplicidad con el nombre, se asigna el tipo usuario esto hará la diferencia para acceder al 100% del sistema o parcialmente, una vez colocado los datos necesarios registra y se brinda las opciones de edición y eliminación.	
Observaciones: La manipulación de la opción de tipo de usuario se asigna al administrador principal que recae al encargado de la unidad de sistemas de información y estadística SIE.	

Fuente: Elaboración Propia

- ✓ **Gestión Usuario:** Este módulo realiza la función de asignar usuario al sistema de acuerdo al tipo de usuario, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.8: Historia de Usuario, Gestión Usuario

Historia de Usuario	
Numero: 02	Usuario: Administrador Principal
Nombre Historia: Registrar Usuario	

Prioridad en negocio: Alta	Riesgo en Desarrollo: Alto
Puntos estimados: 2	Iteración Asignada: 2°
Programador responsable: Hernan Catari Yujra	
<p>Descripción: Se procede al registro del nuevo usuario buscando primeramente una persona de otra base de datos externa y asignándole un usuario, contraseña, tipo de usuario y estado del usuario, se validan los datos para evitar duplicidad con el usuario y tipo de usuario, se asigna el usuario esto hará la diferencia para acceder al 100% del sistema o parcialmente, una vez colocado los datos necesarios para un usuario se registra y se brinda la opción de edición y eliminación.</p> <p>Como usuario Administrador Principal debe tener la facultad de habilitar o deshabilitar usuarios, asignar permisos, que les permita ingresar solo a determinadas secciones del sistema, es decir no tiene acceso a todas.</p>	
<p>Observaciones: La manipulación de la opción de gestión de usuario se asigna al administrador principal que recae al encargado de la unidad de sistemas de información y estadística SIE.</p> <p>El usuario estará activo siempre y cuando esté trabajando en la institución (UPEA), de caso contrario estará en estado Inactivo.</p>	

Fuente: Elaboración Propia

D. Iteración 4 (Gestión Resoluciones)

A continuación, se dan a conocer las diferentes iteraciones de la gestión de resoluciones en el sistema, que consta de 02 historias de usuario: Tipo Resolución y Resoluciones.

- ✓ **Gestión Tipo Resolución:** Este módulo realiza la función de asignar un tipo de resolución para una resolución que será registrado al sistema, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.9: Historia de Usuario, Gestión Tipo Resolución

Historia de Usuario	
Numero: 01	Usuario: Encargado de Archivo
Nombre Historia: Registrar Tipo Resolución	
Prioridad en negocio: Media	Riesgo en Desarrollo: Media
Puntos estimados: 1	Iteración Asignada: 1°
Programador responsable: Hernan Catari Yujra	
<p>Descripción: El usuario Encargado de Archivo luego de ingresar al sistema, elegirá la opción de menú gestión resoluciones para introducir los datos del nuevo tipo resolución asignándole nombre y descripción, se validan los datos para evitar duplicidad con el tipo de resolución, una vez colocado los datos necesarios registra y se brinda las opciones de edición y eliminación.</p>	
<p>Observaciones: para registrar los datos del tipo de resolución se debe llenar toda la información solicitada, con mucha exactitud y claridad. Solo podrá ser realizada esta acción por el Encargado de Archivo de la unidad de Secretaría General.</p>	

Fuente: Elaboración Propia

- ✓ **Gestión Resolución:** Este módulo realiza la función de asignar resolución al sistema de acuerdo al tipo de resolución, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.10: Historia de Usuario, Gestión Resolución

Historia de Usuario	
Numero: 02	Usuario: Encargado de Archivo
Nombre Historia: Registrar Resolución	
Prioridad en negocio: Media	Riesgo en Desarrollo: Media
Puntos estimados: 2	Iteración Asignada: 4°
Programador responsable: Hernan Catari Yujra	
<p>Descripción: El usuario Encargado de Archivo luego de ingresar al sistema, elegirá la opción menú gestión resoluciones para introducir los</p>	

datos de la nueva resolución asignándole número, gestión, fecha, título, resuelve, estado y también es necesario tener disponible la información del tipo de resolución y unidad, se validan los datos para evitar duplicidad en la resolución, una vez colocado los datos necesarios registra y se brinda las opciones de edición (ingresa archivo de resolución), y eliminación.

Observaciones: para registrar los datos de resolución se debe llenar toda la información solicitada, con mucha exactitud y claridad.

Solo podrá ser realizada esta acción por el Encargado de Archivo de la unidad de Secretaría General.

Fuente: Elaboración Propia

E. Iteración 5 (Gestión de Convenios)

A continuación, se dan a conocer las diferentes iteraciones de la gestión de convenios en el sistema, que consta de 04 historias de usuario: Representante, Institución, Tipo Convenio y Convenios.

- ✓ **Gestión Representante:** Este módulo realiza la función de asignar un representante legal para un convenio que será registrado al sistema, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.11: Historia de Usuario, Gestión Representante

Historia de Usuario	
Numero: 01	Usuario: Encargado de Archivo
Nombre Historia: Registrar Representante	
Prioridad en negocio: Media	Riesgo en Desarrollo: Media
Puntos estimados: 1	Iteración Asignada: 1°
Programador responsable: Hernan Catari Yujra	
Descripción: El usuario Encargado de Archivo luego de ingresar al sistema, elegirá la opción menú gestión convenios para introducir los datos del nuevo representante legal asignándole datos personales y el cargo, se	

validan los datos para evitar duplicidad, una vez colocado los datos necesarios registra y se brinda las opciones de edición y eliminación.

Observaciones: para registrar los datos de representante legal se debe llenar toda la información solicitada, con mucha exactitud y claridad.

Solo podrá ser realizada esta acción por el Encargado de Archivo de la unidad de Secretaría General.

Fuente: Elaboración Propia

- ✓ **Gestión Institución:** Este módulo realiza la función de asignar Instituciones al sistema de acuerdo al representante legal, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.12: Historia de Usuario, Gestión Institución

Historia de Usuario	
Numero: 02	Usuario: Encargado de Archivo
Nombre Historia: Registrar Institución	
Prioridad en negocio: Media	Riesgo en Desarrollo: Media
Puntos estimados: 1	Iteración Asignada: 1°
Programador responsable: Hernan Catari Yujra	
Descripción: El usuario Encargado de Archivo luego de ingresar al sistema, elegirá la opción menú gestión de convenios para introducir los datos de la nueva institución asignándole datos relevantes de la institución, país y ciudad a la que pertenece y su representante legal, se validan los datos para evitar duplicidad en la resolución, una vez colocado los datos necesarios registra y se brinda las opciones de edición y eliminación.	
Observaciones: para registrar los datos de resolución se debe llenar toda la información solicitada, con mucha exactitud y claridad.	
Solo podrá ser realizada esta acción por el Encargado de Archivo de la unidad de Secretaría General.	

Fuente: Elaboración Propia

- ✓ **Gestión Tipo Convenio:** Este módulo realiza la función de asignar tipo de convenio al sistema, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.13: Historia de Usuario, Gestión Tipo Convenio

Historia de Usuario	
Numero: 03	Usuario: Encargado de Archivo
Nombre Historia: Registrar Tipo Convenio	
Prioridad en negocio: Media	Riesgo en Desarrollo: Media
Puntos estimados: 1	Iteración Asignada: 1°
Programador responsable: Hernan Catari Yujra	
Descripción: El usuario Encargado de Archivo luego de ingresar al sistema, elegirá la opción menú gestión de convenios para introducir los datos del nuevo tipo de convenio asignándole nombre y descripción, se validan los datos para evitar duplicidad, una vez colocado los datos necesarios registra y se brinda las opciones de edición y eliminación.	
Observaciones: para registrar los datos de tipo de convenio se debe llenar toda la información solicitada, con mucha exactitud y claridad. Solo podrá ser realizada esta acción por el Encargado de Archivo de la unidad de Secretaría General.	

Fuente: Elaboración Propia

- ✓ **Gestión Convenio:** Este módulo realiza la función de asignar convenios al sistema de acuerdo al tipo de convenio, institución y representante, también se realiza los procesos de altas, bajas y modificaciones correspondientes.

Tabla 3.14: Historia de Usuario, Gestión Convenios

Historia de Usuario	
Numero: 04	Usuario: Encargado de Archivo
Nombre Historia: Registrar Convenio	
Prioridad en negocio: Media	Riesgo en Desarrollo: Media

Puntos estimados: 1	Iteración Asignada: 4°
Programador responsable: Hernan Catari Yujra	
<p>Descripción: El usuario Encargado de Archivo luego de ingresar al sistema, elegirá la opción menú gestión de convenios para introducir los datos del nuevo convenio asignándole número, gestión, título, objetivo, fecha inicio, fecha fin, estado y también es necesario tener disponible la información del tipo de convenio, institución, representante y unidad ejecutora, se validan los datos para evitar duplicidad, una vez colocado los datos necesarios registra y se brinda las opciones de edición (ingresa archivo de convenio) y eliminación.</p>	
<p>Observaciones: para registrar los datos de convenio se debe llenar toda la información solicitada, con mucha exactitud y claridad.</p> <p>Solo podrá ser realizada esta acción por el Encargado de Archivo de la unidad de Secretaría General.</p>	

Fuente: Elaboración Propia

3.3.1.2 Tareas de las historias de usuario

Las tareas se obtienen de la descripción de las historias de usuario y se utilizan para definir de forma concreta el trabajo que deben realizar los miembros del equipo de desarrollo, para realizar una historia de usuario.

En la siguiente tabla se muestran de forma general las tareas de programación y se detalla cada uno de ellos.

Tabla 3.15: Detalle de las tareas por iteración

Historia de Usuario		Tareas
1ra Iteración	Página web	<ul style="list-style-type: none"> • Preparar un interfaz amigable para el usuario final. • Preparar la información concerniente con respecto a la gaceta universitaria. • Diseñar filtros de búsqueda.

2da Iteración	Gestión de Menús	<ul style="list-style-type: none"> Estructurar las tablas de la base de datos para la categoría de menús y url de menús. Identificar las categorías para asignar a un menú. Diseñar la interfaz de Categoría. Diseñar la interfaz para los Menús dinámicos. Realizar las validaciones correspondientes. Preparar la interfaz para adicionar, modificar, eliminar y listar los categoría y menú.
3ra Iteración	Gestión de Usuarios	<ul style="list-style-type: none"> Estructurar las tablas de la base de datos para los usuarios y tipo de usuario. Identificar los tipos de usuarios, permisos y funciones de acuerdo al menú asignado. Diseñar la interfaz de autenticación. Diseñar la interfaz para los usuarios. Realizar la validación de los usuarios. Preparar la interfaz para adicionar, modificar, eliminar y listar los usuarios.
4ta Iteración	Gestión de Resoluciones	<ul style="list-style-type: none"> Estructurar las tablas de la base de datos para los tipos de resoluciones y resoluciones. Identificar los tipos de resoluciones. Diseñar la interfaz de tipo de resolución. Diseñar la interfaz para las resoluciones. Realizar la validación correspondiente. Preparar la interfaz para adicionar, modificar, eliminar y listar los usuarios. Registrar los archivos de la resolución en formato pdf.
5ta Iteración	Gestión de Convenios	<ul style="list-style-type: none"> Estructurar las tablas de la base de datos para los convenios, tipo de convenio, institución y representante. Preparar la interfaz para adicionar, modificar, eliminar y listar los convenios. Diseñar la interfaz de tipo de convenio. Diseñar la interfaz para instituciones. Diseñar la interfaz para representante. Realizar la validación correspondiente a cada interfaz. Registrar los archivos del convenio en formato pdf.

Fuente: Elaboración Propia

3.3.1.3 Planificación de Iteraciones

Como parte del ciclo de vida de un proyecto con la metodología XP, se crea el plan

de duración de cada uno de las iteraciones, este plan se encarga de mostrar las historias de usuario que serán implantadas en cada una de las iteraciones, así como la duración estimada de cada una y el orden en que se implementan. La planificación resultante se divide en 5 iteraciones el cual se muestra en la siguiente tabla.

Tabla 3.16: Planificación de Iteraciones

Nº	Historia de Usuario	Prioridad	Riesgo	Tiempo estimado (semanas)	Iteración asignada
H1	Portal Web	Media	Media	1	3
H2	Gestión Menús	Alta	Alta	3	4
H3	Gestión Usuarios	Alta	Alta	3	4
H4	Gestión Resoluciones	Media	Media	3	5
H5	Gestión Convenios	Media	Media	4	7

Fuente: Elaboración propia

3.3.1.4 Plan de Entregas

Para la planificación de la entrega, se debe determinar la velocidad de desarrollo, así mismo el número de iteraciones, posterior a ello se asignarán las historias de usuario a las iteraciones y su entrega se estimará en tiempo ideal.

La finalización de las entregas terminara, cuando el software es implantado en producción, donde el usuario hace uso del software.

Cada iteración representa un ciclo de trabajo en la cual se puede encontrar una o más historias de usuario, en este entendido se lo debe planificar para conocer la distribución de tiempo en el desarrollo de cada historia.

A continuación, se detalla el plan de entrega establecidas en cada iteración y fecha tentativas para su ejecución.

Tabla 3.17: Planificación de Iteraciones

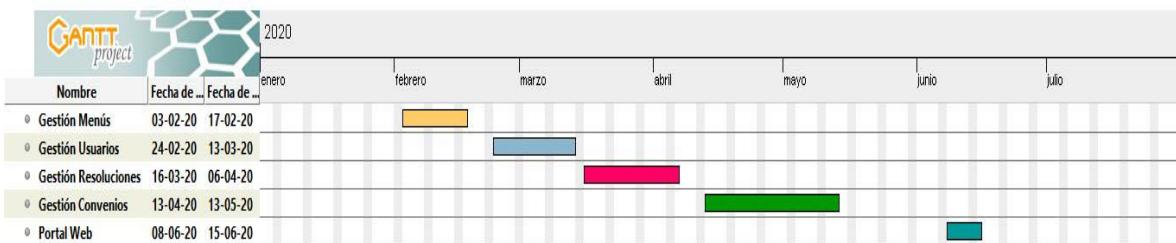
Iteración	Historia de Usuario	Duración	Fecha Inicio	Fecha Fin
-----------	---------------------	----------	--------------	-----------

1	Portal Web	1 semana	08/06/2020	15/06/2020
2	Gestión Menús	3 semanas	03/02/2020	17/02/2020
3	Gestión Usuarios	3 semanas	24/02/2020	13/03/2020
4	Gestión Resoluciones	3 semanas	16/03/2020	06/04/2020
5	Gestión Convenios	4 semanas	13/04/2020	13/05/2020

Fuente: Elaboración propia

En la siguiente tabla se muestra la descripción de las iteraciones mediante un cronograma de fecha de entrega.

Tabla 3.18: Planificación de Iteraciones



Fuente: Elaboración propia

3.3.2 Fase II: Diseño

La fase diseño en la construcción de software es muy importante, ya que es la parte fundamental de la aplicación en donde se conoce el funcionamiento del mismo. Primero se debe diseñar la Base de Datos la cual se la va a realizar en mysql.

Dentro de la misma generamos todas las siguientes tablas de nuestro sistema para la Unidad de Secretaría General, donde se encuentran todas las tablas para todos los módulos que se trabajarán.

Luego de esto con código SQL vamos a generar nuestra base de datos la cual es la siguiente y se genera el siguiente modelo de Datos entre los cuales tenemos el

modelo físico.

3.3.2.1 Modelado UWE

En la parte de modelado haremos una combinación de UWE, ya que nos brinda un conjunto de modelos especializados para el diseño web.

Todos los diagramas fueron realizados con MagicDraw 18.0 con su complementación de su plugin MagicUWE, porque simplifica el modelado con UWE en las diferentes etapas.

a) Modelo Conceptual o Contenido

El modelo conceptual tiene por propósito mostrar las principales entidades del sistema.

Figura 3.2: Diagrama de contenido



Fuente: Elaboración propia

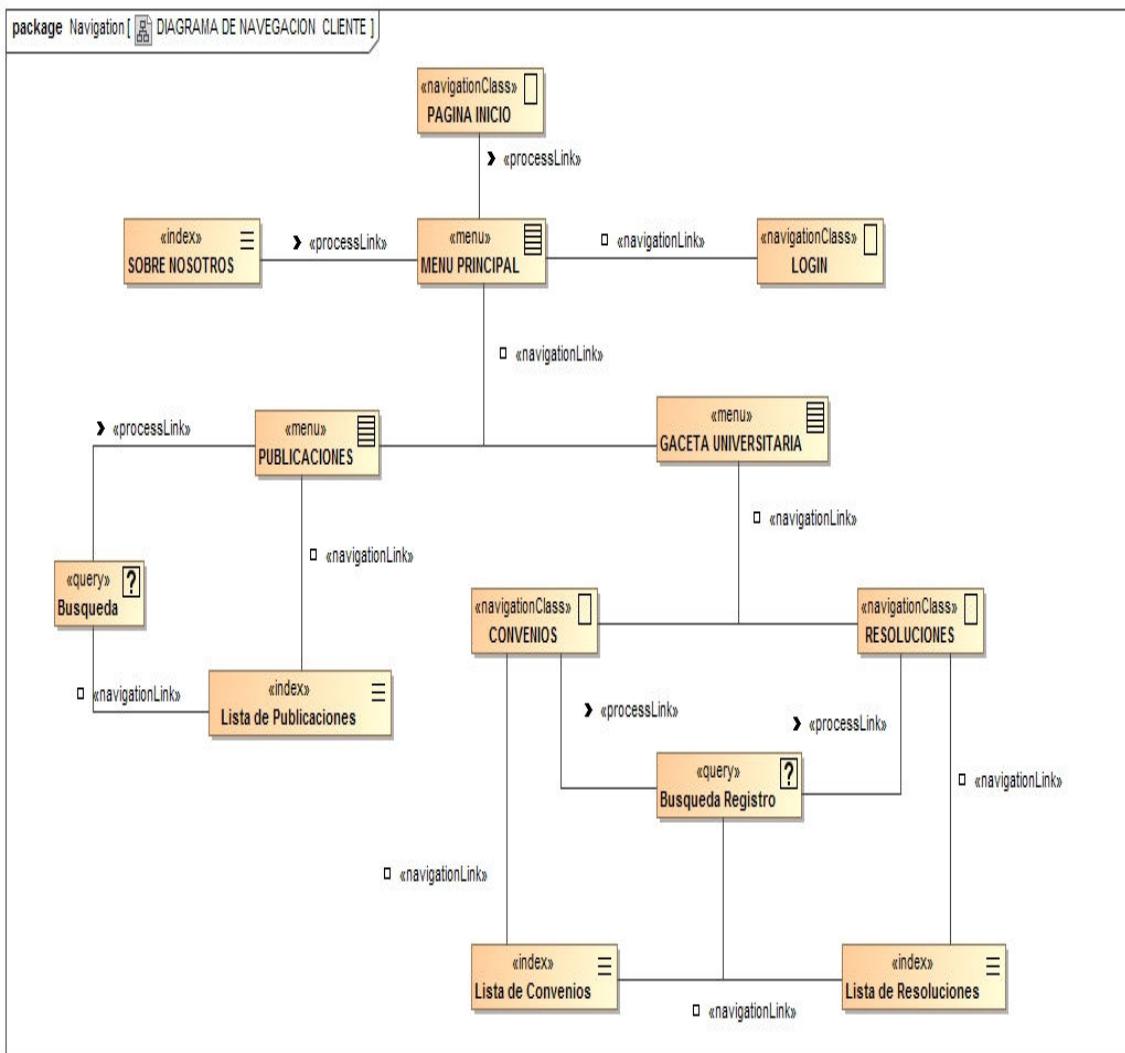
En anexo c) se presenta el modelo relacional de la base de datos.

b) Modelo de Navegación

Como segundo modelo uwe, se presentará a continuación, los diagramas de navegación para las clases relevantes del sistema como: diagramas de cliente, administrador principal y encargado archivo.

- Cliente

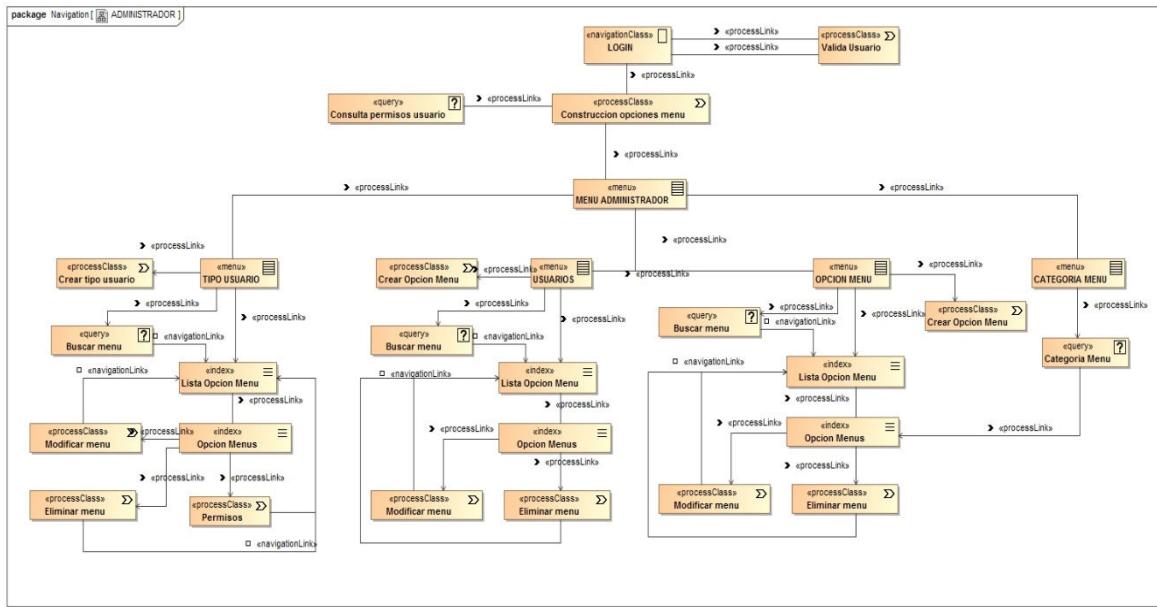
Figura 3.3: Diagrama de navegación cliente



Fuente: Elaboración propia

- Administrador Principal

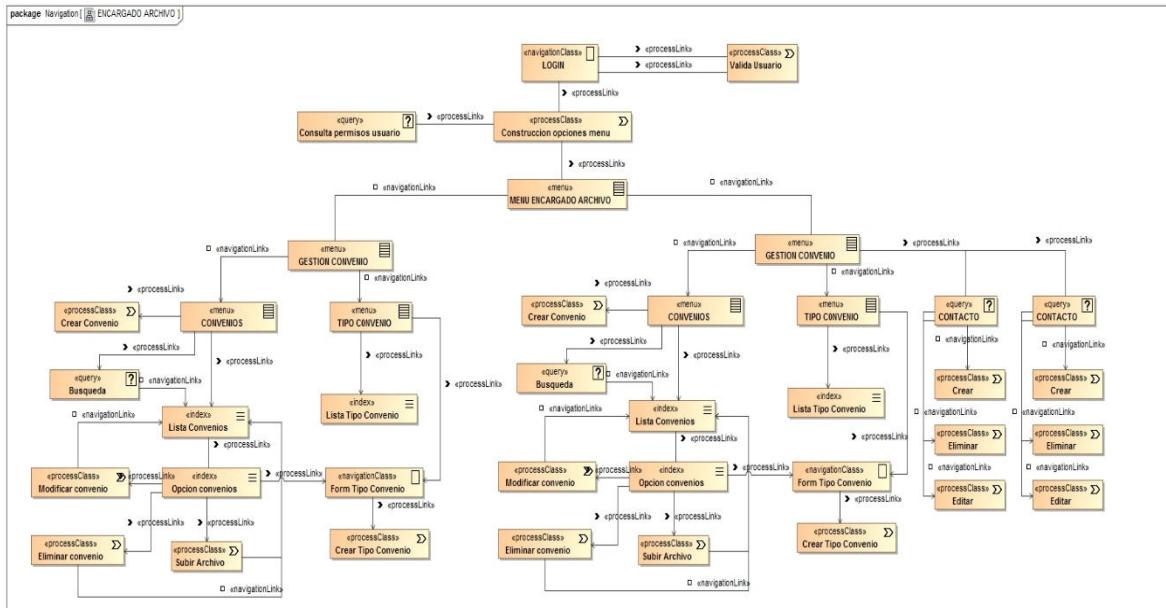
Figura 3.4: Diagrama de navegación Administrador Principal



Fuente: Elaboración propia

- Encargado Archivo

Figura 3.5: Diagrama de navegación Encargado Archivo

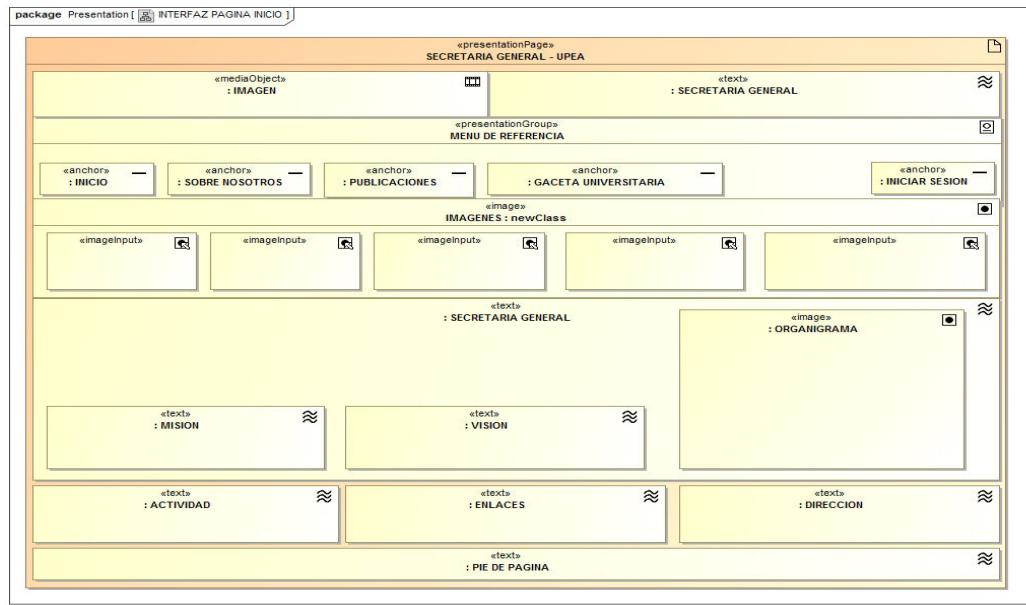


Fuente: Elaboración propia

c) Modelo de presentación

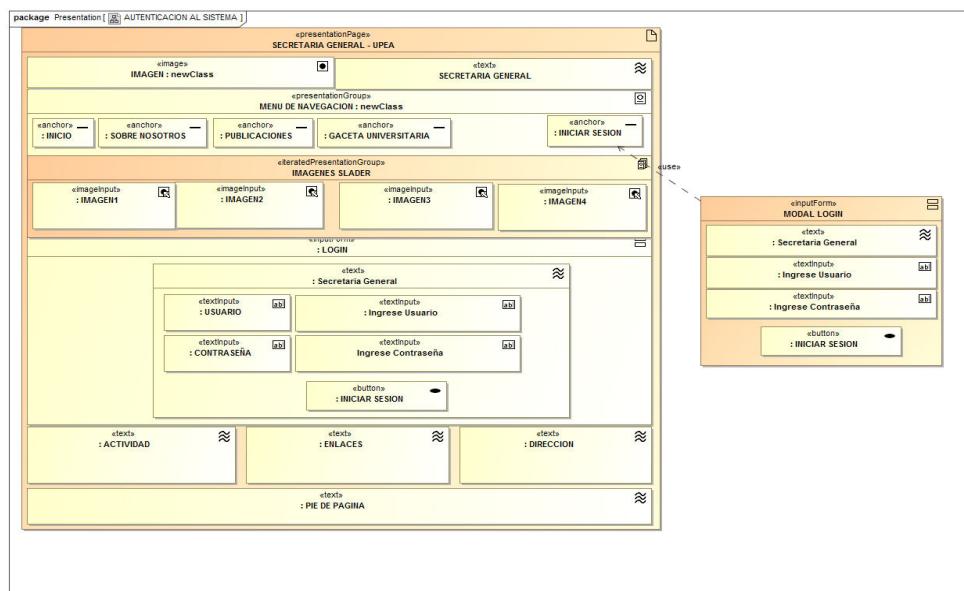
Como tercer modelo UWE se presentarán los diagramas de presentación correspondientes a un esquema de las vistas.

Figura 3.6: Diagrama de presentación: interfaz web página de inicio



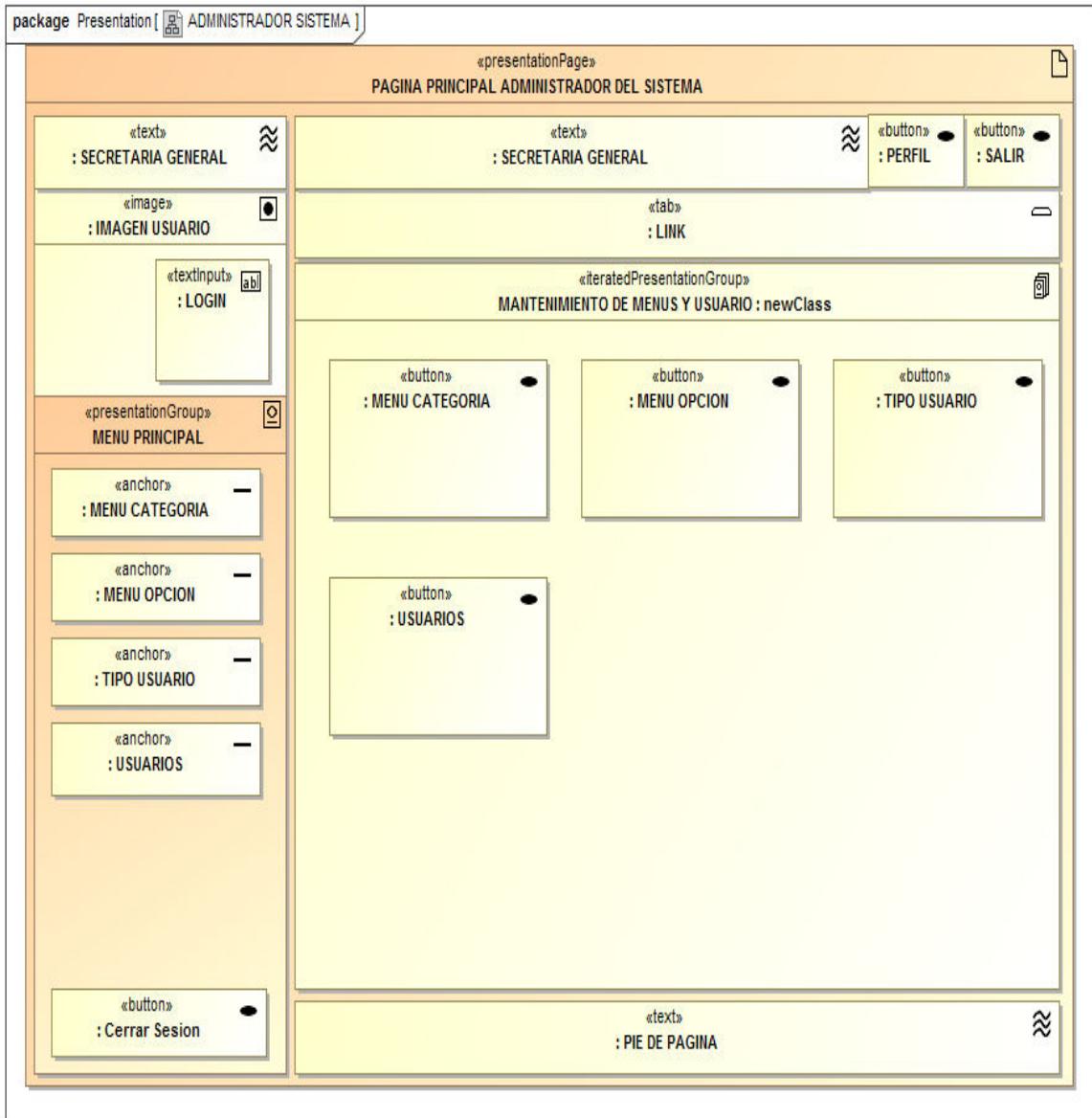
Fuente: Elaboración propia

Figura 3.7: Diagrama de presentación: autenticación al sistema



Fuente: Elaboración propia

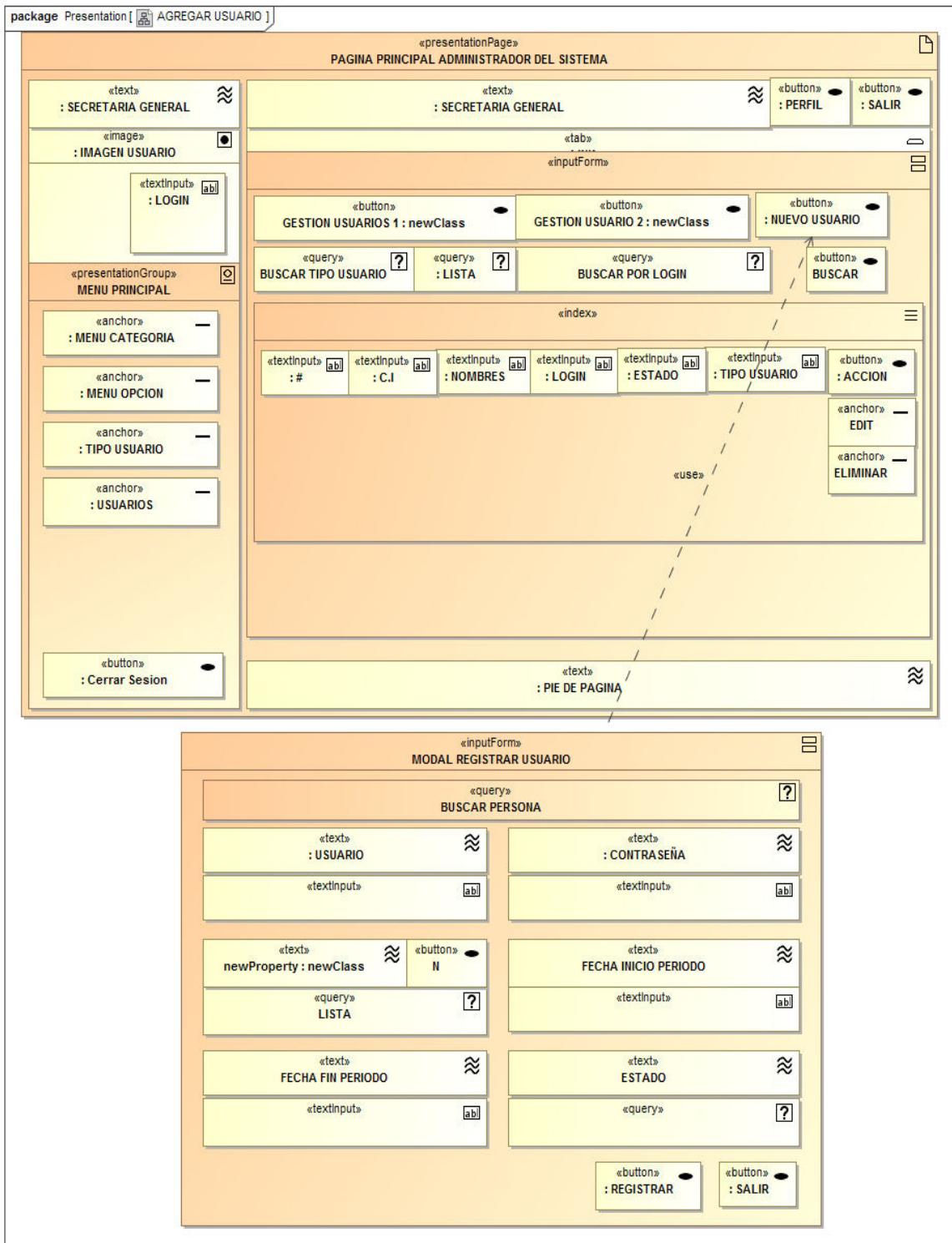
Figura 3.8: Diagrama de presentación: página principal administrador



Fuente: Elaboración propia

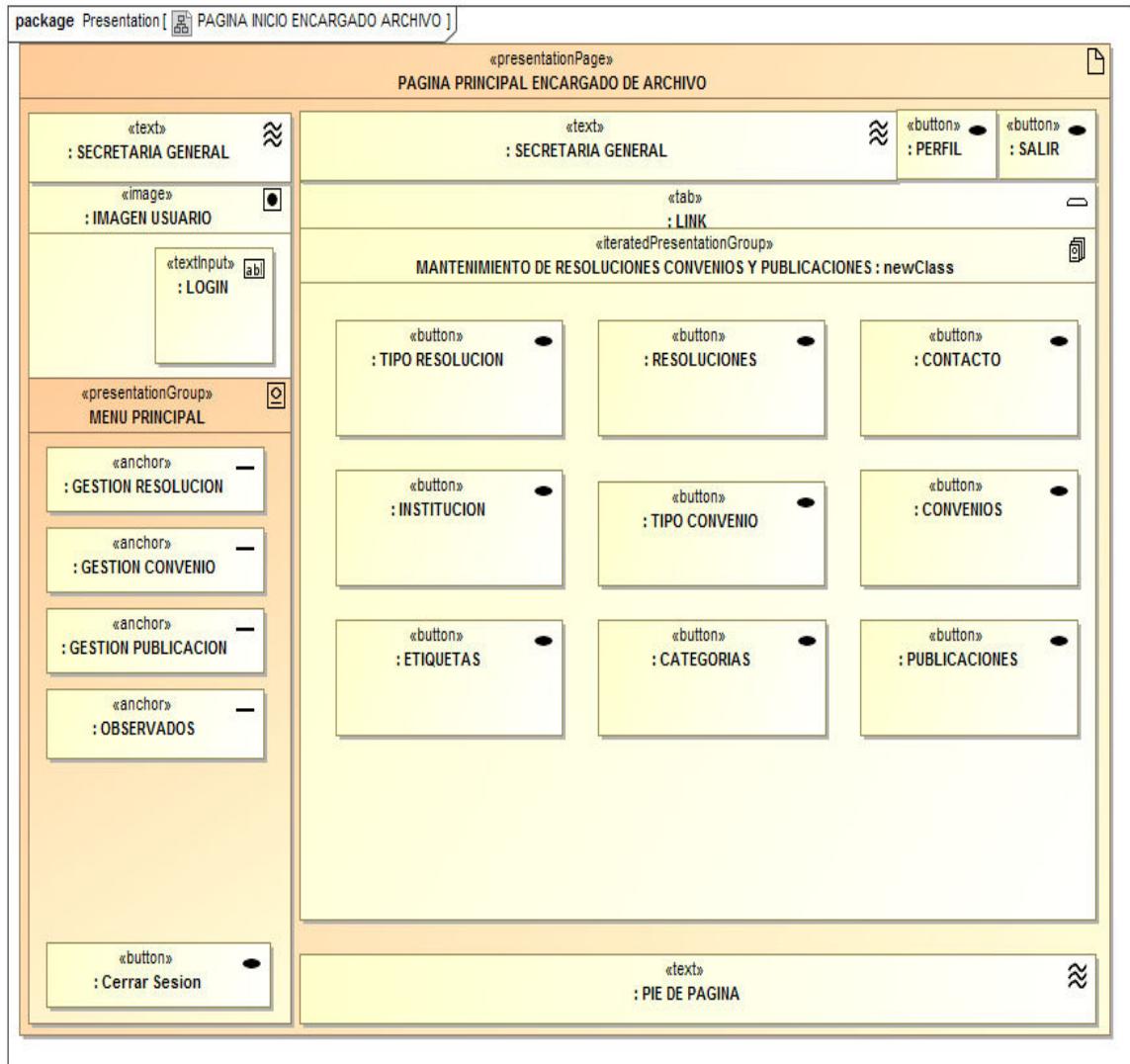
En la figura 3.9, es una ventana básica del componente usuario que tiene las opciones de adicionar o agregar, listar o ver, editar o modificar y eliminar algún registro y se repite para la mayoría de los objetos en el sistema en el módulo de administrador del sistema, esto puede apreciarse en el siguiente diagrama de presentación.

Figura 3.9: Diagrama de presentación: Agregar Usuario



Fuente: Elaboración propia

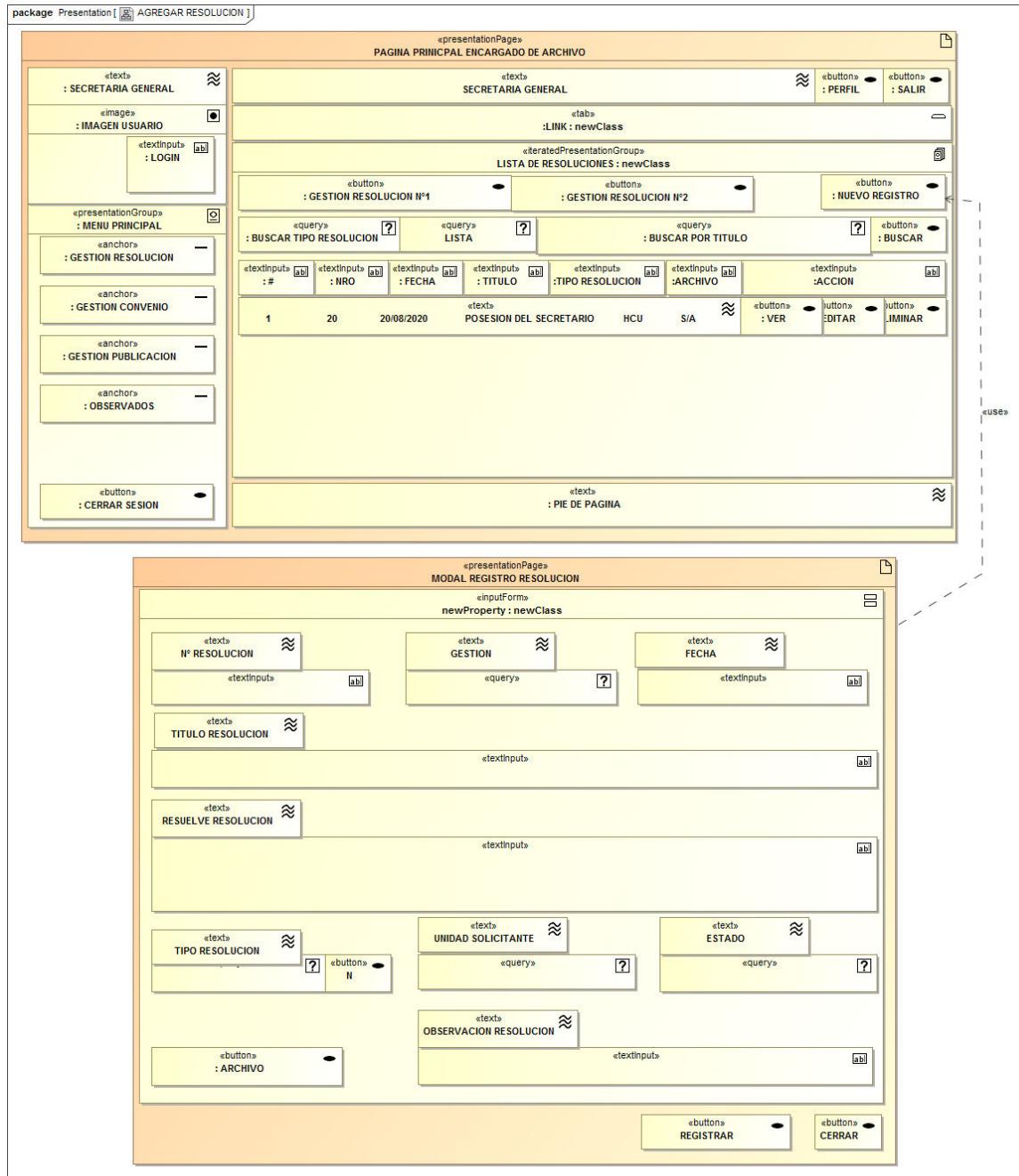
Figura 3.10: Diagrama de presentación: página principal encargado archivo



Fuente: Elaboración propia

En la figura 3.11, es una ventana básica del componente resolución que tiene las opciones de adicionar o agregar, listar o ver, editar o modificar y eliminar algún registro y se repite para la mayoría de los objetos en el sistema del módulo de encargado de archivo, esto puede apreciarse en el siguiente diagrama de presentación.

Figura 3.11: Diagrama de presentación: Agregar Resolución



Fuente: Elaboración propia

Se diseñaron los diagramas de presentación los más relevantes, en todo caso el sistema tiene el mismo modelo de presentación, para todas las tablas de la base de datos.

3.3.3 Fase III: Codificación

En esta fase la codificación de cada uno de las historias de usuario se realizará con el cliente, permitiendo de esa forma una retroalimentación de lo que el cliente quiere.

A continuación, se describe y muestra algunas capturas de pantallas más sobresalientes, que son el resultado de las historias de usuario:

En la figura 3.12 se muestra la página principal del ingreso al sistema.

Figura 3.12: Página de inicio del sistema



Fuente: Elaboración propia

Figura 3.13: Menú de inicio de sesión donde el usuario ingresa



Fuente: Elaboración propia

Figura 3.14: Página principal del administrador del sistema

Fuente: Elaboración propia

Figura 3.15: Pagina de registro de un usuario

Fuente: Elaboración propia

Figura 3.16: Lista de usuarios del sistema

The screenshot shows a web application interface for managing users. At the top, there is a header with the logo of 'Universidad Pública de El Alto' and several navigation links. On the left, a sidebar menu includes 'MENU PRINCIPAL' and 'GESTIÓN ADMIN'. The main content area has two tabs: 'GESTIÓN USUARIOS N°1' and 'GESTIÓN USUARIOS N°2'. The first tab is active, displaying a table of user data. The table columns are: #, C.I., NOMBRES, LOGIN, ESTADO, TIPO USUARIO, and ACCIONES. The data in the table is as follows:

#	C.I.	NOMBRES	LOGIN	ESTADO	TIPO USUARIO	ACCIONES
1	6144513 LP.	JUAN REGIS MUÑEZ SIRPA	SG-6144513	Activo	ADMINISTRADOR PRINCIPAL	
2	390823 LP.	LIDIA IRMA ZUAZO YUJRA	SG-390823	Inactivo	SECRETARIA	
3	4838414 LP.	HERNAN CATARI YUJRA	SG-4838414	Activo	AUXILIAR OFICINA	

At the bottom of the page, there are buttons for 'Cerrar sesión' (Logout) and '© 2020 - Versión 2.0.'

Fuente: Elaboración propia

Figura 3.17: Modificar menú categoría

The screenshot shows a modal dialog titled 'MODIFICAR - MENU CATEGORÍA'. The form contains the following fields:

- Categoría: A dropdown menu labeled 'Seleccione una categoría'.
- Nombre: A text input field with a red asterisk indicating it is required. The value 'GESTIÓN ADMIN' is entered.
- Orden: A text input field with a red asterisk. The value '1' is entered.
- Posicion: A dropdown menu labeled 'Vertical'.
- Icono: A text input field containing the icon code 'mdi-playlist-check'.
- Buttons at the bottom: 'Modificar' (with a checkmark icon) and 'Cerrar' (with a warning icon).

Fuente: Elaboración propia

Figura 3.18: Modificar menú opción

MODIFICAR - MENU OPCIÓN

Categoría: * GESTIÓN ADMIN

Nombre: * MENU OPCIÓN

Link: * opcionmenu

Orden: * 2

Icono: * fa fa-angle-right

Modificar Cerrar

Fuente: Elaboración propia

Figura 3.19: Permisos Administrador principal

PERMISOS: ADMINISTRADOR PRINCIPAL

GESTIÓN ADMIN

Nota: Si asigna alguna opción de "GESTIÓN ADMIN" es su responsabilidad ...!

MENU CATEGORÍA	<input checked="" type="checkbox"/>
MENU OPCIÓN	<input checked="" type="checkbox"/>
TIPO USUARIO	<input checked="" type="checkbox"/>
USUARIOS	<input checked="" type="checkbox"/>
USUARIO HISTORIAL	<input checked="" type="checkbox"/>

GESTIÓN RESOLUCIÓN

Nota: Si asigna alguna opción de "GESTIÓN RESOLUCIÓN" es su responsabilidad ...!

GESTIÓN CONVENIO

Nota: Si asigna alguna opción de "GESTIÓN CONVENIO" es su responsabilidad ...!

GESTIÓN OBSERVADOS

Nota: Si asigna alguna opción de "GESTIÓN OBSERVADOS" es su responsabilidad ...!

GESTIÓN PUBLICACIÓN

Nota: Si asigna alguna opción de "GESTIÓN PUBLICACIÓN" es su responsabilidad ...!

Guardar Cancelar

Fuente: Elaboración propia

Figura 3.20: Registrar Resolución

REGISTRAR - RESOLUCIÓN

Nº RESOLUCIÓN: 007 GESTIÓN: 2015 FECHA: 28/01/2015

TÍTULO RESOLUCIÓN: DESCONSIDERACIÓN DE CAJAS DE PAGO

RESUELVE RESOLUCIÓN:

ARTICULO ÚNICO. - Se dispone la creación de cajas de pagos de obligaciones económicas al interior de la Universidad Pública de El Alto, desconcentrándose cajas de pago para docentes, cajas de pago estudiantes y cajas de pago administrativos. La Máxima Autoridad Ejecutiva a través de sus unidades respectivas, es la encargada de dar cumplimiento a la presente resolución.

TIPO RES.: 1 - HCU
Seleccione 1 - HCU
2 - RECTORAL
3 - ADMINISTRATIVA
4 - I CONGRESO 2004
5 - II CONGRESO 2007
6 - II CONGRESO 2008
7 - III CONGRESO 2012

UNIDAD SOLICITANTE: RECTORADO

ESTADO: APROBAD

OBSERVACION RESOLUCIÓN: Ingrese observación Ejm. Ninguna

Registrar Cerrar

Fuente: Elaboración propia

Figura 3.21: Registrar Archivo PDF de la Resolución

MODIFICAR - RESOLUCIÓN

Nº RESOLUCIÓN: 007 GESTIÓN: 2015 FECHA: 28/01/2015

TÍTULO RESOLUCIÓN: DESCONSIDERACIÓN DE CAJAS DE PAGO

RESUELVE RESOLUCIÓN:

ARTICULO ÚNICO. - Se dispone la creación de cajas de pagos de obligaciones económicas al interior de la Universidad Pública de El Alto, desconcentrándose cajas de pago para docentes, cajas de pago estudiantes y cajas de pago administrativos. La Máxima Autoridad Ejecutiva a través de sus unidades respectivas, es la encargada de dar cumplimiento a la presente resolución.

TIPO RES.: HCU
Seleccione Archivo HCU 007-2015.pdf

UNIDAD SOLICITANTE: RECTORADO

ESTADO: APROBAD

OBSERVACION RESOLUCIÓN: Ingrese observación Ejm. Ninguna

Modificar Cerrar

Fuente: Elaboración propia

Figura 3.22: Lista de Resoluciones registradas

The screenshot shows a web-based application for managing university resolutions. At the top, there's a header with the university's name and a navigation bar with links for 'INICIO' and 'HOME'. On the left, a sidebar titled 'MENU PRINCIPAL' includes 'GESTIÓN RESOLUCIÓN', 'GESTIÓN CONVENIO', 'GESTIÓN OBSERVADOS', and 'GESTIÓN PUBLICACIÓN'. The main content area has two tabs: 'GESTIÓN RESOLUCIÓN N°1' and 'GESTIÓN RESOLUCIÓN N°2'. Below these tabs is a search and filter section with fields for 'Tipo Resolución' (Todos), 'Filas a Mostrar' (10), and 'Buscar por Título'. A table lists the first resolution, which is titled 'DESCONSIDERACIÓN DE CAJAS DE PAGO'. The table columns include '#', 'Nro.', 'FECHA', 'TÍTULO', 'RESUELVE', 'TIPO RES.', 'ARCHIVO', and 'ACCIÓN'. The resolution content describes the creation of payment boxes for staff, students, and administrative staff. Below the table, there are sections for 'ESTADO' (APROBADO) and 'UNIDAD' (RECTORADO). At the bottom right, there are buttons for 'Total' and '1'.

Fuente: Elaboración propia

Figura 3.23: Ver resolución PDF registrada

This screenshot shows a PDF document titled 'b475TzCYegokApthrxmURxlbx9EsiPypNCfNcWvN.pdf'. The document is a scanned copy of a resolution from the 'Universidad Pública de El Alto'. It features a circular seal at the top left and a formal title at the top right. The text is in Spanish and discusses the creation of payment boxes for different groups within the university. The PDF viewer interface includes a toolbar with icons for zooming and a status bar at the bottom indicating the file name and size.

Fuente: Elaboración propia

3.3.4 Fase IV: Pruebas

La metodología XP ofrece una utilidad al instante de realizar pruebas de aceptación ya que mediante esto se puede realizar una regresión al proceso de las iteraciones para su corrección, haciendo que sea bastante dinámica y flexible. En esta fase se realiza una descripción de las pruebas de aceptación, y corresponden a la comprobación del correcto funcionamiento de la aplicación.

3.3.4.1 Pruebas de aceptación

Las pruebas de aceptación se hacen para determinar el correcto funcionamiento del sistema y los requerimientos del cliente. Estas pruebas se hacen directamente con el cliente, cuando el sistema esté finalizado en su totalidad.

Tabla 3.19: Ejecución de las pruebas de aceptación

Historia de Usuario	Prueba de aceptación ejecutadas	Resultado de la prueba
Página web	<ul style="list-style-type: none">- Se mostró una interfaz elegante para el usuario final con la manipulación de vistas de resoluciones, convenios, publicaciones y objetivos de la unidad.- Se verificó los campos a mostrar sobre la información concerniente con respecto a la gaceta universitaria.- Controlar que la búsqueda de registros cumpla con las expectativas del usuario final.	Aprobado

Gestión de Menús	<ul style="list-style-type: none"> - Se mostró una interfaz elegante para la manipulación de adicionar, modificar, eliminar y listar las categorías de menú y opción de menú del sistema de acuerdo al permiso de logeo correspondiente de tipo de usuario y/o rol del sistema. - Controlar que en la tabla menu_categoria en nombre sea único. - Controlar y verificar que en la tabla menu_opcion la url sea único y corresponda a una categoría correctamente. - Controlar que no se permita registrar en el formulario de menú categoría y menú opción, si los campos obligatorios no están presentes o llenados como corresponde. - Controlar que los campos obligatorios sean debidamente llenados, mostrando alertas y señalando los campos que deben ser ingresados. - Se listaron todas las categorías de menú y opción de menú del sistema. - Se eliminaron y modificaron categorías de menú y opción de menú. - Se mostró un mensaje de confirmación una vez que se guardó un registro correctamente. 	Aprobado
------------------	--	----------

Gestión de Usuarios	<ul style="list-style-type: none"> - Se mostró una interfaz elegante para la manipulación de adicionar, modificar, eliminar y listar los usuarios y tipo de usuario y/o rol de acuerdo al permiso de logeo correspondiente de tipo de usuario y/o rol del sistema. - Controlar en el campo login y tipo de usuario sea único en la tabla usuarios. - Controlar que no se permita registrar en el formulario de usuario y tipo de usuario, si los campos obligatorios no están presentes o llenados como corresponde. - Validar campos numéricos y de texto para evitar errores en el sistema. - Controlar que los campos obligatorios sean debidamente llenados, mostrando alertas y señalando los campos que deben ser ingresados. - Se registra y asigna el tipo de usuario y/o rol a los usuarios del sistema. - Se listaron todos los usuarios y tipos de usuario del sistema. - Se eliminaron y modificaron a los usuarios y tipo de usuario. - Se mostró un mensaje de confirmación una vez que se guardó un registro correctamente. 	Aprobado
---------------------	---	----------

Gestión de Resoluciones	<ul style="list-style-type: none"> - Se mostró una interfaz elegante para la manipulación de adicionar, modificar, eliminar y listar los resoluciones y tipo de resolución de acuerdo al permiso de logeo correspondiente de tipo de usuario y/o rol del sistema. - Controlar en el campo número, gestión y tipo de resolución sean únicos en la tabla resoluciones. - Controlar que no se permita registrar en el formulario de resoluciones y tipo de resolución, si los campos obligatorios no están presentes o llenados como corresponde. - Validar campos numéricos y de texto para evitar errores en el sistema. - Controlar que los campos obligatorios sean debidamente llenados, mostrando alertas y señalando los campos que deben ser ingresados. - Se registra y asigna el tipo resolución, unidad solicitante a una resolución del sistema. - Controlar que la subida se archivos cumpla con las especificaciones de extensión de archivo y tamaño de archivo PDF correctamente en el formulario de edición caso contrario no se registra el archivo, se muestra de error. - Se listaron todas las resoluciones y tipos de resolución del sistema. - Se eliminaron y modificaron a los resoluciones y tipos de resolución. - Se mostró un mensaje de confirmación una vez que se guardó un registro correctamente. 	Aprobado
-------------------------	--	----------

Gestión de Convenios	<ul style="list-style-type: none"> - Se mostró una interfaz elegante para la manipulación de adicionar, modificar, eliminar y listar los contactos, institución, tipo de convenio y convenios de acuerdo al permiso de logeo correspondiente de tipo de usuario y/o rol del sistema. - Controlar en el campo número, gestión y tipo de convenio sean únicos en la tabla convenios. - Controlar que no se permita registrar en el formulario de convenios, tipo convenio, contactos y institución si los campos obligatorios no están presentes o llenados como corresponde. - Validar campos numéricos y de texto para evitar errores en el sistema. - Controlar que los campos obligatorios sean debidamente llenados, mostrando alertas y señalando los campos que deben ser ingresados. - Se registra y asigna el tipo convenio, unidad ejecutoria, contacto institución e institución a un convenio del sistema. - Controlar que la subida se archivos cumpla con las especificaciones de extensión de archivo y tamaño de archivo PDF correctamente en el formulario de edición caso contrario no se registra el archivo, se muestra de error. - Se listaron todos los convenios, tipo convenios, contactos e institución del sistema. - Se eliminaron y modificaron a los convenios, tipo convenios, contactos e institución. - Se mostró un mensaje de confirmación una vez que se guardó un registro correctamente. 	Aprobado
----------------------	---	----------

Fuente: Elaboración Propia

CAPÍTULO IV

CALIDAD, COSTO Y

SEGURIDAD DEL SOFTWARE

4.1 METODOLOGÍA WEBQEM

Esta metodología de evaluación de calidad de sitios web, propone un enfoque sistemático, disciplinado y cuantitativo que se adecue a la evaluación, comparación y análisis de calidad de sistemas web. WebQEM toma las características del modelo de calidad ISO 9126, las cuales son: usabilidad, funcionalidad, confiabilidad, eficiencia, portabilidad y mantenibilidad (Olsina, 1999).

- **Usabilidad:** Se define como un conjunto de atributos que otorgan el esfuerzo necesario para su uso, y en la evaluación individual de dicho uso, mediante un conjunto de usuarios declarados implícitos.
- **Funcionalidad:** Se define como un conjunto de atributos que otorgan la existencia de un conjunto de funciones y sus propiedades específicas. Las funciones son aquellas que satisfacen conjuntos de usuarios declarados implícitos.
- **Confiabilidad:** Se define como un conjunto de atributos de habilidad del software para mantener condiciones de establecer su propio nivel de desempeño por un periodo determinado.
- **Eficiencia:** Se define como un conjunto de atributos que otorgan la relación entre el nivel de rendimiento del software y la cantidad de recursos usados por el usuario, bajo las condiciones establecidas.
- **Mantenibilidad:** Se define como un conjunto de atributos que otorgan el esfuerzo necesario para hacer modificaciones específicas.
- **Portabilidad:** Se define como un conjunto de atributos que otorgan la habilidad de software para ser transferido de un entorno a otro.

Para la evaluación de calidad utilizare solo cuatro de las seis características prescritas en el estándar (**usabilidad, funcionalidad, confiabilidad y eficiencia**)

- **Tipo de criterio elemental**

La preferencia elemental se interpreta como una variable lógica continua. El valor 0 denota que X_i no satisface los requisitos, y el valor 1 denota una satisfacción perfecta de requisitos. En consecuencia, todas las preferencias son frecuentemente interpretadas como un porcentaje de satisfacción.

Cada característica obtiene un valor 0 % a 100% que nos indica el nivel de satisfacción de la misma, este se calcula promediando el valor de sus atributos.

A continuación, veremos los distintos tipos del criterio elemental del sistema:

- **Criterio de variable normalizada (CVN):**

$$\text{CVN: } CE = (X/Y) * 100$$

Donde con $X = \sum$ Puntaje obtenido; $Y = \sum$ Puntaje Máximo.

- **Criterio Binario (CB):**

CB: $CE = 0$ si No Existe; $CE = 1$ si existe.

- **Criterio de Preferencia Directa (CPD):**

CPD: Sujeto a la objetividad del observador.

- **Criterio de Multiniveles (CMN):**

CMN: $CE = 0 \approx 0$ Ausente; $CE = 1 \approx 60$ Presencia Parcial; $CE = 2 \approx 100$ Presente

4.1.1 Usabilidad

Se toma en cuenta la capacidad del software para ser comprendido, utilizado y atractivo para el usuario el cual se utiliza en determinadas condiciones.

A continuación, se ha procedido a la evaluación de la usabilidad:

Tabla 4.1: Resultados de Preferencia elemental – Usabilidad

Código	Característica	Criterio Elemental	Preferencia
1.1	Comprendibilidad global del sistema	CVN	100,00
1.1.1	Esquema de organización global	CVN	100,00
1.1.1.1	Mapa del Sitio	CB	1 ≈ 100
1.1.1.2	Tabla de contenidos	CB	1 ≈ 100
1.1.1.3	Índice Alfabético	CB	1 ≈ 100
1.2.2	Calidad en el sistema de etiquetado	CVN	100
1.2.2.1	Etiquetado textual	CPD	1 ≈ 100
1.2.2.2	Etiquetado con iconos	CPD	1 ≈ 100
1.2	Comentarios y características de ayuda	CVN	40
1.2.1	Características de calidad de la ayuda	CVN	80
1.2.1.1	Ayuda explicativo al visitante	CPD	60
1.2.1.2	Ayuda a la búsqueda	CPD	1 ≈ 100
1.2.2	Indicador de ultima actualización del sistema	CVN	0
1.2.2.1	Global	CMN	0 ≈ 0
1.3	Interfaz y características estéticas	CVN	80,42
1.3.1	Coercividad al agrupar los objetivos del control principal	CPD	1 ≈ 100
1.3.2	Permanencia y estabilidad en la presentación de los controles principales	CVN	75
1.3.2.1	Permanencia de controles directos	CPD	75
1.3.2.2	Permanencia de controles indirectos	CPD	75
1.3.2.3	Estabilidad	CPD	75
1.3.3	Aspectos de estilos	CVN	86,67
1.3.3.1	Uniformidad en el color de enlaces	CMN	2 ≈ 100
1.3.3.2	Uniformidad en el estilo global	CMN	2 ≈ 100
1.3.3.3	Guía de estilo global	CMN	1 ≈ 60
1.3.4	Preferencias estéticas	CPD	1 ≈ 60
1.4	Características varias	CVN	50
1.4.1	Soporte de idioma extranjero	CB	0
1.4.2	Características de descarga	CB	1 ≈ 100

Fuente: Elaboración propia

La usabilidad de la aplicación evaluada estará determinada por el promedio de las características anteriormente mencionadas como muestra la siguiente tabla:

Tabla 4.2: Evaluación total de Usabilidad

Nro.	Características	Resultado
1	Comprensibilidad global del sistema	100
2	Comentarios y características de ayuda	40
3	Interfaz y características estéticas	80,42
4	Características varias	50
Evaluación total de usabilidad		67,61

Fuente: Elaboración propia

4.1.2 Funcionalidad

Es una métrica orientada a la función del sistema y al proceso por el cual se desarrolla y se centra en la funcionalidad o utilidad del programa. A continuación, se tiene la evaluación elemental para la funcionalidad.

Tabla 4.3: Resultados de Preferencia elemental – Funcionabilidad

Código	Características	Criterio Elemental	Preferencia
2.1	Aspectos de búsqueda y recuperación	CVN	80
2.1.1	Mecanismo de búsqueda en el sitio web	CVN	100
2.1.1.1	Búsqueda específica	CB	1 ≈ 100
2.1.1.2	Búsqueda global	CMN	1 ≈ 100
2.1.2	Mecanismos de recuperación	CVN	60
2.1.2.1	Nivel de retroalimentación en la recuperación	CMN	1 ≈ 60
2.2	Aspectos de navegación y exploración	CVN	65
2.2.1	Navegabilidad	CVN	30
2.2.1.1	Orientación	CVN	60
2.2.1.1.1	Indicador del camino	CB	1 ≈ 60
2.2.1.1.2	Etiqueta de la posición actual	CPD	1 ≈ 60
2.2.1.2	Promedio de enlaces por pagina	CMN	0 ≈ 0

2.2.2	Objetos de control de navegacion	CVN	100
2.2.2.1	Permanencia y estabilidad en la presentación de los controles contextuales	CVN	100
2.2.2.1.1	Permanencia de los controles contextuales	CMN	1 ≈ 100
2.2.2.1.2	Estabilidad de los controles contextuales	CMN	1 ≈ 100
2.2.2.2	Nivel de desplazamiento	CVN	100
2.2.2.2.1	Desplazamiento vertical	CB	1 ≈ 100
2.2.2.2.2	Desplazamiento horizontal	CB	1 ≈ 100
2.3	Aspectos del dominio orientados al personal técnico	CVN	90
2.3.1	Relevancia de contenido	CVN	90
2.3.1.1	Información de la gaceta universitaria	CVN	100
2.3.1.1.1	Historia y situación actual de resoluciones	CB	1 ≈ 100
2.3.1.1.2	Historia y situación actual de los convenios	CB	2 ≈ 100
2.3.1.2	Información de solicitud de resoluciones	CVN	80
2.3.1.2.1	Información de registros	CMN	2 ≈ 100
2.3.1.2.2	Información de bajas	CMN	1 ≈ 60
2.3.1.3	Información de solicitud de convenios	CVN	80
2.3.1.3.1	Información de registros	CMN	2 ≈ 100
2.3.1.3.2	Información de bajas	CMN	1 ≈ 60
2.3.1.4	Información de publicaciones On-Line	CPD	1 ≈ 100

Fuente: Elaboración propia

La funcionalidad de la aplicación evaluada estará determinada por el promedio de las características anteriormente mencionadas como muestra la siguiente tabla:

Tabla 4.4: Evaluación total de funcionalidad

Nro.	Características	Resultado
1	Aspectos de búsqueda y recuperación	80
2	Aspectos de navegación y exploración	65
3	Aspectos del dominio orientados al personal técnico	90
Evaluación total de Funcionalidad		78,33

Fuente: Elaboración propia

4.1.3 Confiabilidad

La medición de esta característica está definida por el complemento de los casos de deficiencia encontradas en la aplicación.

Tabla 4.5: Resultados de preferencia elemental – Confiabilidad

Código	Características	Criterio Elemental	Preferencia
3.1	No deficiencia	CVN	90,00
3.1.1	Errores de enlaces		100,00
3.1.1.1	Enlaces rotos	CMN	2 ≈ 100
3.1.1.2	Enlaces inválidos	CMN	1 ≈ 100
3.1.1.3	Enlaces no implementados	CMN	1 ≈ 100
3.1.2	Errores o deficiencias varias	CVN	80
3.1.2.1	Deficiencia o cualidades ausentes debido a diferentes navegadores (browsers)	CMN	1 ≈ 60
3.1.2.2	Deficiencia o resultados inesperados independientes de browsers (p.ej. errores de búsqueda imprevistos, deficiencia con marcos (frames), etc.)	CMN	1 ≈ 60
3.1.2.3	Nodos destinos (inesperadamente) en construcción	CMN	2 ≈ 100
3.1.2.4	Nodos web muertos (sin enlaces de retornos)	CMN	2 ≈ 100

Fuente: Elaboración propia

La evaluación total de la confiabilidad es de 90%.

4.1.4 Eficiencia

Es una característica de calidad del sistema de alto nivel que puede medirse mediante cálculo a partir de métricas directas e indirectas.

Tabla 4.6: Resultados de preferencia elemental – Eficiencia

Código	Características	Criterio Elemental	Preferencia
4.1	Rendimiento	CVN	90
4.1.1	Páginas de acceso rápido	CDP	90

4.2	Accesibilidad	CVN	95
4.2.1	Accesibilidad de la información	CVN	90
4.2.1.1	Soporte de versión sólo texto	CB	1 ≈ 100
4.2.1.2	Legibilidad al desactivar la propiedad imagen del browser	CVN	80
4.2.1.2.1	Imagen con título	CB	1 ≈ 100
4.2.1.2.2	Legibilidad global	CMN	1 ≈ 60
4.2.2	Accesibilidad de ventanas	CMN	2 ≈ 100

Fuente: Elaboración propia

La eficiencia de la aplicación evaluada estará determinada por el promedio de las características anteriormente mencionadas, como muestra la siguiente tabla:

Tabla 4.7: Evaluación total de Eficiencia

Nro.	Características	Resultado
1	Rendimiento	90
2	Accesibilidad	95
Evaluación total de Eficiencia		92,50

Fuente: Elaboración propia

La calidad total de la aplicación web estará determinada por el promedio de las características de usabilidad, funcionalidad, confiabilidad y eficiencia como se muestra en la siguiente tabla:

Tabla 4.8: Resultados de la calidad total

Nro.	Nombre	Preferencia
1	Usabilidad	67,61
2	Funcionalidad	78,33
3	Confiabilidad	90,00
4	Eficiencia	92,50
Evaluación de la Calidad Total		82,11

Fuente: Elaboración propia

- ✓ **Análisis de resultados:** A continuación, se hace la comparación de las preferencias de calidad elementales, parciales y globales además la

justificación del resultado. Para un mejor análisis se presenta la siguiente tabla con la escala de medición de aceptabilidad:

Tabla 4.9: Escala de medición de aceptabilidad

Criterio	Escalas en %
Insatisfactorio	0 – 40
Aceptable marginal	40 – 60
Satisfactorio	60 – 100

Fuente: Elaboración propia

Por lo tanto, acorde a la valoración de calidad del sistema, aplicando la metodología WebQEM, el valor de la calidad del sistema es de **82,11%** como se observa en la tabla 4.9, esto indica que se tiene un nivel de aceptabilidad satisfactorio, ubicándose en los márgenes de aceptabilidad de (60% a 100%).

4.2 ESTIMACIÓN DE COSTOS

El método COCOMO es aquel que permite realizar estimaciones en función del tamaño del software, y de un conjunto de factores de costo y beneficio. Los factores de costo describen aspectos relacionados con la naturaleza del producto, hardware utilizado, personal involucrado, y características propias del proyecto. COCOMO posee tres modelos denominados: modelo básico, **modelo intermedio** y modelo avanzado o detallado. Cada uno de ellos orientados a sectores específicos del mercado de desarrollo de software y las distintas etapas del desarrollo de software.

Por otro lado, también se define tres modos de desarrollo o tipo de proyectos: Orgánico, semi-acoplado y empotrado. Estos métodos de estimación de costos se engloban en el grupo de modelos algorítmicos que tratan de establecer una relación matemática la cual permite estimar el esfuerzo y tiempo requerido para desarrollar un producto.

4.2.1 Modelo de desarrollo intermedio

En este modelo se introducen 15 atributos de costo para tener en cuenta el entorno

de trabajo. Estos atributos se utilizan para ajustar el costo nominal del proyecto al entorno real, incrementando la precisión de la estimación.

Para nuestro caso el modelo intermedio será el que usare, dado que realiza las estimaciones con bastante precisión.

4.2.2 Estimación de costos

Las ecuaciones que se usaran para la estimación de costos son las siguientes:

$$\text{Esfuerzo (E)} \quad E = a * (KLDC)^b * ME$$

$$\text{Tiempo (T)} \quad T = c * (E)^d$$

$$\text{Personal (P)} \quad P = E/T$$

Donde:

E: Es el esfuerzo requerido por el proyecto, en persona/mes.

T: Es el tiempo requerido por el proyecto, en meses.

P: Es el número de personas requerido por el proyecto, en personas.

a, b, c y d: Son constantes con valores definidos en una tabla, según cada sub modelo.

KLDC: Es la cantidad de líneas de código, en miles.

ME: Es el multiplicador de esfuerzo que depende de 15 atributos

a) Cálculo de Esfuerzo

Para calcular el esfuerzo, necesitamos hallar la variable KLDC(Kilo-linias de código). Este proyecto se implementa un aproximado de 3500 líneas de código (LDC) en el lenguaje PHP.

Aplicando la formula se tiene:

$$KLDC = \frac{LCD}{1000} = \frac{4000}{1000}$$

$$KLDC = 4,000$$

Este modo de desarrollo es de tipo orgánico porque es el más apropiado en este caso porque el número de líneas de código no supera los 50 KLDC y además el proyecto no es muy complejo, entonces por consiguiente los coeficientes que se usarán serán los siguientes valores detallados en la siguiente tabla:

Tabla 4.10: Modo de desarrollo, COCOMO Intermedio

Modo de desarrollo	a	b	c	d
Orgánico	3,2	1,05	2,5	0,38
Semiacoplado	3	1,12	2,5	0,35
Empotrado	2,8	1,2	2,5	0,32

Fuente: (Boehm, 1983)

Por otro lado, para el valor de la variable ME, la cual se obtiene mediante la multiplicación de los valores en los diferentes 15 conductores de costos. Para el cálculo de los atributos SE ME utilizara la tabla 4.11 y se define las variables a usar pintando de un color, porque son las que se multiplicarán.

Tabla 4.11: Cálculo de los atributos Multiplicador de Esfuerzo

Multiplicadores de esfuerzo (ME)		Muy Bajo	Bajo	Nominal	Alto	Muy Alto	Extra Alto
Atributos del Producto							
1	Garantía de funcionamiento requerida al software	0,75	0,88	1	1,15	1,4	-
2	Tamaño de la base de datos	-	0,94	1	1,08	1,16	-
3	Complejidad del producto	0,7	0,85	1	1,15	1,3	1,65
Atributos del Hardware							
4	Restricción de tiempo de ejecución	-	-	1	1,11	1,3	1,66
5	Restricción del almacenamiento principal			1	1,06	1,21	1,56
6	Volatilidad de la máquina virtual		0,87	1	1,15	1,3	-
7	Tiempo de respuesta del ordenador		0,87	1	1,07	1,15	-
Atributos del Personal							
8	Capacidad del analista	1,46	1,19	1	0,86	0,71	-
9	Experiencia en la aplicación	1,29	1,13	1	0,91	0,82	-
10	Capacidad del programador	1,42	1,17	1	0,86	0,7	-
11	Experiencia en máquina virtual	1,21	1,1	1	0,9	-	-
12	Experiencia en lenguaje de programación	1,14	1,07	1	0,95	-	-
Atributos del proyecto							
13	Prácticas de programación modernas	1,24	1,1	1	0,91	0,83	-
14	Utilización de herramientas software	1,24	1,1	1	0,91	0,82	-
15	Plan de desarrollo requerido	1,23	1,08	1	1,04	1,1	-
TOTAL		0,69					

Fuente: (Boehm, 1983)

$$\begin{aligned}
ME &= 1,15 * 1,00 * 0,85 * 1,11 * 1,00 * 1,15 * 1,07 * 1,00 * 0,82 * 0,86 * 1,00 * 0,95 * \\
&0,82 * 0,91 * 1,04 \\
ME &= 0,69
\end{aligned}$$

Por lo tanto, el multiplicador de esfuerzo evaluados en la tabla 4.11 es: 0,69

Aplicando y reemplazando los valores de la tabla de modo de desarrollo y el multiplicador de esfuerzo a la fórmula de esfuerzo, se tiene:

$$\begin{aligned}
E &= a(KLDC)^b * ME(Persona/Mes) \\
E &= 3,2 * (4,000)^{1,05} * 0,69(\text{Persona/Mes}) \\
E &= 9,46(\text{Persona/Mes})
\end{aligned}$$

b) Cálculo del tiempo de desarrollo

$$\begin{aligned}
T &= c * E^d (\text{meses}) \\
T &= 2,5 * 9,46^{0,38} (\text{meses}) \\
T &= 5,9 (\text{meses})
\end{aligned}$$

c) Cálculo del personal

$$\begin{aligned}
P &= E/T (\text{Personas}) \\
P &= 9,46/5,9 (\text{Personas}) \\
P &= 1,6 \approx 2(\text{Personas})
\end{aligned}$$

d) Cálculo de productividad

$$\begin{aligned}
PR &= LDC/E \\
PR &= 4000/9,46 LCD/\text{Personas mes} \\
PR &= 422,83 LCD/\text{Personas mes}
\end{aligned}$$

e) Costo Total del proyecto

En el supuesto que se cuente con un promedio de \$us. 250.- mes por persona entonces:

$$CT = \text{Sueldo Mes} * P * T$$

$$CT = 250 * 2 * 5,9$$

$$CT = 2950.- \text{ $us. } \approx 20,496.- \text{ Bs.}$$

Por lo tanto, con los resultados obtenidos diríamos que el “Sistema de Gestión para la Gaceta Universitaria On-Line utilizado herramientas Open Source” debería desarrollarse aproximadamente en **5,9 meses por 2 personas** y el costo **total del desarrollo del software es de \$us. 2950.- equivalentes en bolivianos a Bs. 20.496.-**

4.3 SEGURIDAD

La seguridad de la información se puede definir como conjunto de medidas técnicas, organizativas y legales que permiten a la organización asegurar la confidencialidad, integridad y disponibilidad del sistema.

La seguridad informática es importante para cualquier institución pública o privada que maneje sistemas de información, ya que en la actualidad todo se maneja en el ámbito computarizado, por esto se han creado tantos métodos para proteger todo tipo de información ya sea en el desarrollo o en la programación de estas. Gracias a ello, todos los sistemas corren menos riesgos al momento de ser operados, lo que se busca es estar lo más protegido para evitar que nadie sea afectado al momento de operar el sistema.

Para la seguridad del sistema se considerarán las siguientes precauciones:

4.3.1. Seguridad a nivel de Base de datos

La base de datos es lo más importante que contiene un sistema, sin ella no tendría ninguna funcionalidad, es por eso que la Unidad de Sistemas de Información y Estadística (SIE) hace copias de seguridad a los datos cada cierto tiempo y estas las guarda en diferentes formas físicas como lógicas, cada cierto tiempo se guardan en pendrives y otras técnicas de resguardo, por norma estas copias deben seguir las normas de seguridad informática, así se podrá garantizar la seguridad de los datos que posee la institución no solo del sistema actual sino de toda su información.

Seguridad en los datos: Los datos se alojan en servidores ubicados en Data Centers con toda la infraestructura necesaria para asegurar la protección de datos y el funcionamiento constante de las aplicaciones.

También el acceso a la base de datos debe ser protegido, para evitar perdida de información para este caso se toma en cuenta los siguientes aspectos:

- El acceso a la información de la BD, para los usuarios, será a través de la aplicación, por medio de la autenticación de los mismos.
- Manejo de permisos y tipo de usuarios o rol en el gestor de base de datos.
- Hash de Laravel proporciona un hashing seguro a través de la función Bcrypt para almacenar las contraseñas de los usuarios. Esto a través del controlador AuthController que se incluye con las aplicaciones de Laravel.

4.3.2. Seguridad a nivel de aplicación

Los niveles de seguridad de la aplicación determinan si un usuario tiene la autoridad para realizar dicha operación, se cuenta con el módulo de autenticación que administra el acceso al sistema, así como los permisos asignados según sus permisos y tipo de usuario o roles, por tanto, se puede mencionar que:

- Laravel provee una estructura muy sólida para crear aplicaciones web, una herramienta que incorpora de forma predeterminada es el sistema de autenticación basado en el uso de tokens.
- Un token es un valor único (clave) que se asigna a cada usuario que tiene acceso al sistema sus función básica es: Encriptar el password del usuario y guardarlo en la base de datos.
- La utilización de Tokens en todos los formularios que proporcionan datos de entrada al sistema, eliminando de esta manera los ataques CSRF (Cross Site Request Forgery).
- Los filtros de acceso, como Laravel implementa Middleware, que tienen una utilidad similar a un filtro de acceso que nos permiten proteger rutas y acciones de acceso no autorizado. El Middleware, como el nombre lo indica, se sitúa en el medio entre la petición del usuario (Request) y las acciones del controlador que arman y envían la respuesta (Response).
- Cierre de sesión automático, en caso de no tener actividad en una determinada cantidad de tiempo, termina la sesión de usuario.

CAPÍTULO V

CONCLUSIONES Y

RECOMENDACIONES

CONCLUSIONES Y RECOMENDACIONES

El objetivo general del Proyecto de titulación fue el desarrollar el Sistema de Gestión para la Gaceta Universitaria On-Line utilizando herramientas Open Source, para administrar los recursos de la Unidad de Secretaría General de la Universidad Pública de El Alto, y luego de concluido el trabajo, se puede observar que el objetivo se ha cumplido, brindando un nuevo Proyecto de Grado.

5.1 CONCLUSIONES

Luego de plantear el problema, diseñar, desarrollar y probar el sistema web para la Unidad de Secretaría General de la Universidad Pública de El Alto, aplicando todas las metodologías de análisis y diseño de software, se logró cumplir el objetivo general planteado a un inicio, y el desarrollo del sistema fue exitoso.

Tomando en cuenta los objetivos previamente planteados se llegó a las siguientes conclusiones:

- Se desarrolló el módulo de administrador principal para gestión de: usuarios, tipo de usuario, categorías de menú y opciones de menú del sistema que serán manipuladas por la Unidad de Sistemas de Información y Estadística.
- Se desarrolló el módulo de Encargado de Archivo para gestión de: resoluciones, tipo resolución, convenios, tipo convenio, contacto convenio, institución, publicaciones y entre otros que son enteramente internas, que estará a cargo del encargado de resguardo de documentación de la unidad de Secretaría General.
- Se desarrolló los módulos para Secretario General, Secretaría de Oficina y Auxiliar de Oficina de acuerdo al rol o tipo de usuario con vistas de gaceta universitaria de: resoluciones, convenios y publicaciones, que estará a cargo del personal de la unidad de Secretaría General.
- El sistema nos permite la automatización del proceso de búsqueda y revisión de la información de los convenios y resoluciones, por lo tanto, el acceso a ellos es más rápido.

- El sistema permite una mejor organización de la información de los convenios, resoluciones y con ello se pueden tomar acciones de publicación al público en general.
- Se logró desarrollar un sistema con interfaz atractivo y amigable para los usuarios del sistema.

5.2 RECOMENDACIONES

A la culminación del presente proyecto se efectúan las siguientes recomendaciones:

- Utilizar el Framework PHP Lavarel para el desarrollo de aplicaciones web de pequeño y mediano tamaño, porque además de ser libre está basado completamente en la arquitectura MVC (Modelo Vista Controlador).
- Para el uso del sistema es recomendable cambiar las contraseñas de administradores mensualmente, por razones de seguridad del sistema.
- Que, los usuarios mantengas su contraseña y nombre de usuario como privacidad.
- En caso de que se requiera la ampliación y creación de nuevos módulos no contemplados en el presente proyecto, se recomienda al a unidad de Sistemas de Información y Estadística, que pueda ampliar el sistema puesto que presenta elementos reutilizables que podrían ser utilizados en nuevos módulos.
- Realizar copias de seguridad de la base de datos si es posible diariamente o en su defecto semanalmente, ya que la información que contiene es muy importante para el buen funcionamiento del sistema, ya que el sistema no genera copias de seguridad automáticamente.
- Para la publicación de documentos institucionales en formato PDF al público en general, se recomienda que sean autorizadas por el inmediato superior para la publicación respectiva.

BIBLIOGRAFÍA

FUENTES CONSULTADAS

Bibliográficos

Boehm,B., Abst,C., Brown, .W., Chulani,S., Clark,B,K., Horowitz,K., Madachy, R., Reifer, D., & Steele,B.(2000). Software Cost Estimation With COCOMO II. Presentice Hall.

Gallego, S. A. J. (2016), Laravel 5: The PHP Framework For Web Artisans.

Gallego, S. A. J. (2015), BOOTSTRAP 3 Introducción al diseño Responsive.

Garcia, R., (2012). Calidad en el Desarrollo y Mantenimiento del Software.

Fernandez, G. (2002). Introducción a Extreme Programming.

Kendall,K.E & Kendall, Julie E.(2011) "Analisis y diseño de sistemas" 8va edición, Mexico: Pearson Education.

Minguez, D., & Garcia, E., (2011). Metodologías para el Desarrollo de Aplicaciones Web: UWE.

Minguez D, & Garcia E., (2011). Metodologías para el Desarrollo de Aplicaciones Web: UWE.

Olsina,L(2002). Tutorial: Metricas, Criterios y Estrategias para Evaluar Calidad Web. GIDIS, Facultad de Ingenieria, UNLPam, Argentina.

Pressman, R. S. (2010). Ingeniería de software - un enfoque práctico, 7ma. Edición. Mexico:Mc Graw Hill.

Sanz, D. M. (2014). Metodologías para el Desarrollo de Aplicaciones Web.

Webgrafía

Bootstrap, Documentación oficial de bootstrap, 2020. [En línea]. Disponible en: <https://getbootstrap.com/docs/4.4/getting-started/introduction> [Accedido, marzo 2020].

Cocomo, [En línea]. Disponible en: <https://es.wikipedia.org/wiki/COCOMO> [Accedido, mayo 2020].

Laravel, Documentación en español, [En línea]. Disponible en: <https://laravel.montogeek.com/5.1> [Accedido, marzo 2020].

Que es MariaDB, [En línea]. Disponible en: <https://www.nerion.es/soporte/que-es-mariadb-y-mejoras-sobre-mysql> [Accedido, marzo 2020].

Universidad Carlos III de Madrid, «Estudio de UWE (UML - based Web Engineering)». [En línea]. Disponible en: <https://es.scribd.com/doc/44936310/Estudio-de-UWE-Metodologia-de-Desarrollo-Web> [Accedido, mayo 2020].

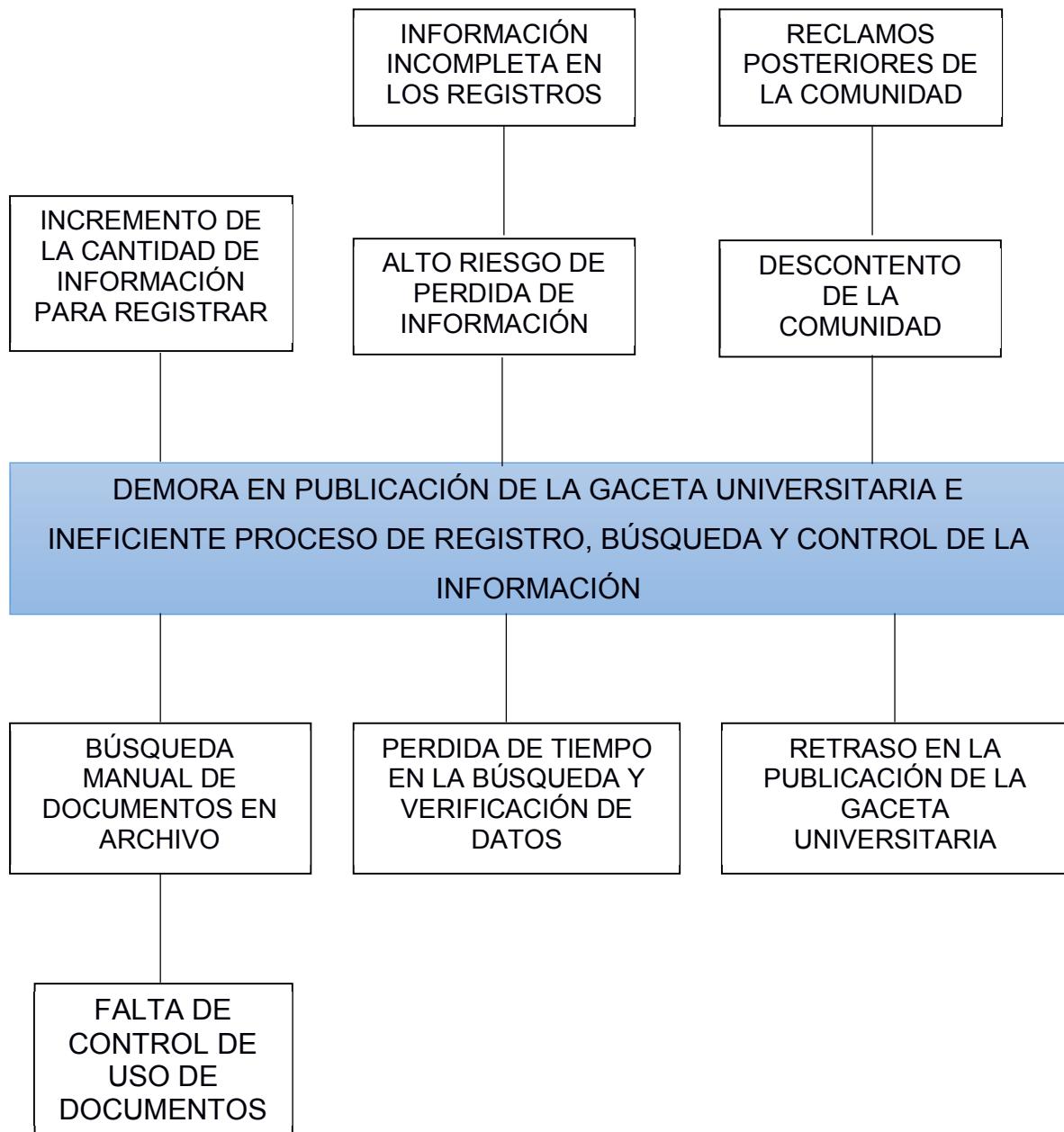
ANEXOS

ANEXO A

ÁRBOL DE PROBLEMAS

ANEXO A

ÁRBOL DE PROBLEMAS

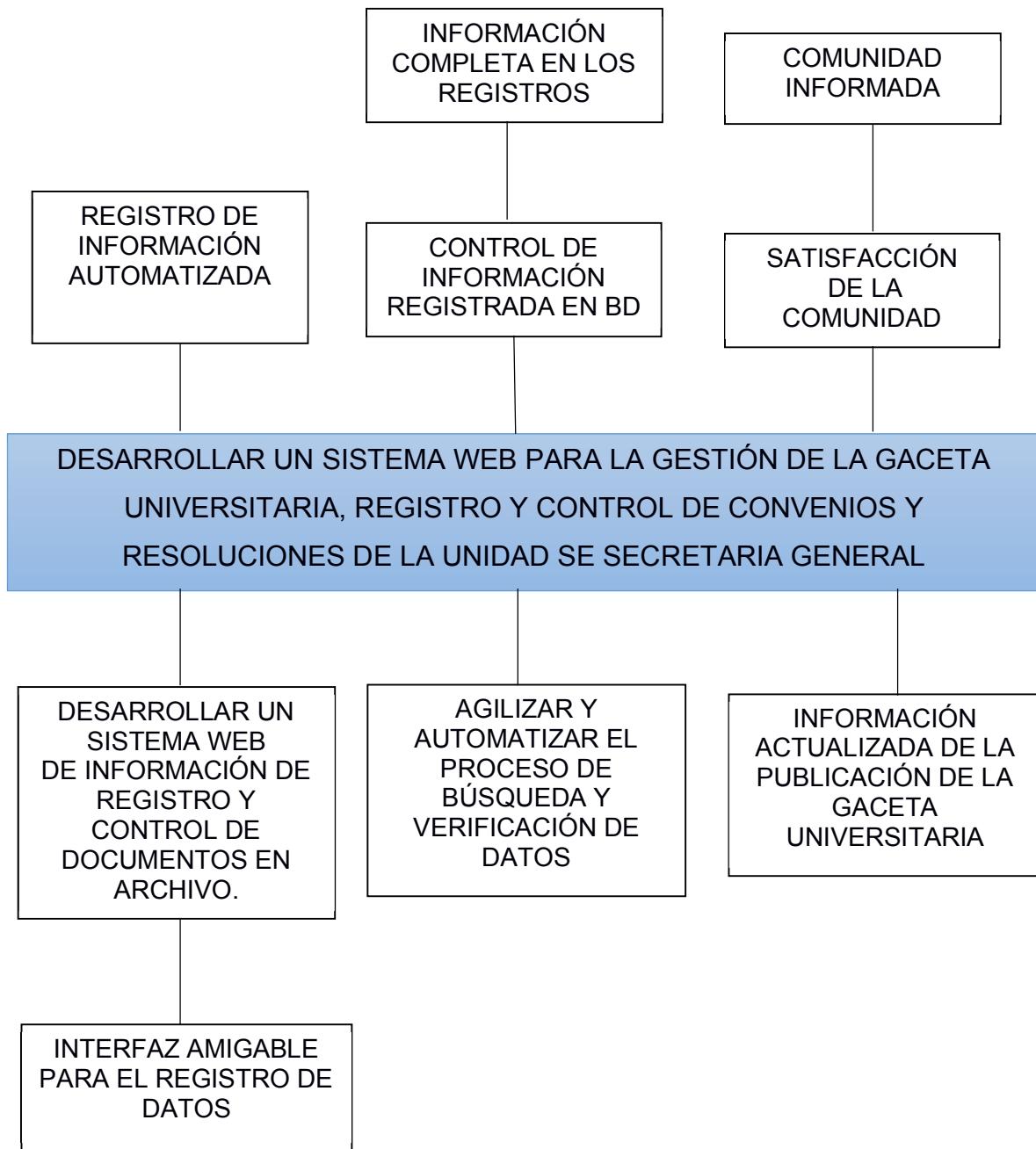


Fuente: Elaboración propia

ANEXO B

ÁRBOL DE OBJETIVOS

ANEXO B
ÁRBOL DE OBJETIVOS



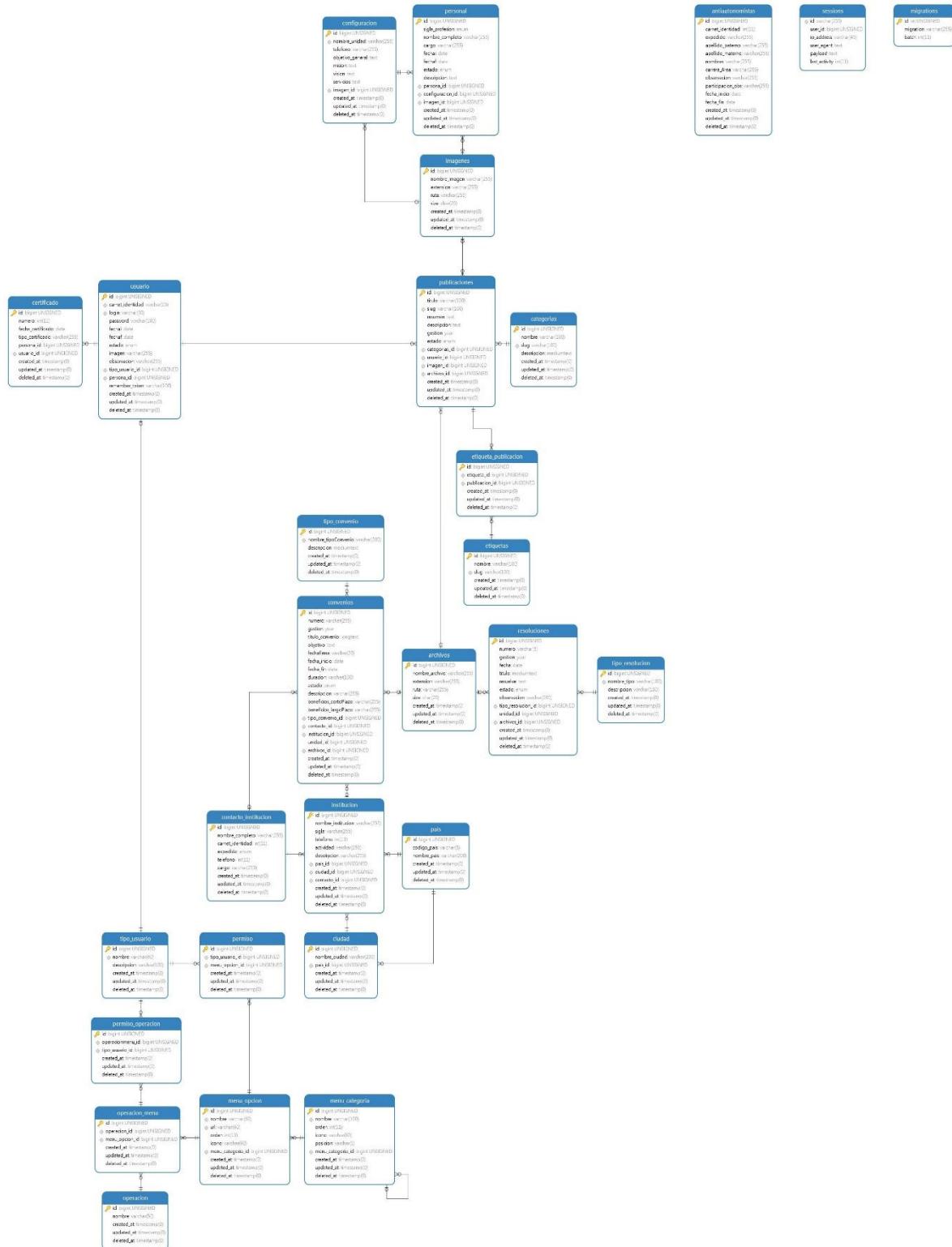
Fuente: Elaboración propia

ANEXO C

MODELO RELACIONAL DE

LA BASE DE DATOS

ANEXO C: Modelo relacional de la base de datos



Fuente: Elaboración propia

ANEXO D

REQUERIMIENTO DE

HARDWARE Y SOFTWARE

D.1 Requerimientos Software: La aplicación funciona con el framework laravel 7.x, por ello son necesarios los siguientes requerimientos del lado del servidor de aplicación apache:

- PHP >= 7.2.5
- BCMath PHP Extension
- Ctype PHP Extension
- Fileinfo PHP extension
- JSON PHP Extension
- Mbstring PHP Extension
- OpenSSL PHP Extension
- PDO PHP Extension
- Tokenizer PHP Extension
- XML PHP Extension
- composer (opcional)
- MYSQL >=8 o MariaDB 10
- Sistema operativo (Windows o Linux)
- Navegador Firefox, Chrome, Opera (Actuales).

D.2 Requerimientos hardware: Los requerimientos de hardware variaran de acuerdo a la cantidad de usuarios y transacciones que se realizan en la aplicación, inicialmente se estima el uso de un servidor con las siguientes características:

- Procesador Intel core i5 o superior
- Memoria ram 2 gb o superior
- 500 gb de disco duro
- Tarjeta de video 1gb

ANEXO E

MANUAL DE USUARIO

UNIVERSIDAD PÚBLICA DE EL ALTO



MANUAL DE USUARIO

SISTEMA DE GESTIÓN PARA LA GACETA UNIVERSITARIA
ON-LINE UTILIZANDO HERRAMIENTAS OPEN SOURCE

HERNAN CATARI YUJRA

LA PAZ - BOLIVIA

1. INTRODUCCIÓN

El manual de usuario busca brindar una serie de pasos para manipular el sitio web Para la Unidad de Secretaría General de la Universidad Pública de El Alto, el cual es un proyecto dirigido a los usuarios finales. Con esta solución web se pretende administrar la gaceta universitaria y el documento sirve como fuente de capacitación para el manejo de los diferentes módulos que componen el sistema.

2. INICIO DE SESIÓN

Para ingresar a la aplicación debemos ingresar a través de un navegador web (Chrome, Firefox u Opera) a la dirección asignada al servidor, en esta dirección se abrirá inmediatamente la pantalla inicial del sistema.

Figura 1: Pantalla de Inicio



En esta interfaz se visualizará un encabezado con el nombre de la aplicación, las opciones de “Iniciar Sesión o login”, presionamos clic en la opción “Iniciar Sesión” la cual nos direccionará a la interfaz de inicio de sesión.

Figura 2: Modal Inicio se Sesión



En esta interfaz debemos de ingresar nuestro Usuario y contraseña en los campos correspondientes, terminando de completar los campos correspondientes presionamos clic en la opción “Iniciar Sesión”.

Iniciada sesión se visualizará la interfaz del módulo del menú principal de la aplicación de acuerdo al permiso, tipo de usuario y/o rol.

3. CERRAR SESIÓN

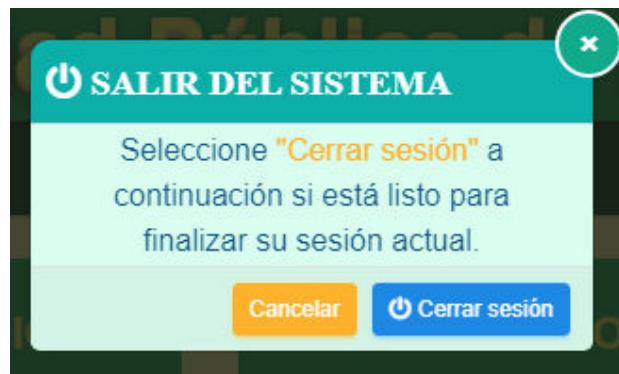
Para cerrar sesión nos ubicamos en el menú superior o inferior del menú principal de la aplicación una vez iniciada sesión.

Figura 3: Pantalla de inicio de sesión según modulo



En esta interfaz en la parte superior derecha presionamos clic sobre nuestro nombre de usuario lo cual desplegará un menú que muestra la respectiva opción de cierre de sesión o salir.

Figura 4: Modal de cerrar sesión



Presionamos clic sobre la opción “Cerrar sesión” y de esta forma el sistema realizara el respectivo cierre de sesión de usuario.

4. MÓDULO ADMINISTRADOR PRINCIPAL

Una vez iniciada la sesión como administrador principal, se abrirá inmediatamente la pantalla inicial de la aplicación.

Figura 5: Pantalla módulo administrador principal



En esta interfaz se visualizará un encabezado con el nombre de la aplicación, las opciones de menú principal donde se puede apreciar que solo tiene acceso a: Menú, Categoría, Menú Opción, Tipo Usuario, Usuarios y Usuario Historial, si presionamos en cualquiera de las opciones del menú principal nos dirigirá a la interfaz a dicho menú.

4.1 Interfaz Menú Categoría

En esta interfaz se visualizará los registros creados del menú categoría con los botones de crear, editar, eliminar y buscar registros.

Figura 6: Listado de menú categoría

GESTIÓN MENU CATEGORÍA N°1						GESTIÓN MENU CATEGORÍA N°2		Nuevo Registro	
Filas a Mostrar:		10	Buscar por Nombre			Buscar Registro			
#	NOMBRES	ORDEN	CATEGORIA	POSICION	ACCIONES	Total: 0	1		
1	GESTIÓN ADMIN	1	--	Vertical					
2	GESTIÓN RESOLUCIÓN	2	--	Vertical					
3	GESTIÓN CONVENIO	3	--	Vertical					
4	GESTIÓN OBSERVADOS	4	--	Vertical					
5	GESTIÓN PUBLICACIÓN	5	--	Vertical					

4.1.1 Crear nuevo registro de menú categoría

Al hacer clic en la opción de crear nuevo registro se abrirá el modal de creación o registrar - menú categoría.

Figura 7: Modal Registrar – menú categoría

The modal window has a teal header bar with the title "REGISTRAR - MENU CATEGORÍA". Inside, there are four input fields: "Categoria:" (dropdown menu "Seleccione una categoría"), "Nombre: *" (text input "Ingresar nombre" with a red border), "Orden: *" (text input "Ingresar orden" with a red border), and "Posicion: *" (dropdown menu "Vertical"). Below these is an "Icono: *" field with the value "mdi-playlist-play". A pink error message box at the bottom left says "Por favor corrija los siguientes errores:" with two items: "El campo nombre es obligatorio." and "El campo orden es obligatorio.". At the bottom right are two buttons: a blue "Registrar" button with a checkmark icon and an orange "Cerrar" button with an exclamation mark icon.

En esta interfaz debemos de llenar los campos correspondientes, según los campos requeridos.

4.2 Interfaz Menú Opción

En esta interfaz se visualizará los registros creados del menú opción con los botones de crear, editar, eliminar y buscar registros.

Figura 8: Listado de menú opción

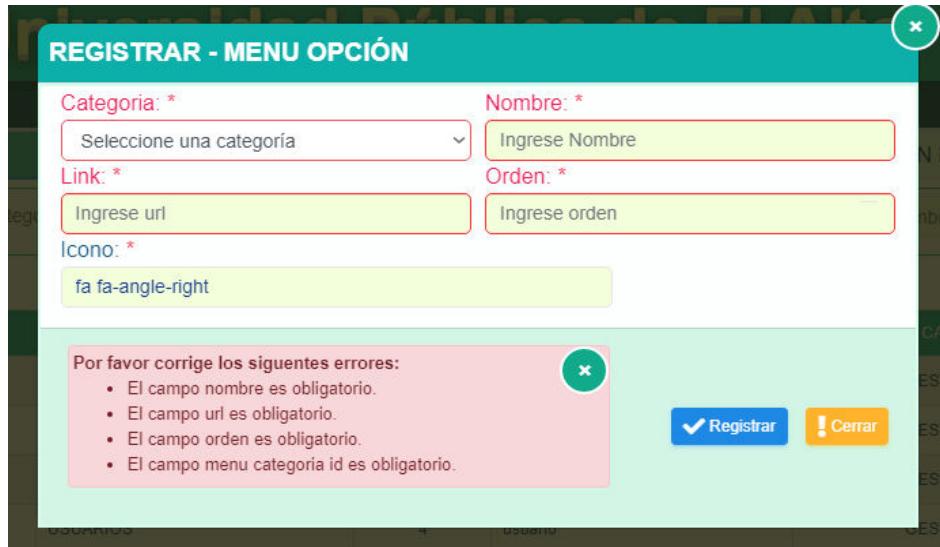
The table has a green header bar with tabs "GESTIÓN MENU OPCIÓN N°1" and "GESTIÓN MENU OPCIÓN N°2" and a "Nuevo Registro" button. Below the header are search/filter fields: "Categoría: Todos", "Filas a Mostrar: 10", "Buscar por Nombre", and a "Buscar Registro" button. At the top right are buttons for "Total: 15", page numbers "1" and "2", and navigation icons. The main body is a grid with columns: "#", "NOMBRE", "ORDEN", "URL", "CATEGORÍA", and "ACCIONES". Each row contains a set of edit and delete icons. The data rows are:

#	NOMBRE	ORDEN	URL	CATEGORÍA	ACCIONES
1	MENU CATEGORÍA	1	categoriaopcionmenu	GESTIÓN ADMIN	
2	MENU OPCIÓN	2	opcionmenu	GESTIÓN ADMIN	
3	TIPO USUARIO	3	tipousuario	GESTIÓN ADMIN	
4	USUARIOS	4	usuario	GESTIÓN ADMIN	
5	USUARIO HISTORIAL	5	sessions	GESTIÓN ADMIN	
6	TIPO RESOLUCIÓN	1	tiporesolucion	GESTIÓN RESOLUCIÓN	
7	RESOLUCIONES	2	resoluciones	GESTIÓN RESOLUCIÓN	
8	CONTACTO	1	representantes	GESTIÓN CONVENIO	
9	INSTITUCIÓN	2	institucion	GESTIÓN CONVENIO	
10	TIPO CONVENIO	3	tipoconvenio	GESTIÓN CONVENIO	

4.2.1 Crear nuevo registro de menú opción

Al hacer clic en la opción de crear nuevo registro se abrirá el modal de creación o registrar - menú opción.

Figura 9: Modal Registrar – menú opción



En esta interfaz debemos de llenar los campos correspondientes, según los campos requeridos.

4.3 Interfaz Tipo Usuario

En esta interfaz se visualizará los registros creados del tipo Usuario con los botones de crear, editar, eliminar y buscar registros.

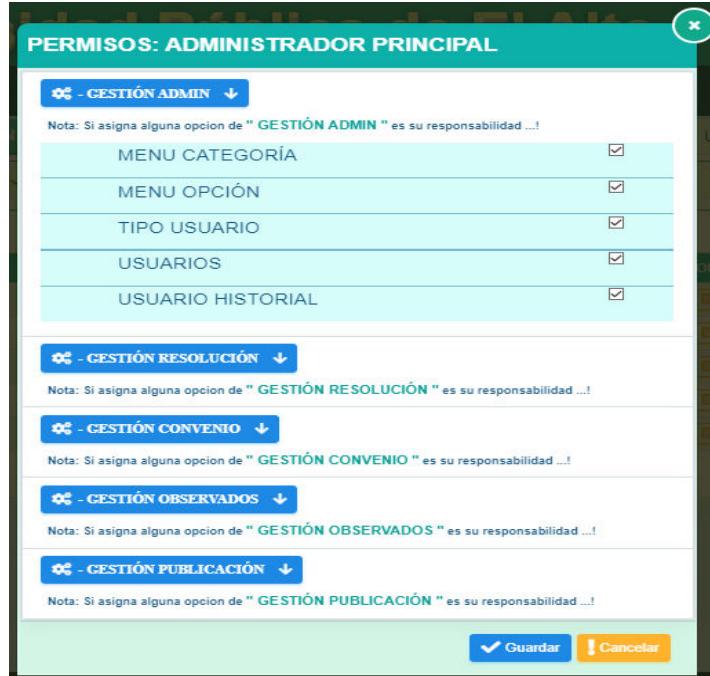
Figura 10: Listado tipo usuario

GESTIÓN TIPO USUARIO N°1		GESTIÓN TIPO USUARIO N°2			
				Nuevo Registro	
Filas a Mostrar: 10		Buscar por Nombre		Buscar Registro	
#	NOMBRES	OPCIONES	ACCIONES	Total: 3	1
1	ADMINISTRADOR PRINCIPAL	<input type="button" value="Permisos"/> <input type="button" value="Operaciones"/>	<input type="button" value="Edit"/> <input type="button" value="Delete"/>	Total: 3	1
2	AUXILIAR OFICINA	<input type="button" value="Permisos"/> <input type="button" value="Operaciones"/>	<input type="button" value="Edit"/> <input type="button" value="Delete"/>	Total: 3	1
3	ENCARGADO ARCHIVO	<input type="button" value="Permisos"/> <input type="button" value="Operaciones"/>	<input type="button" value="Edit"/> <input type="button" value="Delete"/>	Total: 3	1
4	SECRETARIA	<input type="button" value="Permisos"/> <input type="button" value="Operaciones"/>	<input type="button" value="Edit"/> <input type="button" value="Delete"/>	Total: 3	1
5	SECRETARIO GENERAL	<input type="button" value="Permisos"/> <input type="button" value="Operaciones"/>	<input type="button" value="Edit"/> <input type="button" value="Delete"/>	Total: 3	1

4.3.1 Asignar permiso a un usuario

Al hacer clic en permisos en la columna opciones de listado de tipo de usuario se abrirá un modal donde se puede asignar permisos a los usuarios del sistema de acuerdo a la categoría menú.

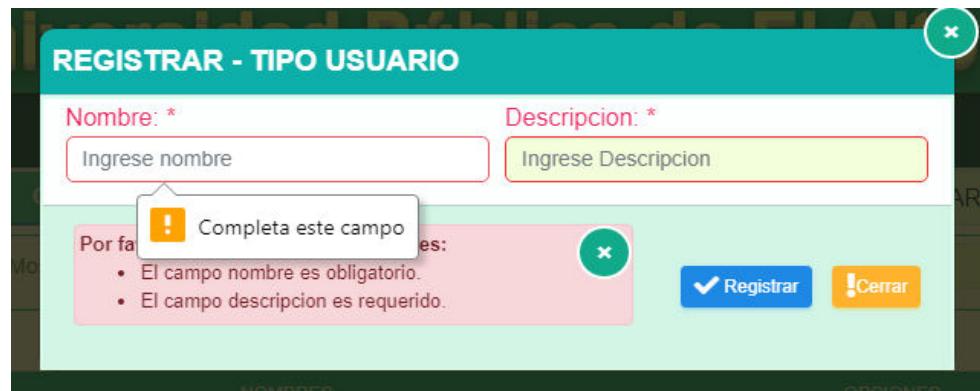
Figura 11: Modal de asignación de permisos



4.3.2 Crear nuevo registro de tipo usuario

Al hacer clic en la opción de crear nuevo registro se abrirá el modal de creación o registrar – tipo usuario.

Figura 12: Modal Registrar tipo usuario



En esta interfaz debemos de llenar los campos correspondientes, según los campos requeridos.

4.4 Interfaz Usuarios

En esta interfaz se visualizará los registros creados de Usuarios del Sistema con los botones de crear, editar, eliminar y buscar registros.

Figura 13: Listado de usuarios

GESTIÓN USUARIOS N°1		GESTIÓN USUARIOS N°2				Nuevo Registro	
Tipo Usuario: Todos		Filas a Mostrar: 10	Buscar por Login		Buscar Registro		Total: 0 1
#	C.I.	NOMBRES	LOGIN	ESTADO	TIPO USUARIO	ACCIONES	
1	6144513 LP.	JUAN REGIS MUÑEZ SIRPA	SG-4838414	Activo	ADMINISTRADOR PRINCIPAL		
2	8437517 LP.	DEYVID MAMANI MAMANI	SG-8437517	Activo	ENCARGADO ARCHIVO		
3	390823 LP.	LIDIA IRMA ZUAZO YUJRA	SG-390823	Inactivo	SECRETARIA		
4	4838414 LP.	HERNAN CATARI YUJRA	SG-48384142	Activo	AUXILIAR OFICINA		

4.4.1 Crear nuevo registro de usuarios

Al hacer clic en la opción de crear nuevo registro se abrirá el modal de creación o registrar - menú opción.

Figura 14: Modal Registrar usuarios

REGISTRAR - USUARIO

4838414	4838414 LP - HERNAN CATARI YUJRA
USUARIO: *	CONTRASEÑA: *
SG 4838414	Ingrese contraseña
TIPO USUARIO: *	INICIO PERÍODO: *
5 - AUXILIAR OFICINA	dd/mm/aaaa
FIN PERÍODO:	ESTADO: *
dd/mm/aaaa	Seleccione
Nota: * es un campo obligatorio.	
Por favor corrige los siguientes errores:	
<ul style="list-style-type: none">El campo password es obligatorio.El campo Inicio Periodo es obligatorio.La persona ya está asignado a un usuario activo y/o eliminadoEl tipo usuario ya está asignado a un usuario activo y/o eliminadoEl campo Estado es obligatorio.	
<input checked="" type="button"/> Registrar	<input type="button"/> Cerrar

En esta interfaz debemos de llenar los campos correspondientes, según los campos requeridos de:

- **Persona:** Buscar persona por el número de carnet para asignar a un usuario del sistema.
- **Usuario:** El campo usuario se creará automáticamente con el carnet de identidad de la persona buscada.
- **Contraseña:** Se debe asignar la contraseña que debe tener más de 8 caracteres.
- **Tipo de usuario:** Se debe asignar un tipo de usuario según la lista para asignar a un usuario en caso de que no exista un tipo de usuario puede crear uno nuevo en la pestaña más.
- **Inicio periodo:** Se debe asignar una fecha de inicio de asignación del usuario de acuerdo al memorándum de designación de funciones emitida por recursos humano de la UPEA.
- **Fin periodo:** Se debe asignar una fecha de finalización de asignación del usuario.
- **Estado:** Se debe asignar un estado de activo o Inactivo al usuario.

Nota: si existe datos ya existentes en la tabla usuarios se mostrará un mensaje de error de ya está asignado este campo, si todo está correcto presionamos nuevamente en el botón “Registrar” y de esta forma quedará creado nuestro usuario.

5. MÓDULO ENCARGADO ARCHIVO

Una vez iniciada la sesión como encargado archivo, se abrirá inmediatamente la pantalla inicial de la aplicación.

En esta interfaz se visualizará un encabezado con el nombre de la aplicación, las opciones de menú principal donde se puede apreciar que solo tiene acceso a: gestión resolución con un sub menú Tipo Resolución y resoluciones; gestión convenio con un sub menú contacto, institución, tipo convenio y convenios; gestión publicación con un sub menú etiquetas, categorías y publicaciones, si hacemos clic en cualquiera de las opciones del menú principal nos dirigirá a la interfaz a dicho menú.

Figura 15: Pantalla modulo encargado archivo



5.1 Interfaz gestión resolución – tipo resolución

En esta interfaz se visualizará los registros creados de tipo resolución del Sistema con los botones de crear, editar, eliminar y buscar registros.

Figura 16: Listado tipo resolución

GESTIÓN TIPO RESOLUCIÓN N°1			GESTIÓN TIPO RESOLUCIÓN N°2		Nuevo Registro
Filas a Mostrar:	10	▼	Buscar por Nombre	Buscar Registro	
#	NOMBRES	DESCRIPCION	ACCIÓN	Total: 7	1
1	III CONGRESO 2012	Resolución III Congreso 2012			
2	II CONGRESO 2008	Resolución I Congreso 2008			
3	II CONGRESO 2007	Resolución II Congreso 2007			
4	I CONGRESO 2004	Resolución I Congreso 2004			
5	ADMINISTRATIVA	Resolución Administrativa Rectoral			
6	RECTORAL	Resolución Rectoral			
7	HCU	Honorable Consejo Universitario			

5.1.1 Crear nuevo registro tipo resolución

Al hacer clic en la opción de crear nuevo registro se abrirá el modal de creación o registrar – tipo resolución.

Figura 17: Modal Registrar tipo resolución



5.1.2 Interfaz gestión resolución – resoluciones

En esta interfaz se visualizará los registros creados de resoluciones del Sistema con los botones de crear, editar, eliminar, ver y buscar registros.

Figura 18: Listado resoluciones

GESTIÓN RESOLUCIÓN N°1				GESTIÓN RESOLUCIÓN N°2				Nuevo Registro			
Tipo Resolución:		Todos		Filas a Mostrar:		10		Buscar por Título		Buscar Registro	
#	Nro.	FECHA	TITULO	RESUELVE		TIPO RES.	ARCHIVO	ACCIÓN			
1	007 / 2015	28-01-2015	DESCONCENTRACION DE LAS CAJA DE PAGO	ARTICULO ÚNICO. - Se dispone la creación de cajas de pagos de obligaciones económicas al interior de la Universidad Pública de El Alto, descentrando cajas de pago para docentes, cajas de pago estudiantes y cajas de pago administrativos. La Máxima Autoridad Ejecutiva a través de sus unidades respectivas, es la encargada de dar cumplimiento a la presente resolución.		HCU	345.3 KB				
										Total: 1	

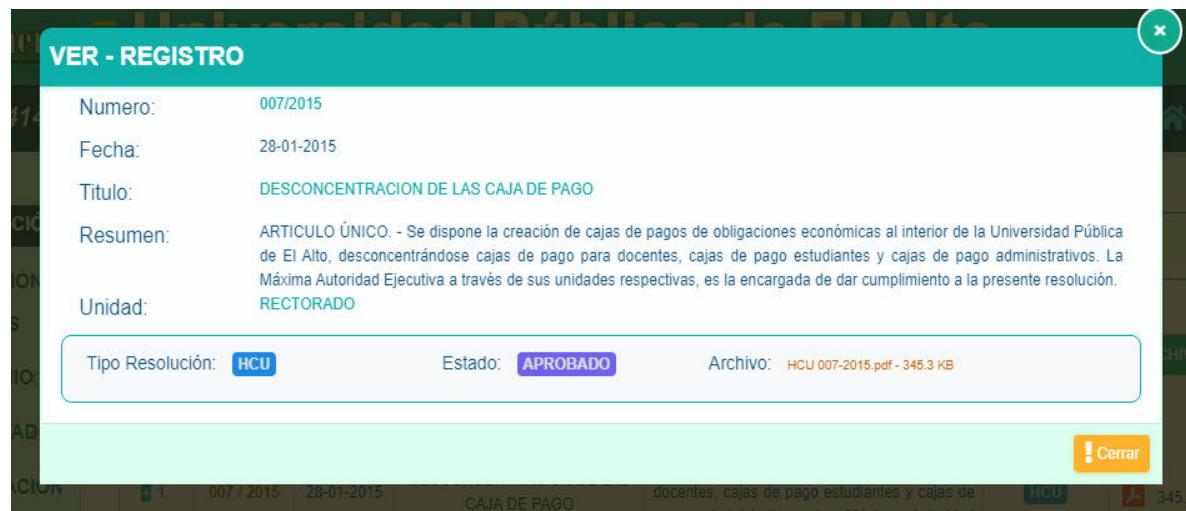
Al hacer clic en el ícono pdf en la columna archivo de listado de resoluciones se abrirá un modal donde se puede ver el archivo asignado a una resolución con marca de agua “no valido para trámites”.

Figura 19: Modal ver-archivo resolución



Al hacer clic en el icono ver en la columna acción de listado de resoluciones se abrirá un modal donde se puede ver el detalle de la resolución.

Figura 20: Modal ver-registro resolución



5.1.3 Crear nuevo registro resolución

Al hacer clic en la opción de crear nuevo registro se abrirá el modal de creación o registrar – resoluciones.

Figura 21: Modal Registrar resolución

The screenshot shows a modal window titled 'REGISTRAR - RESOLUCIÓN'. It contains several input fields and dropdown menus:

- Nº RESOLUCIÓN:** A text input field with placeholder 'Ingrese numero'.
- GESTIÓN:** A dropdown menu with placeholder 'Seleccione Gestión'.
- FECHA:** A date input field with placeholder 'dd/mm/aaaa'.
- TITULO RESOLUCIÓN:** A text input field with placeholder 'Ingrese título'.
- RESUELVE RESOLUCIÓN:** A text input field with placeholder 'Ingrese resuelve'.
- TIPO RES.:** A dropdown menu with placeholder 'Seleccione'. A sub-menu is open, showing options: 'Seleccione', '1 - HCU', '2 - RECTORAL', '3 - ADMINISTRATIVA', '4 - I CONGRESO 2004', '5 - II CONGRESO 2007', '6 - II CONGRESO 2008', and '7 - III CONGRESO 2012'. The '1 - HCU' option is highlighted.
- UNIDAD SOLICITANTE:** A dropdown menu with placeholder 'Ingresar unidad solicitante'.
- ESTADO:** A dropdown menu with placeholder 'Seleccione'.
- OBSERVACION RESOLUCIÓN:** A text input field with placeholder 'Ingrese observacion Ejm. Ninguna'.

At the bottom right of the modal are two buttons: 'Registrar' (with a checkmark icon) and 'Cerrar' (with a close icon).

En esta interfaz debemos de llenar los campos correspondientes, según los campos requeridos de:

- **Nº Resolución:** Llenar el campo número de resolución.
- **Gestión:** Llenar el campo gestión de la resolución.
- **Fecha:** Se debe asignar una fecha de resolución.
- **Titulo resolución:** Se debe asignar un título de la resolución.
- **Resuelve resolución:** Se debe asignar un resuelve de la resolución.
- **Tipo resolución:** Se debe asignar un tipo resolución según la lista para asignar a una resolución, en caso de que no exista un tipo resolución se puede crear uno nuevo en la pestaña más abriendose otro modal para crear.
- **Unidad Solicitante:** Se debe asignar la unidad solicitante de la resolución.
- **Estado:** Se debe asignar un estado de aprobado, borrador, abrogada o derogada a la resolución.
- **Archivo:** Subir un archivo pdf opcional en modo creación.

Nota: si existe datos ya existentes en la tabla resoluciones se mostrará un mensaje de error de ya está asignado este campo, si todo está correcto presionamos nuevamente en el botón “Registrar” y de esta forma quedará creado nuestra

resolución.

5.1.4 Edición registro resolución

Al hacer clic en icono editar en la lista de resoluciones de la columna acción se abrirá el modal de editar resolución, donde se puede subir el archivo de la resolución con extensión pdf.

Figura 22: Modal Registrar buscar archivo

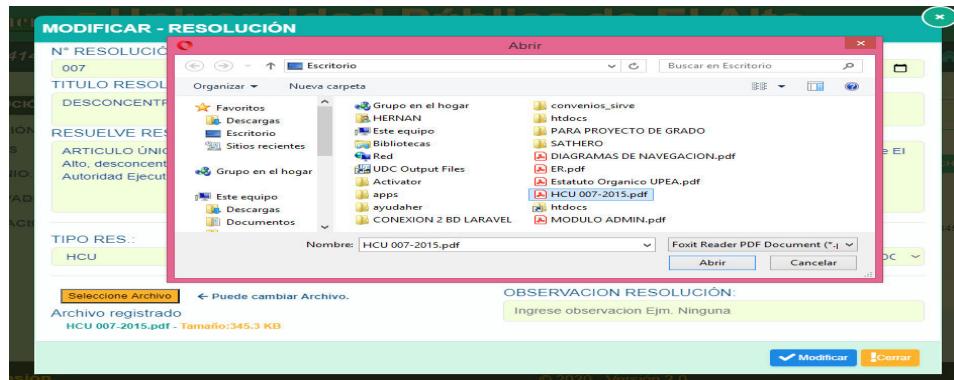


Figura 23: Modal asignar archivo pdf



Para modificar un campo se debe seleccionar los campos o campo que vea conveniente como se puede observar en seleccione archivo se ve en vista previa el

archivo seleccionado en formato pdf, como la unidad y entre otros para luego hacer en clin en el botón modificar la resolución.

5.2 Interfaz gestión convenios – tipo convenio

En esta interfaz se visualizará los registros creados de tipo convenio del Sistema con los botones de crear, editar, eliminar y buscar registros.

Figura 24: Listado tipo convenio

GESTIÓN TIPO CONVENIO N°1		GESTIÓN TIPO CONVENIO N°2		Nuevo Registro
Filas a Mostrar:	5	Buscar por Nombre	Buscar Registro	
#	NOMBRES	DESCRIPCION	ACCIÓN	
1	COOPERACIÓN	Convenio Cooperacion: Este convenio (también conocido como convenio o acuerdo marco) es una expresión de la voluntad de las partes por desarrollar y favorecer acciones de cooperación en ámbitos de mutuo interés y beneficio.		
2	ESPECIFICO	Convenio Especifico: Son convenios cuyo objetivo es establecer actividades específicas de un área académica, científica y/o administrativa en particular, con la finalidad de facilitar la ejecución del convenio.		
3	MARCO	Convenio Marco: Es un acuerdo por el que se crea un espacio de colaboración común entre las partes firmantes, que posteriormente, se concretará en acciones determinadas; a través de la firma de convenios específicos.		

5.2.1 Crear nuevo registro tipo convenio

Al hacer clic en la opción de crear nuevo registro se abrirá el modal de creación o registrar – tipo convenio.

Figura 25: Modal Registrar tipo convenio

REGISTRAR - TIPO CONVENIO

Tipo Convenio: *

Ingrese nombre tipo convenio

Descripcion: *

Ingrese Descripcion

Por favor corrige los siguientes errores:

- El campo nombre tipo convenio es obligatorio.
- El campo descripción es obligatorio.

Registrar Cerrar

5.2.2 Interfaz gestión convenios – convenios

En esta interfaz se visualizará los registros creados de convenios del Sistema con los botones de crear, editar, eliminar, ver y buscar registros.

Figura 26: Listado convenios

The screenshot shows a web-based application titled 'GESTIÓN CONVENIOS'. It has two tabs: 'Nº1' and 'Nº2'. The 'Nº1' tab is active, displaying a list of conventions. At the top, there are filters: 'Tipo Convenio: ESPECIFICO', 'Filas a Mostrar: 2', and search fields 'Buscar por Titulo' and 'Buscar Registro'. Below the filters, a table lists two conventions:

#	NUMERO	TITULO	OBJETIVO	FECHA INICIO	FECHA FIN	TIPO CONV.	ARCHIVO	ACCIONES
1	17 / 2015	CONVENIO ESPECIFICO ENTRE LA UNIVERSIDAD PUBLICA DE ALTO(BOLIVIA) Y LA UNIVERSIDAD NACIONAL DE GENERAL SARMIENTO(ARGENTINA)	La finalidad del presente convenio específico es la de establecer el marco legal para la realización de viajes educativos y tránsidos de los estudiantes de cada una de las Universidades firmantes.	24/04/2015	24/04/2020	ESPECIFICO	Sin/Doc	
2	13 / 2015	CONVENIO DE ESPECIFICO DE COOPERACION INTERINSTITUCIONAL ENTRE EL CONSEJO DE LA MAGISTRATURA, TRIBUNAL DEPARTAMENTAL DE JUSTICIA DE LA PAZ Y LA UNIVER ...	El presente convenio tiene por objeto normar institucionalmente el proceso de selección de los universitarios y/o egresados de la Universidad Pública de El Alto. Carrera de Derecho, Carrera de Ingeniería de Sistemas y Carrera de Administración de Empresas, para que los mismos puedan realizar PASANTIAS Y/O TRABAJOS DIRIGIDOS.	07/05/2015	07/05/2017	ESPECIFICO	Sin/Doc	

At the bottom right of the table, there are pagination controls: 'Total: 3', '1', '2', '3'.

Al hacer clic en el ícono pdf en la columna archivo de listado de convenios se abrirá un modal donde se puede ver el archivo asignado a un convenio con marca de agua “no valido para trámites”, en caso de no tener archivo se mostrará “Sin/Doc”.

Al hacer clic en el ícono ver en la columna acción de listado de convenios se abrirá un modal donde se puede ver el detalle de la resolución.

Figura 27: Modal ver-registro convenio

The modal window is titled 'VER - REGISTRO'. It displays the following details for a specific convention:

- Numero:** 13/2015
- Fecha Firma:** 07/05/2015
- Titulo Convenio:** CONVENIO DE ESPECIFICO DE COOPERACION INTERINSTITUCIONAL ENTRE EL CONSEJO DE LA MAGISTRATURA, TRIBUNAL DEPARTAMENTAL DE JUSTICIA DE LA PAZ Y LA UNIVERSIDAD PUBLICA DE EL ALTO CARRERA DE DERECHO-INGENIERIA DE SISTEMAS Y ADMINISTRACION DE EMPRESAS.
- Objetivo Convenio:** El presente convenio tiene por objeto normar institucionalmente el proceso de selección de los universitarios y/o egresados de la Universidad Pública de El Alto. Carrera de Derecho, Carrera de Ingeniería de Sistemas y Carrera de Administración de Empresas, para que los mismos puedan realizar PASANTIAS Y/O TRABAJOS DIRIGIDOS.
- Fecha Inicio:** 07/05/2015 **Fecha Fin:** 07/05/2017 **Duración Convenio:**
- Beneficios corto plazo:**
- Beneficios largo plazo:**
- Observaciones:** Lic. Freddy Sanabria Taboada CONSEJO MAGISTRATURA, Lic. Miguel Angel Vargas Castellon RECTOR-UPEA, y directores mencionados anteriormente.
- Institución Convenio:**
- Contacto Institución:**
- Unidad beneficiaria:**
- Tipo Convenio:** ESPECIFICO **Estado:** Inactivo **Archivo:** Sin/Doc

At the bottom right of the modal, there is a button labeled 'Cerrar'.

5.2.3 Crear nuevo registro convenio

Al hacer clic en la opción de crear nuevo registro se abrirá el modal de creación o registrar – convenios.

Figura 21: Modal Registrar convenio

The screenshot shows the 'Registrar - CONVENIOS' modal. At the top right, a red box displays the message 'CAMPOS VACIOS. ! Por favor llenar todos los campos requeridos.' (Empty fields. ! Please fill all required fields). The form contains several input fields with validation errors:

- Nº Convenio:** A red border surrounds the input field, and a tooltip says 'Completa este campo' (Fill this field).
- Gestión:** A dropdown menu labeled 'Selección Gestión' has a red border.
- Tipo Convenio:** A dropdown menu labeled 'Selección' has a red border.
- Titulo Convenio:** An input field with a red border and the placeholder 'Ingrese título' (Enter title).
- Objetivo Convenio:** An input field with a red border and the placeholder 'Ingrese objetivo' (Enter objective).
- Fecha Firma:** A date input field with a red border and placeholder 'dd/mm/aaaa'.
- Fecha Inicio:** A date input field with a red border and placeholder 'dd/mm/aaaa'.
- Fecha Conclusión:** A date input field with a red border and placeholder 'dd/mm/aaaa'.
- Contacto Institución:** A dropdown menu labeled 'Selección' has a red border.
- Institución:** A dropdown menu labeled 'Selección' has a red border.
- Unidad Ejecutora:** A dropdown menu labeled 'Ingrese unidad solicitante' has a red border.
- Beneficios a corto P:** An input field with a red border and placeholder 'Ingrese Beneficios a corto plazo'.
- Beneficios a largo P:** An input field with a red border and placeholder 'Ingrese Beneficios a largo plazo'.
- Observaciones:** An input field with a red border and placeholder 'Ingrese observaciones'.
- ESTADO:** A dropdown menu labeled 'Selección' has a red border.

A pink callout box at the bottom left lists validation errors:

- El campo numero es obligatorio.
- El campo gestión es obligatorio.
- El campo título convenio es obligatorio.
- El campo objetivo es obligatorio.
- El campo fechafirma es obligatorio.
- El campo fecha inicio es obligatorio.
- El campo estado es obligatorio.
- El campo descripción es obligatorio.
- El campo tipo convenio id es obligatorio.
- El campo institucion id es obligatorio.
- El campo contacto id es obligatorio.
- El campo unidad id es obligatorio.

At the bottom right are two buttons: a blue 'Registrar' button with a checkmark icon and an orange 'Cerrar' button with a close icon.

En esta interfaz debemos de llenar los campos correspondientes, según los campos requeridos de:

- **Nº convenio:** Llenar el campo número de convenio.
- **Gestión:** Llenar el campo gestión de la resolución.
- **Tipo convenio:** Se debe asignar un tipo convenio según la lista para asignar a un convenio, en caso de que no exista un tipo convenio se puede crear uno nuevo en la pestaña más abriendose otro modal para crear.
- **Titulo convenio:** Se debe asignar un título del convenio.
- **Objetivo convenio:** Se debe asignar un objetivo de convenio.
- **Fecha firma:** Se debe asignar una fecha de firma del convenio.
- **Fecha Inicio:** Se debe asignar una fecha de inicio convenio.
- **Fecha conclusión:** Se debe asignar una fecha de conclusión convenio.
- **Contacto Institución:** Se debe asignar un contacto de referencia de la institución a firmar.

- **Institución:** Se debe asignar una institución al que pertenece el contacto.
- **Unidad Ejecutora:** Se debe asignar la unidad ejecutora al que pertenece el convenio.
- **Beneficios a corto y largo plazo:** Se debe asignar los beneficios del convenio.
- **Estado:** Se debe asignar un estado de aprobado, borrador, abrogada o derogada a la resolución.
- **Archivo:** Subir un archivo pdf opcional en modo creación.

Nota: si existe datos ya existentes en la tabla convenios se mostrará un mensaje de error de “ya está asignado este campo”, si todo está correcto presionamos nuevamente en el botón “Registrar” y de esta forma quedará creado el convenio.

5.2.4 Edición registro convenio

Al hacer clic en icono editar en la lista de resoluciones de la columna acción se abrirá el modal de editar convenio, donde se puede subir el archivo del convenio con extensión pdf.

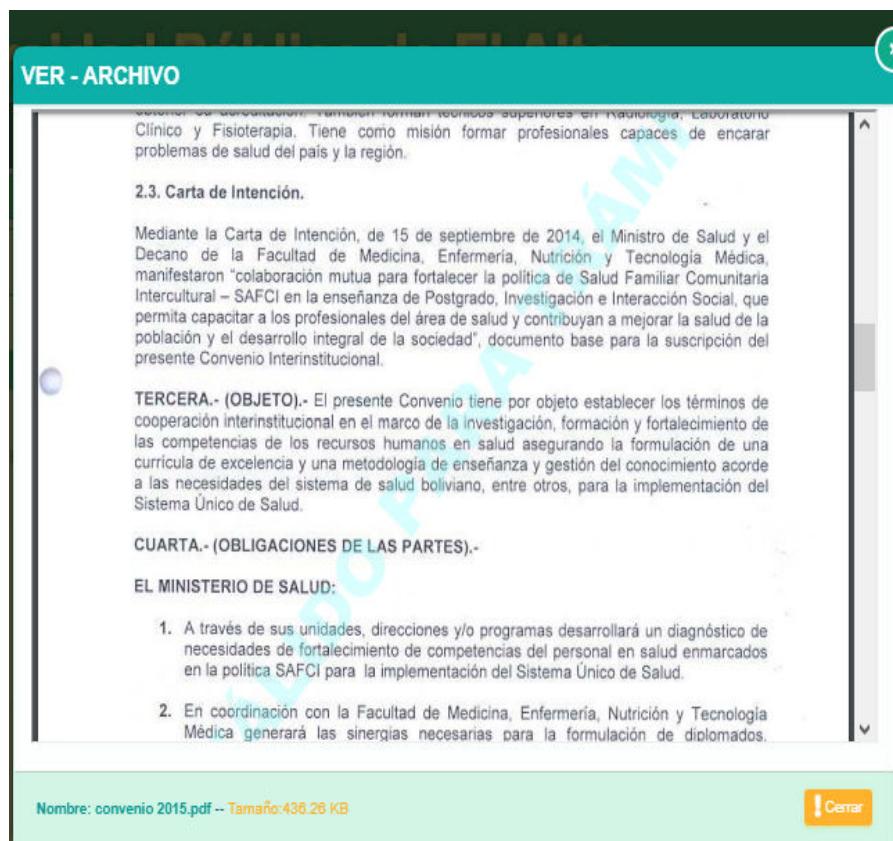
Figura 22: Modal Editar Convenio

Para modificar un campo se debe seleccionar los campos o campo que vea conveniente como se puede observar en seleccione archivo se ve en vista previa el archivo seleccionado en formato pdf, como la unidad solicitante y entre otros para luego hacer en clic en el botón modificar convenio.

5.2.5 Ver archivo de convenio

Al hacer clic en el ícono pdf en la columna archivo de listado de convenios se abrirá un modal donde se puede ver el archivo asignado a dicho convenio con marca de agua “no válido para trámites”.

Figura 23: Modal ver archivo Convenio



6. MÓDULO DE SECRETARIO GENERAL, SECRETARIA Y AUXILIAR DE OFICINA

Una vez iniciada la sesión como uno de los usuarios del módulo mencionado en el punto 6, se abrirá inmediatamente la pantalla inicial de la aplicación de solo en vistas.

Figura 24: Pantalla módulo Secretario General, Secretaria y Auxiliar de Oficina



En esta interfaz se visualizará un encabezado con el nombre de la aplicación, las opciones de menú principal donde se puede apreciar que solo tiene acceso a: gestión resolución y gestión convenios, si presionamos en cualquiera de las opciones del menú principal nos dirigirán a la interfaz a dicho menú.

Figura 24: Pantalla módulo gestión convenios

GESTIÓN CONVENIOS N°1				GESTIÓN CONVENIOS N°2					
Tipo Convenio:		Todos	Filas a Mostrar:	2	Buscar por Título	Buscar Registro			
Total: 281	1	2	3	...	131				
#	NUMERO	TITULO	OBJETIVO	FECHA INICIO	FECHA FIN	TIPO CONV.	ARCHIVO	ACCIONES	
1	1 / 2020	CONVENIO INTERINSTITUCIONAL ENTRE EL MINISTERIO DE SALUD Y LA UNIVERSIDAD PÚBLICA DE EL ALTO	El presente convenio tiene por objetivo establecer los términos de cooperación interinstitucional entre la universidad pública de el alto y el ministerio de salud, para favorecer al área salud de la UPEA.	22/08/2020		ESPECIFICO	436.26 KB		
2	11 / 2015	CONTRATO DE COMODATO	El comodante declara ser poseedor de buena fe del inmueble ubicado en la zona no seco ex tranca de esta ciudad... Al presente el comodante entrega 10 ambientes de la segunda planta del inmueble, destinados al uso de consultorios del Seguro Universitario de la UPEA a favor del Comodatario....	26/05/2015	31/12/2015	CONTRATO DE COMODATO	Sin/Doc		

Como se puede observar no tiene las opciones de crear, editar ni eliminar registros, solo puede tener la opción de lectura y vista.

DOCUMENTACIÓN

La Paz, 28 de agosto de 2020

Señor

Ing. David Carlos Mamani Quispe

DIRECTOR DE CARRERA INGENIERÍA DE SISTEMAS - UPEA

Presente. -

REF.: AVAL DE CONFORMIDAD

Distinguido Ingeniero:

Mediante la presente tengo en bien comunicarle mi conformidad del proyecto de grado “**SISTEMA DE GESTIÓN PARA LA GACETA UNIVERSITARIA ON-LINE UTILIZADO HERRAMIENTAS OPEN SOURCE**” CASO: SECRETARIA GENERAL DE LA UNIVERSIDAD PUBLICA DE EL ALTO, que propone el postulante Univ. **Hernan Catari Yujra**, con cedula de identidad **4838414 LP.**, para su defensa publica, evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba saludos cordiales.

Atentamente,

Ing. Enrique Flores Baltazar
TUTOR METODOLOGICO
TALLER DE LICENCIATURA II

La Paz, 28 de agosto de 2020

Señor

Ing. Enrique Flores Baltazar

DOCENTE – TALLER DE LICENCIATURA II

CARRERA INGENIERÍA DE SISTEMAS - UNIVERSIDAD PUBLICA DE EL ALTO

Presente. –

REF.: AVAL DE CONFORMIDAD

Distinguido Ingeniero:

Mediante la presente tengo en bien comunicarle mi conformidad del proyecto de grado "**SISTEMA DE GESTIÓN PARA LA GACETA UNIVERSITARIA ON-LINE UTILIZADO HERRAMIENTAS OPEN SOURCE**" CASO: SECRETARIA GENERAL DE LA UNIVERSIDAD PUBLICA DE EL ALTO, que propone el postulante Univ. **Hernan Catari Yujra**, con cedula de identidad **4838414 LP.**, para su defensa publica, evaluación correspondiente a la materia de Taller de Licenciatura II, de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba saludos cordiales.

Atentamente,



Ing. Isaac Magno Quispe Mamani
TUTOR ESPECIALISTA

La Paz, 28 de agosto de 2020

Señor

Ing. Enrique Flores Baltazar

DOCENTE – TALLER DE LICENCIATURA II

CARRERA INGENIERÍA DE SISTEMAS - UNIVERSIDAD PÚBLICA EL ALTO

Presente. –

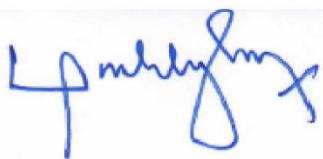
REF.: AVAL DE CONFORMIDAD

Distinguido Ingeniero:

Mediante la presente tengo en bien comunicarle mi conformidad del proyecto de grado “**SISTEMA DE GESTIÓN PARA LA GACETA UNIVERSITARIA ON-LINE UTILIZADO HERRAMIENTAS OPEN SOURCE**” CASO: SECRETARIA GENERAL DE LA UNIVERSIDAD PUBLICA EL ALTO, que propone el postulante Univ. **Hernán Catari Yujra**, con cedula de identidad **4838414 LP.**, para su defensa publica, evaluación correspondiente a la materia de **Taller de Licenciatura II**, de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba saludos cordiales.

Atentamente,



Lic. Fredy Alanoca Coareti

TUTOR REVISOR



Universidad Pública de El Alto

Creada por Ley 2125 del 5 de Septiembre de 2000 y Autónoma por Ley 2556 del 12 de Noviembre de 2003

La Paz, 28 de agosto de 2020

Señor

Ing. Enrique Flores Baltazar

DOCENTE – TALLER DE LICENCIATURA II

CARRERA INGENIERÍA DE SISTEMAS - UNIVERSIDAD PUBLICA DE EL ALTO

Presente. –

A petición del Interesado:

CERTIFICA

Por la presente tengo a bien comunicarle mi conformidad del proyecto de grado que lleva como título “**SISTEMA DE GESTIÓN PARA LA GACETA UNIVERSITARIA ON-LINE UTILIZADO HERRAMIENTAS OPEN SOURCE**” CASO: SECRETARIA GENERAL DE LA UNIVERSIDAD PUBLICA DE EL ALTO, que propone el postulante Univ. **Hernan Catari Yujra**, con cedula de identidad **4838414 LP.**, de tal forma cabe recalcar que el sistema satisface los requerimientos de la unidad de Secretaria General.

Por tal motivo certifico para fines consiguientes del interesado, cumplimiento de esta manera con las formalidades y requisitos necesarios para su defensa publica y evaluación correspondiente.

Sin otro particular, reciba usted saludos cordiales.

Atentamente,




Dr. Juan Carlos Saavedra Villamil
SECRETARIO GENERAL
UNIVERSIDAD PÚBLICA DE EL ALTO