UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA

PROYECTO DE GRADO

SISTEMA DE INFORMACIÓN DE GESTIÓN Y CONTROL DOCUMENTAL PARA EL INSTITUTO NACIONAL DE LABORATORIOS DE SALUD "NÉSTOR MORALES VILLAZÓN – INLASA"

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERIA DE SISTEMAS INFORMÁTICOS

POSTULANTE: RICHARD PAUCARA MACHACA

TUTORA METODOLÓGICA: LIC. BHYLENIA YHASMYNA RIOS MIRANDA

ASESOR: M. SC. CARLOS MULLISACA CHOQUE

LA PAZ – BOLIVIA
2016

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA

LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria

A mis padres Roberto y Paulina
por estar siempre a mi lado
brindándome su apoyo incondicional,
a Miriam, Gisel, Anabel y Danilo
por ser la razón de mis esfuerzos.

AGRADECIMIENTOS

Agradezco a Dios por darme la vida y por permitirme llegar a este punto de mi vida, la felicidad que siento la quiero compartir contigo, tú siempre me acompañas y haces que pueda conseguir mis metas.

A mi Tutora Metodológica Lic. Bhylenia Yhasmyna Rios Miranda, quien con paciencia y gran calidad de persona, profesionalismo y experiencia acompaño el desarrollo de este trabajo desde sus inicios aportando siempre con sus observaciones y consejos oportunos.

A mi Asesor M. Sc. Carlos Mullisaca Choque, quien con su excelente calidad profesional realizo el seguimiento a este trabajo, sus observaciones, correcciones, acompañados de su experiencia fueron oportunos en la elaboración de este proyecto de grado.

A toda mi familia por nunca dudar de mis capacidades.

RESUMEN

La creciente de manda en la búsqueda de soluciones prácticas y exitosas en entidades públicas en la actualidad genera la necesidad de contar con sistemas que permitan la gestión eficaz de los recursos de información y documentación, el presente Proyecto de Grado responde al desarrollo de la temática Gestión Documental como línea de trabajo en el Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA".

Los Sistemas de Gestión de Archivos son sistemas de información específicos para el tratamiento archivístico de los archivos e incorporan todas las prestaciones para la informatización integral de las funciones propias de un archivo: descripción y recuperación de fondos, gestión documental, control de depósitos, gestión de la consulta y el préstamo.

La intención del presente proyecto es desarrollar un sistema web que pueda dar solución a la demora las practicas archivísticas, el poco de tiempo en la atención a las solicitudes vinculadas al servicio que brinda el Área de Archivo Central en INLASA, razón por la cual se considera sistematizar una gran parte de los procesos en la práctica archivista a través de un sistema de gestión de archivo.

El proyecto que desarrollara un producto de software denominado Sistema de Información de Gestión y Control Documental en adelante conocido como SIGCD será desarrollado desde una óptica archivística, aplicará las fases de la metodología SCRUM para modelar los requerimientos en historias de usuario y las actividades a realizar, las etapas de la metodología web OOHDM para el modelado web. A la conclusión del proyecto se procederá con una evaluación de calidad del mismo utilizando la ISO-9126, una estimación del costo aproximado del proyecto a través del método paramétrico PF Lite y finalizando con las conclusiones del proyecto y sus recomendaciones.

Índice del Proyecto

CAPÍTULO I MARCO REFERENCIAL

1.1 Introducción…… 1

1.2 Antecedentes……..... 3

1.2.1 Antecedentes de La Institución……..…… 3

1.2.2 Antecedentes de Proyectos Pasados……..…. 3

1.3 Planteamiento del Problema…….. 4

1.4 Objetivos…… 5

1.4.1 Objetivo General……..……..... 5

1.4.2 Objetivos Específicos……..…….. 5

1.5 Justificación…….. 6

1.5.1 Social……..…… 6

1.5.2 Técnica…… 6

1.5.3 Económica……..…… 7

1.6. Límites y Alcances…….…… 8

1.6.1 Delimitación Temática……...……. 8

1.6.2 Delimitación Espacial……...……..... 8

1.6.3 Delimitación Temporal……..……. 8

1.6.4 Alcances……..…… 8

1.7 Aportes…… 9

1.8 Metodología…….…….……..... 9

1.8.1 Metodología Scrum…….……..... 9

1.8.2 Beneficios de Scrum……. 10

1.8.3 Proceso y Roles de Scrum……. 11

1.8.3.1 El Proceso Scrum…….…….. 11

1.8.3.2 Roles de Scrum……..…… 13

1.9. Marco Teórico……..…….……..…… 14

1.9.1 Técnicas de Información……..…….…….….…… 14

1.9.2 Herramientas……...…….…… 15

1.9.2.1 OOHDM- Object-Oriented Hypermedia Design Method…….. 15

1.9.2.2 Fases de OOHDM….……..……. 16

1.9.3 PHP……..……..…….. 18

1.9.3.1 Características de PHP…… 21

1.9.4 MYSQL…….…….. 22

1.9.4.1 Uso de MYSQL…….…….. 23

1.9.4.2 Características de MYSQL…….…… 23

1.9.5 Adobe Dreamweaver…….…..…… 23

1.9.6 Xampp…….…… 24

1.9.6.1 Características y Requisitos de Xampp...……..... 25

1.9.7 Requerimientos del Software…….……...…… 25

CAPÍTULO II MARCO TEÓRICO

2.1 Ingeniería de Software……. . 26

2.1.1 Metodologías de Desarrollo…..……. 26

2.1.2 El Manifiesto Ágil……. 27

2.1.3 Principios del Manifiesto Ágil…… 28

2.1.4 Historia de Usuario……. 29

2.1.5 SCRUM…….. 31

2.1.6 Teoría de SCRUM……..……..…… 32

2.1.6.1 Transparencia……..…… 32

2.1.6.2 Inspección……..……. 32

2.1.6.3 Adaptación……..…… 33

2.1.7 El Equipo SCRUM (Scrum Team)……. 33

2.1.7.1 El Dueño de Producto (Product Owner)…..…….…..... 34

2.1.7.2 El Equipo de Desarrollo (Development Team)……..……. 35

2.1.7.3 Tamaño del Equipo de Desarrollo……..……. 35

2.1.7.4 El Dueño del Equipo (SCRUM Master)……..…… 36

2.1.7.5 El Servicio del SCRUM Master al Dueño del Producto…… … 36

2.1.7.6 El Servicio del SCRUM Master al Equipo de Desarrollo..……. 37

2.1.7.7 El Servicio del SCRUM Master a la Organización…….. 37

2.1.8 Eventos de SCRUM……. 37

2.1.8.1 El Sprint……..……. 38

2.1.8.2 Cancelación de un Sprint…….. 39

2.1.8.3 Reunión de Planificación de Sprint……. 39

2.1.8.4 Objetivos del Sprint……..……..... 41

2.1.8.5 SCRUM Diario……..……. 42

2.1.8.6 Revisión de Sprint…… 43

2.1.8.7 Retrospectiva de Sprint……..……. 44

2.1.9 Artefactos de SCRUM……. 45

2.1.9.1 Lista de Producto (Product Backlog)……..…… 44

2.1.9.2 Lista de Pendientes del Sprint (Sprint Backlog)……..…… 48

2.1.9.3 Incremento……..……. 49

2.1.10 Transparencia de los Artefactos…… 49

2.1.11 Definición de Terminado (Definition of “Done”)……. 50

2.2 Metodología OOHDM……. 51

2.2.1 Fases de OOHDM…..…….……. 51

2.2.1.1 Diseño Conceptual……..……. 51

2.2.1.2 Diseño Navegacional…….…….. 52

2.2.1.3 Diseño Interfaz Abstracta……..……. 54

2.2.1.4 Implementación……..……. 55

2.3 Calidad del Producto Software con ISO-9126……. 56

2.3.1 Características Norma ISO-9126……. 56

2.4 Estimación del Tamaño por Punto Función…… 57

2.4.1 Medición de PF con el Método PF LITE……. 58

2.4.2 Estimación en base a PF…….. 60

2.5 Los Documentos y su Importancia……. 61

2.5.1 Definición del Término Documento…… 62

2.5.2 Características Esenciales de los Documentos…… 63

2.5.3 Ciclo de Vida de los Documentos…… 64

2.5.3.1 Primera Etapa Documentación Activa (Archivo de Oficina)…….. 64

2.5.3.2 Segunda Etapa Documentación Semi-activa (Archivo Central)…. 64

2.5.3.3 Tercera Etapa Documentación Inactiva (Archivo Histórico)…….. 64

2.5.4 Calidad de los Documentos……. 65

2.6 La Archivística…… 66

2.6.1 Practica Archivística…… 67

2.6.2 Concepto de Archivo…….…….. 68

2.6.3 Fases y Operaciones del Tratamiento Archivístico……. 69

2.6.4 Instrumentos de Descripción…… 70

2.6.4.1 Guía…….. 70

2.6.4.2 Inventario……. 70

2.6.4.3 Catálogo……..……. 71

2.6.4.4 Índice…… 71

2.7 Norma ISAD (G)……..…… 71

2.7.1 Definición y Alcance……..…….……. 73

2.7.2 Descripción Multinivel…….….…….. 73

2.7.3 Reglas de Descripción Multinivel……..…….……. 74

2.7.3.1 Descripción de lo General a lo Particular……...…….. 74

2.7.3.2 Información Pertinente para el Nivel de Descripción……..……. 74

2.7.3.3 Vinculación de las Descripciones……..…….. 74

2.7.3.4 No Repetición de la Información……..…… 74

2.7.4 Estructura y Contenido de la ISAD (G)……...……. 75

CAPÍTULO III MARCO APLICATIVO

3.1 Análisis del Sistema Actual……. 80

3.2 Análisis de Requerimientos (Product Backlog)……..... 80

3.2.1 Identificación de Usuarios……..... 81

3.2.2 Definición de los Roles de los Actores…… 82

3.2.3 Requerimientos de Usuario…….. 82

3.3 Planificación de la Entrega…….. 83

3.4 Iteraciones (Sprints)……. 84

3.4.1 Primera Iteración – Módulo Usuarios……. 85

3.4.1.1 Diseño de la Base Data……..……. 86

3.4.1.2 Diseño Conceptual – Módulo Usuarios…… 87

3.4.1.3 Diseño Navegacional – Módulo Usuarios…… 88

3.4.1.4 Diseño Interfaz Abstracta – Módulo Usuarios……. 89

3.4.1.5 Implementación - Módulo Usuarios…….……..... 90

3.4.2 Segunda Iteración – Módulo Administración…….. 91

3.4.2.1 Diseño Conceptual – Módulo Administración……..……. 91

3.4.2.2 Diseño Navegacional – Módulo Administración……..…….. 93

3.4.2.3 Diseño Interfaz Abstracta – Módulo Administración…….…… 94

3.4.2.4 Implementación - Módulo Administración…….……..... 95

3.4.3 Tercera Iteración – Módulo de Descripción…… 97

3.4.3.1 Diseño Conceptual – Módulo de Descripción……..…… 98

3.4.3.2 Diseño Navegacional – Módulo de Descripción……..…… 98

3.4.3.3 Diseño Interfaz Abstracta – Módulo de Descripción…….……. 100

3.4.3.4 Implementación - Módulo de Descripción…….……..... 101

3.4.4 Cuarta Iteración – Módulo de Solicitudes…… 103

3.4.4.1 Diseño Conceptual – Módulo de Solicitudes..……..…… 103

3.4.4.2 Diseño Navegacional – Módulo de Solicitudes……..……..…… 105

3.4.4.3 Diseño Interfaz Abstracta – Módulo de Solicitudes..…….……. 106

3.4.4.4 Implementación - Módulo de Solicitudes…….……..……..... 107

3.4.4.5 Diseño Conceptual – Módulo de Búsqueda..……...…… 109

3.4.4.6 Diseño Navegacional – Módulo de Búsqueda…….……..……. 110

3.4.4.7 Diseño Interfaz Abstracta – Módulo de Búsqueda….…….…… 111

3.4.4.8 Implementación - Módulo de Búsqueda…..…….……..... 112

CAPÍTULO IV CALIDAD DE SOFTWARE

4.1 Calidad de Software…….. 113

4.2 Calidad ISO / IEC 9126……..... 113

4.2.1 Funcionalidad……. 113

4.2.2 Confiabilidad…….. 116

4.2.3 Usabilidad…… 118

4.2.4 Capacidad de Mantenimiento……..... 119

4.2.5 Portabilidad……. 121

4.2.6 Eficiencia…… 122

4.3 Seguridad…… 123

4.3.1 Seguridad de la Base de Datos…… 123

4.3.1.1 Encriptación de Datos…….. 123

4.3.1.2 Autenticación de Usuarios…….. 123

4.3.1.3 Manejo de Sesiones…… 123

4.4 Análisis de Costos…….. 124

4.4.1 Estimación del Esfuerzo……. 125

4.4.2 Estimación de la Duración……. 125

4.4.3 Estimación del Equipo de Trabajo…….. 125

4.4.4 Estimación del Costo de Desarrollo……. 126

4.5 Análisis de Beneficios……. … 126

CAPÍTULO V CONCLUSIONES Y RECOMENDACIONES

Conclusiones……..…….…….…….. 127

Recomendaciones……..…… 127

Bibliografía…… 129

ANEXO A – ÁRBOL DE PROBLEMAS……. 131

ANEXO B – ÁRBOL DE OBJETIVOS…….. 132

ANEXO C – ENCUESTAS……..... 133

ANEXO D – HISTORIAS DE USUARIO…… 135

ANEXO E – ORGANIGRAMA GENERAL INLASA…… 139

ANEXO F – GLOSARIO DE TÉRMINOS……. 140

ÍNDICE DE FIGURAS

Figura 1.1 Estructura Central de SCRUM……. 10

Figura 1.2 Proceso Scrum……. 12

Figura 1.3 Roles de Scrum……. 14

Figura 2.1 Niveles de tamaño con los que trata los requisitos de la Gestión Ágil. 31

Figura 2.2 Diagrama de Ciclo Iterativo de SCRUM……. 46

Figura 2.3 Ejemplo Diseño Conceptual……. 52

Figura 2.4 Ejemplo Diseño Navegacional……. 53

Figura 2.5 Ejemplo de diseño de ADVs…… 54

Figura 2.6 Relaciones entre los diseños conceptuales, navegacional y los objetos de interfaz……. 55

Figura 2.7 Proceso para calcular PF de FP Lite……. 59

Figura 2.8 Ciclo de vida de los Documentos……. 65

Figura 2.9 Modelo de los niveles de un fondo…….. 72

Figura 3.1 Fases y Metodologías utilizadas para el desarrollo del SIGCD…….. 79

Figura 3.2 Identificación de Usuarios…….. 81

Figura 3.3 Diseño de la Base de Datos – Primera Iteración……. 86

Figura 3.4 Diseño Conceptual – Módulo de Usuarios…….. 87

Figura 3.5 Diseño Navegacional – Módulo de Usuarios…….. 88

Figura 3.6 Diseño Interfaz Abstracta - Módulo de Usuarios…… 89

Figura 3.7 Interfaz de Autenticación - Módulo de Usuarios……. 90

Figura 3.8 Interfaz Registro de Usuarios - Módulo de Usuarios…….. 90

Figura 3.9 Diseño Conceptual – Módulo de Administración…… 92

Figura 3.10 Diseño Navegacional – Módulo de Administración…….. 93

Figura 3.11 ADV Módulo de Administración…….. 94

Figura 3.12 ADVs contenidos en el Módulo de Administración…….. 94

Figura 3.13 Interfaz de Administración…… 95

Figura 3.14 Interfaz de Inventario de Expedientes…… 96

Figura 3.15 Interfaz de Registro de Documentos…….. 96

Figura 3.16 Subir y Descargar Documentos……. 97

Figura 3.17 Diseño Conceptual – Módulo de Descripción…… 98

Figura 3.18 Diagrama de Contextos Navegacionales – Módulo de Descripción.. 99

Figura 3.19 ADV Navegación Multinivel - Módulo de Descripción…… 100

Figura 3.20 ADV Descripción ISAD (G)- Módulo de Descripción……. 100

Figura 3.21 Implementación – Módulo de Descripción ISAD – (G)…… 101

Figura 3.22 Editar Módulo de Descripción ISAD – G…….. 102

Figura 3.23 Diseño Conceptual – Módulo de Solicitudes……. 104

Figura 3.24 Diseño Navegacional – Módulo de Solicitudes……. 105

Figura 3.25 ADV Módulo de Solicitudes…….. 106

Figura 3.26 Módulo de Solicitudes – Descargas…… 106

Figura 3.27 Diseño de Implementación – Módulo de Solicitudes……. 107

Figura 3.28 Interfaz – Módulo de Solicitudes – descargas…… 108

Figura 3.29 Diseño Conceptual – Módulo de Búsqueda…….. 109

Figura 3.30 Diseño Navegacional – Módulo de Búsqueda…….. 110

Figura 3.31 Diseño Navegacional – Módulo de Búsqueda…….. 111

Figura 3.32 Implementación – Módulo de Búsqueda – Fechas…… 112

Figura 3.33 Implementación – Módulo de Búsqueda – Dependencias…….. 112

Figura 4.1 Diagrama de Transferencia…….. 117

ÍNDICE DE TABLAS

Tabla 1.1 Características de la computadora…… 7

Tabla 2.1 Diferencias entre Metodologías Ágiles y las Tradicionales…….. 29

Tabla 2.2 Complejidad media para cada elemento funcional…… 59

Tabla 2.3 Diferencias entre un almacén de papel y un archivo de documentos… 68

Tabla 2.4 Fases y Operaciones del Tratamiento Archivístico……... 69

Tabla 2.5 Equivalencia entre agrupaciones documentales y los instrumentos de. descripción archivística 73

Tabla 2.6 Estructura de la norma ISAD (G)…….. 77

Tabla 3.1 Requerimientos de Usuario…… 83

Tabla 3.2 Plan de Entrega……. 84

Tabla 3.3 Primera Iteración…… 85

Tabla 3.4 Segunda Iteración…….. 91

Tabla 3.5 Tercera Iteración…… 97

Tabla 3.6 Cuarta Iteración…… 103

Tabla 4.1 Cálculo de Puntos Función…… 114

Tabla 4.2 Cálculo de Confiabilidad…….. 118

Tabla 4.3 Cálculo de Usabilidad…….. 119

Tabla 4.4 Cálculo de la Eficiencia……. 122

Tabla 4.5 Puntos función no ajustados…….. 124

CAPÍTULO I
MARCO REFERENCIAL

1.1 INTRODUCCIÓN

La Organización de un Archivo es una de las actividades administrativas que no gustan a la mayoría de los empleados públicos por razones de distinto índole pero frecuentemente llegan a un mismo punto poner orden a un desorden de un mundo de documentos en los cuales no les encontramos "ni pies ni cabezas" cayendo entonces en un círculo vicioso en el que solo abunda la acumulación de documentos, dificultad en la búsqueda de información o perdida de la misma, reproducción de información en las diferentes oficinas administrativas, generando cantidades innecesarias de papelería provocando a su vez más acumulación de papel, convirtiéndose así en un cuento de nunca acabar.

En ocasiones terminamos rompiendo este círculo eliminando papelería acumulada pero en la mayoría de los casos sin previa revisión de la misma provocando la perdida inclusive de documentos de gran valor por el contenido de los mismos o terminando en el abandono en las orillas de los sitios más lejanos de las edificaciones y áreas administrativas, provocando su deterioro y posteriormente siendo eliminados, perdiendo cuantiosos y valiosos documentos.

Esto se podría evitar si todas las Dependencias Gubernamentales, Oficinas e Instituciones tanto Públicas como Privadas contaran con un Archivo Organizado, obteniendo cómo resultado de ello una rápida y eficaz información acerca del documento que se esté buscando, que sea atendido por personal autorizado que cuente con conocimientos Archivísticos para que pueda ser accesible para la ciudadanía al momento de atender sus inquietudes y darle una pronta resolución de sus problemas, esto apoyado de material apropiado como mobiliario requerido. Así como también contarán con normas internas que permitan normar y regular la administración de los archivos, así como la preservación, conservación y difusión de los documentos.

Se considera el papel esencial de la gestión de documentos, tramites y con transparencia para apoyar la eficiencia de la administración, la rendición de cuentas y el estado de derecho.

Toda institución pública o privada cuenta con documentos para distintos trámites y actos jurídicos propios de la institución o en su relacionamiento con otras instituciones y con la sociedad en general, el manejo de toda esta documentación debe responder a procesos denominados de gestión documental o administrativa, para tener un verdadero control, saber y conocer en que instancia se encuentra cada trámite, que documentos acompañan al mismo, cual es el estado del trámite y acto relacionado al documento, de tal manera que la institución cuente con sistemas y procedimientos organizados de control que aseguren la adecuación, identificación, disponibilidad y conservación de información como parte de la transparencia institucional, la memoria institucional, respondiendo a la seguridad jurídica que deben tener las instituciones tanto en su tramitación interna como en los actos jurídicos y relacionamiento hacia afuera.

Los cada vez más exigentes estándares de calidad demandan una organización y acceso a la información cada vez más segura rápida y efectiva, el uso de sistemas informáticos muestra grandes beneficios en toda organización. la necesidad de la descripción archivística se ha basado siempre sencillamente en la imposibilidad de que el usuario de archivo pueda examinar físicamente todos los documentos custodiados en el mismo, hasta encontrar aquellos que sean de su interés. Los sistemas apoyan a la aplicación de sistemas de gestión archivística e instrumentos de descripción documental en las distintas entidades respecto de la organización de archivos en instituciones.

Es por esta razón que se presenta el Proyecto de Grado denominado Sistema de Información de Gestión y Control Documental para El Instituto Nacional de Laboratorios de Salud “Néstor Morales Villazón – INLASA”, institución desconcentrada del Ministerio de Salud, ya que en la actualidad no existe un sistema que controle el movimiento de los documentos y el mismo será de gran ayuda para todo el personal que trabaja en esta dependencia y no solo de ellos, sino que servirá a todos los ciudadanos al momento de necesitar de alguna consulta y con ello facilitarle la localización de un expediente.

1.2 ANTECEDENTES

1.2.1 ANTECEDENTES DE LA INSTITUCIÓN

El Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA", pertenece al Ministerio de Salud como unidad desconcentrada, fue fundada el año 1908 en el departamento de La Paz, está ubicada en la zona de Miraflores, Pasaje Rafael Zubieta Nro. 1880, al lado del Hospital del Niño del departamento de La Paz, dentro de su historia se creó una biblioteca pero hasta ahora no cuenta con un Archivo Central organizado, presenta documentación con más de 50 años de la cual mucha documentación administrativa se perdió, a causa de esto corren con altas multas económicas y el riesgo de perder su memoria institucional, ya que dentro de los diferentes laboratorios se investiga científicamente y se hicieron aportes de relevancia nacional que hasta ahora pocos conocen.

El Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA" a diario recibe entre 20 y 30 solicitudes de trámites, además de la correspondencia externa, con los diferentes centros que se relaciona y con el Ministerio de Salud, significa un movimiento que requiere de un adecuado control, gestión de trámites y documentación.

Es en este sentido que se identificó inmediatamente la necesidad de establecer un Sistema de Gestión y Control Documental, que regularice y permita un eficaz control en el registro y manipulación de documentos, además de varios trámites como parte de la transparencia del instituto en su responsabilidad como institución pública.

1.2.2 ANTECEDENTES DE PROYECTOS PASADOS

A continuación se muestra proyectos similares de sistemas de información de archivo desarrollados en la Universidad Mayor de San Andrés e implementados en nuestro país.

El Proyecto de Grado Sistema de Descripción Archivística elaborado por Lunasco Cusi Augusto C. fue implementado en Archivo de La Paz el 2008, utiliza como base la Norma Internacional de Descripción Archivística - ISAD (G) para la organización y descripción de los documentos.

El Proyecto de Grado Sistema Web de Gestión Documental Digital y Administración de Archivos desarrollado por Vega Miguel Ángel el año 2015, se lo implementó en el Fondo Nacional de Desarrollo Regional con el fin de atender a la problemática de una gestión documental poco eficiente entre los cuales podemos citar el almacenamiento físico de archivos y documentos.

El Proyecto de Grado Sistema de Administración de Archivos Estudiantiles Kardex – Informática implantado por Loza Moya Javier Alex el año 2007 describe el manejo de los archivos de cada estudiante en la Universidad Mayor de San Andrés utilizando metodologías agiles para el desarrollo del sistema y pruebas de evaluación que verifican el correcto funcionamiento del producto.

El Proyecto de Grado Diseño de un modelo para la Implantación de un Sistema de Gestión Documental en Áreas u Organizaciones Jurídicas desarrollado por Contreras Henao Felipe Andrés y Forero Guzmán Felipe el año 2005 está dedicado al funcionamiento de entidades jurídicas y como estas manejan los procesos donde se involucran documentos de uso activo y diario.

1.3 PLANTEAMIENTO DEL PROBLEMA

La rápida evolución de las tecnologías de la información, ha obligado a asimilar grandes novedades tecnológicas a gran velocidad y ha permitido la aparición de nuevos productos y herramientas informáticas para la gestión electrónica de documentos, pero al momento de aproximarse a estas nuevas tecnologías y al querer implantar un sistema de estos, se llega a una realidad completamente diferente, por un lado se encuentra el mundo tecnológico, en el que se manejan conceptos como, archivos digitales, intranet, e-mail, etc., y por otro lado se encuentra el mundo de profesionales o especialistas en tecnologías de información como ingenieros de sistema, electrónicos, etc.

Los problemas que se encontraron según la encuesta (Ver Anexo A) realizada a los funcionarios del INLASA son los siguientes:

- Demoras en atención a solicitudes.
- Inexistencia de Respaldos que garanticen su resguardo.
- Procesos administrativos incompletos.
- Inexistencia de normas de clasificación de Documentos.

- Gestión administrativa deficiente.

En síntesis la institución no cuenta con un sistema optimizado para el manejo de recepción, registro y control de documentos. Es así que nace el problema principal al que se debe dar solución con la implementación del sistema:

¿Cómo ayudará el Sistema de Información de Gestión y Control Documental en el manejo de los Documentos Institucionales y anexos al Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA"?

1.4 OBJETIVOS

Los objetivos son deducidos según el Árbol de Objetivos. (Ver Anexo B)

1.4.1 OBJETIVO GENERAL

Implementar un Sistema de Información que optimice el control y seguimiento de documentos en formato digital para el Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA".

1.4.2 OBJETIVOS ESPECÍFICOS.

- Realizar procesos de registro, digitalización, clasificación de documentos.
- Diseñar procesos de consulta, préstamo y transferencia de documentos.
- Mejorar la atención de los procesos administrativos.
- Mejorar la Gestión Administrativa.
- Clasificación de Documentos según normas archivísticas.

1.3 JUSTIFICACIÓN.

1.3.1 JUSTIFICACIÓN SOCIAL

Se justifica socialmente porque se realizará un análisis y optimización de los procesos que ocasionan retardación y pérdida en los documentos de correspondencia y trámites administrativos. Asimismo brindará transparencia al proceso de circulación y procesamiento de los documentos internos y externos.

El software brindará apoyo en procesos de registro, búsqueda, consulta y descripción, beneficiará directamente al responsable del área de Archivo, a los solicitantes de documentos y a personas ajenas al instituto. El mismo permitirá mejor accesibilidad de información rápida y oportuna al momento que se la requiera. El sistema contribuirá a toda persona u organización que solicite documentación logrando una mejor difusión de los mismos.

1.3.2 JUSTIFICACIÓN TÉCNICA.

El personal y dependientes del Área de Archivo del Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA" necesitan herramientas que faciliten la recepción, registro, manejo y acceso de documentos.

El Proyecto a desarrollar, se realiza con la implementación de estándares y normas internacionales de descripción archivística en el Área de Archivo Central, optimizando así los servicios que presta el mismo.

El Instituto cuenta con los requerimientos de Hardware y Software, de esta manera el desarrollo de este proyecto no exige condiciones técnicas con las que no dispone actualmente la entidad, pues existen los medios necesarios para realizar el análisis, diseño, implementación y mantenimiento del Sistema para su funcionamiento. El sistema se desarrollara con tecnología cliente/servidor para compartir o distribuir la información utilizando medidas de seguridad para el resguardo adecuado de la información.

1.3.3 JUSTIFICACIÓN ECONÓMICA

En la actualidad el acceso a una computadora e internet son más económicos que en décadas pasadas, por tal razón lejos de pensar en el valor económico que conlleva un sistema complejo de Archivo se enfatiza el valor cualitativo que ésta genera.

El proyecto presenta una justificación económica, puesto que desarrollar el Sistema de Información de Gestión y Control Documental permitirá mejorar los procesos relacionados con la organización y localización de documentos ahorrando recursos y tiempo para de esta manera brindar mayor accesibilidad de la documentación resguardada por el Área de Archivo Central del Instituto.

A continuación detallamos el requerimiento en tecnología necesaria para la implementación del presente proyecto:

- Computadora con características descritas en la tabla 1.1:

Procesador	Pentium 4 o superior
Memoria Ram	256 Mb o superior
Tarjeta Madre	Intel Integrada
Monitor	Pantalla de 17 “ cualquier marca
Disco Duro	20 Gb o superior
Lector de DVD	LG multiformato o cualquier marca
Combo Case	Incluye Teclado y Mouse

Tabla 1.1 Características de la computadora
Fuente: Elaboración Propia

- Escáner a elección, con preferencia HP Scanjet N9120 (L2683A). que es para uso en documentos de gran cantidad de páginas, que además cuente con el software necesario para la conexión con sistemas operativos actuales.

1.4 LÍMITES Y ALCANCES

1.4.1 DELIMITACIÓN TEMÁTICA.

El presente proyecto se desarrollará en el Área de Archivo Central del Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA", el sistema solo se enfocara en la información de documentos de archivo, no tendrá control en la derivación de documentos hacia otras unidades pues este proceso ya cuenta con un software de correspondencia institucional.

1.4.2 DELIMITACIÓN ESPACIAL.

El Proyecto de Grado será desarrollado en el Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA" del Departamento de La Paz específicamente en el Área de Archivo Central, no contará con conexiones a otros sistemas de información que tuviera la institución u otros que se implementen.

1.4.3 DELIMITACIÓN TEMPORAL.

Se estima que el Proyecto de Grado se desarrollará entre el 10 de Septiembre y el 25 de Noviembre del presente año, tiempo establecido para el análisis de requerimientos, lista de tareas necesarias para llevar a cabo, presentación de demos y procesos que necesiten ser perfeccionados y por ultimo realizar las pruebas del sistema y el costo beneficio que se obtenga en la evaluación del sistema.

1.4.4 ALCANCES

Desarrollar un prototipo del sistema, el cual tendrá como dominio específico el manejo de documentos relacionados al Área de Archivo, una interfaz del sistema que permita comunicarse con el usuario de manera interactiva durante el proceso de registro, consulta y gestión de documentos, dar respuesta de forma sistemática y ordenada a los distintos requerimientos que tenga el usuario, estar orientado a aquellos usuarios con o sin conocimiento en el manejo de sistemas ya que su manejo será sencillo.

El presente sistema brindará soporte a la gestión de los documentos resguardados mediante normas de descripción estandarizadas extendiéndose su uso por motivos relacionados con la necesidad de uniformizar las descripciones para propiciar el intercambio de información y para mejorar la

accesibilidad a los documentos. Se elaborarán módulos de administración, descripción, digitalización y búsquedas que permitan un acceso rápido a los documentos.

1.5 APORTES

Con la implementación del Sistema de Información de Gestión y Control Documental la institución contará con nuevas herramientas que faciliten el trabajo de tal forma que mejoren el desempeño y rendimiento de las funciones que realicen a diario evitando el retardo en procesos de circulación de documentos, atención a solicitudes de documentos resguardados, coadyuvando a la organización y el acceso a la documentación del Área de Archivo Central dentro de la institución.

1.6 METODOLOGÍA

Considerando que el logro de los objetivos de una investigación requiere de la utilización de métodos adecuados que posibiliten aquello, es importante poner especial énfasis en la selección de tales instrumentos, puesto que permitirán realizar un trabajo sistemático y acorde a un cronograma previamente establecido para cada una de las etapas de la investigación, desde la revisión bibliográfica, recolección de información, procesamiento y análisis de resultados.

Existen varios métodos para el desarrollo de un sistema a continuación describiremos la Metodología SCRUM que es el que se eligió para el desarrollo del sistema.

1.6.1 METODOLOGÍA SCRUM

La metodología SCRUM es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa. Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación. Esta metódica de trabajo promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades, a continuación en la figura 1.1 veremos la estructura central de la metodología SCRUM.

Figura 1.1 Estructura Central de SCRUM
Fuente: Ken Schwaber (2004)

Con la metodología Scrum el cliente se entusiasma y se compromete con el proyecto dado que lo ve crecer iteración a iteración. Asimismo le permite en cualquier momento realinear el software con los objetivos de negocio de su empresa, ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración sin ningún problema.

1.6.2 BENEFICIOS DE SCRUM

Los beneficios de utilizar la metodología Scrum son los siguientes:

- **Cumplimento de expectativas:** El cliente establece sus expectativas indicando el valor que le aporta cada requisito del proyecto, el equipo los estima y con esta información el Product Owner establece su prioridad. De manera regular en las demos de Sprint el Product Owner comprueba que efectivamente los requisitos se han cumplido y transmite su feedback al equipo.
- **Flexibilidad a cambios:** Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado. La metodología está diseñada para adaptarse a los cambios de requerimientos que conllevan los proyectos complejos.

- **Reducción del Time to Market:** El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo.
- **Mayor calidad del software:** La metódica de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.
- **Mayor productividad:** Se consigue entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
- **Maximiza el retorno de la inversión:** Producción de software únicamente con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión.
- **Predicciones de tiempos:** Mediante esta metodología se conoce la velocidad media del equipo por sprint con lo que consecuentemente es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está en el Backlog.
- **Reducción de riesgos:** El hecho de llevar a cabo las funcionalidades de más valor en primer lugar y de conocer la velocidad con que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

1.6.3 PROCESO Y ROLES DE SCRUM

1.6.3.1 EL PROCESO SCRUM

El desarrollo se realiza de forma iterativa e incremental, cada iteración denominada Sprint tiene una duración preestablecida de entre 2 y 4 semanas obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. En cada nuevo Sprint se va ajustando la funcionalidad ya construida y se añaden nuevas prestaciones priorizándose siempre aquellas que aporten mayor valor de negocio. A continuación se describe cada uno de los procesos de Scrum en la figura 1.2 donde lo veremos de forma gráfica.

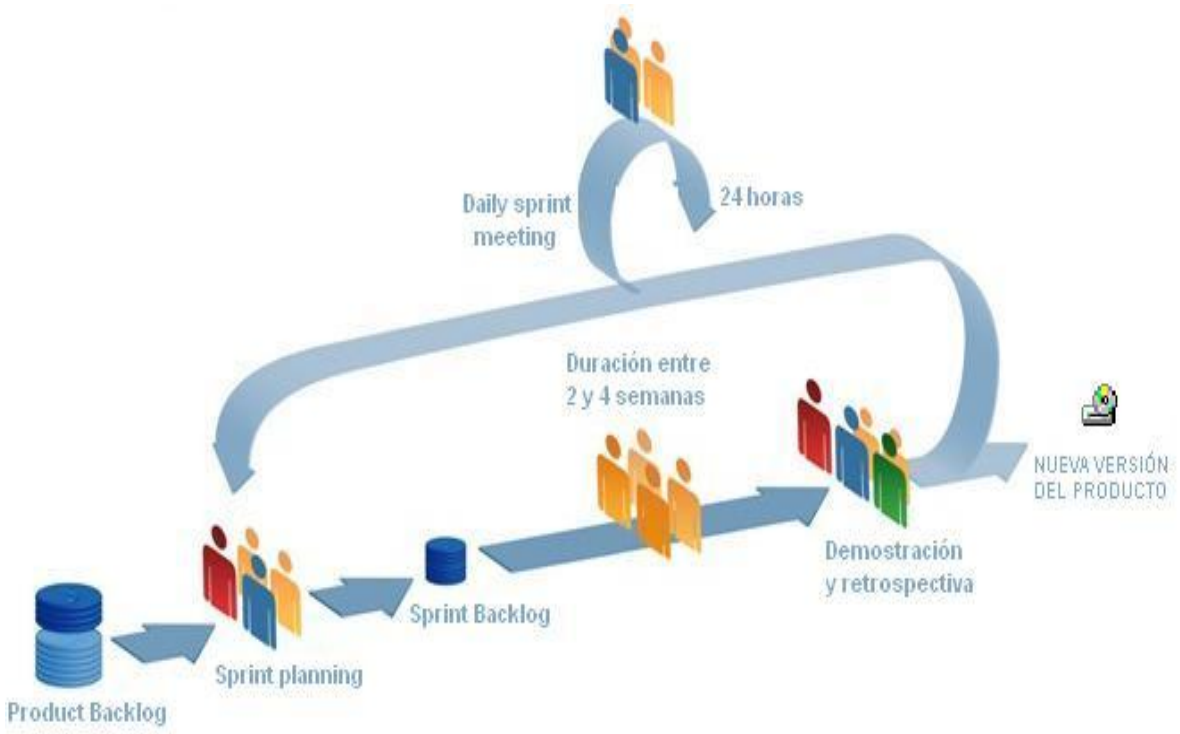


Figura 1.2 Proceso Scrum
Fuente: Ken Schwaber (2004)

- **Product Backlog (Lista de Requerimientos):** Es el conjunto de requisitos denominados historias de usuario descritos en un lenguaje no técnico y priorizados por valor de negocio o lo que es lo mismo por retorno de inversión considerando su beneficio y coste. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares.
- **Sprint Planning (Planeación de la Iteración):** Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint, para en una segunda parte de la reunión decidir y organizar cómo lo va a conseguir.
- **Sprint (Iteración de Trabajo):** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo.
- **Sprint Backlog (Lista de Iteraciones):** Lista de las tareas necesarias para llevar a cabo las historias del sprint.

- **Daily sprint meeting (Reuniones diarias):** Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada. Cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos.
- **Demo y retrospectiva:** Reunión que se celebra al final del sprint y en la que el equipo presenta las historias conseguidas mediante una demonstración del producto, después de la retrospectiva el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos.

1.6.3.2 ROLES DE SCRUM

En Scrum, el equipo se focaliza en construir software de calidad. La gestión de un proyecto Scrum se centra en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en vencer cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo.

El equipo Scrum está formado por los siguientes roles:

- **Scrum master:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el Product Owner para maximizar la inversión.
- **Product owner:** Representante de los accionistas y clientes que usan el software. Se focaliza en la parte de negocio y él es responsable de la inversión económica del proyecto. Traslada la visión del proyecto al equipo, formaliza las prestaciones en historias a incorporar en el Product Backlog y las re-prioriza de forma regular.
- **Team:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.

La figura 1.3 muestra a los actores que componen los roles de Scrum de forma general, en la que se puede observar la jerarquía de trabajo.

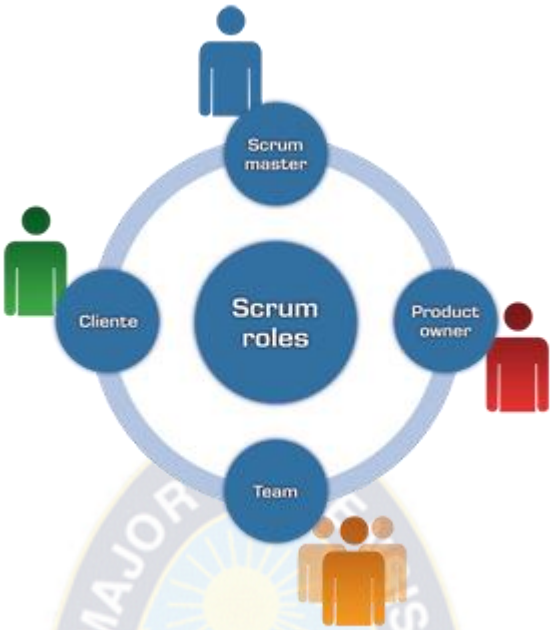


Figura 1.3 Roles de Scrum
Fuente: Ken Schwaber (2004)

1.7 MARCO TEÓRICO

1.7.1 TÉCNICAS DE INFORMACIÓN

Se utilizarán las siguientes técnicas de recolección de información:

- **Entrevista estructurada:** Estará dirigida al personal ejecutivo y administrativo del INLASA con la finalidad de conocer su percepción respecto de la necesidad de crear un Sistema de Información para la Gestión y Control de Documentos para su Institución. Esta forma de entrevista se realizara sobre la base de un cuestionario previamente formulado y preparado y estrictamente normalizado, a través de una lista de preguntas que se establecieron con anterioridad. Considerando que la población es reducida no se utilizará ningún criterio de muestreo, sino que se considerará a la totalidad de sujetos de la población, es decir a los funcionarios del Instituto.
- **Observación documental**: Esta técnica permitirá recopilar información bibliográfica relativa al proceso de gestión y otros temas inherentes al problema de investigación, así como documentos, disposiciones legales y otros relativos a la administración pública. Esta técnica se refiere a la

investigación bibliográfica realizada en diversos tipos de escritos, tales como libros, documentos académicos, actas o informes, revistas, etc. Población y Muestra, la población de estudio estará conformada por funcionarios del nivel ejecutivo y administrativo del Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón".

1.7.2 HERRAMIENTAS

1.7.2.1 OOHDM- OBJECT-ORIENTED HYPERMEDIA DESIGN METHOD

OOHDM es una metodología de desarrollo propuesta por Rossi y Schwabe (Rossi, 1996) para la elaboración de aplicaciones multimedia. OOHDM está basada en HDM, en el sentido de que toma muchas de las definiciones, sobre todo en los aspectos de navegación, planteadas en el modelo de HDM. Sin embargo, OOHDM supera con creces a su antecesor, ya que no es simplemente un lenguaje de modelado, sino que define unas pautas de trabajo, centrado principalmente en el diseño, para desarrollar aplicaciones multimedia de forma metodológica. OOHDM ha evolucionado bastante desde su nacimiento. Actualmente está siendo utilizado por sus autores para el desarrollo de aplicaciones en la web (Schwabe 1996). OOHDM como ya se ha comentado es una metodología de desarrollo para aplicaciones multimedia. Antes de comenzar a detallar cada una de las fases que propone, es necesario resaltar algunas de sus características.

La primera de ellas es que OOHDM está basada en el paradigma de la orientación a objetos, en esto se diferencia de su antecesor HDM, a la hora de la representación de las clases y los diagramas OOHDM utilizan la nomenclatura propuesta por OMT. Sin embargo en las últimas versiones ya se ha asumido UML.

Otra característica de OOHDM es que a diferencia de HDM no sólo propone un modelo para representar a las aplicaciones multimedia, sino que propone un proceso predeterminado para el que indica las actividades a realizar y los productos que se deben obtener en cada fase del desarrollo, sin embargo OOHDM no puede considerarse una metodología en el amplio sentido ya que aunque se detalla el proceso a seguir en lo que sería el diseño de la aplicación no toma parte en otras fases como pueden ser la captura de requisitos o el análisis. Según OOHDM estas fases son idénticas a las que se deben realizar en la propuesta de OMT.

Fundamentalmente OOHDM toma como partida el modelo de clases que se obtiene en la fase de diseño de objetos de OMT o en sus últimas versiones en el análisis del Proceso Unificado de UML. A este modelo lo denomina modelo conceptual, el proceso anterior a éste es decir el análisis y el diseño de sistemas, sería idéntico al que se realizaría en el desarrollo de un sistema clásico según OMT o UML. Partiendo de este modelo conceptual OOHDM propone ir añadiendo características que permitan incorporar a esta representación del sistema todos los aspectos propios de las aplicaciones multimedia.

En una segunda etapa de diseño, se parte de ese modelo conceptual y se añade a éste todos los aspectos de navegación, obteniéndose un nuevo modelo de clases denominado modelo navegacional. Por último este modelo sirve como base para definir el modelo de interfaz abstracta que representa la visión que tendrá cada usuario del sistema.

1.7.2.2 FASES DE OOHDM

En OOHDM se proponen 4 fases de desarrollo:

- **Diseño Conceptual:** Se construye un modelo orientado a objetos que represente el dominio de la aplicación usando las técnicas propias de la orientación a objetos. La finalidad principal durante esta fase es capturar el dominio semántico de la aplicación en la medida de lo posible, teniendo en cuenta el papel de los usuarios y las tareas que desarrollan. El resultado de esta fase es un modelo de clases relacionadas que se divide en subsistemas.
- **Diseño Navegacional:** En OOHDM una aplicación se ve a través de un sistema de navegación. En la fase de diseño navegacional se debe diseñar la aplicación teniendo en cuenta las tareas que el usuario va a realizar sobre el sistema. Para ello, hay que partir del esquema conceptual desarrollado en la fase anterior. Hay que tener en cuenta que sobre un mismo esquema conceptual se pueden desarrollar diferentes modelos navegacionales (cada uno de los cuales dará origen a una aplicación diferente).

  La estructura de navegación de una aplicación hipermedia está definida por un esquema de clases de navegación específica, que refleja una posible vista elegida. En OOHDM hay una serie de

clases especiales predefinidas, que se conocen como clases navegacionales: Nodos, Enlaces y Estructuras de acceso, que se organizan dentro de un Contexto Navegacional. Mientras que la semántica de los nodos y los enlaces son comunes a todas las aplicaciones hipermedia, las estructuras de acceso representan diferentes modos de acceso a esos nodos y enlaces de forma específica en cada aplicación.

1- Nodos: Los nodos son contenedores básicos de información de las aplicaciones hipermedia. Se definen como vistas orientadas a objeto de las clases definidas durante el diseño conceptual usando un lenguaje predefinido y muy intuitivo, permitiendo así que un nodo sea definido mediante la combinación de atributos de clases diferentes relacionadas en el modelo de diseño conceptual. Los nodos contendrán atributos de tipos básicos (donde se pueden encontrar tipos como imágenes o sonidos) y enlaces.

2- Enlaces: Los enlaces reflejan la relación de navegación que puede explorar el usuario. Ya sabemos que para un mismo esquema conceptual puede haber diferentes esquemas navegacionales y los enlaces van a ser imprescindibles para poder crear esas vistas diferentes.

OOHDM hace una definición de clase enlace que contiene un único atributo. Las clases enlaces sirven para especificar los atributos de enlaces y estos a su vez para representar enlaces entre clases nodos o incluso entre otros enlaces. En cualquier caso, el enlace puede actuar como un objeto intermedio en un proceso de navegación o como un puente de conexión entre dos nodos.

3- Estructuras de Acceso: Las estructuras de acceso actúan como índices o diccionarios que permiten al usuario encontrar de forma rápida y eficiente la información deseada. Los menús, los índices o las guías de ruta son ejemplos de estas estructuras. Las estructuras de acceso también se modelan como clases, compuestas por un conjunto de referencias a objetos que son accesibles desde ella y una serie de criterios de clasificación de las mismas.

4- Contexto Navegacional: Para diseñar bien una aplicación hipermedia, hay que prever los caminos que el usuario puede seguir, así es como únicamente podremos evitar información redundante o que el usuario se pierda en la navegación. En OOHDM un contexto navegacional está compuesto por un conjunto de nodos, de enlaces, de clases de contexto y de otros contextos navegacionales. Estos son introducidos desde clases de navegación (enlaces, nodos o estructuras de acceso), pudiendo ser definidas por extensión o de forma implícita.

5- Clase de Contexto: Es otra clase especial que sirve para complementar la definición de una clase de navegación. Por ejemplo, sirve para indicar qué información está accesible desde un enlace y desde dónde se puede llegar a él.

- **Diseño de Interfaz Abstracto:** Una vez definida la estructura navegacional, hay que prepararla para que sea perceptible por el usuario y esto es lo que se intenta en esta fase. Esto consiste en definir qué objetos de interfaz va a percibir el usuario y en particular el camino en el cuál aparecerán los diferentes objetos de navegación, qué objeto de interfaz actuará en la navegación, la forma de sincronización de los objetos multimedia y el interfaz de transformaciones. Al haber una clara separación entre la fase anterior y esta fase, para un mismo modelo de navegación se pueden definir diferentes modelos de interfaces, permitiendo así que el interfaz se ajuste mejor a las necesidades del usuario.
- **Implementación:** Después de determinar de forma abstracta los requerimientos pasamos a mostrar de forma gráfica el comportamiento de los anteriores modelos.

1.7.3 PHP

PHP tal y como se conoce hoy en día es en realidad el sucesor de un producto llamado PHP/FI. Creado en 1994 por Rasmus Lerdorf, la primera encarnación de PHP era un conjunto simple de ficheros binarios Common Gateway Interface (CGI) escritos en el lenguaje de programación C. Originalmente utilizado para rastrear visitas de su currículum online, llamó al conjunto de scripts "Personal Home Page Tools", más frecuentemente referenciado como "PHP Tools". Con el paso del tiempo se quiso más funcionalidad, y Rasmus reescribió PHP Tools, produciendo una implementación más grande y rica.

Este nuevo modelo fue capaz de interaccionar con bases de datos, y mucho más, proporcionando un entorno de trabajo sobre cuyos usuarios podían desarrollar aplicaciones web dinámicas sencillas tales como libros de visitas. En junio de 1995, Rasmus publicó el código fuente de PHP Tools, lo que permitió a los desarrolladores usarlo como considerasen apropiado. Esto también permitió -y animó- a los usuarios a proporcionar soluciones a los errores del código, y generalmente a mejorarlo.

En septiembre de ese mismo año, Rasmus amplió PHP y -por un corto periodo de tiempo- abandonó el nombre de PHP. Ahora, refiriéndose a las herramientas como FI (abreviatura de "Forms Interpreter"), la nueva implementación incluía algunas de las funciones básicas de PHP tal y como la conocemos hoy. Tenía variables como las de Perl, interpretación automática de variables de formulario y sintaxis incrustada HTML. La sintaxis por sí misma era similar a la de Perl, aunque mucho más limitada, simple y algo inconsistente.

De hecho, para embeber el código en un fichero HTML, los desarrolladores tenían que usar comentarios de HTML. Aunque este método no era completamente bien recibido, FI continuó gozando de expansión y aceptación como una herramienta CGI pero todavía no completamente como lenguaje. Sin embargo, esto comenzó a cambiar al mes siguiente; en octubre de 1995 Rasmus publicó una versión nueva del código. Recordando el nombre PHP, ahora era llamado Personal Home Page Construction Kit," y fue la primera versión que presumía de ser en aquel momento considerada como una interfaz de scripts avanzada. El lenguaje fue deliberadamente diseñado para asemejarse a C en estructura, haciéndolo una adopción sencilla para desarrolladores familiarizados con C, Perl, y lenguajes similares. Habiendo sido así bastante limitado a sistemas UNIX y compatibles con POSIX, el potencial para una implementación de Windows NT estaba siendo explorada.

El código fue completamente rehecho de nuevo en abril de 1996, combinando los nombres de versiones anteriores, Rasmus introdujo PHP/FI. Esta implementación de segunda generación comenzó realmente a desarrollar PHP desde un conjunto de herramientas dentro de un lenguaje de programación de derecho propio, incluía soporte interno para DBM, mSQL, y bases de datos Postgres95, cookies, soporte para funciones definidas por el usuario, y mucho más. Ese mes de junio, PHP/FI brindó una versión 2.0. Sin embargo, un interesante hecho sobre esto, es que sólo había una única versión completa de PHP 2.0. Cuando finalmente pasó de la versión beta en noviembre de 1997, el motor de análisis subyacente ya estaba siendo reescrito por completo.
Aunque vivió una corta vida de desarrollo, continuó gozando de un crecimiento de popularidad en el aún joven mundo del desarrollo. En 1997 y 1998, PHP/FI tenía un culto de varios miles de usuarios en todo el mundo. Una encuesta de Netcraft en mayo de 1998 indicó que cerca de 60,000 dominios reportaron que tenían cabeceras que contenían "PHP", indicando en efecto que el servidor host lo

tenía instalado. Este número se correspondía con aproximadamente el 1% de todos los dominios de Internet del momento. A pesar de estas impresionantes cifras, la maduración de PHP/FI estaba condenada por limitaciones; mientras había varios contribuidores menores, aún era desarrollado principalmente por un individuo.

PHP 3.0 fue la primera versión que más se parecía al PHP que existe hoy. Encontrando todavía PHP/FI 2.0 ineficiente y falto de las características que necesitaban para impulsar una aplicación de comercio electrónico que estaban desarrollando para un proyecto de universidad, Andi Gutmans y Zeev Suraski, de Tel Aviv, Israel, comenzaron otra nueva versión del analizador subyacente en 1997. Proponiendo Rasmus online, discutieron varios aspectos de la implementación actual y su redesarrollo de PHP. En un esfuerzon para mejorar el motor y comenzar a construir sobre la base de usuario de PHP/FI existente, Andi, Rasmus y Zeev decidieron colaborar en el desarrollo de un nuevo e independiente lenguaje de programación.

Una de las mejores características de PHP 3.0 era su gran extensibilidad. Además de proveer a los usuarios finales de una interfaz madura para múltiples bases de datos, protocolos, y APIs, la sencillez de ampliar el lenguaje mismo atrajo a docenas de desarrolladores que presentaron variedad de módulos. Podría decirse que esta fue la clave para el tremendo éxito de PHP 3.0. Otras características clave introducidas en PHP 3.0 incluían el soporte para programación orientada a objetos y una sintaxis de lenguaje mucho más potente y consistente.

En junio de 1998, con muchos nuevos desarrolladores de todo el mundo unidos al esfuerzo, PHP 3.0 fue anunciado por el nuevo Equipo de Desarrollo de PHP como el sucesor oficial de PHP/FI 2.0. El desarrollo activo de PHP/FI 2.0, que estaba casi parado desde noviembre del año anterior, fue oficialmente finalizado. Después de aproximadamente nueve meses de pruebas públicas, cuando el anuncio de la versión oficial de PHP 3.0 vino, ya estaba instalado en más de 70,000 dominios de todo el mundo, y ya no estaba limitado a sistemas operativos compatibles con POSIX. Una relativamente pequeña parte de de los dominios que tenían instalado PHP estaban albergados en servidores que ejecutaban Windows 95, 98, y NT, y Macintosh. En este punto, PHP 3.0 estaba instalado en aproximadamente el 10% de los servidores web de Internet.

En el invierno de 1998, poco después del lanzamiento oficial de PHP 3.0, Andi Gutmans y Zeev Suraski comenzaron a trabajar en una nueva versión del núcleo de PHP. Los objetivos de diseño fueron mejorar la ejecución de aplicaciones complejas y mejorar la modularidad del código base de PHP. Estas aplicaciones se hicieron posibles por las nuevas características de PHP 3.0 y el apoyo de una gran variedad de bases de datos y APIs de terceros, pero PHP 3.0 no estaba diseñado para un mantenimiento tan complejo de aplicaciones eficientemente.

El nuevo motor, apodado 'Motor Zend' (proviene de sus nombres de pila, Zeev y Andi), alcanzó estos objetivos de diseño satisfactoriamente, y se introdujo por primera vez a mediados de 1999. PHP 4.0, basado en este motor, y asociado con un gran rango de nuevas características adicionales, fue oficialmente publicado en Mayo del 2000, casi dos años después que su predecesor. Además de la mejora de rendimiento de esta versión, PHP 4.0 incluía otras características clave como el soporte para la mayoría de los servidores Web, sesiones HTTP, buffers de salida, formas más seguras de controlar las entradas de usuario y muchas nuevas construcciones de lenguaje.

PHP 5 fue lanzado en Julio del 2004 después de un largo desarrollo y varios pre-releases. Está básicamente impulsado por su núcleo, Zend Engine *2.0* que contiene un nuevo modelo de objetos y docenas de nuevas opciones.

El equipo de desarrollo de PHP incluye docenas de desarrolladores, así como docenas de otras personas trabajando en proyectos relacionados y de soporte para PHP, como PEAR, PECL, y documentación, y una infraestructura en red subyacente de más de cien servidores web individuales en seis de los siete continentes del mundo. Aunque es solo una estimación basada en estadísticas de años anteriores, es seguro suponer que PHP ahora está instalado en diez o quizá cien millones de dominios en todo el mundo.

1.7.3.1 CARACTERÍSTICAS DE PHP

Orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una base de datos. Es considerado un lenguaje fácil de aprender ya que en su desarrollo se simplificaron

distintas especificaciones, como es el caso de la definición de las variables primitivas ejemplo que se hace evidente en el uso de php arrays.

El código fuente escrito en PHP es invisible al navegador web y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.

Capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL. Capacidad de expandir su potencial utilizando módulos. Posee una amplia documentación en su sitio web 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. Permite aplicar técnicas de programación orientada a objetos. No requiere definición de tipos de variables aunque sus variables se pueden evaluar también por el tipo que estén manejando en tiempo de ejecución. Tiene manejo de excepciones desde PHP5.

Si bien PHP no obliga a quien lo usa a seguir una determinada metodología a la hora de programar, aun haciéndolo, el programador puede aplicar en su trabajo cualquier técnica de programación o de desarrollo que le permita escribir código ordenado, estructurado y manejable. Un ejemplo de esto son los desarrollos que en PHP se han hecho del patrón de diseño Modelo Vista Controlador (MVC), que permiten separar el tratamiento y acceso a los datos, la lógica de control y la interfaz de usuario en tres componentes independientes. Debido a su flexibilidad ha tenido una gran acogida como lenguaje base para las aplicaciones WEB de manejo de contenido, y es su uso principal.

1.9.4 MYSQL

Mysql es un sistema de gestión de bases de datos (SGBD) multiusuario, multiplataforma y de código abierto bajo licencia GNU como también bajo una variedad de acuerdos propietarios. MySQL pertenecía a la compañía sueca MySQL AB, que tenía casi todos los derechos del código fuente y se encargaba de desarrollar y mantener el sistema, vender soporte y servicios, como

también las licencias para usar MySQL. Actualmente pertenece a Oracle Corporation tras comprarla en 2008.

Existen versiones de pago con funcionalidades adicionales. Para julio de 2013 era el segundo SGBD más utilizado del mundo. Su nombre es una combinación de "My", nombre de la hija del fundador Michael Widenius, y "SQL" (Structured Query Language).

1.9.4.1 USO DE MYSQL

MySQL es muy utilizado en bases de datos para aplicaciones web, en especial en la combinación LAMP (Linux, Apache, MySQL, Perl/PHP/Python) y otras como MAMP, WAMP, etc. Probablemente la combinación más popular sea MySQL y el lenguaje PHP. Algunas aplicaciones populares que emplean base de datos MySQL son: TYPO3, MODx, Joomla, WordPress, phpBB, MyBB, Drupal, entre otras.

1.9.4.2 CARACTERÍSTICAS DE MYSQL

- MySQL está escrito en C y C++
- Emplea el lenguaje SQL para consultas a la base de datos.
- MySQL Server está disponible como freeware bajo licencia GPL.
- MySQL Enterprise es la versión por suscripción para empresas, con soporte las 24 horas.
- Trabaja en múltiples plataformas como: AIX, BSDi, FreeBSD, HP-UX, GNU/Linux, Mac OS X, NetBSD, Novell NetWare, OpenBSD, OS/2 Warp, QNX, SGI IRIX, Solaris, SunOS, SCO OpenServer, SCO UnixWare, Tru64, Microsoft Windows.
- Algunos de los lenguajes que se pueden usar son los siguientes: C, C++, Visual Net, PHP, entre otros.

1.9.5 ADOBE DREAMWEAVER

Adobe Dreamweaver es una aplicación en programa de estudio, está destinada a la construcción, diseño y edición de sitios, vídeos y aplicaciones Web basados en estándares. Creado por Macromedia es uno de los programas más utilizados en el sector del diseño y la programación web

por sus funcionalidades, su integración con otras herramientas como Adobe Flash y recientemente por su soporte de los estándares del World Wide Web Consortium.

Tiene soporte tanto para edición de imágenes como para animación a través de su integración con otras. Hasta la versión MX, fue duramente criticado por su escaso soporte de los estándares de la web, ya que el código que generaba era con frecuencia sólo válido para Internet Explorer y no validaba como HTML estándar. Esto se ha ido corrigiendo en las versiones recientes, se vende como parte de Adobe Creative Suite. A partir de la compra de Macromedia por parte de Adobe.

La gran ventaja de este editor sobre otros es su gran poder de ampliación y personalización, puesto que en este programa sus rutinas hechas en Javascript-C lo que le ofrece una gran flexibilidad en estas materias. Esto hace que los archivos del programa no sean instrucciones de C++ sino rutinas de Javascript que hace que sea un programa muy fluido y todo ello permite que programadores y editores web hagan extensiones para su programa y lo pongan a su gusto.

1.9.6 XAMPP

Es un servidor independiente de plataforma software libre, que consiste principalmente en el sistema de gestión de bases de datos MySQL, el nombre proviene del acrónimo de X para cualquiera de los diferentes sistemas operativos, Apache, MySQL, PHP, Perl. Desde la versión "5.6.15", XAMPP cambió la base de datos de MySQL A MariaDB. El cual es un fork de MySQL con licencia GPL.

El programa se distribuye bajo la licencia GNU y actúa como un servidor web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris y Mac OS X.

1.9.6.1 CARACTERÍSTICAS Y REQUISITOS XAMPP

XAMPP solamente requiere descargar y ejecutar un archivo exe con unas pequeñas configuraciones en alguno de sus componentes que el servidor Web necesitará. XAMPP se actualiza regularmente para incorporar las últimas versiones de Apache/MySQL/PHP y Perl. También incluye otros

módulos como OpenSSL y phpMyAdmin. Para instalar XAMPP se requiere solamente una pequeña fracción del tiempo necesario para descargar y configurar los programas por separado.

Los diseñadores de XAMPP fueron Baiker y Anthony Coorporation los cuales solo pretendían su uso como una herramienta de desarrollo, para permitir a los diseñadores de sitios webs y programadores testear su trabajo en sus propios ordenadores sin ningún acceso a Internet.

1.9.7 REQUERIMIENTOS DEL SOFTWARE

Los requerimientos para el correcto funcionamiento de XAMPP son los siguientes:

- MySQL 5.0.41.
- PHP 5.
- phpMyAdmin 2.10.1.
- Sql Yog: es un editor de base de datos con el cual simularemos las funciones del sistema, apto para la elaboración de bakups.
- Bootstrap templates: es un framework o conjunto de herramientas de Código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como, extensiones de JavaScript opcionales adicionales.
- Angular JS: Angular ha sido definido como un framework Model-View-Whatever, es decir Modelo-Vista-*Lo-que-sea*, la aplicación se organiza entorno a Controllers responsables de construir los ViewModels que se enlazarán a las vistas, en angular el controller puede construir un ViewModel que luego toma el control completo de las operaciones realizadas en la vista, por lo que básicamente actúan como factorías de ViewModels.

CAPITULO II

MARCO TEÓRICO

2.1 INGENIERÍA DE SOFTWARE

La evolución de la disciplina de la ingeniería de software ha traído consigo propuesta diferentes para mejorar los resultados del proceso de construcción. Las metodologías tradicionales hacen énfasis en la planeación, mientras las metodologías agiles hacen énfasis en la adaptabilidad del proceso, delinean las principales propuestas presentes en la literatura. De manera paralela, el tema de modelos para el mejoramiento de los procesos de desarrollo ocupa un lugar importante en la búsqueda de la metodología adecuada para producir software de calidad en cualquier contexto de desarrollo (Letelier & Penadés, 2005).

Esta disciplina trasciende la actividad de programación que es el pilar fundamental a la hora de crear una aplicación. El ingeniero de software se encarga de toda la gestión del proyecto para que éste se pueda desarrollar en un plazo determinado y con el presupuesto previsto. La ingeniería de software por lo tanto incluye el análisis previo de la situación, el diseño del proyecto, el desarrollo del software, las pruebas necesarias para confirmar su correcto funcionamiento y la implementación del sistema.

2.1.1 METODOLOGÍAS DE DESARROLLO

Según (Letelier & Penadés, 2005) existen numerosas propuestas de metodología para desarrollar software, tradicionalmente estas metodologías se centran en el control del proceso, estableciendo rigurosamente las actividades, herramientas y notaciones al respecto, dado estas reglas las metodologías se caracterizan por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas.

Este enfoque no resulta ser el más adecuado para muchos de los proyectos actuales donde el entorno del sistema es muy cambiante y en donde se exige reducir drásticamente los tiempos de desarrollo pero manteniendo una alta calidad. En febrero de 2001, tras una reunión celebrada en Utah-EEUU,

nace el término ágil aplicado al desarrollo de software.

Su objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto. Se pretendía ofrecer una alternativa a los procesos de desarrollo de software tradicionales, caracterizados por ser rígidos y dirigidos por la documentación que se genera en cada una de las actividades desarrolladas.

2.1.2 EL MANIFIESTO ÁGIL

Las metodologías ágiles son una familia de metodologías, no un enfoque individual de desarrollo de software. En 2001 destacadas en el campo del desarrollo ágil llamado entonces metodologías de peso ligero, se juntaron para tratar el tema de la unificación de sus metodologías, a raíz de esto crearon el Manifestó Ágil, ampliamente considerado como la definición canónica del desarrollo ágil (Inteco, 2000).

Según el Manifiesto se valora:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente.
- Desarrollar software que funciona más que conseguir una buena documentación. La regla es no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante. Estos documentos deben ser cortos y centrarse en lo fundamental.
- La colaboración con el cliente más que la negociación de un contrato. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a los largo del proyecto, cambios en los requisitos, en la tecnología, en el equipo, etc., determina también el éxito o fracaso del mismo. Por lo tanto la planificación no debe ser estricta sino flexible y abierta.

2.1.3 PRINCIPIOS DEL MANIFIESTO ÁGIL

Según (Inteco, 2000) los valores anteriores inspiran los doce principios del manifiesto que 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 tiene que ver con el proceso a seguir y con el equipo de desarrollo, en cuanto a metas a seguir y organización del mismo.

La prioridad más alta es satisfacer al cliente a través de la entrega temprana y continua de software de valor.

- Los cambios en los requisitos son bienvenidos, incluso tarde, en el desarrollo. Los procesos ágiles aprovechan los cambios como ventaja competitiva del cliente.
- Entregar software que funcione con frecuencia, desde un par de semanas hasta un par de meses, con preferencia de escalas de tiempo cortas con el menor intervalo de tiempo posible.
- Las personas de negocio y los desarrolladores deben trabajar juntos diariamente durante todo el proyecto.
- Construir proyectos alrededor de individuos motivados. Darles el entorno y el soporte necesario, y confiar en que ellos harán el trabajo.
- El método más eficiente y efectivo de hacer llegar información a o dentro de un equipo de desarrollo es en una conversación cara a cara.
- Software que funciona es la medida principal del progreso.
- Los procesos ágiles promueven desarrollo sostenible. Los sponsors, desarrolladores y usuarios deberían ser capaces de mantener un ritmo constante indefinido.
- La atención continua a la excelencia técnica y los buenos diseños aumentan la agilidad.
- Simplicidad, el arte de maximizar la cantidad de trabajo que no hay que hacer, es esencial.
- Las mejores arquitecturas, requisitos y diseño surgen de equipos organizados por sí mismos.
- A intervalos regulares, los equipos reflexionan sobre cómo ser más efectivos, entonces afinan y ajustan su comportamiento de acuerdo con ello.

La tabla 2.1 recoge esquemáticamente las principales diferencias de las metodologías ágiles con respecto a las tradicionales. Estas diferencias que afectan no sólo al proceso en sí, sino también al contexto del equipo así como a su organización.

METODOLOGÍAS ÁGILES	METODOLOGÍAS TRADICIONALES
Basadas en heurísticas provenientes de prácticas de producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.
Especialmente preparados para cambios durante el proyecto.	Cierta resistencia a los cambios.
Impuestas internamente por el equipo.	Impuestas externamente.
Proceso menos controlado, con pocos principios.	Proceso mucho más controlado, con Numerosas políticas y normas.
No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.
Grupos pequeños de menos de 10 integrantes y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos artefactos.	Más artefactos.
Pocos roles.	Más roles.
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos.

Tabla 2.1 Diferencias entre Metodologías Ágiles y las Tradicionales
Fuente: (Letelier & Penadés, 2005)

2.1.4 HISTORIAS DE USUARIO

Las historias de usuario son utilizadas en los métodos ágiles para la especificación de requisitos, son una descripción breve de una funcionalidad software tal y como la percibe el usuario (Cohn, 2004).

Describen lo que el cliente o el usuario quiere que se implemente y se escriben con una o dos frases utilizando el lenguaje común del usuario. El pensamiento de las personas se estructura siguiendo una

narrativa, una historia, así es como entendemos el mundo. Estamos capacitados para comprender personajes, deseos y motivaciones, por tanto la forma en que más facilidad tenemos para adquirir y retener conocimiento es a través de las historias. Nos metemos dentro de los protagonistas de forma que vivimos como tales la historia que nos cuentan, empatizamos, y a todos los niveles que toman decisiones, hasta a nivel de desarrollador, tomamos decisiones más acertadas.

Cada historia de usuario debe ser limitada, esta debería poderse memorizar fácilmente y escribir sobre una tarjeta o post-it (card). Poco antes de ser implementadas, estas van acompañadas de conversaciones con los usuarios y la definición de los criterios de validación asociados. Como los cambios son bienvenidos en agilidad, no vale la pena profundizar antes, ya que en el momento de la implementación estas pueden haber cambiado desde que fueron escritas. Los criterios de validaación permiten al propietario del producto/usuario de negocio confirmar que el equipo ha recogido correctamente los requisitos, al equipo realizar las pruebas adecuadas y/o desarrollar guiados por pruebas con TDD, y finalmente comprobar que la historia ha sido completada. Las historias de usuario forman parte de la fórmula de captura de funcionalidades definida en 2001 por Ron Jeffries de las tres C's: Card - Conversation – Confirmation.

Estas son una forma ágil de administrar los requisitos de los usuarios sin tener que elaborar gran cantidad de documentos formales y sin requerir de mucho tiempo para administrarlos.
Ventajas que aportan las historias de usuario:

- Al ser muy cortas, estas representan requisitos del modelo de negocio que pueden implementarse rápidamente (días o semanas).
- Necesitan poco mantenimiento.
- Mantienen una relación cercana con el cliente.
- Permiten dividir los proyectos en pequeñas entregas.
- Permiten estimar fácilmente el esfuerzo de desarrollo.
- Son ideales para proyectos con requisitos volátiles o no muy claros.

El origen de las historias de usuario viene de XP “eXtremeProgramming” o programación extrema, donde las historias de usuario deben ser escritas por los clientes. Esta metodología fue creada por

Kent Beck y descrita por primera vez en 1999 en su libro eXtreme Programming Explained. Las historias de usuario se aplican en la mayoría de las metodologías ágiles, siendo así una herramienta muy importante también en SCRUM, se lo ve de forma general en la figura 2.1.

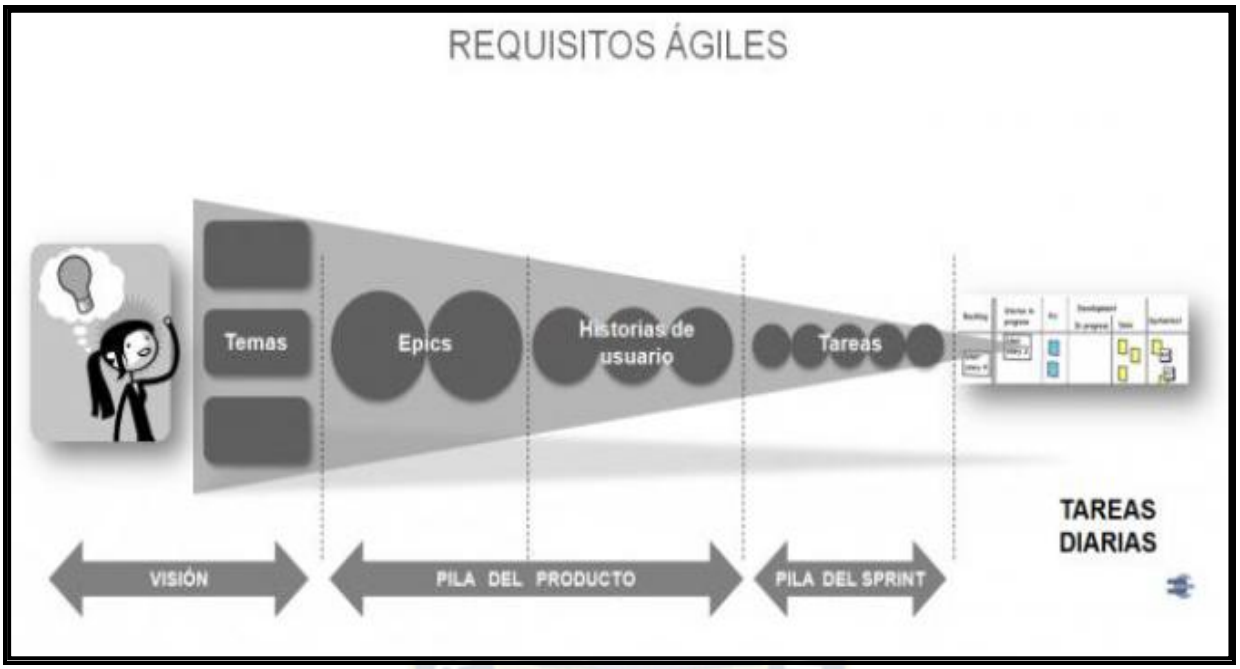


Figura 2.1 Niveles de tamaño con los que trata los requisitos de la Gestión Ágil
Fuente: (Beck, 1999)

2.1.5 SCRUM

Scrum es un marco de trabajo por el cual las personas pueden acometer problemas complejos adaptativos, a la vez que entregar productos del máximo valor posible productiva y creativamente. Según (Schwaber, 2004) Scrum es:

- Ligero.
- Fácil de entender.
- Extremadamente difícil de llegar a dominar.

Scrum es un marco de trabajo de procesos que ha sido usado para gestionar el desarrollo de productos complejos desde principios de los años 90. Scrum no es un proceso o una técnica para construir productos, en lugar de eso es un marco de trabajo dentro del cual se pueden emplear varias técnicas y procesos, muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo de modo que podamos mejorar.

El marco de trabajo Scrum consiste en los Equipos Scrum, roles, eventos, artefactos y reglas asociadas. Cada componente dentro del marco de trabajo sirve a un propósito específico esencial para el éxito de Scrum y para su uso. Las reglas de Scrum relacionan los eventos, roles y artefactos, gobernando las relaciones e interacciones entre ellos. Las reglas de Scrum se describen en el presente documento. Las estrategias específicas para usar el marco de trabajo Scrum son diversas y están descritas en otros lugares.

2.1.6 TEORÍA DE SCRUM

Scrum se basa en la teoría de control de procesos empírica o empirismo. El empirismo asegura que el conocimiento procede de la experiencia *y* de tomar decisiones basándose en lo que se conoce. Scrum emplea un enfoque iterativo e incremental para optimizar la predictibilidad y el control del riesgo. Tres pilares soportan toda la implementación del control de procesos empírico: transparencia, inspección y adaptación.

2.1.6.1 TRANSPARENCIA

Los aspectos significativos del proceso deben ser visibles para aquellos que son responsables del resultado. La transparencia requiere que dichos aspectos sean definidos por un estándar común, de tal modo que los observadores compartan un entendimiento común de lo que se está viendo. Por ejemplo:

- Todos los participantes deben compartir un lenguaje común para referirse al proceso.
- Aquellos que desempeñan el trabajo y aquellos que aceptan el producto de dicho trabajo deben compartir una definición común de “Terminado”.

2.1.6.2 INSPECCIÓN

Los usuarios de Scrum deben inspeccionar frecuentemente los artefactos de Scrum y el progreso hacia un objetivo, para detectar variaciones. Su inspección no debe ser tan frecuente como para que interfiera en el trabajo. Las inspecciones son más beneficiosas cuando se realizan de forma diligente por inspectores expertos, en el mismo lugar de trabajo.

2.1.6.3 ADAPTACIÓN

Si un inspector determina que uno o más aspectos de un proceso se desvían de límites aceptables y que el producto resultante no será aceptable, el proceso o el material que está siendo procesado deben ser ajustados, dicho ajuste debe realizarse cuanto antes para minimizar desviaciones mayores.

Scrum prescribe cuatro eventos formales, contenidos dentro del Sprint, para la inspección y adaptación tal y como se describen en la sección *Eventos de Scrum* del presente documento.

- Reunión de Planificación del Sprint (*Sprint Planning Meeting*).
- Scrum Diario (*Daily Scrum*).
- Revision del Sprint (*Sprint Review*).
- Retrospectiva del Sprint (*Sprint Retrospective*).

2.1.7 EL EQUIPO SCRUM (*SCRUM TEAM*)

El Equipo Scrum consiste en un Dueño de Producto (*Product Owner*), el Equipo de Desarrollo (*Development Team*) y un Scrum Master. Los Equipos Scrum son auto organizados y multifuncionales.

Los equipos auto organizados eligen la mejor forma de llevar a cabo su trabajo y no son dirigidos por personas externas al equipo. Los equipos multifuncionales tienen todas las competencias necesarias para llevar a cabo el trabajo sin depender de otras personas que no son parte del equipo. El modelo de equipo en Scrum está diseñado para optimizar la flexibilidad, la creatividad y la productividad.

Los Equipos Scrum entregan productos de forma iterativa e incremental maximizando las oportunidades de obtener retroalimentación. Las entregas incrementales de producto “Terminado” aseguran que siempre estará disponible una versión potencialmente útil y funcional del producto.

2.1.7.1 EL DUEÑO DE PRODUCTO (*PRODUCT OWNER*)

El Dueño de Producto es el responsable de maximizar el valor del producto y del trabajo del Equipo de Desarrollo. El cómo se lleva a cabo esto podría variar ampliamente entre distintas organizaciones, Equipos Scrum e individuos.

El Dueño de Producto es la única persona responsable de gestionar la Lista del Producto (*Product Backlog*). La gestión de la Lista del Producto incluye:

- Expresar claramente los elementos de la Lista del Producto.
- Ordenar los elementos en la Lista del Producto para alcanzar los objetivos y misiones de la mejor manera posible.
- Optimizar el valor del trabajo desempeñado por el Equipo de Desarrollo.
- Asegurar que la Lista del Producto es visible, transparente y clara para todos, y que muestra aquello en lo que el equipo trabajará a continuación.
- Asegurar que el Equipo de Desarrollo entiende los elementos de la Lista del Producto al nivel necesario.

El Dueño de Producto podría hacer el trabajo anterior, o delegarlo en el Equipo de Desarrollo. Sin embargo, en ambos casos el Dueño de Producto sigue siendo el responsable de dicho trabajo, es una única persona, no un comité. El Dueño de Producto podría representar los deseos de un comité en la Lista del Producto, pero aquellos que quieran cambiar la prioridad de un elemento de la Lista deben hacerlo a través del Dueño de Producto.

Para que el Dueño de Producto pueda hacer bien su trabajo, toda la organización debe respetar sus decisiones, las decisiones del Dueño de Producto se reflejan en el contenido y en la priorización de la Lista del Producto, no está permitido que nadie pida al Equipo de Desarrollo que trabaje en base a un conjunto diferente de requerimientos y por último el Equipo de Desarrollo no debe actuar en base a lo que diga cualquier otra persona, inclusive los que forman parte de equipo de trabajo.

2.1.7.2 EL EQUIPO DE DESARROLLO (*DEVELOPMENT TEAM*)

El Equipo de Desarrollo consiste en los profesionales que desempeñan el trabajo de entregar un Incremento de producto "Terminado", que potencialmente se pueda poner en producción al final de cada Sprint. Solo los miembros del Equipo de Desarrollo participan en la creación del Incremento. Los Equipos de Desarrollo son estructurados y empoderados por la organización para organizar y gestionar su propio trabajo. La sinergia resultante optimiza la eficiencia y efectividad del Equipo de Desarrollo.

Los Equipos de Desarrollo tienen las siguientes características:

- Son auto-organizados. Nadie (ni siquiera el Scrum Master) indica al Equipo de Desarrollo cómo convertir elementos de la Lista del Producto en Incrementos de funcionalidad potencialmente desplegables.
- Los Equipos de Desarrollo son multifuncionales, contando como equipo con todas las habilidades necesarias para crear un Incremento de producto.
- Scrum no reconoce títulos para los miembros de un Equipo de Desarrollo, todos son Desarrolladores, independientemente del trabajo que realice cada persona; no hay excepciones a esta regla.
- Scrum no reconoce sub-equipos en los equipos de desarrollo, no importan los dominios particulares que requieran ser tenidos en cuenta, como pruebas o análisis de negocio; no hay excepciones a esta regla.
- Los Miembros individuales del Equipo de Desarrollo pueden tener habilidades especializadas y áreas en las que estén más enfocados, pero la responsabilidad recae en el Equipo de Desarrollo como un todo.

2.1.7.3 TAMAÑO DEL EQUIPO DE DESARROLLO

El tamaño óptimo del Equipo de Desarrollo es lo suficientemente pequeño como para permanecer ágil y lo suficientemente grande como para completar una cantidad de trabajo significativa. Tener menos de tres miembros en el Equipo de Desarrollo reduce la interacción y resulta en ganancias de productividad más pequeñas. Los Equipos de Desarrollo más pequeños podrían encontrar

limitaciones en cuanto a las habilidades necesarias durante un Sprint, haciendo que el Equipo de Desarrollo no pudiese entregar un Incremento que potencialmente se pueda poner en producción. Tener más de nueve miembros en el equipo requiere demasiada coordinación.

Los Equipos de Desarrollo grandes generan demasiada complejidad como para que pueda gestionarse mediante un proceso empírico. Los roles de Dueño de Producto y Scrum Master no cuentan en el cálculo del tamaño del equipo a menos que también estén contribuyendo a trabajar en la Lista de Pendientes de Sprint (Schwaber, 2004).

2.1.7.4 EL DUEÑO DEL EQUIPO (SCRUM MASTER)

El Scrum Master es el responsable de asegurar que Scrum es entendido y adoptado. Los Scrum Masters hacen esto asegurándose de que el Equipo Scrum trabaja ajustándose a la teoría, prácticas y reglas de Scrum.

El Scrum Master es un líder que está al servicio del Equipo Scrum. El Scrum Master ayuda a las personas externas al Equipo Scrum a entender qué interacciones con el Equipo Scrum pueden ser de ayuda y cuáles no. El Scrum Master ayuda a todos a modificar estas interacciones para maximizar el valor creado por el Equipo Scrum.

2.1.7.5 EL SERVICIO DEL SCRUM MASTER AL DUEÑO DE PRODUCTO

El Scrum Master da servicio al Dueño de Producto de varias formas, incluyendo:

- Encontrar técnicas para gestionar la Lista de Producto de manera efectiva.
- Ayudar al Equipo Scrum a entender la necesidad de contar con elementos de Lista de Producto claros y concisos.
- Entender la planificación del producto en un entorno empírico.
- Asegurar que el Dueño de Producto conozca cómo ordenar la Lista de Producto para maximizar el valor.
- Entender y practicar la agilidad.
- Facilitar los eventos de Scrum según se requiera o necesite.

2.1.7.6 EL SERVICIO DEL SCRUM MASTER AL EQUIPO DE DESARROLLO

El Scrum Master da servicio al Equipo de Desarrollo de varias formas, incluyendo:

- Guiar al Equipo de Desarrollo en ser auto organizado y multifuncional.
- Ayudar al Equipo de Desarrollo a crear productos de alto valor.
- Eliminar impedimentos para el progreso del Equipo de Desarrollo.
- Facilitar los eventos de Scrum según se requiera o necesite.
- Guiar al Equipo de Desarrollo en el entorno de organizaciones en las que Scrum aún no ha sido adoptado y entendido por completo.

2.1.7.7 EL SERVICIO DEL SCRUM MASTER A LA ORGANIZACIÓN

El Scrum Master da servicio a la organización de varias formas, incluyendo:

- Liderar y guiar a la organización en la adopción de Scrum.
- Planificar las implementaciones de Scrum en la organización.
- Ayudar a los empleados e interesados a entender y llevar a cabo Scrum y el desarrollo empírico de producto.
- Motivar cambios que incrementen la productividad del Equipo Scrum.
- Trabajar con otros Scrum Masters para incrementar la efectividad de la aplicación de Scrum en la organización.

2.1.8 EVENTOS DE SCRUM

Según (Schwaber, 2004) en Scrum existen eventos predefinidos con el fin de crear regularidad y minimizar la necesidad de reuniones no definidas en Scrum. Todos los eventos son bloques de tiempo, de tal modo que todos tienen una duración máxima. Una vez que comienza un Sprint, su duración es fija y no puede acortarse o alargarse. Los demás eventos pueden terminar siempre que se alcance el objetivo del evento, asegurando que se emplee una cantidad apropiada de tiempo sin permitir desperdicio en el proceso.

Además del propio Sprint, que es un contenedor del resto de eventos, cada uno de los eventos de Scrum constituye una oportunidad formal para la inspección y adaptación de algún aspecto. Estos eventos están diseñados específicamente para habilitar las vitales transparencia e inspección. La falta de alguno de estos eventos da como resultado una reducción de la transparencia y constituye una oportunidad perdida para inspeccionar y adaptarse.

2.1.8.1 EL SPRINT

El corazón de Scrum es el Sprint, es un bloque de tiempo (*time-box*) de un mes o menos durante el cual se crea un incremento de producto "Terminado", utilizable y potencialmente desplegable. Es más conveniente si la duración de los Sprints es consistente a lo largo del esfuerzo de desarrollo. Cada nuevo Sprint comienza inmediatamente después de la finalización del Sprint previo.

Los Sprints contienen y consisten de la Reunión de Planificación del Sprint (Sprint Planning Meeting), los Scrums Diarios (Daily Scrums), el trabajo de desarrollo, la Revisión del Sprint (Sprint Review), y la Retrospectiva del Sprint (Sprint Retrospective).

Durante el Sprint:

- No se realizan cambios que puedan afectar al Objetivo del Sprint (Sprint Goal).
- Los objetivos de calidad no disminuyen.
- El alcance puede ser clarificado y renegociado entre el Dueño de Producto y el Equipo de Desarrollo a medida que se va aprendiendo más.

Cada Sprint puede considerarse un proyecto con un horizonte no mayor de un mes. Al igual que los proyectos, los Sprints se usan para lograr algo. Cada Sprint tiene una definición de qué se va a construir, un diseño y un plan flexible que guiará la construcción y el trabajo y el producto resultante.

Los Sprints están limitados a un mes calendario, cuando el horizonte de un Sprint es demasiado grande la definición de lo que se está construyendo podría cambiar, la complejidad podría elevarse y el riesgo podría aumentar.

Los Sprints habilitan la predictibilidad al asegurar la inspección y adaptación del progreso al menos en cada mes calendario. Los Sprints también limitan el riesgo al costo de un mes calendario debido al tamaño del proyecto.

2.1.8.2 CANCELACIÓN DE UN SPRINT

Un Sprint puede ser cancelado antes de que el bloque de tiempo llegue a su fin. Solo el Dueño de Producto tiene la autoridad para cancelar el Sprint, aunque puede hacerlo bajo la influencia de los interesados del Equipo de Desarrollo o del Scrum Master.

Un Sprint se cancelaría si el Objetivo del Sprint llega a quedar obsoleto. Esto podría ocurrir si la compañía cambia la dirección o si las condiciones del mercado o de la tecnología cambian, en general un Sprint debería cancelarse si no tuviese sentido seguir con él dadas las circunstancias, pero debido a la corta duración de los Sprints rara vez la cancelación tiene sentido.

Cuando se cancela un Sprint se revisan todos los Elementos de la Lista de Producto que se hayan completado y "Terminado". Si una parte del trabajo es potencialmente entregable, el Dueño de Producto normalmente lo acepta. Todos los Elementos de la Lista de Producto no completados se vuelven a estimar y se vuelven a introducir en la Lista de Producto. El trabajo finalizado en ellos pierde valor con rapidez y frecuentemente debe volverse a estimar.

Las cancelaciones de Sprint consumen recursos ya que todos deben reagruparse en otra Reunión de Planificación de Sprint para empezar otro Sprint. Las cancelaciones de Sprint son a menudo traumáticas para el Equipo Scrum y son muy poco comunes.

2.1.8.3 REUNIÓN DE PLANIFICACIÓN DE SPRINT

El trabajo a realizar durante el Sprint se planifica en la Reunión de Planificación de Sprint, este plan se crea mediante el trabajo colaborativo del Equipo Scrum completo, la Reunión de Planificación de Sprint tiene un máximo de duración de ocho horas para un Sprint de un mes.

Para Sprints más cortos el evento es usualmente más corto, el Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito, el Scrum Master enseña al Equipo Scrum a mantenerse dentro del bloque de tiempo.

La Reunión de Planificación de Sprint responde a las siguientes preguntas:

- ¿Qué puede entregarse en el Incremento resultante del Sprint que comienza?
- ¿Cómo se conseguirá hacer el trabajo necesario para entregar el Incremento?

Tema Uno: ¿Qué puede ser terminado en este Sprint?
El Equipo de Desarrollo trabaja para proyectar la funcionalidad que se desarrollará durante el Sprint. El Dueño de Producto discute el objetivo que el Sprint debería lograr y los Elementos de la Lista de Producto que, si se completan en el Sprint, lograrían el Objetivo del Sprint. El Equipo Scrum completo colabora en el entendimiento del trabajo del Sprint.

La entrada a esta reunión está constituida por la Lista de Producto, el último Incremento de producto, la capacidad proyectada del Equipo de Desarrollo para el Sprint, y el rendimiento pasado del Equipo de Desarrollo. El número de elementos de la Lista de Producto seleccionados para el Sprint depende únicamente del Equipo de Desarrollo. Solo el Equipo de Desarrollo puede evaluar qué es capaz de lograr durante el Sprint que comienza.

Después de que el Equipo de Desarrollo proyecta qué elementos de la Lista de Producto entregará en el Sprint, el Equipo Scrum elabora un Objetivo del Sprint (*Sprint Goal*). El Objetivo del Sprint debería lograrse durante el Sprint a través de la implementación de la Lista, y provee una guía al equipo de desarrollo de por qué se está construyendo el incremento.

Tema Dos: ¿Cómo se conseguirá completar el trabajo seleccionado?
Una vez que se ha establecido el objetivo y seleccionado los elementos de la Lista de Producto para el Sprint, el Equipo de Desarrollo decide cómo construirá esta funcionalidad para formar un Incremento de producto “Terminado”. Los elementos de la Lista de Producto seleccionados para

este Sprint, más el plan para terminarlos, recibe el nombre de Lista de Pendientes del Sprint (*Sprint Backlog*).

El Equipo de Desarrollo por lo general comienza diseñando el sistema y el trabajo necesario para convertir la Lista de Producto en un Incremento de producto funcional. El trabajo podría ser de tamaño o esfuerzo estimado variables. Sin embargo, durante la Reunión de Planificación del Sprint, se planifica suficiente trabajo como para que el Equipo de Desarrollo pueda hacer una proyección de lo que cree que puede completar en el Sprint que comienza. Para el final de esta reunión, el trabajo planificado por el Equipo de Desarrollo para los primeros días del Sprint es descompuesto en unidades de un día o menos. El Equipo de desarrollo se auto-organiza para asumir el trabajo de la Lista de Pendientes de Sprint, tanto durante la reunión de Planificación de Sprint como a lo largo del Sprint.

El Dueño de Producto puede ayudar a clarificar los elementos de la Lista de Producto seleccionados y hacer concesiones. Si el Equipo de Desarrollo determina que tiene demasiado trabajo o que no tiene suficiente trabajo, podría renegociar los elementos de la Lista de Producto seleccionados con el Dueño de Producto. El Equipo de Desarrollo podría también invitar a otras personas a que asistan con el fin de que proporcionen asesoría técnica o relacionada con el dominio. Al finalizar la Reunión de Planificación de Sprint, el Equipo de Desarrollo debería ser capaz de explicar al Dueño de Producto y al Scrum Master cómo pretende trabajar como un equipo auto organizado para lograr el Objetivo del Sprint y crear el Incremento esperado.

2.1.8.4 OBJETIVO DEL SPRINT

El Objetivo del Sprint es una meta establecida para el Sprint que puede ser alcanzada mediante la implementación de la Lista de Producto. Proporciona una guía al Equipo de Desarrollo acerca de por qué está construyendo el incremento. Es creado durante la reunión de Planificación del Sprint. El objetivo del Sprint ofrece al equipo de desarrollo cierta flexibilidad con respecto a la funcionalidad implementada en el Sprint. Los elementos de la Lista del Producto seleccionados ofrecen una función coherente, que puede ser el objetivo del Sprint. El objetivo del Sprint puede representar otro

nexo de unión que haga que el Equipo de Desarrollo trabaje en conjunto y no en iniciativas separadas.

A medida que el equipo de desarrollo trabaja, se mantiene el objetivo del Sprint en mente. Con el fin de satisfacer el objetivo del Sprint se implementa la funcionalidad y la tecnología. Si el trabajo resulta ser diferente de lo que el Equipo de Desarrollo espera, ellos colaboran con el Dueño del Producto para negociar el alcance de la Lista de pendientes del Sprint (*Sprint Backlog*).

2.1.8.5 SCRUM DIARIO

El Scrum Diario es una reunión con un bloque de tiempo de 15 minutos para que el Equipo de Desarrollo sincronice sus actividades y cree un plan para las siguientes 24 horas. Esto se lleva a cabo inspeccionando el trabajo avanzado desde el último Scrum Diario y haciendo una proyección acerca del trabajo que podría completarse antes del siguiente.

El Scrum Diario se realiza a la misma hora y en el mismo lugar todos los días para reducir la complejidad. Durante la reunión, cada miembro del Equipo de Desarrollo explica:

- ¿Qué hice ayer que ayudó al Equipo de Desarrollo a lograr el Objetivo del Sprint?
- ¿Qué haré hoy para ayudar al Equipo de Desarrollo a lograr el Objetivo del Sprint?
- ¿Veo algún impedimento que evite que el Equipo de Desarrollo o yo logremos el Objetivo del Sprint?

El Equipo de Desarrollo usa el Scrum Diario para evaluar el progreso hacia el Objetivo del Sprint y para evaluar qué tendencia sigue este progreso hacia la finalización del trabajo contenido en la Lista del Sprint. El Scrum Diario optimiza las posibilidades de que el Equipo de Desarrollo cumpla el Objetivo del Sprint. Cada día, el Equipo de Desarrollo debería entender cómo intenta trabajar en conjunto como un equipo auto organizado para lograr el Objetivo del Sprint y crear el Incremento esperado hacia el final del Sprint.

El Equipo de Desarrollo o los miembros del equipo a menudo se vuelven a reunir inmediatamente después del Scrum Diario, para tener discusiones detalladas, o para adaptar, o replanificar el resto del trabajo del Sprint.

El Scrum Master se asegura de que el Equipo de Desarrollo tenga la reunión, pero el Equipo de Desarrollo es el responsable de dirigir el Scrum Diario. El Scrum Master enseña al Equipo de Desarrollo para que mantenga el Scrum Diario en los límites del bloque de tiempo de 15 minutos. El Scrum Master se asegura de que se cumpla la regla de que solo los miembros del Equipo de Desarrollo participan en el Scrum Diario.

Los Scrum Diarios mejoran la comunicación, eliminan la necesidad de mantener otras reuniones, identifican y eliminan impedimentos relativos al desarrollo, resaltan y promueven la toma de decisiones rápida, y mejoran el nivel de conocimiento del Equipo de Desarrollo. El Scrum Diario constituye una reunión clave de inspección y adaptación.

2.1.8.6 REVISIÓN DE SPRINT

Al final del Sprint se lleva a cabo una Revisión de Sprint para inspeccionar el Incremento y adaptar la Lista de Producto si fuese necesario. Durante la Revisión de Sprint, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint. Basándose en esto y en cualquier cambio a la Lista de Producto durante el Sprint, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor. Se trata de una reunión informal, no una reunión de seguimiento y la presentación del Incremento tiene como objetivo facilitar la retroalimentación de información y fomentar la colaboración.
Se trata de una reunión restringida a un bloque de tiempo de cuatro horas para Sprints de un mes. Para Sprints más cortos se reserva un tiempo proporcionalmente menor. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña a todos a mantener el evento dentro del bloque de tiempo fijado.

La Revisión de Sprint incluye los siguientes elementos:

- Los asistentes son el Equipo Scrum y los interesados clave invitados por el Dueño de Producto.

- El Dueño de Producto explica qué elementos de la Lista de Producto se han “Terminado” y cuales no se han “Terminado”.
- El Equipo de Desarrollo habla acerca de qué fue bien durante el Sprint, qué problemas aparecieron y cómo fueron resueltos esos problemas.
- El Equipo de Desarrollo demuestra el trabajo que ha “Terminado” y responde preguntas acerca del Incremento.
- El Dueño de Producto habla acerca de la Lista de Producto en el estado actual. Proyecta fechas de finalización probables en el tiempo basándose en el progreso obtenido hasta la fecha (si es necesario).
- El grupo completo colabora acerca de qué hacer a continuación, de modo que la Revisión del Sprint proporcione información de entrada valiosa para Reuniones de Planificación de Sprints subsiguientes.
- Revisión de cómo el mercado o el uso potencial del producto podría haber cambiado lo que es de más valor para hacer a continuación.
- Revisión de la línea de tiempo, presupuesto, capacidades potenciales y mercado para la próxima entrega prevista del producto.

El resultado de la Revisión de Sprint es una Lista de Producto revisada que define los elementos de la Lista de Producto posibles para el siguiente Sprint.

Es posible además que la Lista de Producto reciba un ajuste general para enfocarse en nuevas oportunidades.

2.1.8.7 RETROSPECTIVA DE SPRINT

La Retrospectiva de Sprint es una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y crear un plan de mejoras que sean abordadas durante el siguiente Sprint, la Retrospectiva de Sprint tiene lugar después de la Revisión de Sprint y antes de la siguiente Reunión de Planificación de Sprint, se trata de una reunión restringida a un bloque de tiempo de tres horas para Sprints de un mes. Para Sprints más cortos se reserva un tiempo proporcionalmente menor. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum

Master enseña a todos a mantener el evento dentro del bloque de tiempo fijado. El Scrum Master participa en la reunión como un miembro del equipo ya que la responsabilidad del proceso Scrum recae sobre él.

El propósito de la Retrospectiva de Sprint es:

- Inspeccionar cómo fue el último Sprint en cuanto a personas, relaciones, procesos y herramientas.
- Identificar y ordenar los elementos más importantes que salieron bien y las posibles mejoras.
- Crear un plan para implementar las mejoras a la forma en la que el Equipo Scrum desempeña su trabajo.

El Scrum Master alienta al equipo para que mejore, dentro del marco de proceso Scrum, su proceso de desarrollo y sus prácticas para hacerlos más efectivos y amenos para el siguiente Sprint. Durante cada Retrospectiva de Sprint, el Equipo Scrum planifica formas de aumentar la calidad del producto mediante la adaptación de la Definición de "Terminado" (*Definition of "Done"*) según sea conveniente.

Para el final de la Retrospectiva de Sprint el Equipo Scrum debería haber identificado mejoras que implementará en el próximo Sprint. El hecho de implementar estas mejoras en el siguiente Sprint, constituye la adaptación subsecuente a la inspección del Equipo de Desarrollo a sí mismo. Aunque las mejoras pueden implementarse en cualquier momento, la Retrospectiva de Sprint ofrece un evento dedicado para este fin, enfocado en la inspección y la adaptación.

2.1.9 ARTEFACTOS DE SCRUM

Según (Schwaber, 2004) los artefactos de Scrum representan trabajo o valor en diversas formas que son útiles para proporcionar transparencia y oportunidades para la inspección y adaptación. Los artefactos definidos por Scrum están diseñados específicamente para maximizar la transparencia de la información clave, que es necesaria para asegurar que todos tengan el mismo entendimiento del artefacto, como se ve en la Figura 2.2.

2.1.9.1 LISTA DE PRODUCTO (*PRODUCT BACKLOG*)

La Lista de Producto es una lista ordenada de todo lo que podría ser necesario en el producto, es la única fuente de requisitos para cualquier cambio a realizarse en el producto. El Dueño de Producto (*Product Owner*) es el responsable de la Lista de Producto, incluyendo su contenido, disponibilidad y ordenación.

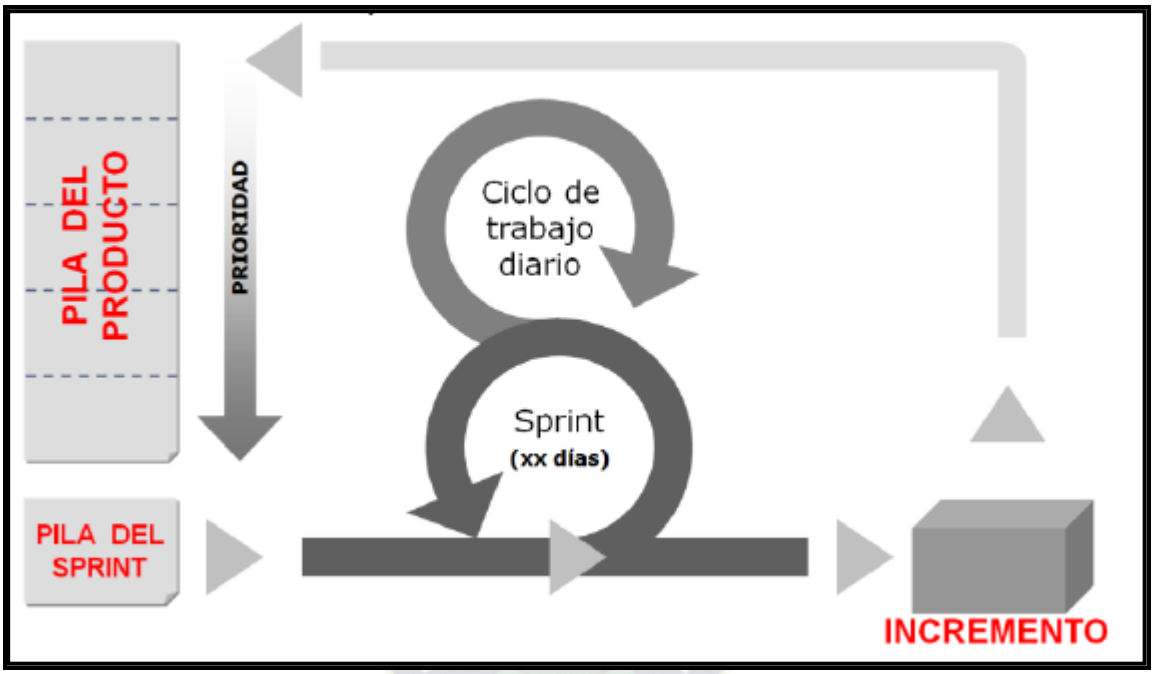


Figura 2.2 Diagrama de Ciclo Iterativo de SCRUM
Fuente: (Schwaber, 2004)

Una Lista de Producto nunca está completa. El desarrollo más temprano de la misma solo refleja los requisitos conocidos y mejor entendidos al principio. La Lista de Producto evoluciona a medida de que el producto y el entorno en el que se usará también lo hacen. La Lista de Producto es dinámica; cambia constantemente para identificar lo que el producto necesita para ser adecuado, competitivo y útil. Mientras el producto exista, su Lista de Producto también existe.

La Lista de Producto enumera todas las características, funcionalidades, requisitos, mejoras y correcciones que constituyen cambios a ser hechos sobre el producto para entregas futuras. Los elementos de la Lista de Producto tienen como atributos la descripción, la ordenación, la estimación y el valor.

A medida que un producto es utilizado y se incrementa su valor, y el mercado proporciona retroalimentación, la Lista de Producto se convierte en una lista más larga y exhaustiva. Los requisitos nunca dejan de cambiar, así que la Lista de Producto es un artefacto vivo. Los cambios en los requisitos de negocio, las condiciones del mercado o la tecnología podrían causar cambios en la Lista de Producto.

A menudo, varios Equipos Scrum trabajan juntos en el mismo producto. Para describir el trabajo a realizar sobre el producto, se utiliza una única Lista de Producto. En ese caso podría emplearse un atributo de la Lista de Producto para agrupar los elementos.

El refinamiento de la Lista de Producto es el acto de añadir detalle, estimaciones y orden a los elementos de la Lista de Producto. Se trata de un proceso continuo, en el cual el Dueño de Producto y el Equipo de Desarrollo colaboran acerca de los detalles de los elementos de la Lista de Producto. Durante el refinamiento de la Lista de Producto, se examinan y revisan sus elementos. El Equipo Scrum decide cómo y cuándo se hace el refinamiento. Este usualmente consume no más del 10% de la capacidad del Equipo de Desarrollo. Sin embargo, los elementos de la Lista de Producto pueden actualizarse en cualquier momento por el Dueño de Producto o a criterio suyo.

Los elementos de la Lista de Producto de orden más alto son generalmente más claros y detallados que los de menor orden. Se realizan estimaciones más precisas basándose en la mayor claridad y detalle; cuanto más bajo es el orden, menor es el detalle. Los elementos de la Lista de Producto de los que se ocupará el Equipo de Desarrollo en el siguiente Sprint tienen una granularidad mayor, habiendo sido descompuestos de forma que cualquier elemento puede ser “Terminado” dentro de los límites del bloque de tiempo del Sprint. Los elementos de la Lista de Producto que pueden ser “Terminados” por el Equipo de Desarrollo en un Sprint son considerados “preparados” o “accionables” para ser seleccionados en una reunión de Planificación de Sprint. Los elementos de la Lista de Producto normalmente adquieren este grado de transparencia mediante las actividades de refinamiento descritas anteriormente.

El Equipo de Desarrollo es el responsable de proporcionar todas las estimaciones. El Dueño de Producto podría influenciar al Equipo ayudándoles a entender y seleccionar soluciones de compromiso, pero las personas que harán el trabajo son las que hacen la estimación final.

En cualquier momento, es posible sumar el trabajo total restante para alcanzar el objetivo. El Dueño de Producto hace seguimiento de este trabajo restante total al menos en cada Revisión de Sprint. El Dueño de Producto compara esta cantidad con el trabajo restante en Revisiones de Sprint previas, para evaluar el progreso hacia la finalización del trabajo proyectado en el tiempo deseado para el objetivo. Esta información se muestra de forma transparente a todos los interesados.

Varias prácticas de proyección sobre tendencias se han utilizado para predecir el progreso, como trabajo consumido (*burndown*), avanzado (*burnup*) y flujo acumulado (*cumulative flow*). Estas se han revelado como útiles. Sin embargo, no remplazan la importancia del empirismo. En entornos complejos, se desconoce lo que ocurrirá. Solo lo que ya ha ocurrido puede utilizarse para la toma de decisiones con miras al futuro.

2.1.9.2 LISTA DE PENDIENTES DEL SPRINT (*SPRINT BACKLOG*)

La Lista de Pendientes del Sprint es el conjunto de elementos de la Lista de Producto seleccionados para el Sprint, más un plan para entregar el Incremento de producto y conseguir el Objetivo del Sprint. La Lista de Pendientes del Sprint es una predicción hecha por el Equipo de Desarrollo acerca de qué funcionalidad formará parte del próximo Incremento y del trabajo necesario para entregar esa funcionalidad en un Incremento “Terminado”.

La Lista de Pendientes del Sprint hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para alcanzar el Objetivo del Sprint, La Lista de Pendientes del Sprint es un plan con un nivel de detalle suficiente como para que los cambios en el progreso se puedan entender en el Scrum Diario. El Equipo de Desarrollo modifica la Lista de Pendientes del Sprint durante el Sprint y esta Lista de Pendientes del Sprint emerge a lo largo del Sprint. Esto ocurre a medida que el Equipo de Desarrollo trabaja sobre el plan y aprende más acerca del trabajo necesario para conseguir el Objetivo del Sprint.

Según se requiere nuevo trabajo, el Equipo de Desarrollo lo añade a la Lista de Pendientes del Sprint. A medida que el trabajo se ejecuta o se completa, se va actualizando la estimación de trabajo restante. Cuando algún elemento del plan pasa a ser considerado innecesario, es eliminado. Solo el Equipo de Desarrollo puede cambiar su Lista de Pendientes del Sprint durante un Sprint.
La Lista de Pendientes del Sprint es una imagen visible en tiempo real del trabajo que el Equipo de Desarrollo planea llevar a cabo durante el Sprint, y pertenece únicamente al Equipo de Desarrollo.

En cualquier momento durante un Sprint, es posible sumar el trabajo restante total en los elementos de la Lista de Pendientes del Sprint. El Equipo de Desarrollo hace seguimiento de este trabajo restante total al menos en cada Scrum Diario para proyectar la posibilidad de conseguir el Objetivo del Sprint. Haciendo seguimiento del trabajo restante a lo largo del Sprint, el Equipo de Desarrollo puede gestionar su progreso.

2.1.9.3 INCREMENTO

El Incremento es la suma de todos los elementos de la Lista de Producto completados durante un Sprint y el valor de los incrementos de todos los Sprints anteriores. Al final de un Sprint, el nuevo Incremento debe estar “Terminado”, lo cual significa que está en condiciones de ser utilizado y que cumple la Definición de “Terminado” del Equipo Scrum. El incremento debe estar en condiciones de utilizarse sin importar si el Dueño de Producto decide liberarlo o no.

2.1.10 TRANSPARENCIA DE LOS ARTEFACTOS

Scrum se basa en la transparencia. Las decisiones para optimizar el valor y controlar el riesgo se toman basadas en el estado percibido de los artefactos. En la medida en que la transparencia sea completa, estas decisiones tienen unas bases sólidas. En la medida en que los artefactos no son completamente transparentes, estas decisiones pueden ser erróneas, el valor puede disminuir y el riesgo puede aumentar.

El Scrum Master debe trabajar con el Dueño de Producto, el Equipo de Desarrollo y otras partes involucradas para entender si los artefactos son completamente transparentes. Hay prácticas para hacer frente a la falta de transparencia; el Scrum Master debe ayudar a todos a aplicar las prácticas más apropiadas si no hay una transparencia completa.

Un Scrum Master puede detectar la falta de transparencia inspeccionando artefactos, reconociendo patrones, escuchando atentamente lo que se dice y detectando diferencias entre los resultados esperados y los reales.

La labor del Scrum Master es trabajar con el Equipo Scrum y la organización para mejorar la transparencia de los artefactos. Este trabajo usualmente incluye aprendizaje, convicción y cambio. La transparencia no ocurre de la noche a la mañana, sino que es un camino.

2.1.11 DEFINICIÓN DE "TERMINADO" (*DEFINITION OF "DONE"*)

Cuando un elemento de la Lista de Producto o un Incremento se describe como "Terminado", todo el mundo debe entender lo que significa "Terminado". Aunque esto varía significativamente para cada Equipo Scrum, los miembros del Equipo deben tener un entendimiento compartido de lo que significa que el trabajo esté completado, para asegurar la transparencia. Esta es la definición de "Terminado" para el Equipo Scrum y se utiliza para evaluar cuándo se ha completado el trabajo sobre el Incremento de producto.

Esta misma definición guía al Equipo de Desarrollo en saber cuántos elementos de la Lista de Producto puede seleccionar durante una reunión de Planificación de Sprint. El propósito de cada Sprint es entregar Incrementos de funcionalidad que potencialmente se puedan poner en producción, y que se ajustan a la Definición de "Terminado" actual del Equipo Scrum.

Los Equipos de Desarrollo entregan un Incremento de funcionalidad de producto en cada Sprint. Este Incremento es utilizable, de modo que el Dueño de Producto podría elegir liberarlo inmediatamente. Si la definición de "Terminado" para un incremento es parte de las convenciones,

estándares o guías de la organización de desarrollo, al menos todos los Equipos Scrum deben seguirla.

Si “Terminado” para un incremento no es una convención de la organización de desarrollo, el Equipo de Desarrollo del Equipo Scrum debe definir una definición de “Terminado” apropiada para el producto. Si hay múltiples Equipos Scrum trabajando en la entrega del sistema o producto, los equipos de desarrolladores en todos los Equipos Scrum deben definir en conjunto la definición de “Terminado”.

Cada Incremento se integra con todos los Incrementos anteriores y es probado exhaustivamente, asegurando que todos los Incrementos funcionan en conjunto. A medida que los Equipos Scrum maduran, se espera que su definición de “Terminado” se amplíe para incluir criterios más rigurosos para una mayor calidad.

2.2 METODOLOGÍA OOHDM

Según (Rossi, 1996). OOHDM (Object Oriented Hypermedia Design Method) fue diseñado por D. Schwabe, G. Rossi y S. D. J. Barbosa, propone el desarrollo de aplicaciones hipermedia a través de un proceso compuesto por cuatro fases: diseño conceptual, diseño de navegación, diseño de interfaces abstractas e implementación.

OOHDM es una mezcla de estilos de desarrollo basado en prototipo, el desarrollo interactivo y el desarrollo incremental. En cada fase se elabora un modelo orientado a objetos conceptual que recoge características a resaltar en la misma incrementando los resaltados de las fase o fases anteriores.

2.2.1 FASES DE OOHDM

OOHDM propone el desarrollo de aplicaciones hipermedia a través de un proceso compuesto por cuatro etapas: diseño conceptual, diseño navegacional, diseño de interfaces abstractas e implementación (Rossi, 1996).

2.2.1.1 DISEÑO CONCEPTUAL

Durante esta actividad se construye un esquema conceptual representado por los objetos del dominio, las relaciones y colaboraciones existentes establecidas entre ellos. En las aplicaciones hipermedia convencionales, cuyos componentes de hipermedia no son modificados durante la ejecución, se podría usar un modelo de datos semántico estructural (como el modelo de entidades y relaciones). De este modo, en los casos en que la información base pueda cambiar dinámicamente o se intenten ejecutar cálculos complejos, se necesitará enriquecer el comportamiento del modelo de objetos En OOHDM, el esquema conceptual está construido por clases, relaciones y subsistemas. Las clases son descritas como en los modelos orientados a objetos tradicionales. Sin embargo, los atributos pueden ser de múltiples tipos para representar perspectivas diferentes de las mismas entidades del mundo real.

En la figura 2.3 se muestra el diseño conceptual de una empresa ofrece servicios de capacitación a distintas empresas.

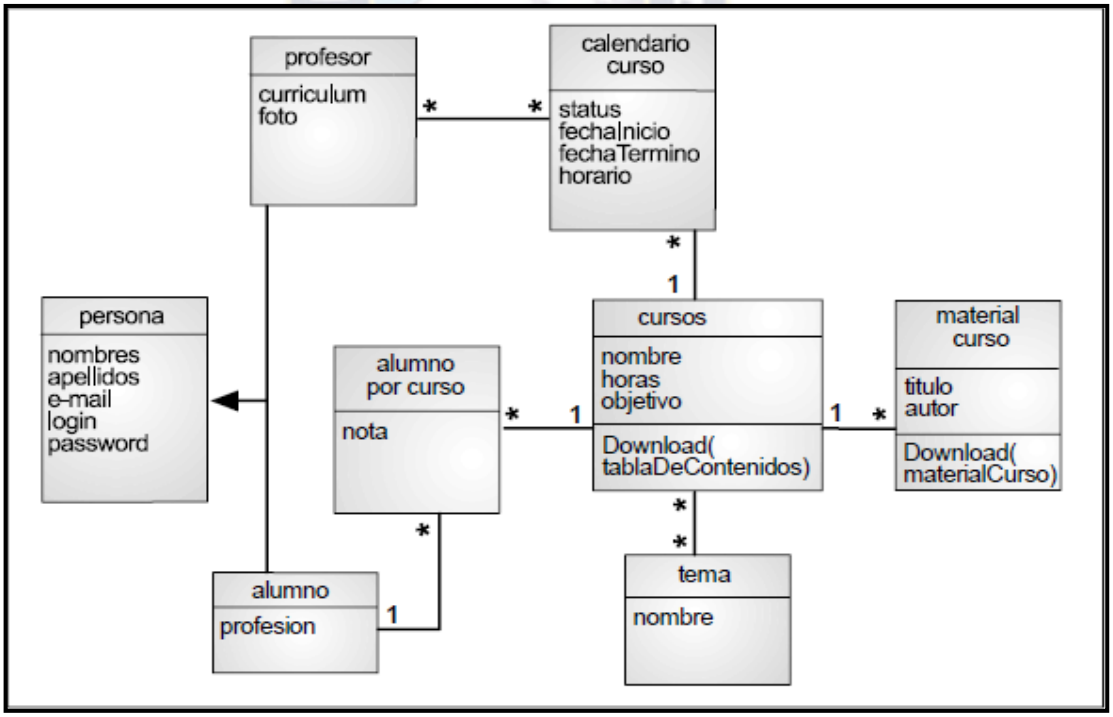


Figura 2.3 Ejemplo Diseño Conceptual
Fuente: (Giorgis, Muños & Horra, 2009)

2.2.1.2 DISEÑO NAVEGACIONAL

En esta fase se definen dos esquemas: el esquema de clases navegacionales y el esquema de contextos navegacionales. En OOHDM existe un conjunto de tipos predefinidos de clases navegacionales tales como nodos, enlaces y estructuras de acceso (índices y visitas guiadas) inducidas del esquema conceptual.

Los nodos son considerados básicos de información de las aplicaciones hipermedia, se definen como vistas orientadas a objeto de las clases definidas durante el diseño conceptual usando un lenguaje basado en query, permitiendo así que un nodo sea definido mediante la combinación de atributos de clases diferentes relacionadas en el diseño conceptual.

Los enlaces reflejan la relación de la navegación que puede explorar el usuario, es importante consideras que para un mismo esquema conceptual puede haber diferentes esquemas navegacionales y los enlaces van a ser imprescindibles para poder crear esas vistas diferentes. El enlace puede actuar como un objeto intermedio en un proceso de navegación como un puente de conexión entre dos nodos.

Estructuras de acceso actúan como índices que permiten al usuario encontrar de forma rápida y eficiente la información deseada. Los menús, los índices son ejemplos de estas estructuras que también se modelan como clases compuestas por un conjunto de referencias a objetos que son accesibles desde ella y una serie de criterios de clasificación de las mismas.

A continuación, el diseñador describe la estructura navegacional en términos de contextos navegacionales como se ve en la figura 2.4. Un contexto navegacional es un conjunto de nodos, enlaces, clases de contextos y otros contextos navegacionales (contextos anidados) que pueden ser definidos por comprensión o extensión, o por enumeración de sus miembros.

Los nodos se enriquecen con un conjunto de clases especiales que permiten presentar atributos así como métodos o comportamientos cuando se navega en un contexto particular. Durante esta etapa, es posible adaptar los objetos navegacionales para cada contexto.

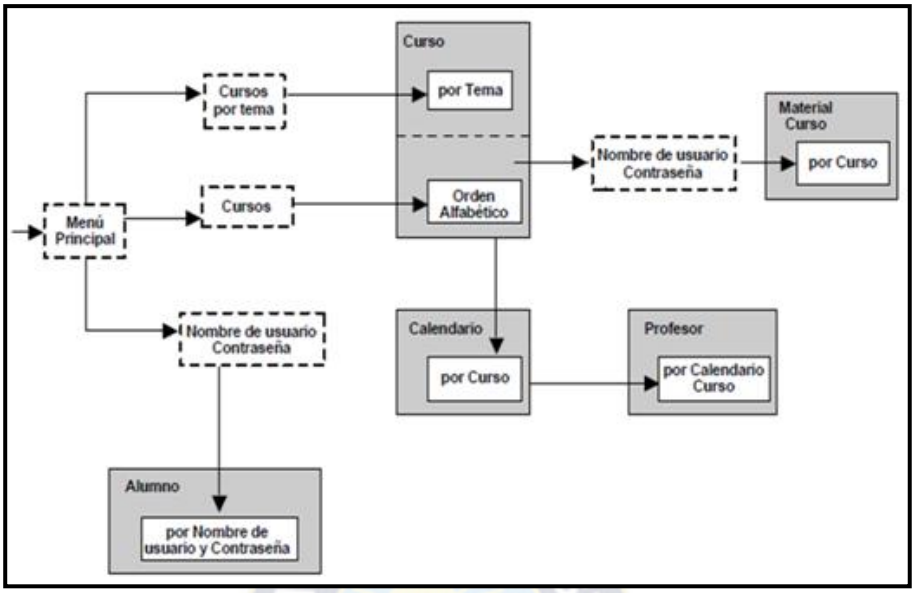


Figura 2.4 Ejemplo Diseño Navegacional
Fuente: (Giorgis, Muños & Horra, 2009)

2.2.1.3 DISEÑO INTERFAZ ABSTRACTA

En OOHDM se usa un acercamiento del diseño de Datos de Vista Abstractos (ADVs), para describir la interfaz del usuario de una aplicación de hipermedia, como se lo muestra en la figura 2.5.

Los ADVs han sido usados para representar interfaces entre medios de comunicación diferentes como usuario, una red, dispositivo o como una interfaz de uno o más Objetos de Datos Abstractos (ADOs) objetos que no soportan externamente eventos generados por el usuario. Los mecanismos de abstracción con los que se estructuran los objetos son la agregación y generalización/especificación.

En esta se define la apariencia de la interfaz de los objetos de navegación y otros objetos de la interfaz útiles como barras de menús y botones.

Así, se define la forma en la cual deben aparecer los contextos navegacionales. También se incluye aquí el modo en que dichos objetos de interfaz activarán la navegación y el resto de funcionalidades de la aplicación, esto es, se describirán los objetos de interfaz y se los asociará con objetos de navegación. La separación entre el diseño navegacional y el diseño de interfaz abstracta permitirá construir diferentes interfaces para el mismo modelo navegacional.

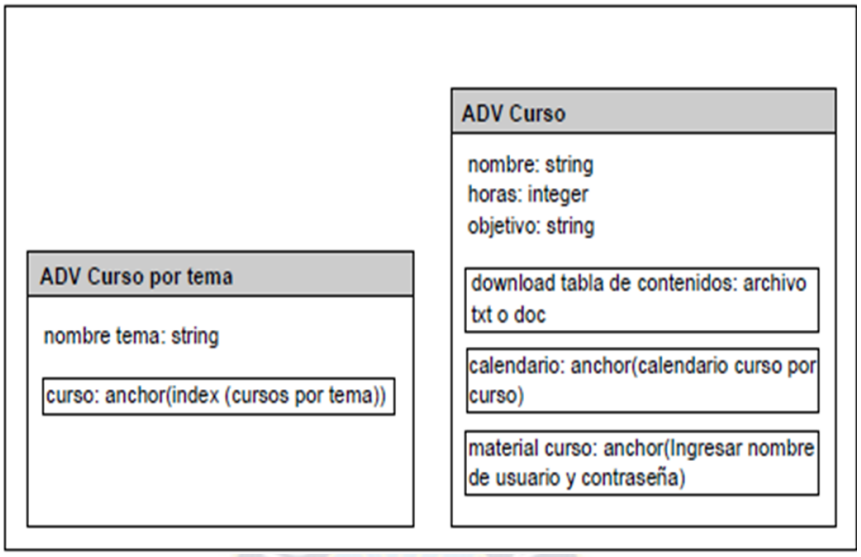


Figura 2.5 Ejemplo de diseño de ADVs
Fuente: (Rossi & Schwave, 2014)

2.2.1.4 IMPLEMENTACIÓN

En esta fase el diseñador debe implementar el diseño, hasta ahora todos los modelos fueron construidos en forma independiente de la plataforma de implementación, en esta fase se tiene en cuenta el entorno particular en el cual se va a correr la aplicación. Al llegar a esta fase el primer paso que debe realizar el diseñador es definir los ítems de información que son parte del dominio del problema. Debe identificar también cómo son organizados los ítems de acuerdo con el perfil del usuario y su tarea; decidir qué interfaz debería ver y cómo debería comportarse, a fin de implementar todo en un entorno web como se ve en la figura 2.6, el diseñador debe decidir además qué información debe ser almacenada.

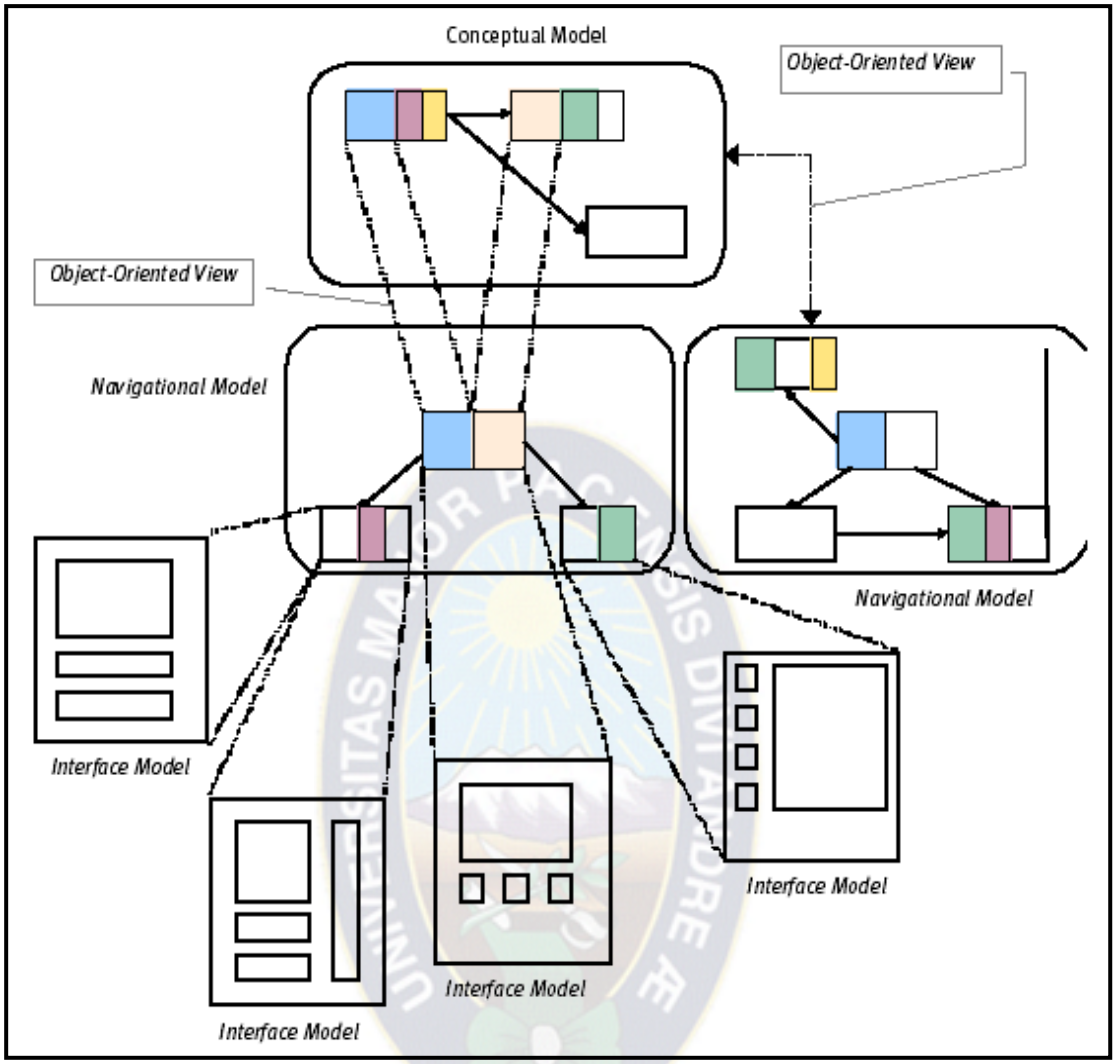


Figura 2.6 Relaciones entre los diseños conceptuales, navegacional y los objetos de interfaz
Fuente: (Rossi & Schwave, 2014)

2.3 CALIDAD DEL PRODUCTO SOFTWARE CON ISO-9126

ISO 9126 es un estándar internacional para la evaluación del Software, fue originalmente desarrollado en 1991 para proporcionar un esquema para la evaluación de calidad del software. La normativa define seis características de la aplicación, estas seis características son dividas en un número de sub- características, las cuales representan un modelo detallado para la evaluación de cualquier sistema informático.

2.3.1 CARACTERÍSTICAS NORMA ISO-9126

El modelo establece diez características, seis que son comunes a las vistas interna y externa y cuatro que son propias de la vista en uso. A continuación se describen las características y sub-características propias de este estándar que se encuentran dentro de las vistas interna y externa, las cuales usaremos para evaluar el software de CMI (Gómez, López, Migani & Otazú, 2009).

- **Funcionalidad:** capacidad del software de proveer los servicios necesarios para cumplir con los requisitos funcionales. Se sub caracteriza por ser idóneo, hace referencia a que si el software desempeña las tareas para las cuales fue desarrollado. Goza de exactitud, la cual evalúa el resultado final que obtiene el software y si tiene consistencia a lo que se espera de él. Es interoperable lo cual consiste en revisar si el sistema puede interactuar con otro sistema independiente y es seguro cuando se verifica si el sistema puede impedir el acceso a personal no autorizado.
- **Fiabilidad:** Capacidad del software de mantener las prestaciones requeridas del sistema, durante un tiempo establecido y bajo un conjunto de condiciones definidas. Goza de madurez, se debe verificar las fallas del sistema y si muchas de estas han sido eliminadas durante el tiempo de pruebas o uso del sistema. Otra sub característica es la recuperabilidad, que es la capacidad de verificar si el software puede reasumir el funcionamiento y restaurar datos perdidos después de un fallo ocasional. La Tolerancia a fallos, evalúa si la aplicación desarrollada es capaz de manejar errores.
- **Usabilidad:** Esfuerzo requerido por el usuario para utilizar el producto satisfactoriamente. El Aprendizaje, determina que tan fácil es para el usuario aprender a utilizar el sistema. La Comprensión, evalúa que tan fácil es para el usuario comprender el funcionamiento del sistema La Operatividad, determina si el usuario puede utilizar el sistema sin mucho esfuerzo. La Atractividad, verifica que tan atractiva se ve la interfaz de la aplicación.
- **Eficiencia:** Relación entre las prestaciones del software y los requisitos necesarios para su utilización. El Comportamiento en el tiempo, verifica la rapidez en que responde el sistema. El Comportamiento de recursos, determina si el sistema utiliza los recursos de manera eficiente.

- **Mantenibilidad:** Esfuerzo necesario para adaptarse a las nuevas especificaciones y requisitos del software. La Estabilidad, verifica si el sistema puede mantener su funcionamiento a pesar de realizar cambios. Facilidad de análisis**,** determina si la estructura de desarrollo es funcional con el objetivo de diagnosticar fácilmente las fallas. La Facilidad de cambio, verifica si el sistema puede ser fácilmente modificado. La Facilidad de pruebas, evalúa si el sistema puede ser probado fácilmente.
- **Portabilidad:** Capacidad del software ser transferido de un entorno a otro. La Capacidad de instalación, verifica si el software se puede instalar fácilmente. Capacidad de reemplazamiento, determina la facilidad con la que el software puede remplazar otro software similar. La Adaptabilidad, el software se puede trasladar a otros ambientes. La Co-Existencia, el software puede funcionar con otros sistemas.

Cada una de las características debe ser evaluada dentro del software basándonos en pruebas de funcionamiento, medición de rendimiento y pruebas con usuarios que harán uso del sistema.

2.4 ESTIMACIÓN DEL TAMAÑO POR PUNTOS FUNCIÓN

Los puntos de función PF, son una métrica para establecer el tamaño y complejidad de los sistemas informáticos basada en la calidad de funcionalidad requerida y entregada a los usuarios, miden el tamaño lógico o funcional de las aplicaciones software basado en los requisitos funcionales del usuario.

A partir de su aparición en 1979 han surgido una serie de diversas propuestas para calcular el tamaño funcional de un proyecto software. Los métodos para la estimación de los PF deben ajustarse a la norma ISO/IEC 14143-1, cualquiera de los métodos que se ajuste a este estándar puede ser usado para la estimación del tamaño funcional.

FP Lite es un método de estimación paramétrico derivado del método de análisis de punto función FPA definido por el Grupo Internacional de Usuarios de Puntos Función IFPUG. Fue creado por la empresa “David Consulting Group”, en esta versión se considera que todos los elementos tiene una complejidad media.

2.4.1 MEDICIÓN DE PF CON EL MÉTODO PF LITE

Los cinco elementos funcionales son:

Datos-ILF: Ficheros Lógicos Internos, es un grupo de datos relacionados lógicamente o información de control, identificable por el usuario referenciado por la aplicación y almacenada en los límites de la aplicación. El objetivo principal es almacenar datos referenciados mediante uno o más procesos elementales, típicamente se corresponden con las entidades que aparecen en un diagrama Entidad-Relación.

Interfaces-ELF: Ficheros Lógicos Externos, es un grupo de datos relacionados lógicamente o información de control, identificable por el usuario pero mantenido dentro de los límites de otra aplicación, cuyo objetivo principal es almacenar datos dentro de los límites de otra aplicación.

Entradas-EI: Entradas Externas, proceso elemental que procesa datos o información de control que proviene de fuera de los límites de la aplicación los datos pueden venir desde otra aplicación de una pantalla de ingreso de datos, el objetivo fundamental es mantener uno o más ILF.

Salidas-EO: Salidas Externas, proceso elemental que envía datos o información de control hacia fuera de los límites de la aplicación el objetivo fundamental es presentar información al usuario mediante un proceso lógico diferente o adicional al de la recuperación de datos.

Consultas-EQ: Consultas Externas, proceso elemental que envía datos o información de control fuera de los límites de la aplicación cuyo principal objetivo es presentar información al usuario a través de la recuperación de datos.

A causa de los problemas para la evaluación de la complejidad de los elementos funcionales se propuso el método PF Lite, en el que se considera que todos los elementos funcionales tiene una complejidad media para de esta manera simplificar en tres pasos la metodología FPA, pasos que se describen en la Figura 2.7.

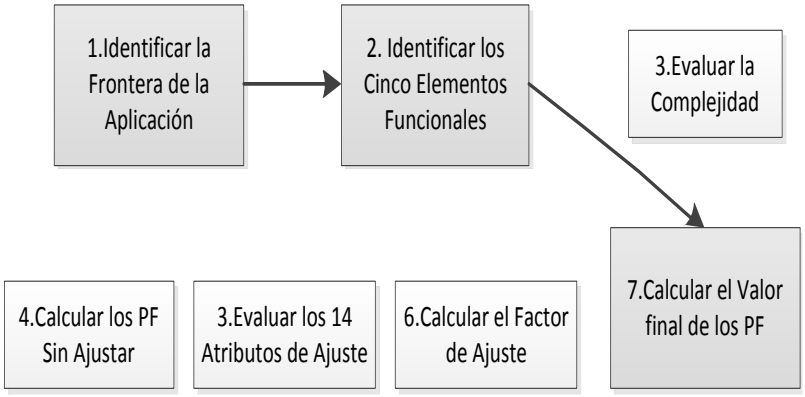


Figura 2.7 Proceso para calcular PF de FP Lite
Fuente: (Plattini, García, Garzás & Genero, 2008)

El valor final de los puntos función para proyectos en desarrollo está dado por la fórmula:

$$PF = \sum_{k=1}^{n} Ei * \mathrm{Ci}$$

Siendo:

- Ei cada elemento funcional (EI, EQ, EO, ILF, ELF)
- Ci la complejidad media considerada para cada uno de los tipos de elementos

La tabla 2.2 muestra el valor de la complejidad media para cada elemento funcional.

Elemento	Peso
ILF	10
ELF	7
EI	4
EO	5
EQ	4

Tabla 2.2 Complejidad media para cada elemento funcional
Fuente: (Plattini, García, Garzás & Genero, 2008)

Al calcular los PF Lite obtenemos una estimación del tamaño funcional del proyecto de desarrollo por lo que se aconseja considerar un rango del valor obtenido PF±20%.

2.4.2 ESTIMACIÓN EN BASE A PF

Para la estimación de un proyecto software utilizando como base a los PF y considerando que no existen datos históricos de proyectos anteriores se puede utilizar las formulas propuestas por el Internacional Software Benchmarking Estándar Group ISBSG.

Estimación del Esfuerzo: Para la estimación del esfuerzo del equipo de desarrollo se utiliza la fórmula:

$$Esfuerzo = C * PF^{E}$$

Dónde PF es la cantidad de puntos función del proyecto que se quiere estimar y C, E son factores de corrección para el cálculo del esfuerzo, las unidades están expresadas en horas.

Estimación de la duración: Para estimar la duración se utilizara la siguiente formula:

$$Duración = C * PF^{E}$$

La duración esta expresada en meses, PF es cantidad de puntos de función del proyecto a desarrollar C, E son factores de corrección. La duración total del proyecto se calcula considerando 20 jornadas laborales de 8 horas por mes.

Si el proyecto no se encuentra dentro de las categorías de la tabla este se puede calcular con la siguiente formula:

$$Duración = 0.411 * PF^{(0.328)}$$

A partir del cálculo anterior es posible conocer el número de personas que se estima podrían trabajar en el proyecto, para ello consideraremos que una persona trabaja 20 días al mes y 8 horas por día.

$$Cantidad\ de\ personas = \frac{Esfuerzo\ Total}{Duracion\ total * 20 * 8}$$

Estimación del costo: Para estimar el costo total del proyecto se utilizara la fórmula:

$$Coste = Esfuerzo * Coste_Medio_Hora$$

Dónde:

Esfuerzo: Es el total de horas estimado.

Costo Medio Hora: El costo promedio de la hora por persona que se asigna al proyecto.

2.5 LOS DOCUMENTOS Y SU IMPORTANCIA

Ninguna institución gubernamental u organización privada podría sobrevivir sin documentar sus actividades, ninguna oficina podría funcionar bien si tuviera que depender solamente de la memoria de sus funcionarios para recordar las transacciones efectuadas.

Sin documentos cualquier administración organizada dejaría rápidamente de funcionar. Los documentos y específicamente la información que contienen, son uno de los recursos fundamentales que las instituciones necesitan para poder realizar sus operaciones eficazmente.

Así como una organización podría seguir funcionando con escasos recursos humanos, económicos o materiales, no podría funcionar si no mantuviera sus documentos y fuera posible el acceso a los mismos.

En el contexto de las actividades administrativas los documentos ayudan a:

- Proporcionar una memoria corporativa.
- Formular políticas.
- Tomar decisiones apropiadas.
- Alcanzar más eficiencia, productividad y coherencia.
- Cumplir con los requisitos legales y las regulaciones vigentes.
- Proteger los intereses de la organización, del personal y sus clientes.
- Reducir los riesgos relacionados con la falta de pruebas de decisiones y acciones.
- Documentar actividades y logros.

En el sector público los documentos son fundamentales dentro el concepto de democracia, la información documentada apoya la protección de los derechos humanos, el estado de derecho y el

trato justo e igual de los ciudadanos, los ciudadanos tienen derecho a esperar que haya pruebas documentales confiables de las decisiones y acciones del gobierno (Alberch & Fugueras, 2003).

Por ejemplo los documentos son imprescindibles para determinar con precisión si los recursos recaudados por el estado en interés de los ciudadanos son gastados de una manera que los beneficie en servicios de salud, educación y otros.

Más aun los documentos son los que protegen los derechos de los ciudadanos, el derecho de propiedad de la tierra documentado mediante registros catastrales y títulos de propiedad, o los derechos a una jubilación o una pensión documentados mediante los expedientes de personal, llamados también legajos personales. Cuando los derechos personales no son respetados son los documentos los que permiten a los ciudadanos solicitar u obtener la reparación legal.

2.5.1 DEFINICIÓN DEL TÉRMINO DOCUMENTO

(a) Un documento es toda aquella información contenida y registrada sobre cualquier soporte material y que es producido, recibido y conservado por las instituciones, organizaciones o personas, durante el desarrollo de sus actividades. Es, por tanto, un testimonio de la actividad humana**.**

(b)

(c) Las definiciones existentes sobre documento cambian según el objeto de estudio de la ciencia que lo utiliza, por ejemplo para los historiadores documento es “cualquier soporte que lleva fijado una información referida a un pasado humano y que haga referencia al hombre en algún sentido”, para los juristas en cambio documento es “todo soporte que lleva fijado un texto y que goza de valor legal para imponer algo o para probar un hecho” (Guerra, 2000).

Los documentos pueden ser textuales como ser: manuscritos, mecanografiados o impresos, iconográficos: mapas, planos, dibujos, fotografías, diapositivas, etc., sonoros: discos, cintas magnetofónicas, discos compactos, audiovisuales: cintas de video, películas cinematográficas, electrónicos: disquetes, CD.ROM, bases de datos, etc., virtuales: webs, blogs, etc., y cualquier otra

propuesta que el avance tecnológico pudiera plantear en el futuro. Los documentos son información registrada en cualquier forma o medio, creados o recibidos, conservados y usados por una organización o un individuo en la transacción de negocios o la realización de asuntos y que se guardan como testimonio de dicha actividad.

El carácter testimonial es lo que distingue a los documentos de otros tipos de datos o informaciones, en primer lugar los documentos son testimonio, la información es el conocimiento al cual se le da sentido por su contexto.

Los datos son la representación de la información de una manera formalizada apropiada para la comunicación, la interpretación y la tramitación. Por ejemplo, los materiales publicados pueden proporcionar información importante para apoyar las actividades administrativas, pero no proporcionan pruebas de tales actividades.

Los documentos son más que los datos que contienen. Los documentos ofrecen datos e informaciones, significado y contexto al estar intrínsecamente ligados a la actividad que documentan y de la que dan testimonio. Sólo los documentos sirven como prueba de la realización de trámites.

El término documento es usado en el presente Proyecto de Grado, en el sentido de una unidad separada de información registrada. Puede ser una entidad física e intelectual como una carta, un memorándum, un informe o una impresión de un documento creado en computadora o bien una entidad intelectual solamente como un documento creado y guardado en forma electrónica en una computadora.

2.5.2 CARACTERÍSTICAS ESENCIALES DE LOS DOCUMENTOS.

Los documentos despejan las actividades de las que son producto. Debido a que derivan de sucesos reales, representan una imagen congelada la que fija una acción en su contexto articular de: función, autoridad, lugar y momento. Es posible establecer ciertas características esenciales:

- Son estáticos en su forma.
- Tienen autoridad.

- Son singulares cuando están en su contexto.
- Son auténticos.

2.5.3 CICLO DE VIDA DE LOS DOCUMENTOS

Los documentos pasan por etapas de vida administrativa que a su vez dan lugar a diferentes niveles de archivo como se puede observar en la figura 2.8.

2.5.3.1 PRIMERA ETAPA DOCUMENTACIÓN ACTIVA (ARCHIVO DE OFICINA)

Durante esta etapa el documento es de circulación y tramitación por los canales regulares, en busca de respuesta y solución al asunto que le ha dado origen y además hay un manejo frecuente de parte del funcionario responsable de su tramitación. Esta documentación forma parte del **"ARCHIVO DE GESTIÓN"**.

2.5.3.2 SEGUNDA ETAPA DOCUMENTACIÓN SEMI-ACTIVA (ARCHIVO CENTRAL)

Una vez recibida la respuesta o solución al asunto iniciado, el documento que la sustenta ha de seguir siendo guardado. En este período formará parte del **"ARCHIVO CENTRAL".**

2.5.3.3 TERCERA ETAPA DOCUMENTACIÓN INACTIVA (ARCHIVO HISTÓRICO)

Finalmente en esta etapa el documento adquiere un valor permanente o histórico, restringiéndose su consulta a su carácter cultural e informativo, principalmente con fines de investigación. Su archivamiento y conservación será definitivo en el **"ARCHIVO HISTÓRICO".**

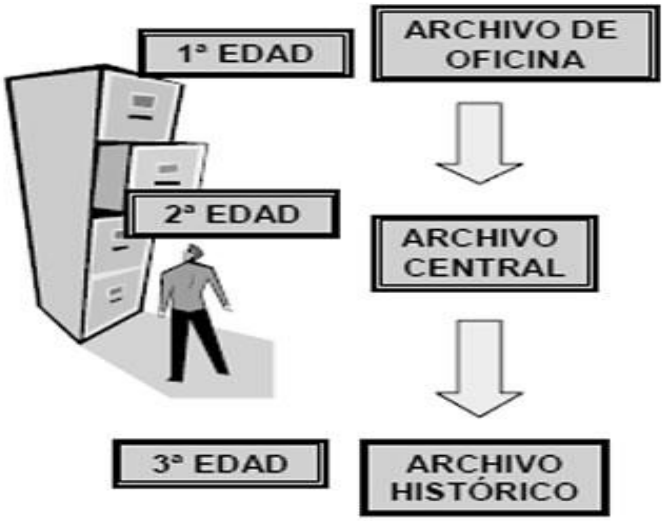


Figura 2.8 Ciclo de vida de los Documentos
Fuente: (Cerdá, 2008)

2.5.4 CALIDAD DE LOS DOCUMENTOS

Así como es necesario crear y conservar los documentos para proporcionar una prueba documental, esos documentos tienen que tener calidad e integridad suficientes. Para que las instituciones estén en condiciones de realizar sus funciones administrativas eficientemente y puedan ser responsables, tienen que mantener documentos íntegros y fieles. Sin ellos los funcionarios no podrían realizar sus cometidos adecuadamente. Para llevar a cabo su propósito de proporcionar una prueba confiable, los documentos deben ser fieles, completos e integrales.

Los documentos, ya sean en papel o en forma electrónica deben ser según (Doyle, 1991):

- **Integrales**: deberá producirse un documento para cada transacción administrativa, de la que dará testimonio.
- **Fiel**: un documento debe registrar con precisión la transacción para la cual fue creado.
- **Adecuado**: un documento debe ser adecuado para los propósitos para los cuales se conserva, es decir el documento debe contener la información necesaria para dar prueba de la transacción que documenta.
- **Completo y significante**: así como debe contener información suficiente para documentar una transacción, un documento debe incluir información suficiente sobre el contexto en el cual fue

creado y usado, sobre su estructura o su forma física y sobre sus vínculos con otros documentos, para que sea posible comprender su contenido.

- **Comprensible y utilizable**: debe ser posible extraer del documento la información que contiene y que pretende comunicar; y ser posible usarlo sin que se pierda información;
- **Auténtico**: debe ser posible comprobar que el documento es lo que dice ser.
- **Inalterado**: ninguna información del documento habrá sido borrada, alterada o perdida, bien sea intencionalmente o por accidente, una vez que la transacción que dio origen al mismo se ha finalizado, en otras palabras los documentos se mantendrán seguros y se evitará el acceso a ellos o a su uso no autorizado.

El documento además cumplirá con cualquier requisito de regulación y de responsabilidad de la organización que lo creó, por ejemplo requisitos de auditoría. De esto se desprende que los documentos deben ser integrales y completos, confiables y auténticos, deber ser administrados con sistemas de gestión que los controlan durante su ciclo de vida, desde su creación hasta su disposición final y su tratamiento en los archivos históricos.

Esto requiere que participen en su cuidado desde el comienzo de su ciclo de vida y que se ocupen de ellos como parte de una atención ininterrumpida y continua, por ejemplo no vale la pena diseñar un esquema de clasificación que no se equipara con los procesos administrativos que fueron causa de que los documentos se clasificaran. Igualmente es ilógico esperar que los documentos lleguen al Archivo sin haberlos evaluado y haber establecido plazos de conservación y determinado qué es lo que debe guardarse.

2.6 LA ARCHIVÍSTICA

El Diccionario de la Terminología Archivística español daba en 1993 la siguiente definición: “Archivística: Disciplina que trata de los aspectos teóricos y prácticos (tipología, organización funcionamiento, planificación, etc.) de los archivos y el tratamiento archivístico de sus fondos”. Es una definición excesivamente corta para abarcar una ciencia tan amplia.

Por ello considerando todos los aspectos del concepto de Archivística, se da esta definición extensiva de la misma abarcando con finalidad didáctica, todos los elementos conceptuales que

constituyen su objetivo y su campo de acción : “Archivística es la ciencia que se ocupa de los archivos en sus aspectos teóricos y prácticos, estableciendo principios inalterables y estudiando técnicas adecuadas de gestión de documentos, administración y tratamiento técnico de archivos, así como la función jurídica, administrativa y científica de los mismos desde un punto de vista archivístico o de ciencias y técnicas diversas, y su relación con las entidades productoras de los conjuntos orgánicos de documentos, a fin de mejorar y hacer más accesible la información de los fondos documentales” (Fuster, 1999).

La archivística moderna es una ciencia en desarrollo relativamente nueva, contrariamente al desarrollo de los archivos que son mucho más antiguos como el mismo oficio del archivero. No se debe confundir la historia de los archivos y documentos con la historia de la archivística. La formulación del francés Natalis de Wally el 24 de abril de 1841 del principio de respeto a los fondos es el punto de partida o la base para que la archivística desarrolle sus propias actividades (Santos, 2006).

El otro principio clásico de la ordenación natural es el principio del orden original. Según él la documentación debe conservarse en el orden que recibió durante su función administrativa" A partir de estos principios, la archivística como teoría y práctica desarrolla innovadores conceptos archivísticos que hoy son fundamentales en la organización documental.

2.6.1 PRÁCTICA ARCHIVÍSTICA

Partiendo del principio archivístico enunciado por Wally sobre el respeto a los fondos se plantea ahora la práctica. Fondo es un conjunto de documentos formados en series acorde a la estructura orgánico funcional de un organismo productor sea este persona física o jurídica, pública o privada.

El punto de partida para cualquier tratamiento archivístico es conocer al que produce, origina o engendra los documentos. En la práctica se encuentra elementos que se deben tomar en cuenta para no confundir una actividad con otra. No es lo mismo decir tratamiento archivístico que procesamiento técnico que se adecua más bien a la bibliotecología. La metodología archivística en el siglo XXI vive cada vez más sistematizada y estandarizada (Santos, 2006).

2.6.2 CONCEPTO DE ARCHIVO

La palabra archivo generalmente tiene tres acepciones: como contenido documental, como institución y como lugar de conservación. El Diccionario de Terminología Archivística presenta tres definiciones de archivo teniendo en cuenta cada uno de los elementos conceptuales.

"ARCHIVO: Conjunto orgánico de documentos productos y/o recibidos en el ejercicio de sus funciones por sus funciones por las personas físicas o jurídicas, publicas y privadas. La institución cultural donde se reúnen, conserva, ordenan y difunden los conjuntos orgánicos de documentos para la gestión administrativa, la información, la investigación y la cultura. El archivo también es el local donde se conservan y consultan los conjuntos orgánicos de documentos" (Fuster, 1999). En la tabla 2.3 se muestra la diferencia entre un almacén de papel y un archivo de documentos.

ALMACÉN DE PAPEL	ARCHIVO DE DOCUMENTOS
Papeles apilados, diarios, boletines, fotocopias...	Documentos importantes que hay que conservar, clasificarlos y ordenarlos.
Pocas personas saben lo que hay, ni como encontrar nada.	Base de datos donde se relaciona el inventario de los documentos con la descripción correspondiente.
Los documentos llegan como pueden y nadie sabe de dónde proceden.	Traslado organizado (transferencia).
Cada persona conserva o elimina lo que le parece oportuno.	Criterios únicos de conservación y eliminación.
Es difícil encontrar y consultar un documento.	Consulta y préstamo de la documentación de forma ágil.
No hay normas de archivo ni clasificación.	Normas internas de tratamiento de la documentación siguiendo las directrices del archivo y cuadro de clasificación corporativo.

Tabla 2.3 Diferencias entre un almacén de papel y un archivo de documentos
Fuente: (García, 2014)

2.6.3 FASES Y OPERACIONES DEL TRATAMIENTO ARCHIVÍSTICO

Con la aplicación de estas fases y operaciones se podría encarar sistemáticamente la organización de los documentos. Seguramente aún no es suficiente la metodología planteada pero posibilita iniciar y controlar la desorganización latente cuando se está frente a frente con los documentos (Cortez, 1981).

En la tabla 2.4 se muestra las fases y opraciones a las cuales son sometidas los documentos además de una descripción que contempla los conceptos a los que se refiere cada fase y operación.

FASES Y OPERACIONES	DEFINICIONES
Identificación	Fase que consiste en conocer las funciones, actividades, estructura y normativa de los órganos productores que generan los documentos. En esta fase se establecen los tipos documentales y las series documentales.
Clasificación	Operación intelectual y física que va separando y agrupando los documentos producidos y recibidos por una persona natural o jurídica de acuerdo a los principios de procedencia y orden original.
Ordenación	Operación física que dispone o coloca los documentos dentro de sus series de acuerdo a su orden natural pueden ser cronológicas, alfabéticas, numéricas, geográficas.
Valoración	Fase que consiste en determinar y señalar los valores: primarios (Administrativo, legal, fiscal, jurídico, contable) y secundarios (Informativo, histórico).
Descripción	Fase que consiste en elaborar instrumentos de control y consulta como por ejemplo los inventarios y relaciones de entrega entre otros.
Valoración	Operación física que consiste en colocar o ubicar los documentos en un depósito determinado. Los documentos pueden encontrase en cajas, legajos, paquetes o encuadernados pero siempre con la signatura topográfica y datos de la serie o tipo documental

Tabla 2.4 Fases y Operaciones del Tratamiento Archivístico
Fuente: (Santos, 2006)

2.6.4 INSTRUMENTOS DE DESCRIPCIÓN

El resultado del proceso de descripción se materializa en los denominados instrumentos de descripción. La norma de descripción ISAD (G*)*, como se verá más adelante, no contempla la forma de presentar los elementos de descripción, por lo que los instrumentos de descripción suelen ser muy variados. Su finalidad en general es dar información sobre la documentación de un archivo (total o parcial) para localizar y recuperar dicha documentación. Al igual que cualquier otro tipo de documento de archivo, los documentos electrónicos también deben ser objeto de descripción y tratamiento archivístico, y su descripción debe estar relacionada con los instrumentos descriptivos disponibles para la documentación cuyo soporte siga siendo el papel (Gavilán, 2009).

2.6.4.1 GUÍA

Instrumento que proporciona información general, exacta y concisa sobre todos o parte de los fondos de uno o varios archivos. Describe globalmente las grandes agrupaciones documentales (fondos), esboza la historia de los organismos productores y facilita información auxiliar básica acerca del archivo y los servicios disponibles: horarios, instrumentos de descripción, publicaciones sobre los fondos, etc. Es el primer instrumento que necesita el usuario para acceder a un archivo. Se trata, por lo tanto, de una herramienta genérica y aproximativa útil para ofrecer al exterior una imagen panorámica de los fondos del archivo.

2.6.4.2 INVENTARIO

Instrumento de consulta que describe las series documentales que componen las unidades archivísticas (libros, expedientes, etc.), dispuestas según el orden que tienen en el cuadro de clasificación y reproduciendo su estructura. El objetivo común de cualquier inventario es la descripción de las series que conforman un fondo para que el usuario pueda conocer el contenido de un fondo y su rápida localización, al tiempo que también permite al personal del archivo controlar la documentación. Para facilitar su consulta y operatividad, los inventarios suelen estar informatizados, en bases de datos generalmente es lo primero que se informatiza en un archivo. Ahora bien pocos se encuentran accesibles a través de Internet, la mayoría son de consulta en el propio archivo.

2.6.4.3 CATÁLOGO

Instrumento de consulta que describe las unidades documentales (documentos o expedientes) de una serie, o parte de ella, que traten de un mismo asunto. Se trata por tanto de un instrumento de descripción que requiere mucho tiempo para su elaboración, pues implica el análisis exhaustivo de los documentos, y por ello sólo se aplica para describir determinadas agrupaciones o categorías de documentos que por su valor estratégico en las actividades de la organización o patrimonial así lo requieran, por ejemplo series históricas o de gran demanda por parte de los investigadores.

2.6.4.4 ÍNDICE

Instrumento de consulta en el que se listan alfabética o numéricamente, términos onomásticos, listados de nombres de personas naturales o jurídicas, toponímicos, listados de nombres de sitios o lugares, cronológicos, listados consecutivo de fechas y temáticos, listas de materias acompañados de referencias para su localización. Actualmente informatizados, los índices son instrumentos de consulta que completan la información de otros instrumentos; no sustituyen en ningún caso a los instrumentos descriptivos: en todo caso, los mejoran y complementan, incrementando y mejorando puntos de acceso. Contienen datos concretos de los documentos que permiten acceder de forma orientada y controlada a la información.

2.7 NORMA ISAD(G)

La norma Internacional General de Descripción Archivística, ISAD (G) se concibe como una guía general para la elaboración de descripciones archivísticas: contiene reglas generales para la descripción que se pueden aplicar a cualquier tipo documental y soporte físico de la documentación, excepto para materiales especiales.

No se concibe como una norma exclusiva sino que ha de ser utilizada conjuntamente con las respectivas normas nacionales. En general ISAD (G) se caracteriza por ser una norma, con una estructura muy definida pero muy flexible en cuanto a nivel de descripción, profundidad de descripción, detalle de descripción de los diferentes elementos (ISAD(G), 2000).

El número de niveles de descripción depende del volumen, complejidad del fondo, demandas de información de los usuarios, valor histórico de los documentos, etc. Las cinco unidades de descripción básicas según la norma ISAD (G) son: Fondo, Sub fondo, Serie, Expediente, Unidad documental. El nivel de descripción fundamental, a partir del cual se establece el plan de descripción de un archivo es el fondo.

La práctica de la descripción en sus diferentes acepciones siempre ha estado presente en archivos y oficinas el recurso más elemental es reflejar en la parte extrema de las cajas o legajos un resumen del contenido de cada unidad de instalación fiando toda la búsqueda a la memoria o institución de la persona responsable, convertido en guía humana, único e insustituible o cualquiera que mediante "visualización" puede localizar el documento requerido, en la figura 2.9 se muestra el modelo de los niveles de un fondo.

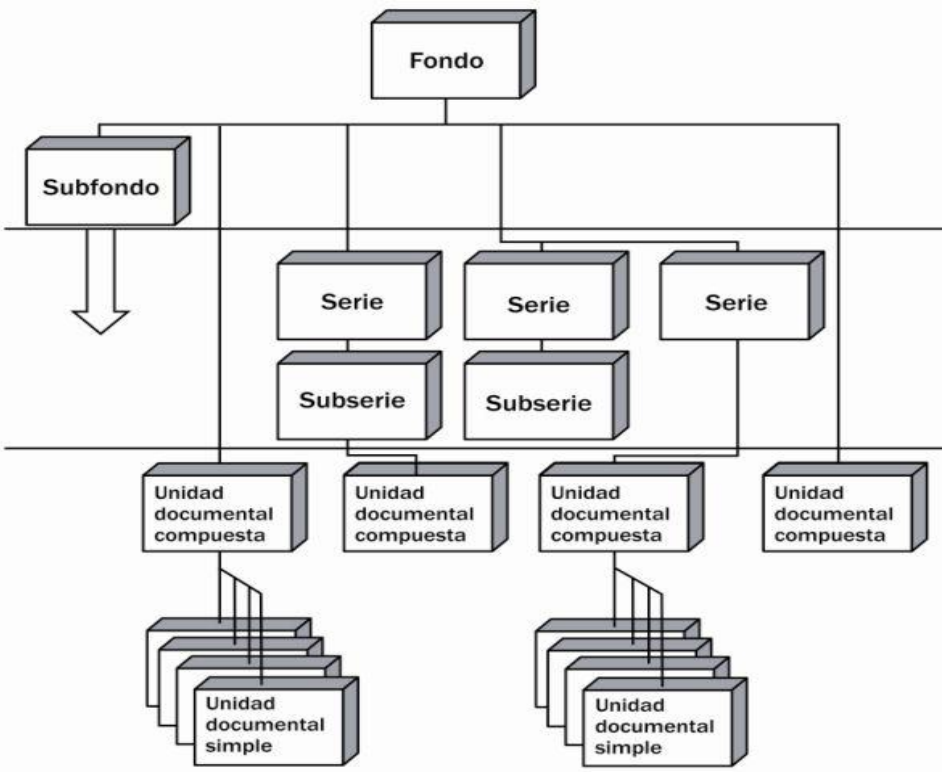


Figura 2.9 Modelo de los niveles de un fondo
Fuente: (ISAD(G), 2000)

Este tipo de prácticas no es un sistema de descripción. Un sistema normalizado debe poder ejercer un control real de acceso a los documentos, tener la capacidad de recuperar de un modo rápido ágil y eficiente cualquier documento o información que se necesite. A continuación en la tabla 2.5 se muestra la equivalencia entre agrupaciones documentales y los instrumentos de descripción archivística (Heredia, 1995).

Nivel	Agrupaciones Documentales	Instrumentos de Descripción
I	Archivo Fondo Sección de Fondo Sección de Archivo	Guías
II	Series Documentales	Inventario
III	Piezas Singulares Unidades Archivísticas	Catálogo
IV		Índice

Tabla 2.5 Equivalencia entre agrupaciones documentales y los instrumentos de descripción archivística
Fuente: (Heredia, 1995)

2.7.1 DEFINICIÓN Y ALCANCE

La norma internacional general de descripción archivística en adelante ISAD (G) constituye una herramienta de gestión muy valiosa para archivos, brindando especificaciones elementales para la elaboración de descripciones archivísticas compatibles a nivel nacional e internacional, que pueden aplicarse con independencia del tipo documental o soporte físico de los documentos de archivo.

2.7.2 DESCRIPCIÓN MULTINIVEL

Si se describe un fondo como un todo, éste debe representarse en una descripción utilizando los elementos de la descripción que se especifican más adelante. Si es necesario describir las partes que integran el fondo, estas pueden describirse por separado, utilizando también estos mismos elementos. La suma total de estas descripciones, jerárquicamente unidas entre sí representa el fondo y las partes descritas. Esta técnica de descripción se denomina descripción multinivel. Al establecer

una jerarquía de descripciones deben aplicarse las cuatro reglas fundamentales de la descripción multinivel (ISAD(G), 2000).

2.7.3 REGLAS DE DESCRIPCIÓN MULTINIVEL

Para la correcta manipulación y uso de los documentos basado en las normas documentales se hace la utilización de cuatro reglas fundamentales y se especifican en los siguientes puntos.

2.7.3.1 DESCRIPCIÓN DE LO GENERAL A LO PARTICULAR

Representar el contexto y la estructura jerárquica del fondo y las partes que lo integran. El nivel siguiente y en los sucesivos, proporcionan información de las partes que se están describiendo. Se debe presentar las descripciones resultantes en una relación jerárquica de la parte al todo procediendo de lo más general, el fondo a lo más particular.

2.7.3.2 INFORMACIÓN PERTINENTE PARA EL NIVEL DE DESCRIPCIÓN

Representar exactamente el contexto y contenido de la unidad de descripción. Solo se debe proporcionar aquella información adecuada al nivel que se está describiendo. Por ejemplo, no dar información detallada del contenido de los expedientes si la unidad de descripción es un fondo; no relatar la historia institucional de todo un departamento si el productor de la unidad de la descripción es una división o sección del mismo.

2.7.3.3 VINCULACIÓN DE LAS DESCRIPCIONES

Evitar la información redundante en las descripciones archivísticas jerárquicamente relacionadas. En el nivel superior que corresponda se debe dar la información común a las partes que lo componen y no repetir en el nivel inferior ya proporcionada en un nivel superior.

2.7.3.4 NO REPETICIÓN DE LA INFORMACIÓN

Evitar la información redundante en las descripciones archivísticas jerárquicamente relacionadas. En el nivel superior que corresponda se debe dar la información común a las partes que lo componen y no repetir en el nivel inferior ya proporcionada en un nivel superior.

2.7.4 ESTRUCTURA Y CONTENIDO DE LA ISAD(G)

La ISAD (G) determina la representación de la información mediante 26 elementos que pueden ser combinados para construir la descripción de una entidad archivística. Estos elementos se hallan agrupados en 7 áreas de información descriptiva como se ve en la tabla 2.6.

ÁREAS	ELEMENTOS DE DESCRIPCIÓN	OBJETIVO
ÁREA DE IDENTIFICACIÓN	Código de referencia	Identificar de un modo único la unidad de descripción y establecer el vínculo con la descripción que la representa.
	Titulo	Denominar la unidad de descripción.
	Fecha	Identificar consignar la(s) fecha(s) de la unidad de descripción.
	Nivel de descripción	Identificar el nivel de organización de la unidad de descripción.
	Volumen y soporte de la unidad de descripción (cantidad, tamaño o dimensiones)	Identificar y describir: La extensión física o lógica y el soporte de la unidad de descripción.
ÁREA DE CONTEXTO	Nombre del o de los Productor(es)	Identificar el productor o los productores de la unidad de descripción.
	Historia institucional/Reseña biográfica	Proporcionar la historia institucional o los datos biográficos del productor o productores de la unidad de descripción para situar la documentación en su contexto para hacerla más compresible
	Historia archivística	Proporcionar información sobre la historia de la unidad de la descripción que sea significativa para su autenticidad, integridad e interpretación.
	Forma de ingreso	Identificar la forma de adquisición o transferencia

ÁREA DE CONTENIDO Y ESTRUCTURA	Alcance y contenido	Proporcionar a los usuarios la información necesaria para apreciar el valor potencial de la unidad de descripción.
	Valoración, Selección y Eliminación	Proporcionar información sobre cualquier acción de valoración, selección y eliminación efectuada.
	Nuevos Ingresos	Información al usuario de los ingresos complementarios previstos relativos a la unidad de descripción.
	Organización	Informar sobre la estructura interna, la ordenación y/o sistema de clasificación de la unidad de descripción.
ÁREA DE CONDICIONES DE ACCESO Y USO	Condiciones de Acceso	Informar sobre la situación jurídica y cualquier otra normativa que restrinja o afecte el acceso a la unidad de descripción.
	Condiciones de Reproducción	Identificar cualquier tipo de restricción relativa a la reproducción de la unidad de descripción.
	Lengua/escritura(s) de la documentación	Identificar la(s) legua(s), escritura(s) y sistemas de símbolos utilizados en la unidad de descripción.
	Características físicas y requisitos técnicos	Informar sobre cualquier característica física o requisito técnico de la importancia que afecte al uso de la unidad de descripción.
	Instrumentos de descripción	Identificar cualquier tipo de instrumento descripción relativa a la unidad de descripción.

ÁREA DE DOCUMENTACIÓN ASOCIADA	Existencia y localización de originales	En el caso de que la unidad de descripción este formada por copias, indicar la existencia, localización, disponibilidad y/o eliminación de los originales.
	Existencia y localización de copias	Indicar la existencia localización y disponibilidad de copias de la unidad de descripción.
	Unidades de descripción relacionadas	Identificar las unidades de descripción relacionadas.
	Notas de publicaciones	Identificar cualquier tipo de publicación que trate o esté basada en el uso, estudio o análisis de la unidad de descripción..
ÁREA DE NOTAS	Notas	Dar información que ni haya podido ser incluida en ninguna de las otras áreas.
ÁREA DE CONTROL DE LA DESCRIPCIÓN	Notas del Archivero	Explicar quién y cómo ha preparado la descripción.
	Reglas o Normas	Identificar la normativa en la que está basada la descripción.
	Fecha(s) de la(s) descripción(es)	Identificar cuando se ha elaborado y/o revisado la descripción.

Tabla 2.6 Estructura de la norma ISAD (G)
Fuente: (ISAD(G), 2000)

CAPÍTULO III
MARCO APLICATIVO

Para comenzar se realizara un análisis de la situación actual donde se identificara los procesos que se aplicaran en el desarrollo del sistema para que el sistema propuesto este orientado al tratamiento archivístico.

Posteriormente se procede con la normalización de los elementos informativos utilizando la descripción archivística como acceso a la información para ello se tiene definida una norma de macro estructura de datos, el International Standard Archival Description (General) – ISAD (G), traducida como Norma Internacional General de Descripción Archivística, con la que se busca facilitar la integración de descripciones en un sistema unificado de información.

La normalización de los registros del Área de Archivo Central constituye un proceso que tiende a unificar los criterios, los métodos de trabajo, las políticas y los instrumentos de descripción. La normalización va más allá de los procesos descriptivos, pero en estos se ha alcanzado un alto nivel de desarrollo internacional.

La normalización por otra parte propicia la generación de una terminología común para el tratamiento archivístico de la información documental, sea en el ambiente tradicional o en los nuevos modelos establecidos por el entorno electrónico.

El Sistema de Gestión de Archivo en adelante conocido como SIGCD aplicará las fases de la metodología SCRUM para organizar el tiempo y las actividades a realizar, las etapas de la metodología OOHDM para el modelado web.

A continuación se muestra en la figura 3.1 el esquema de las fases y la metodología que se utilizarán para el desarrollo del sistema.

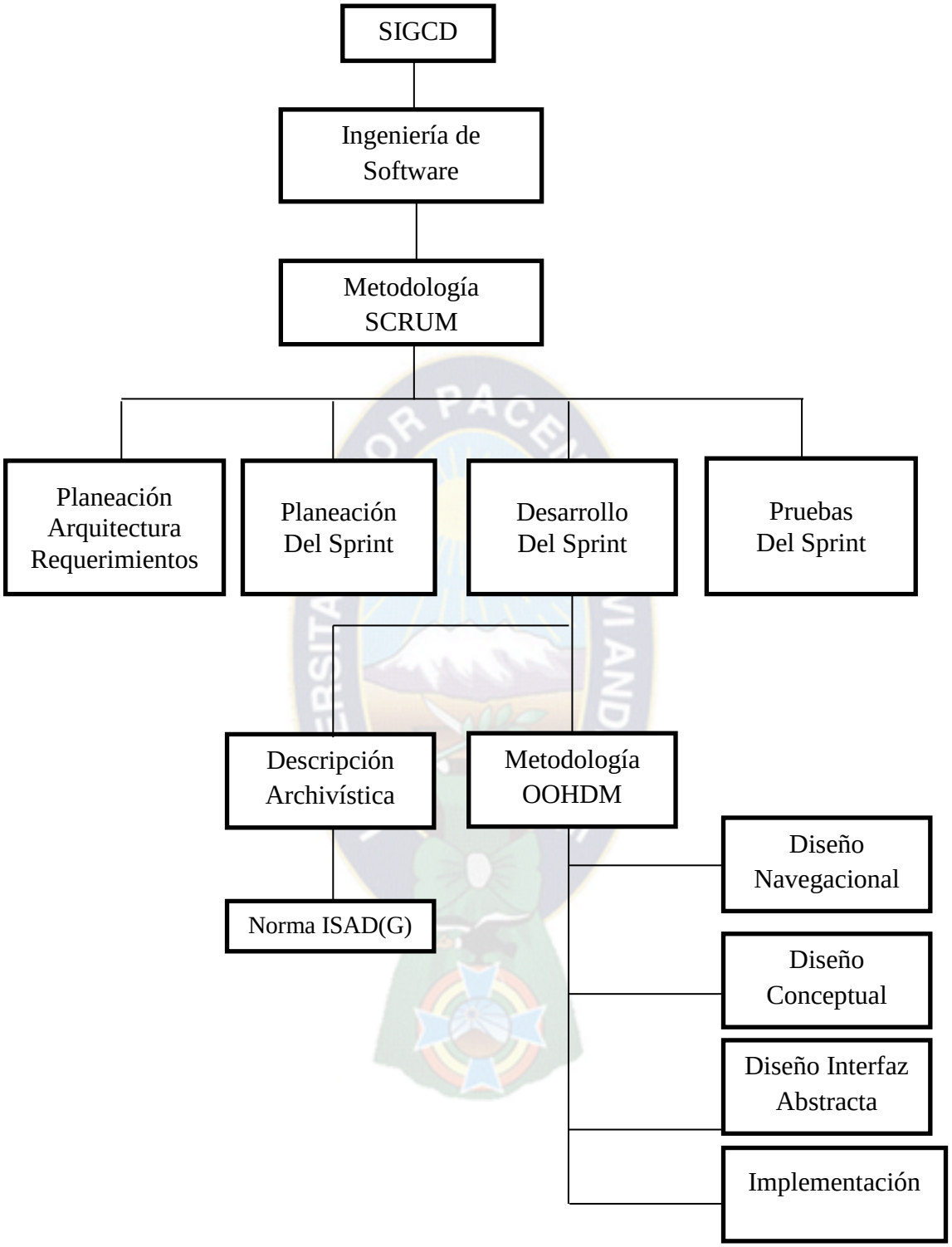


Figura 3.1 Fases y Metodologías utilizadas para el desarrollo del SIGCD
Fuente: Elaboración Propia

3.1 ANÁLISIS DEL SISTEMA ACTUAL

En el Instituto Nacional de Salud INLASA existe un gran movimiento de información y documentación generada por las distintas unidades (ver ANEXO E) que conforman su estructura orgánica, hasta el comienzo del año 2016 se tiene un aproximado de 8000 registros de documentación. El Área de Archivo Central es la encargada del control y el resguardo de esta documentación, de esta manera el personal de esta Áreas debe hacer frente a un número de documentos que va en crecimiento constante sin que el tiempo y los recursos que puedan dedicar a su tratamiento puedan incrementarse en igual proporción.

Para tener un panorama general se describe el ciclo vital de la documentación que pasa por Archivo Central. En esencia este ciclo se define en tres fases por los que pasan los documentos, teniendo en cuenta su valor, es decir valores Primarios (administrativo, fiscal o contable) o Secundarios (informativo, cultural o histórico) son:

- **Documentación activa (unidades que conforman la estructura orgánica de INLASA):** En esta fase la documentación tiene Valores Primarios y su consulta es muy frecuente.
- **Documentación semiactiva (Área de Archivo Central):** Es la fase en que el trámite administrativo ha terminado, tiene valores Primarios y la frecuencia de consulta de esa documentación es menor. La permanencia de los documentos es de tres años, una vez cumplido ese tiempo pasa a su traslado al área de documentación inactiva.
- **Documentación inactiva (Archivo Histórico):** La documentación ha perdido alguno de los valores primarios; con carácter general sólo tiene valor como información e Investigación. Su consulta es puntual. En este momento van adquiriendo valores Secundarios por lo que pasarán a la fase de archivo histórico, aunque existen documentos con perdurabilidad jurídica a lo largo de los siglos.

3.2 ANÁLISIS DE REQUERIMIENTOS (PRODUCT BACKLOG)

En este punto se planteara a grandes rasgos las herramientas, tecnologías y prácticas que se utilizarán en el proyecto, y se exploraran las posibilidades de la arquitectura del sistema.
Las herramientas de desarrollo son las siguientes:

- Lenguaje de Programación: PHP 5, JavaScript, Ajax, JQuery.
- SQL Manager 2011 for SQL.
- Sql yog.
- Bootstrap templates.
- Angular JS.
- MySQL como gestor de Base de Datos.
- Mozilla Firefox o Google Chrome como Navegador.
- Servidor Xampp.
- Windows XP o Windows Seven como sistema operativo.

3.2.1 IDENTIFICACIÓN DE USUARIOS

- **Usuario Administrador:** Es el que tiene todos los privilegios que puede proporcionar el SIGCD, el acceso a la información registrada es total, su rol en el sistema es de administrador y este rol se le otorga al responsable de Archivo o a algún miembro del personal de esa Área.
- **Usuario Normal:** Conformado por dos tipos de usuarios, los funcionarios o las personas externas al ministerio, ambos con una cuenta en el sistema pero con menos privilegios, de esta manera estos tipos de usuario no podrán acceder al módulo de administración del sistema, su rol en el sistema es de usuario normal.

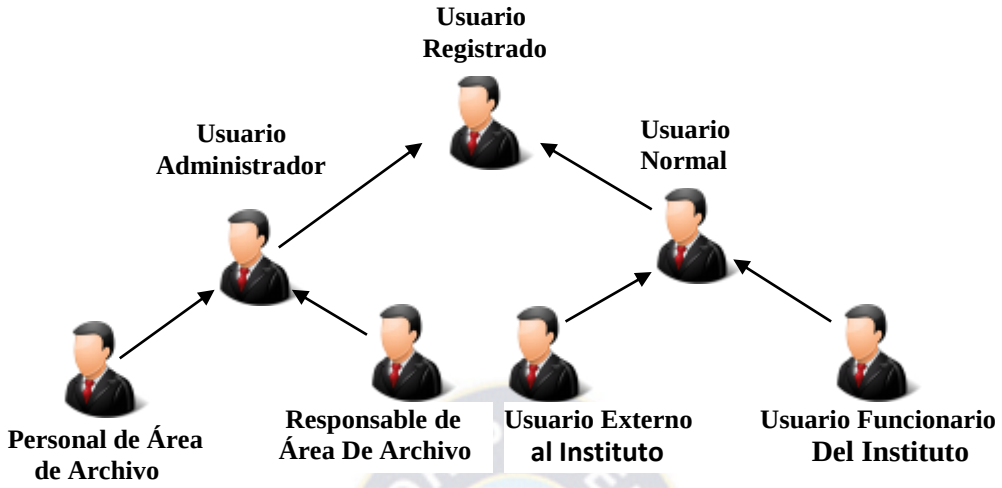


Figura 3.2 Identificación de Usuarios
Fuente: Elaboración Propia

3.2.2 DEFINICIÓN DE ROLES DE LOS ACTORES

En este apartado se describirá los roles de los actores que estarán involucrados de manera directa con el proyecto.

- Cliente: El Cliente será la Dra. Maria Renee Castro Cusicanqui quien es la Directora Ejecutiva del Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón"- INLASA.
- Scrum Master: Estará representado por Univ. Richard Paucara M., quien será el único intercesor entre el cliente y el equipo de trabajo.
- Product Owner: Estará representada por Lic. Miriam Rivas P. ya que es la persona encargada de Archivo Central del Instituto y quien dará a conocer los requerimientos para el desarrollo del sistema.
- Team: Estará representado por Univ. Richard Paucara M. como único programador del equipo de trabajo.

3.2.3 REQUERIMIENTOS DE USUARIO

El propósito de identificar los requerimientos de usuario es representar de forma breve el comportamiento del sistema, utilizando la terminología del cliente dentro de un contexto archivístico. La metodología Scrum específica los requerimientos a través de historias de usuario,

estos requerimientos servirán para la elaboración de diagramas de Interacción del Usuario, luego presiden la creación de las pruebas de aceptación una vez implementadas.

Desde el punto de vista del número de Historias de Usuario, se obtuvo inicialmente un total de doce tareas principales que serán necesarias para el inicio del desarrollo del sistema las cuales fueron de gran utilidad para la estimación de los tiempos requeridos para el desarrollo del proyecto. Las historias de usuario son descompuestas en tareas de programación y asignadas a los programadores para ser implementadas durante una iteración.

A continuación en la tabla 3.1 hacemos la recolección de los requerimientos de usuario que fueron extraídos de las historias de usuario (Ver Anexo D).

ID	DESCRIPCIÓN	MÓDULO	PRIORIDAD	ESTADO
R1	Diseñar una base de datos que facilite el acceso a la documentación.	Administración	Alta	Terminado
R2	Desarrollar una interface que permita el registro, edición y eliminación de documentos resguardados en un sistema unificado.	Administración	Alta	Terminado
R3	Unificar las descripciones archivísticas de los documentos en un sistema de información basado en la norma ISAD-G.	Administración	Media	Terminado
R4	Asociar un recurso digital a los documentos al momento de registro.	Administración	Media	Terminado
R5	Crear una interface de visualización de documentos digitalizados y agrupaciones documentales.	Administración	Alta	Terminado
R6	Incorporar en la base de datos áreas de información descriptiva.	Descripción	Media	Terminado
R7	Desarrollar una interface de búsqueda con criterios múltiples, ya sea por fechas o por dependencias.	Búsqueda	Alta	Terminado
R8	Generar formulario para el préstamo de documentos.	Solicitudes	Alta	Terminado

R9	Adaptar la base de datos a un cuadro de clasificación mediante una emulación de una estructura jerárquica.	Administración	Alta	Terminado
R10	Desarrollar una interface para la gestión y control de las solicitudes de la documentación por parte de los usuarios del sistema.	Solicitudes	Media	Terminado
R11	Diseñar una interface de acceso al sistema, condicionado por su código de acceso mediante privilegios sobre el sistema.	Usuarios	Media	Terminado
R12	Desarrollar una interface que permita el registro, edición y eliminación de usuarios por parte del administrador del sistema.	Usuarios	Media	Terminado

Tabla 3.1 Requerimientos de Usuario
Fuente: Elaboración Propia

3.3 PLANIFICACIÓN DE LA ENTREGA

PLAN DE ENTREGA										
ID	**DESCRIPCIÓN**	**DURACIÓN EN SEMANAS**	**SEPTIEMBRE**				**OCTUBRE**			
R1	Diseñar una base de datos que facilite el acceso a la documentación.	1								
R2	Desarrollar una interface que permita el registro, edición y eliminación de documentos resguardados en un sistema unificado.	1								
R3	Unificar las descripciones archivísticas de los documentos en un sistema de información basado en la norma ISAD-G.	1								
R4	Asociar un recurso digital a los documentos al momento de registro.	1								
R5	Crear una interface de visualización de documentos digitalizados y agrupaciones documentales.	1								
R6	Incorporar en la base de datos áreas de información descriptiva.	1								

R7	Desarrollar una interface de búsqueda con criterios múltiples, ya sea por fechas o por dependencias.	1								
R8	Generar formulario para el préstamo de documentos.	1								
R9	Adaptar la base de datos a un cuadro de clasificación mediante una emulación de una estructura jerárquica.	1								
R10	Desarrollar una interface para la gestión y control de las solicitudes de la documentación por parte de los usuarios del sistema.	1								
R11	Diseñar una interface de acceso al sistema, condicionado por su código de acceso mediante privilegios sobre el sistema.	1								
R12	Desarrollar una interface que permita el registro, edición y eliminación de usuarios por parte del administrador del sistema.	1								

La tabla 3.2 describe el cronograma en el que se harán las entregas de los productos.

Tabla 3.2 Plan de Entrega Fuente: Elaboración Propia

3.4 ITERACIONES (SPRINTS)

Esta sección incluye varias iteraciones sobre el sistema antes de ser entregado. El Plan de entrega está compuesto por iteraciones de no más de 12 días. Cada iteración está conformada por tablas de requerimientos conocidas como product backlog correspondientes a cada historia de usuario descrita en la fase de planificación.

Las iteraciones se desarrollaran combinando las fases de diseño navegacional, diseño conceptual y el diseño de interfaz abstracta propuesta por la metodología OOHDM antes de la implementación según la adaptabilidad en cada una.

3.4.1 PRIMERA ITERACIÓN – MÓDULO USUARIOS

El objetivo de la primera iteración es el diseño de la base de datos que contara con una emulación de una estructura jerárquica para su funcionalidad respecto al cuadro de clasificación, como se ve en la tabla 3.3 que representa a la primera iteración.

		SPRINT	INICIO	DURACIÓN
		1	03/09/2016	12 DÍAS
ID	TAREA	TIPO	DÍAS DE TRABAJO	ESTADO
1.1	Realizar la planificación de la iteración	Desarrollo	3	Terminado
1.2	Analizar los requerimientos de backlog del producto	Desarrollo	2	Terminado
1.3	Diseño de la base de datos	Desarrollo	1	Terminado
1.4	Construcción de diseño navegacional	Desarrollo	1	Terminado
1.5	construcción de diseño conceptual	Desarrollo	1	Terminado
1.6	Construcción de diseño de interfaz abstracta	Desarrollo	1	Terminado
1.7	Desarrollar la página de ingreso al sistema	Desarrollo	2	Terminado
1.8	Desarrollar el módulo de registro de nuevo usuario	Desarrollo	1	Terminado

Tabla 3.3 Primera Iteración
Fuente: Elaboración Propia

3.4.1.1 DISEÑO DE LA BASE DE DATOS

El diseño de la base de datos se la realiza analizando los requerimientos de usuario, es así que después de hacer un análisis exhaustivo se tiene la figura 3.3 que representa a la base de datos del sistema.

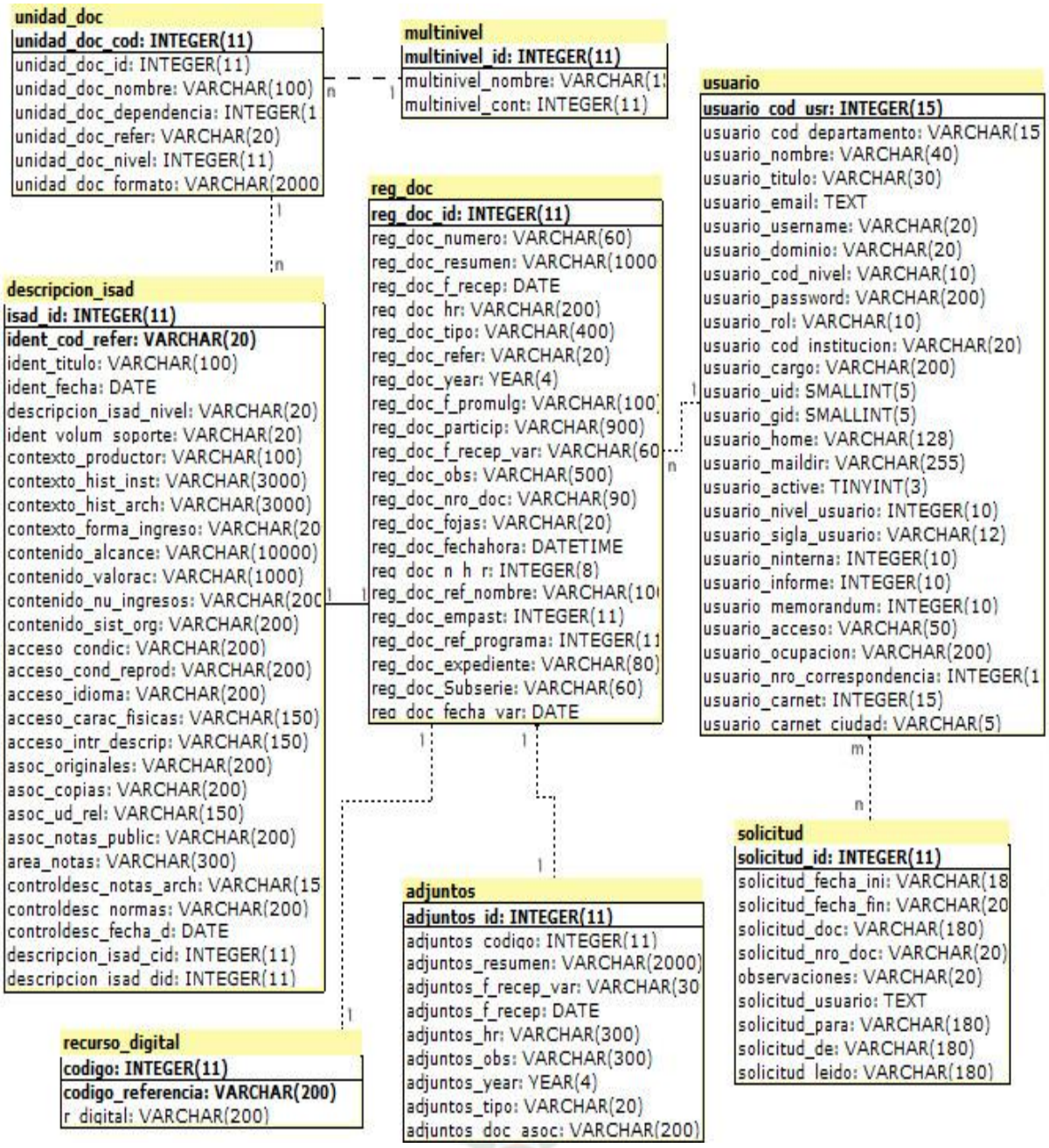


Figura 3.3 Diseño de la Base de Datos – Primera Iteración
Fuente: Elaboración Propia

Luego de haber definido los requerimientos de usuario y la base de datos se procede con el diseño Conceptual, diseño Navegacional y Diseño de Interfaz Abstracta.

3.4.1.2 DISEÑO CONCEPTUAL – MÓDULO USUARIOS

El diseño conceptual de la Figura 3.4 representa la etapa de la metodología OOHDM en la cual se elabora a partir de los requerimientos obtenidos del sistema, constituye las clases relaciones y subsistemas del SIGCD con respecto al módulo de usuarios.

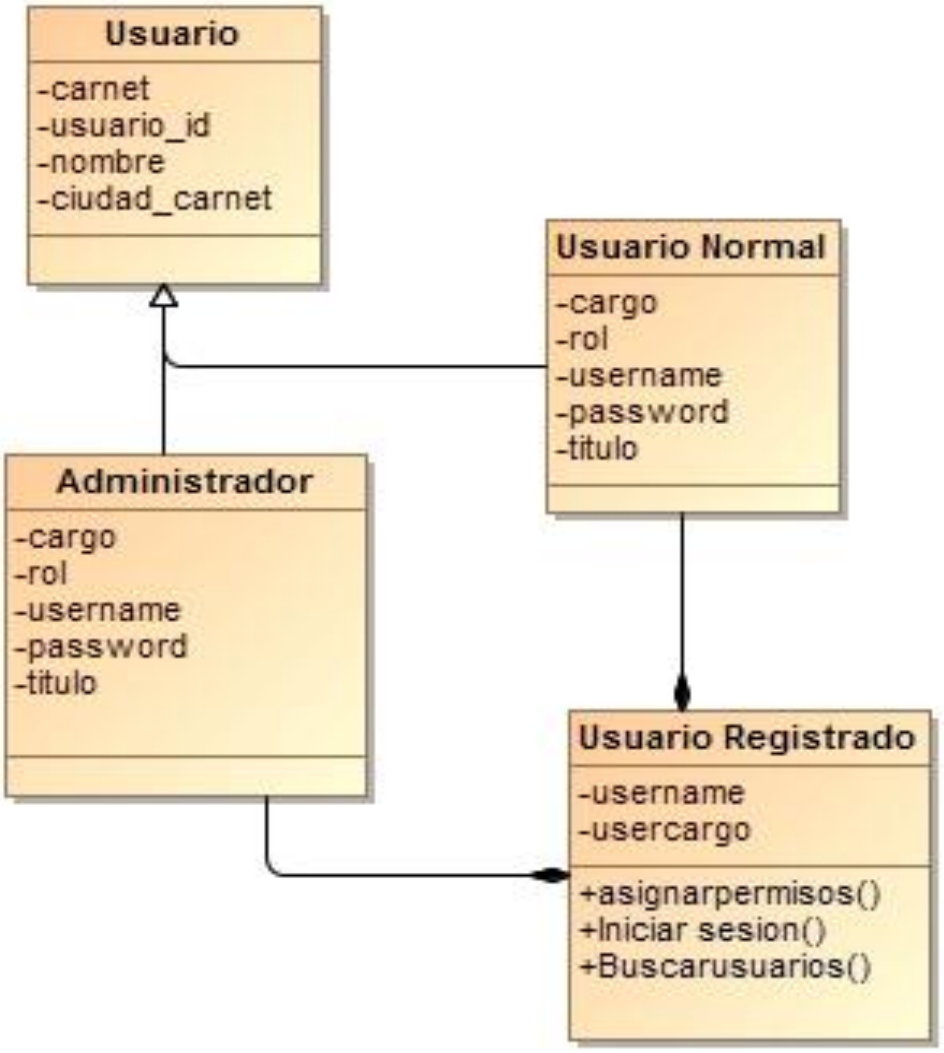


Figura 3.4 Diseño Conceptual – Módulo de Usuarios
Fuente: Elaboración Propia

3.4.1.3 DISEÑO NAVEGACIONAL – MÓDULO USUARIOS

En esta etapa se define clases navegacionales tales como nodos, enlaces y estructuras de acceso (índices y visitas guiadas) inducidas del esquema conceptual, seguidamente se realiza el diagrama de contextos navegacionales representadas en la Figura 3.5.

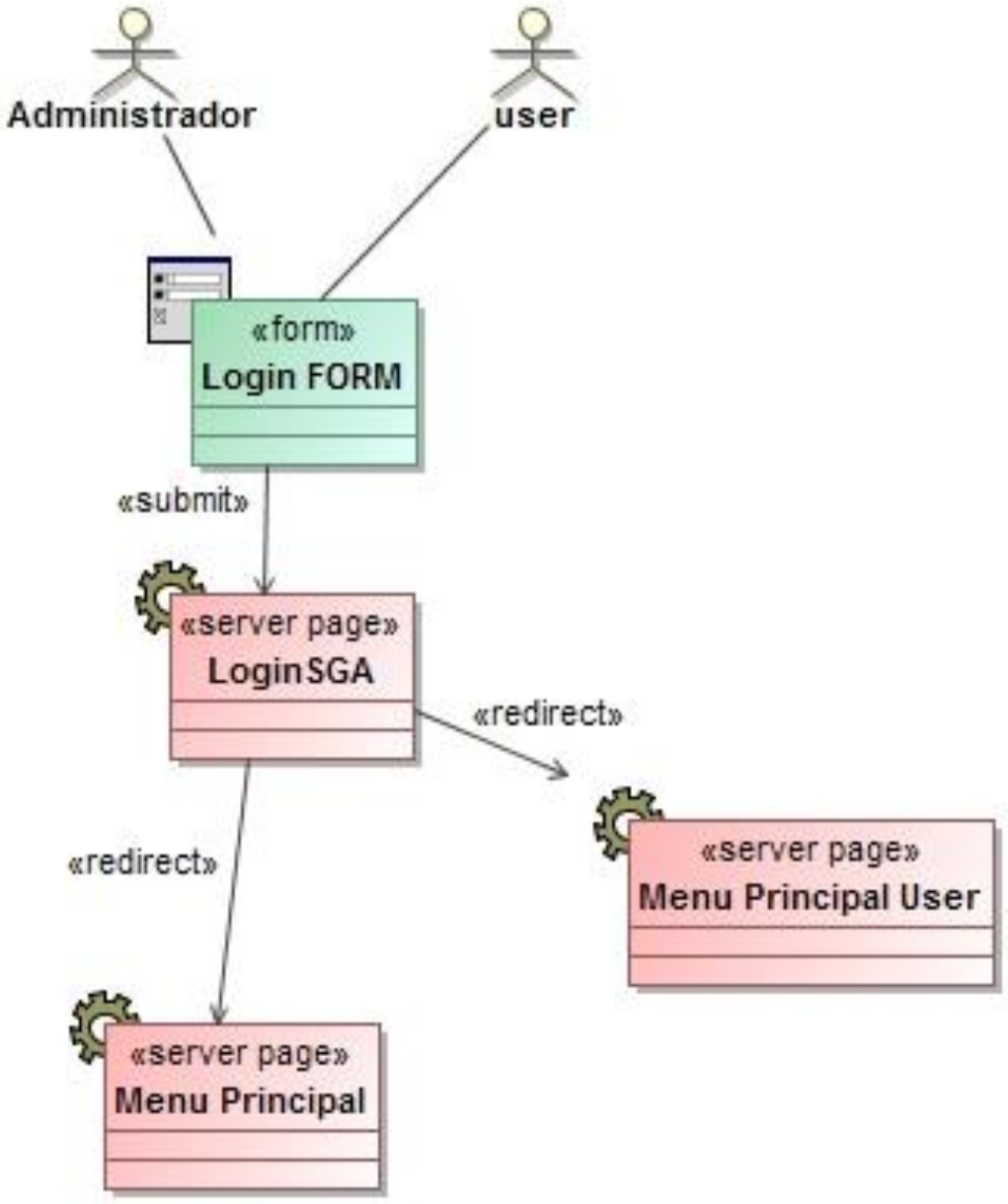


Figura 3.5 Diseño Navegacional – Módulo de Usuarios
Fuente: Elaboración Propia

3.4.1.4 DISEÑO INTERFAZ ABSTRACTA – MÓDULO USUARIOS

Una vez que las estructuras navegacionales son definidas, se pasa a la siguiente etapa de la metodología OOHDM donde se especifica los objetos navegacionales y cuál de estos activaran la navegación, como se ve en la figura 3.6.

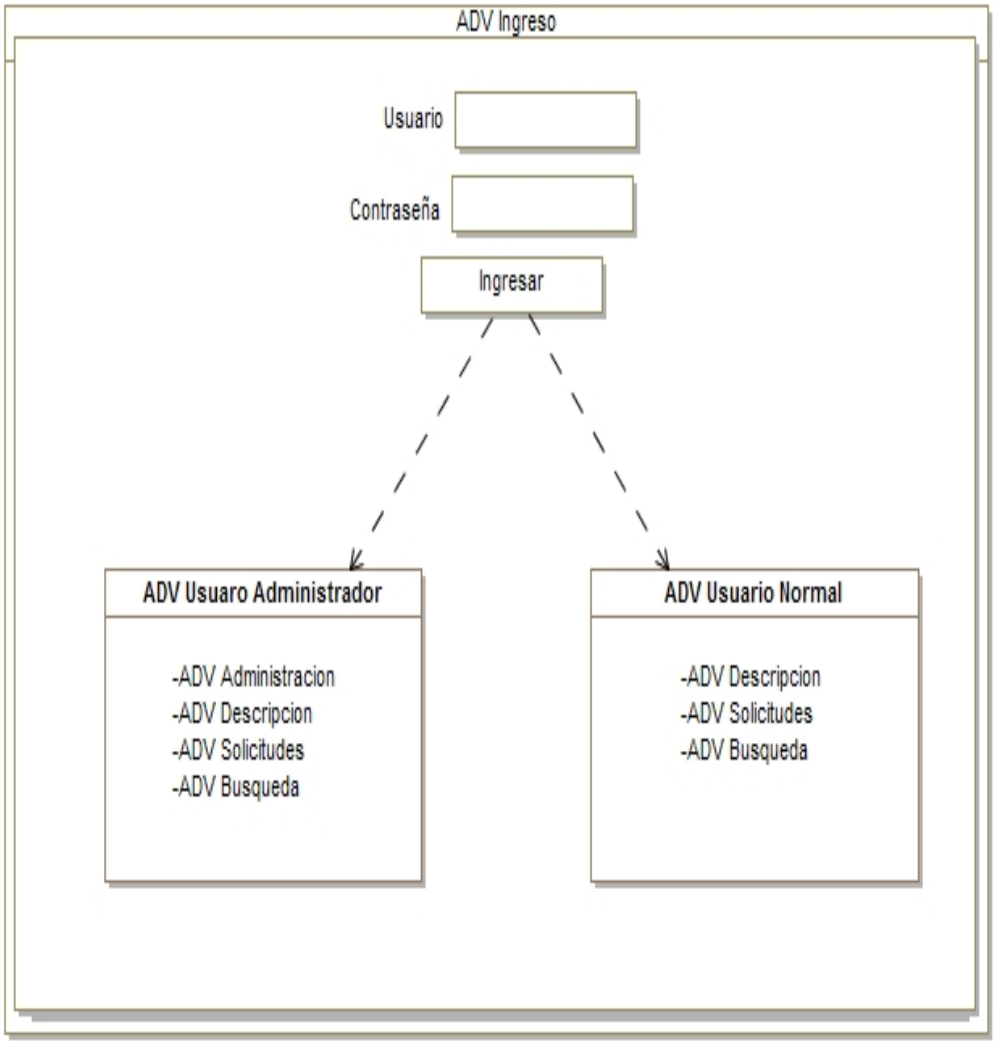


Figura 3.6 Diseño Interfaz Abstracta - Módulo de Usuarios
Fuente: Elaboración Propia

3.4.1.5 IMPLEMENTACIÓN - MÓDULO DE USUARIOS

La Figura 3.7 muestra la interfaz de autenticación de usuarios registrados que re direcciona al SIGCD con los privilegios correspondientes.

Figura 3.7 Interfaz de Autenticación - Módulo de Usuarios
Fuente: Elaboración Propia

La figura 3.8 muestra la forma en que se hace el registro de un nuevo usuario, este proceso solo lo puede hacer el administrador del sistema, quien es el que define el modo en que se registrara al usuario, ya sea de tipo administrador o usuario.

Figura 3.8 Interfaz Registro de Usuarios - Módulo de Usuarios
Fuente: Elaboración Propia

3.4.2 SEGUNDA ITERACIÓN – MÓDULO DE ADMINISTRACIÓN

En la segunda iteración el módulo administración brinda un soporte a los registros de la documentación que se ingresa con la posibilidad de asociar una descripción archivística y un recurso digital a cada expediente como se lo refleja en la tabla 3.4, al cuadro de clasificación en el que define una estructura y descripción multinivel de la documentación partiendo del elemento fondo y descendiendo hasta los niveles de expediente.

		SPRINT	INICIO	DURACIÓN
		2	16/09/2016	12 DÍAS
ID	TAREA	TIPO	DÍAS DE TRABAJO	ESTADO
2.1	Realizar la planificación de la iteración	Desarrollo	1	Terminado
2.2	Analizar los requerimientos de backlog del producto	Desarrollo	2	Terminado
2.3	Complementar diseño navegacional	Desarrollo	1	Terminado
2.4	Complementar diseño conceptual	Desarrollo	1	Terminado
2.5	Complementar diseño de interfaz abstracta	Desarrollo	1	Terminado
2.6	Gestión de cuadros de clasificación	Desarrollo	1	Terminado
2.7	Desarrollar el módulo de registro de documentos	Desarrollo	2	Terminado
2.8	Desarrollar la página de recurso digital.	Desarrollo	3	Terminado

Tabla 3.4 Segunda Iteración
Fuente: Elaboración Propia

3.4.2.1 DISEÑO CONCEPTUAL – MÓDULO ADMINISTRACIÓN

El diseño conceptual es la etapa de la metodología OOHDM en la cual se elabora a partir de los requerimientos obtenidos del sistema, constituye las clases relaciones y subsistemas del SIGCD.

La siguiente figura 3.9 representa el diseño conceptual del módulo de administración que contendrá el sistema.

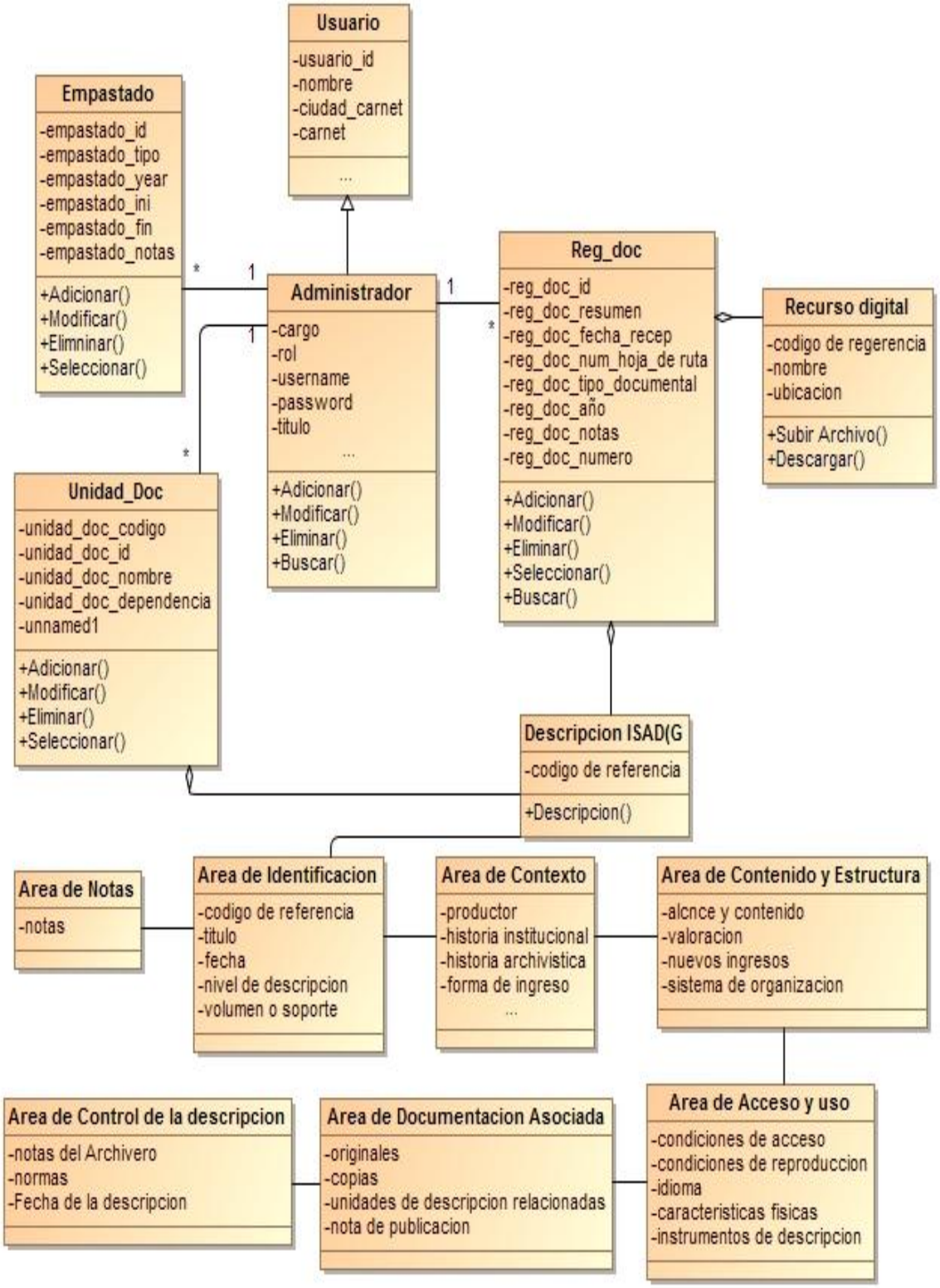


Figura 3.9 Diseño Conceptual – Módulo de Administración
Fuente: Elaboración Propia

3.4.2.2 DISEÑO NAVEGACIONAL – MÓDULO ADMINISTRACIÓN

En esta etapa se define clases navegacionales tales como nodos, enlaces y estructuras de acceso inducidas del esquema conceptual, seguidamente se realiza el diagrama de contextos navegacionales representada en la Figura 3.10.

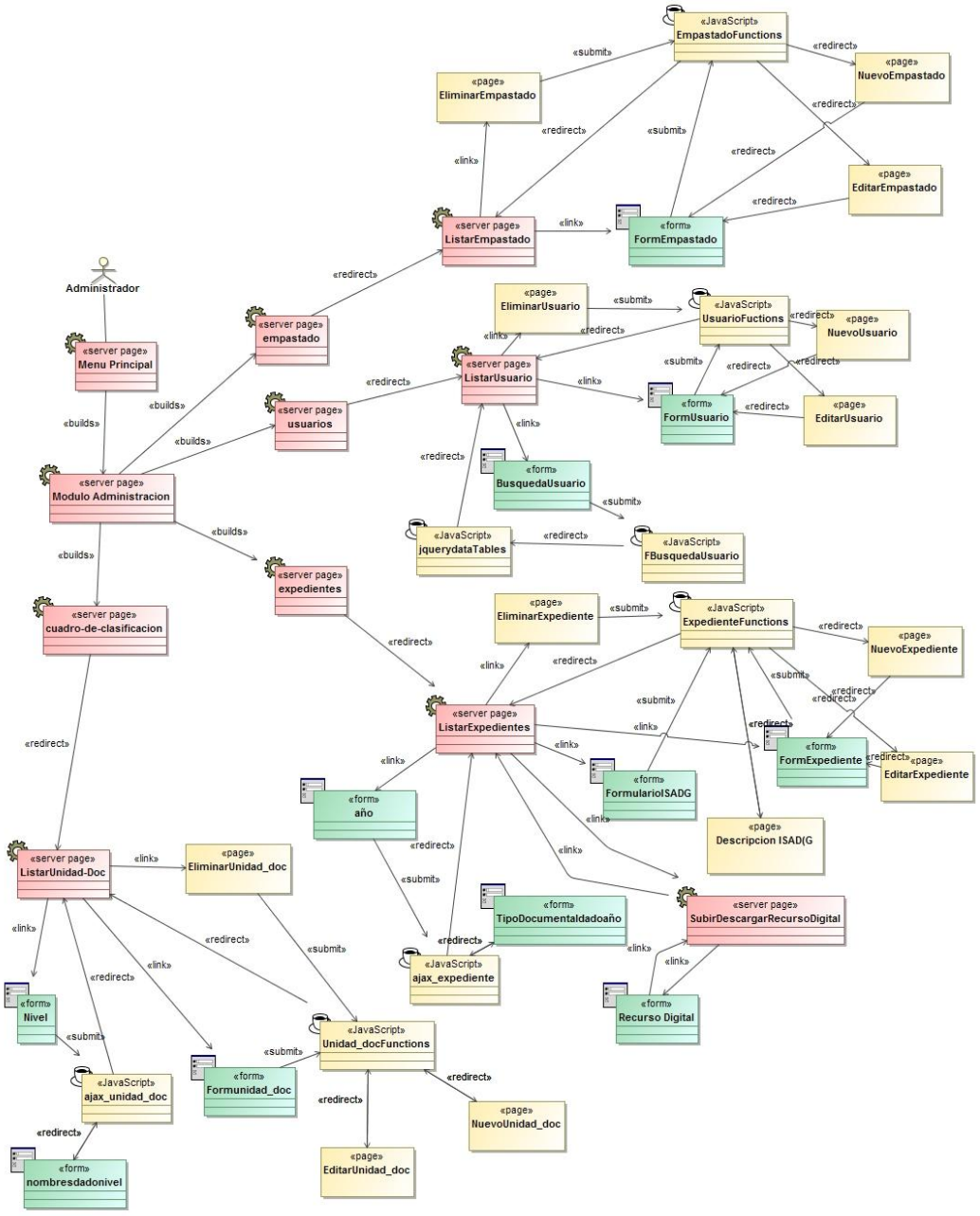


Figura 3.10 Diseño Navegacional – Módulo de Administración
Fuente: Elaboración Propia

3.4.2.3 DISEÑO INTERFAZ ABSTRACTA – MÓDULO ADMINISTRACIÓN

Una vez que las estructuras navegacionales son definidas, se pasa a la siguiente etapa de la metodología OOHDM donde se especifica los objetos navegacionales y cuál de estos activaran la navegación los ADVs correspondientes a este módulo se describen en las Figura 3.11 y Figura 3.12.

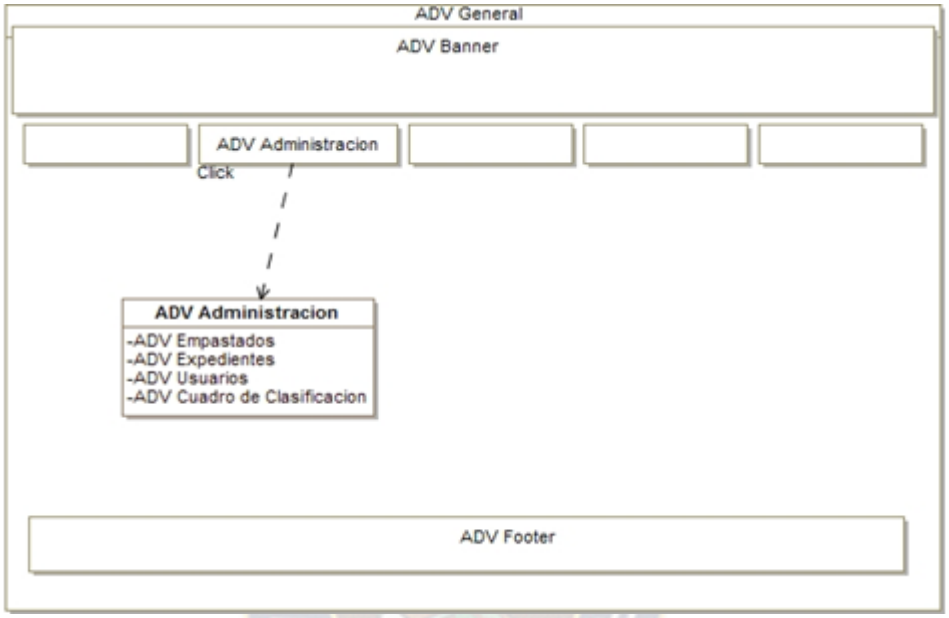


Figura 3.11 ADV Módulo de Administración
Fuente: Elaboración Propia

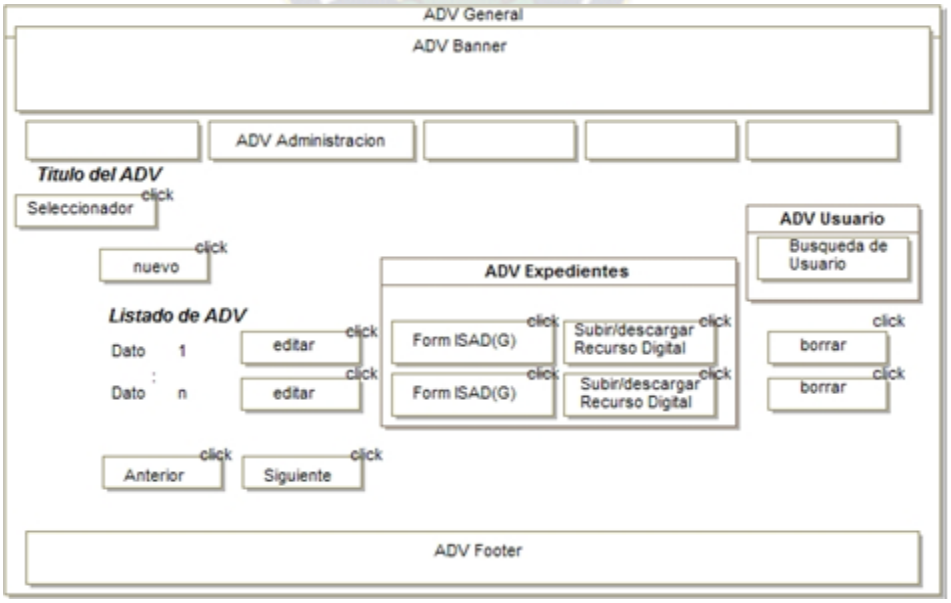


Figura 3.12 ADVs contenidos en el Módulo de Administración
Fuente: Elaboración Propia

Las ADVs (Abstract Data View) o Vista de Datos Abstracta que representan un estado y un interfaz, de esta manera y mediante el uso de ADVs se representara de qué manera los objetos de interfaz activaran la navegación y el resto de la funcionalidad de el Modulo de Administración para su mejor entendimiento antes de la implementación

3.4.2.4 IMPLEMENTACIÓN DEL MÓDULO DE ADMINISTRACIÓN

En esta etapa consiste en la aplicación de los diseños especificados anteriormente se muestra algunas de las pantallas del SIGCD mostrando algunas de las interfaces como se ve en la figura 3.13. Es importante señalar que este módulo es exclusivo del usuario con el rol de administrador puesto que este tipo de usuario tiene acceso a la adición, modificación y eliminación de todas las entidades contempladas en el presente sistema.

Figura 3.13 Interfaz de Administración
Fuente: Elaboración Propia

Existen agrupaciones documentales homogéneas que contienen unidades de instalación, es decir expedientes que ya terminaron su fase de trámite, estos son resguardados en el Área de Archivo Central durante tres gestiones desde el año de su recepción, considerando los puntos anteriores la Figura 3.14 muestra la interfaz de registro de documentos que ya pasaron a un estado inactivo.

Figura 3.14 Interfaz de Inventario de Expedientes
Fuente: Elaboración Propia

La figura 3.15 nos muestra la forma en que se pueden crear carpetas según la unidad que lo requiera.

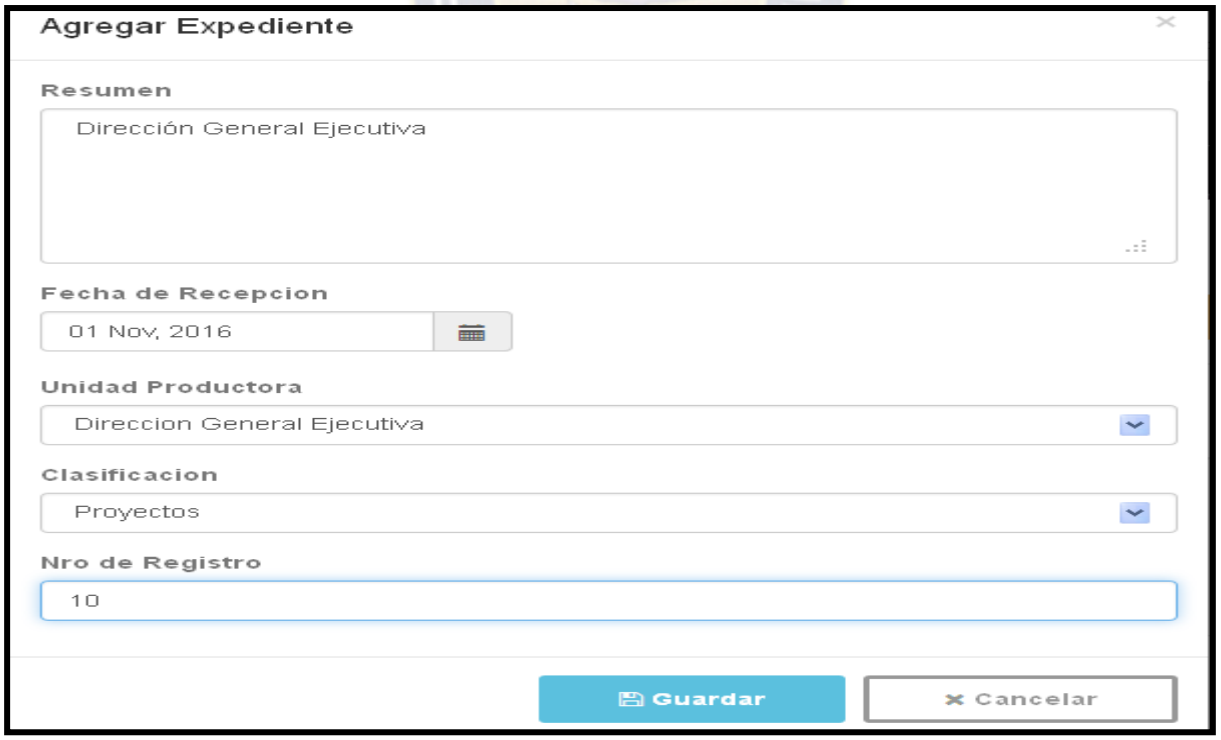


Figura 3.15 Interfaz de Registro de Documentos
Fuente: Elaboración Propia

La figura 3.16 nos muestra la forma de adjuntar expedientes a las carpetas donde se asigna un código de referencia automático.

Figura 3.16 Subir y Descargar Documentos
Fuente: Elaboración Propia

3.4.3 TERCERA ITERACIÓN – MÓDULO DE DESCRIPCIÓN

En esta iteración se hará la implementación de las páginas dentro de un contexto archivístico, además de la Descripción del sistema que contara con una navegación de una lista desplegable del cuadro de clasificación además de la generación integrada de metadatos ISAD (G), como lo vemos en la tabla 3.5 y en la figura 3.17.

		SPRINT	INICIO	DURACIÓN
		3	28/09/2016	10 DÍAS
ID	TAREA	TIPO	DÍAS DE TRABAJO	ESTADO
3.1	Planificación de la iteración	Desarrollo	1	Terminado
3.2	Analizar los requerimientos de backlog del producto	Desarrollo	1	Terminado
3.3	Complementar diseño navegacional	Desarrollo	1	Terminado
3.4	Complementar diseño conceptual	Desarrollo	1	Terminado
3.5	Complementar diseño de interfaz abstracta	Desarrollo	1	Terminado
3.6	Implementación del sistema web	Desarrollo	2	Terminado
3.7	Desarrollar páginas de descripción y navegación multinivel	Desarrollo	2	Terminado

Tabla 3.5 Tercera Iteración
Fuente: Elaboración Propia

3.4.3.1 DISEÑO CONCEPTUAL - MÓDULO DE DESCRIPCIÓN

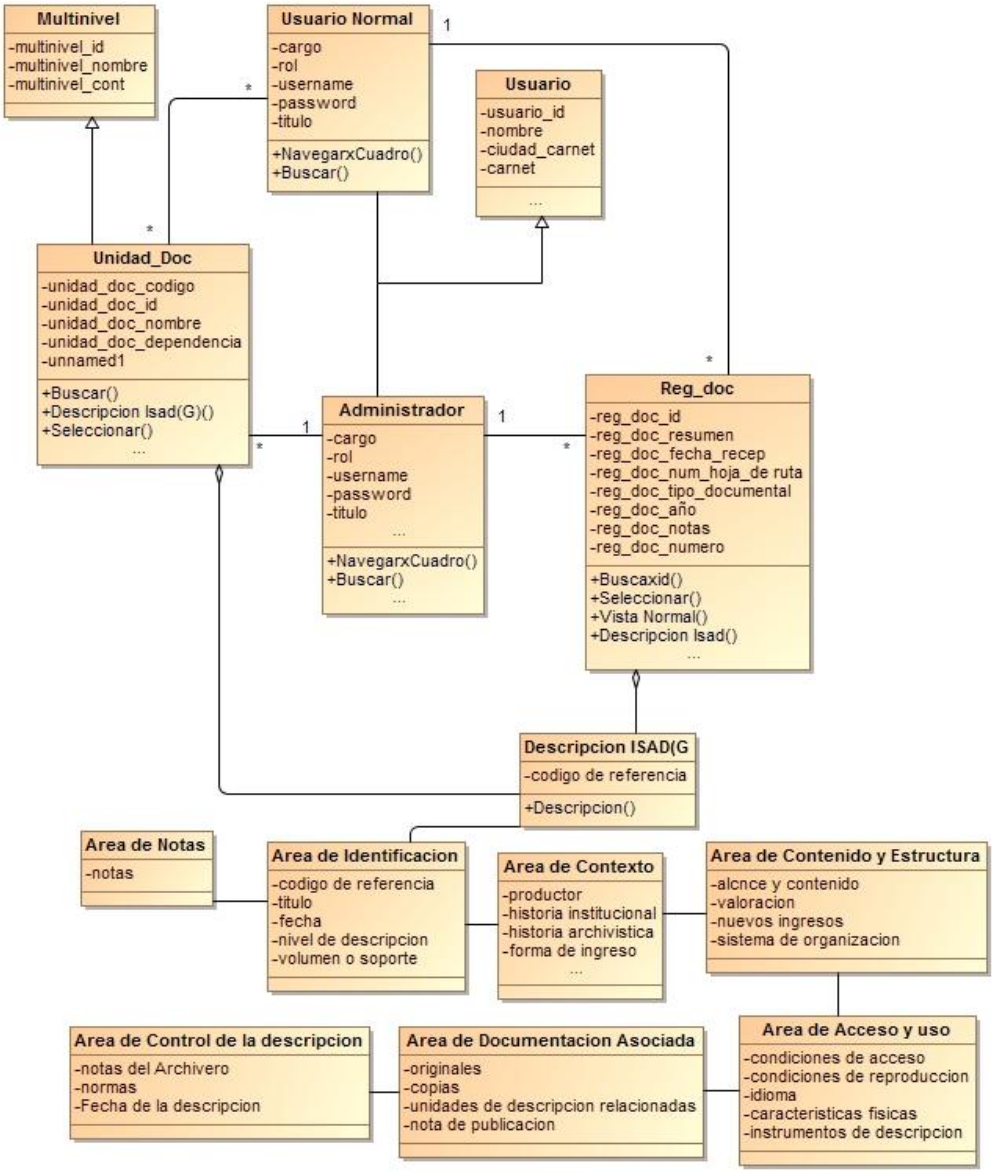


Figura 3.17 Diseño Conceptual – Módulo de Descripción
Fuente: Elaboración Propia

3.4.3.2 DISEÑO NAVEGACIONAL – MÓDULO DE DESCRIPCIÓN

En esta etapa se define clases navegacionales tales como nodos, enlaces y estructuras de acceso (índices y visitas guiadas) inducidas del esquema conceptual, seguidamente se realiza el diagrama de contextos navegacionales representadas en la Figura 3.18.

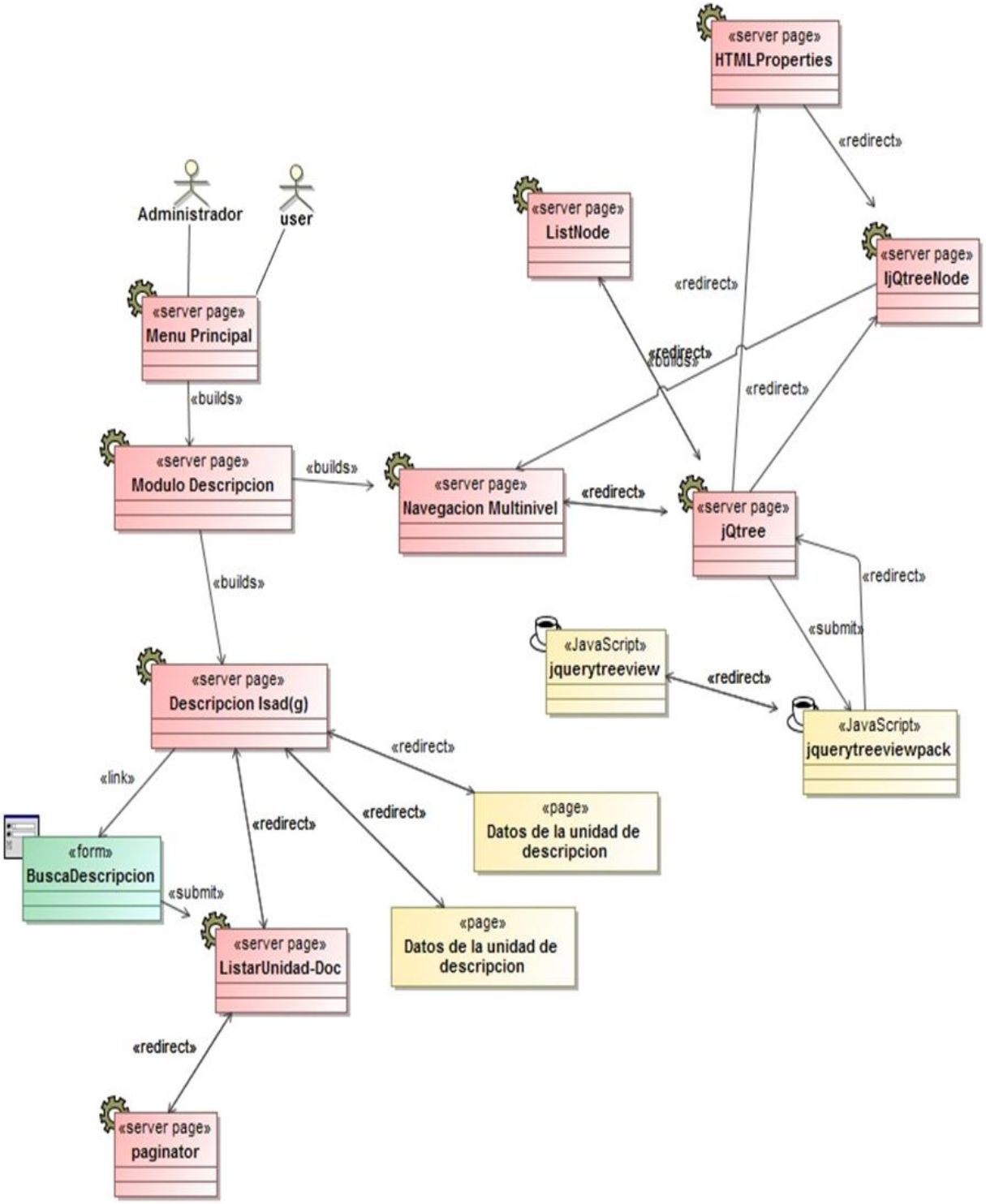


Figura 3.18 Diagrama de Contextos Navegacionales – Módulo de Descripción
Fuente: Elaboración Propia

3.4.3.3 DISEÑO INTERFAZ ABSTRACTA – MÓDULO DE DESCRIPCIÓN

Una vez que las estructuras navegacionales son definidas, se pasa a la siguiente etapa de la metodología OOHDM donde se especifica los objetos navegacionales y cuál de estos activaran la navegación para este módulo se especifican los ADVs en las Figuras 3.19 y Figura 3.20.

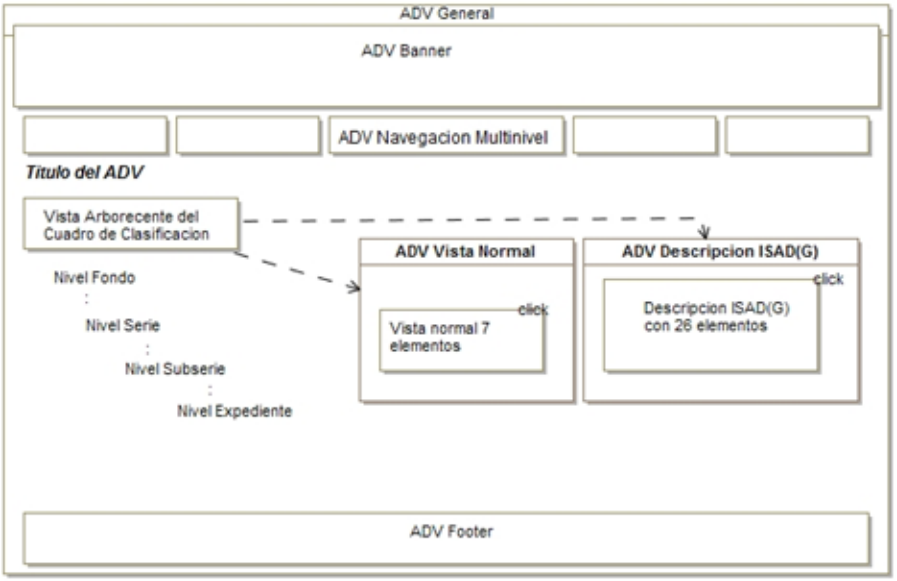


Figura 3.19 ADV Navegación Multinivel - Módulo de Descripción
Fuente: Elaboración Propia

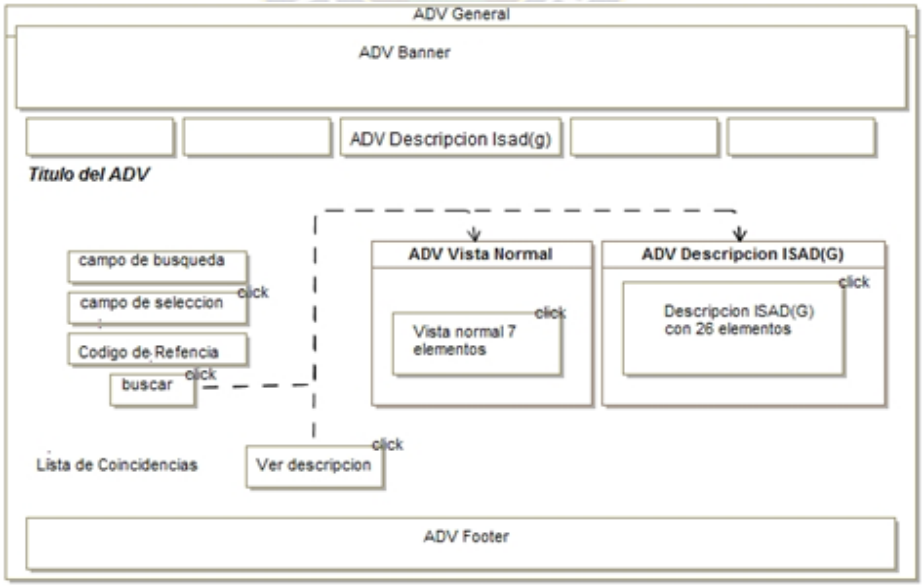


Figura 3.20 ADV Descripción ISAD (G)- Módulo de Descripción
Fuente: Elaboración Propia

3.4.3.4 IMPLEMENTACIÓN DEL MÓDULO DE DESCRIPCIÓN

Esta parte del sistema contara con una navegación multinivel que consiste en una navegación de una lista arborescente desplegable del cuadro de clasificación además de la generación integrada de metadatos ISAD (G), como se ve en la figura 3.21, que nos muestra los niveles de descripción con los que contará cada unidad que forma parte de la institución.

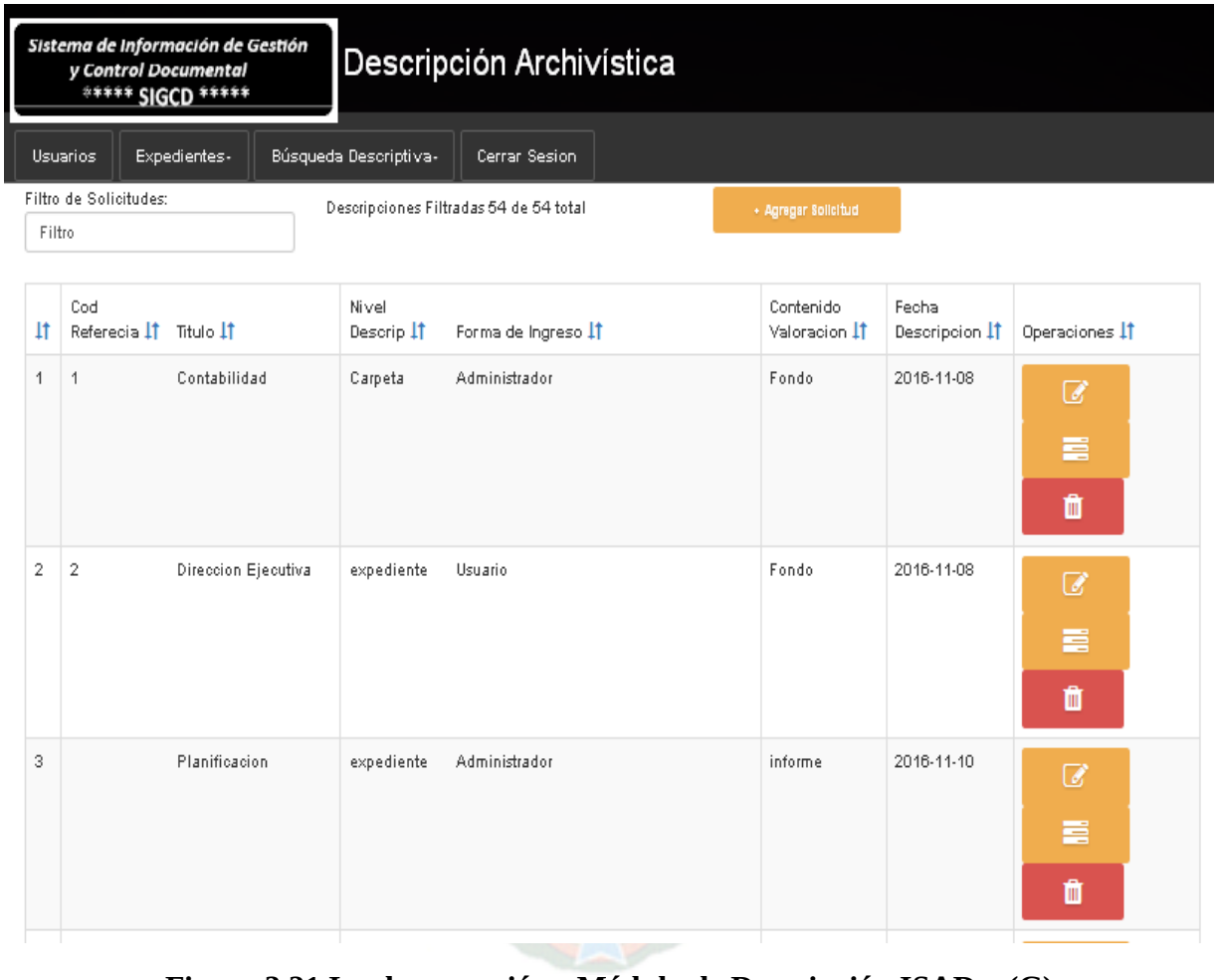


Figura 3.21 Implementación - Módulo de Descripción ISAD – (G)
Fuente: Elaboración Propia

En la figura 3.22 veremos la forma en que se pueden ir creando los niveles de navegación que serán necesarios para el resguardo lógico y físico de los documentos, de esta forma se asegura que los documentos sean registrados en el orden que determina la norma ISAD (G).

Figura 3.22 Editar Módulo de Descripción ISAD - G
Fuente: Elaboración Propia

3.4.4 CUARTA ITERACIÓN – MÓDULO DE SOLICITUDES

En esta iteración se hará la implementación de los módulos de Gestión y control de las solicitudes, que consiste en diseñar una interfaz que permita generar una solicitud formal además de llevar el control de las solicitudes efectuadas a fin de atender de manera personalizada a los usuarios en el momento de que necesiten efectuar una solicitud o realizar reservas de espacios con respecto a la documentación. Además se implementará la búsqueda y recuperación por criterios múltiples.

		SPRINT	INICIO	DURACIÓN
		4	13/10/2016	12 DÍAS
ID	TAREA	TIPO	DÍAS DE TRABAJO	ESTADO
3.1	Realizar la planificación de la iteración	Desarrollo	1	Terminado
3.2	Analizar los requerimientos de backlog del producto	Desarrollo	1	Terminado
3.3	Complementar diseño navegacional	Desarrollo	1	Terminado
3.4	Complementar diseño conceptual	Desarrollo	1	Terminado
3.5	Complementar diseño de interfaz abstracta	Desarrollo	1	Terminado
3.6	Desarrollar la página de gestión y control de solicitudes	Desarrollo	2	Terminado
3.7	Desarrollar páginas de búsqueda y recuperación de documentos por criterios múltiples	Desarrollo	2	Terminado
3.8	Desarrollar la página de gestión y control de préstamos	Desarrollo	2	Terminado

Tabla 3.6 Cuarta Iteración
Fuente: Elaboración Propia

3.4.4.1 DISEÑO CONCEPTUAL – MÓDULO DE SOLICITUDES

En el siguiente diseño conceptual, figura 3.23 se describirán todas las cualidades que destacan los diferentes usuarios y procesos correspondientes al diseño de la página de gestión y control de solicitudes, también servirá para organizar las búsquedas por medio de criterios múltiples y por último nos ayudará a describir la página de gestión y control de préstamos.

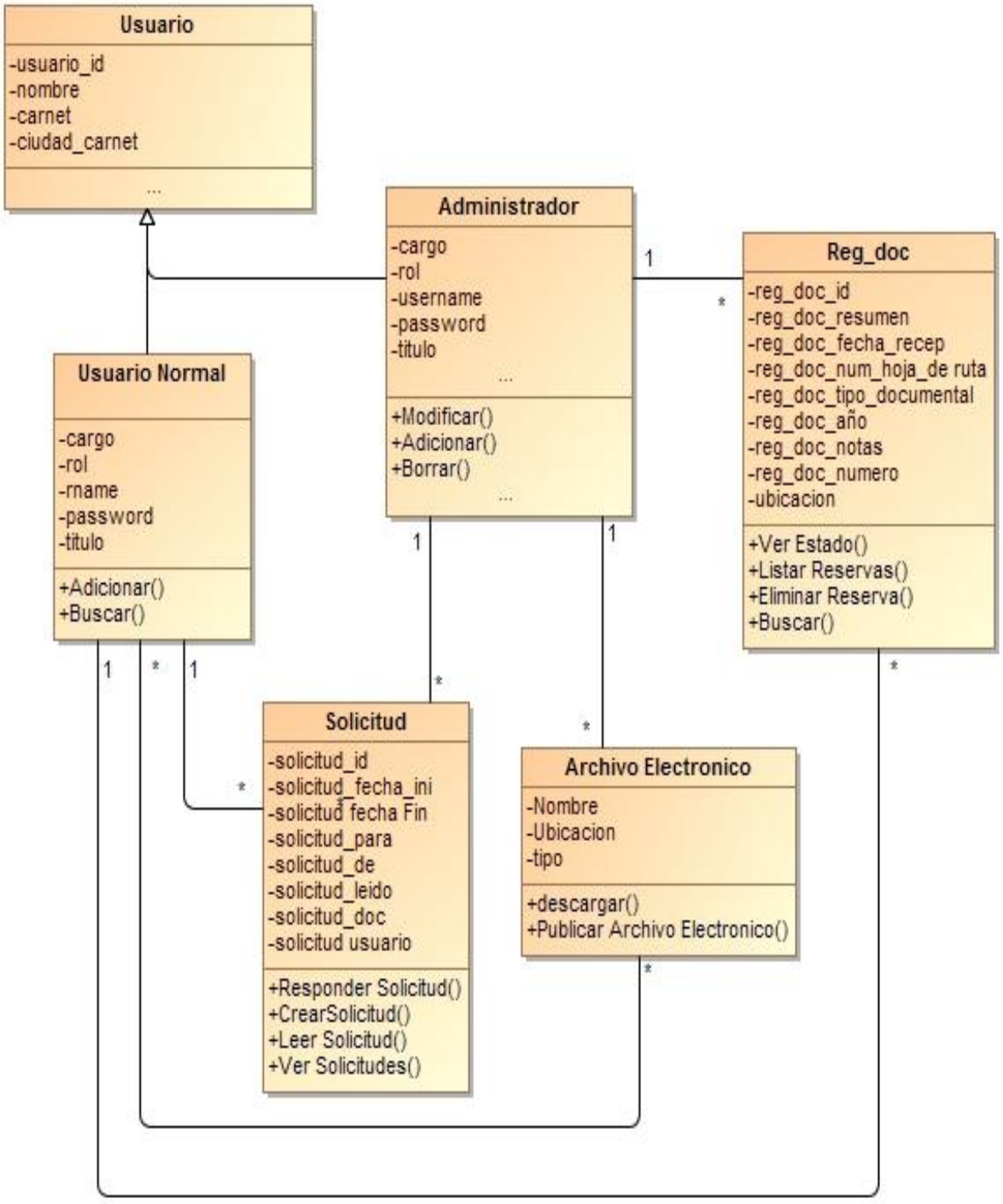


Figura 3.23 Diseño Conceptual – Módulo de Solicitudes
Fuente: Elaboración Propia

3.4.4.2 DISEÑO NAVEGACIONAL – MÓDULO DE SOLICITUDES

En esta etapa se define clases navegacionales tales como nodos, enlaces y estructuras de acceso (índices y visitas guiadas) inducidas del esquema conceptual, seguidamente se realiza el diagrama de contextos navegacionales representadas en la Figura 3.24.

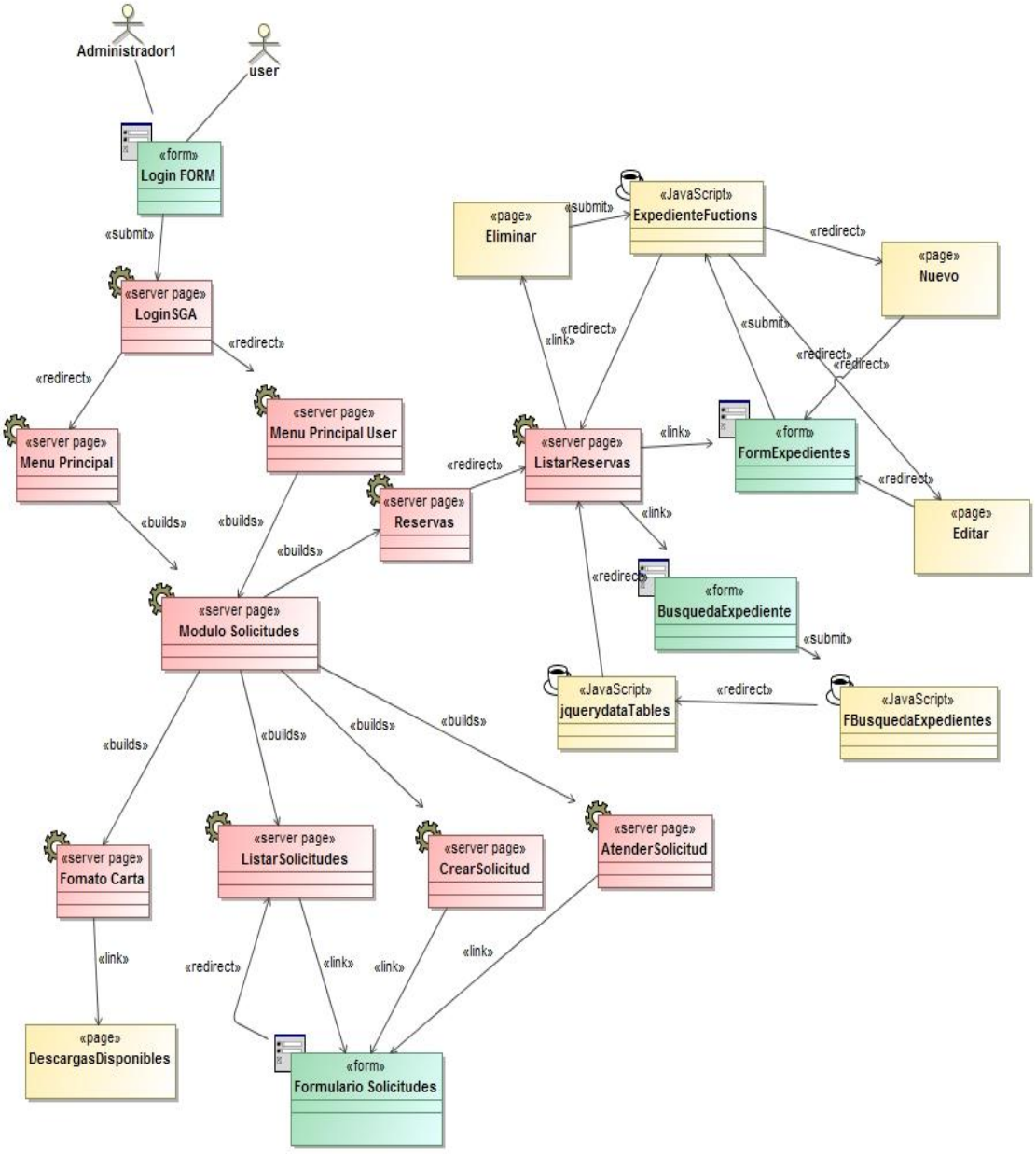


Figura 3.24 Diseño Navegacional – Módulo de Solicitudes
Fuente: Elaboración Propia

3.4.4.3 DISEÑO INTERFAZ ABSTRACTA – MÓDULO DE SOLICITUDES

Una vez que las estructuras navegacionales son definidas, se pasa a la siguiente etapa de la metodología OOHDM donde se especifica los objetos navegacionales y cuál de estos activaran la navegación, para este módulo se especifican los ADVs en la figura 3.25.

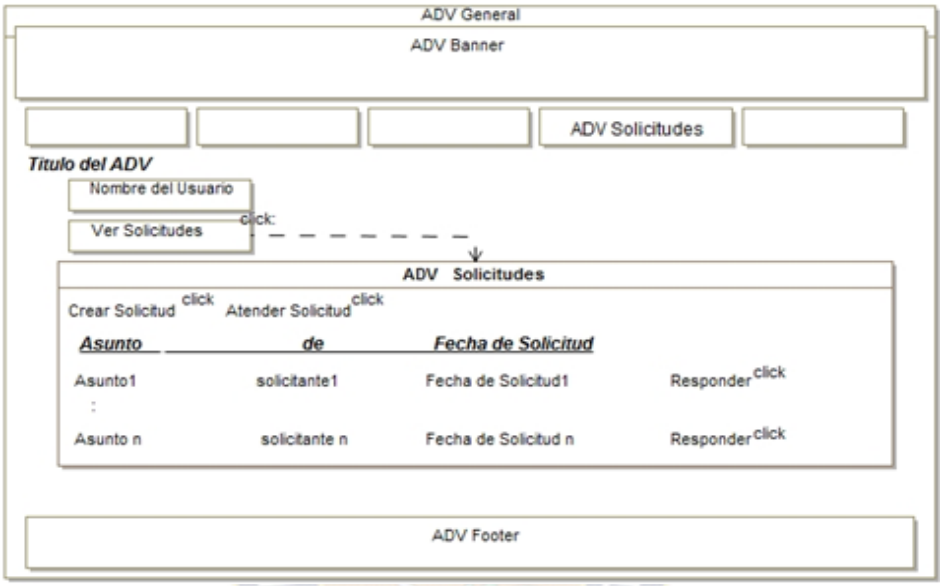


Figura 3.25 ADV Módulo de Solicitudes
Fuente: Creación Propia

La Figura 3.26 muestra la interfaz mediante la cual los usuarios con el rol de usuario normal pueden descargar todo recurso en formato digital que el usuario con el rol de Administrador haya subido a la Web.

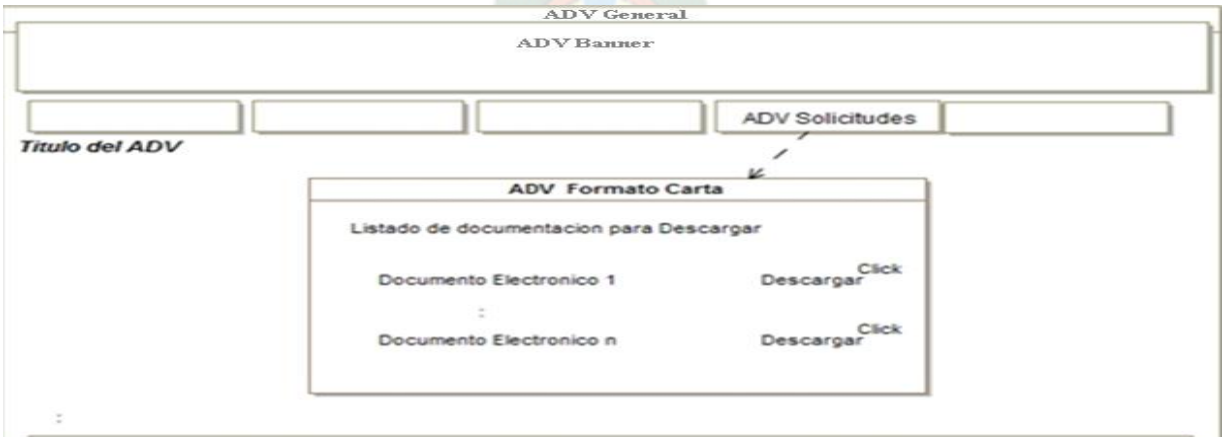


Figura 3.26 Módulo de Solicitudes - Descargas
Fuente: Elaboración Propia

3.4.4.4 IMPLEMENTACIÓN DEL MÓDULO DE SOLICITUDES

En esta etapa consiste en la aplicación de los diseños especificados anteriormente, tales como el diseño conceptual, diseño de navegación e interfaz abstrtacta.

La Figura 3.27 muestra las solicitudes enviadas por parte del usuario administrador al responsable o personal de Archivo Central.

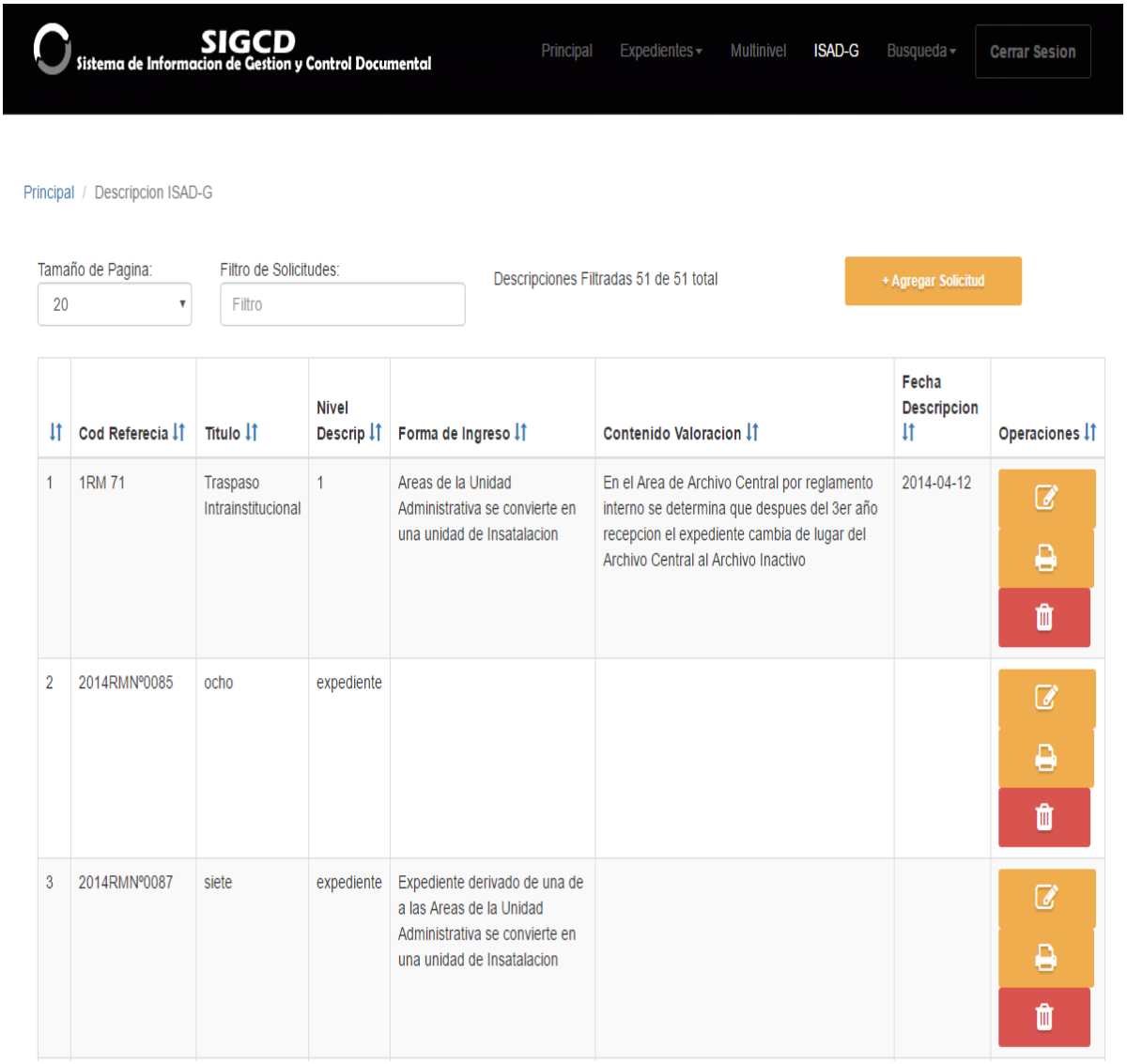
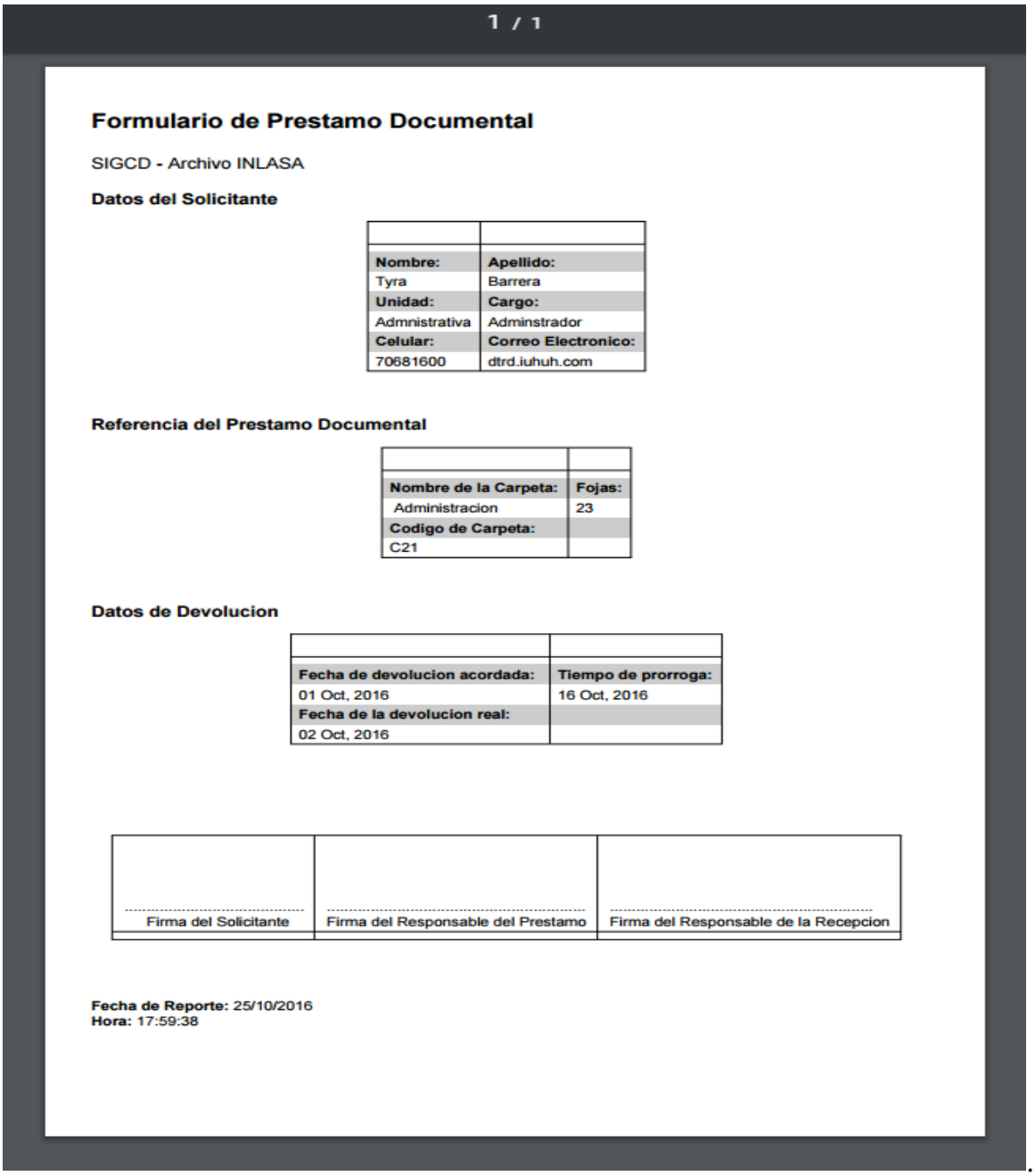


Figura 3.27 Diseño de Implementación – Módulo de Solicitudes
Fuente: Elaboración Propia

Los recursos digitales son aquellas imágenes o expedientes escaneados, algún documento adjunto que este en formato digital puede ser subido y asociado a un recurso digital, esta interfaz se describe en la Figura 3.28.



1 / 1

Formulario de Prestamo Documental

SIGCD - Archivo INLASA

Datos del Solicitante

Nombre:	**Apellido:**
Tyra	Barrera
Unidad:	**Cargo:**
Admnistrativa	Adminstrador
Celular:	**Correo Electronico:**
70681600	dtrd.iuhuh.com

Referencia del Prestamo Documental

Nombre de la Carpeta:	**Fojas:**
Administracion	23
Codigo de Carpeta:	
C21	

Datos de Devolucion

Fecha de devolucion acordada:	**Tiempo de prorroga:**
01 Oct, 2016	16 Oct, 2016
Fecha de la devolucion real:	
02 Oct, 2016	

Firma del Solicitante	Firma del Responsable del Prestamo	Firma del Responsable de la Recepcion

Fecha de Reporte: 25/10/2016
Hora: 17:59:38

Figura 3.28 Interfaz – Módulo de Solicitudes - descargas
Fuente: Elaboración Propia

3.4.4.5 DISEÑO CONCEPTUAL - MÓDULO DE BÚSQUEDA

En esta etapa describiremos las características que nos presenta el módulo de búsquedas, estas búsquedas se las puede realizar vasados en diferentes criterios, ya sea por fecha, por nombre de documento, por dependencia o inclusive por palabra clave tal como se ve en la siguiente figura 3.29.

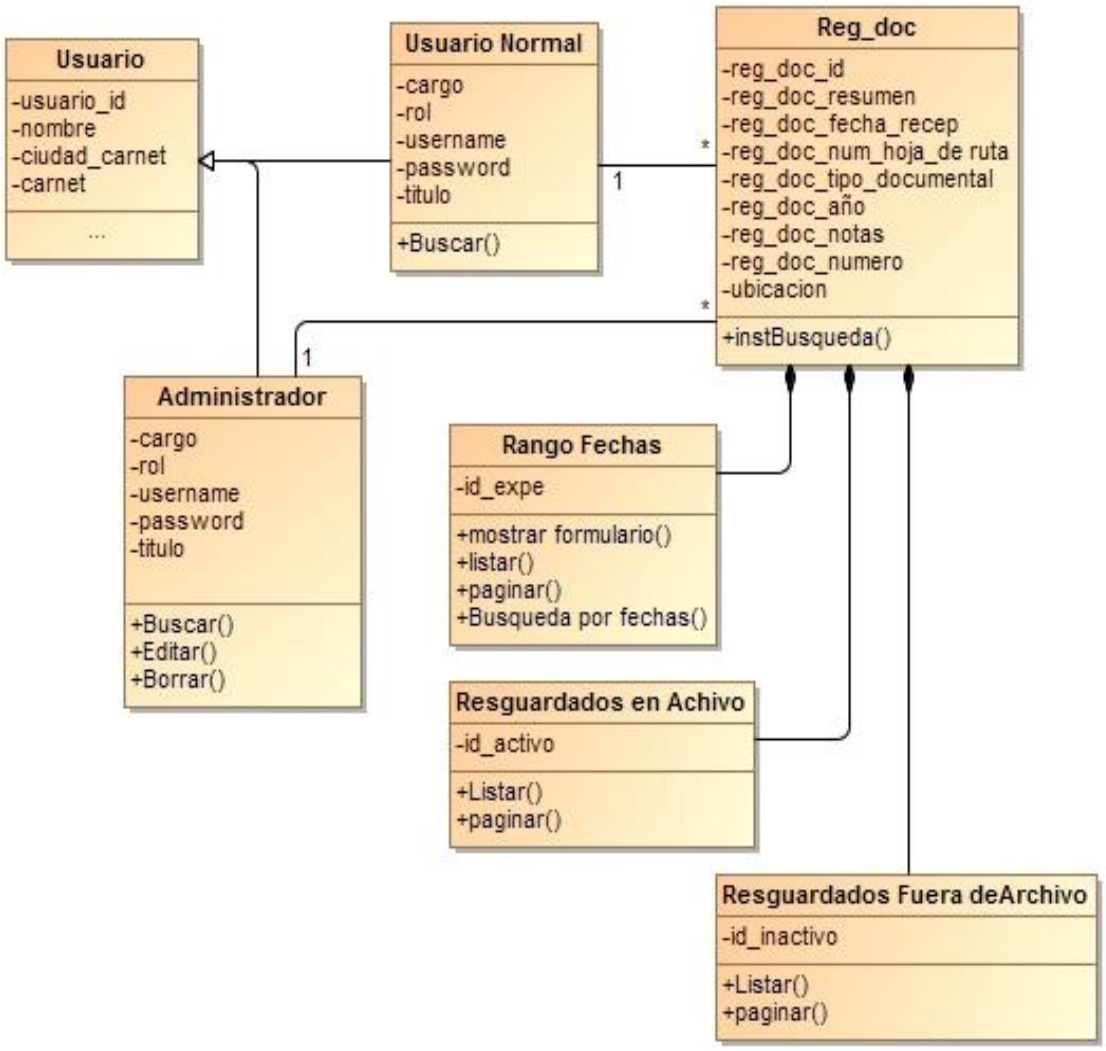


Figura 3.29 Diseño Conceptual – Módulo de Búsqueda
Fuente: Elaboración Propia

3.4.4.6 DISEÑO NAVEGACIONAL – MÓDULO DE BÚSQUEDA

En esta etapa se define clases navegacionales tales como nodos, enlaces y estructuras de acceso (índices y visitas guiadas) inducidas del esquema conceptual, seguidamente se realiza el diagrama de contextos navegacionales representadas en la Figura 3.30.

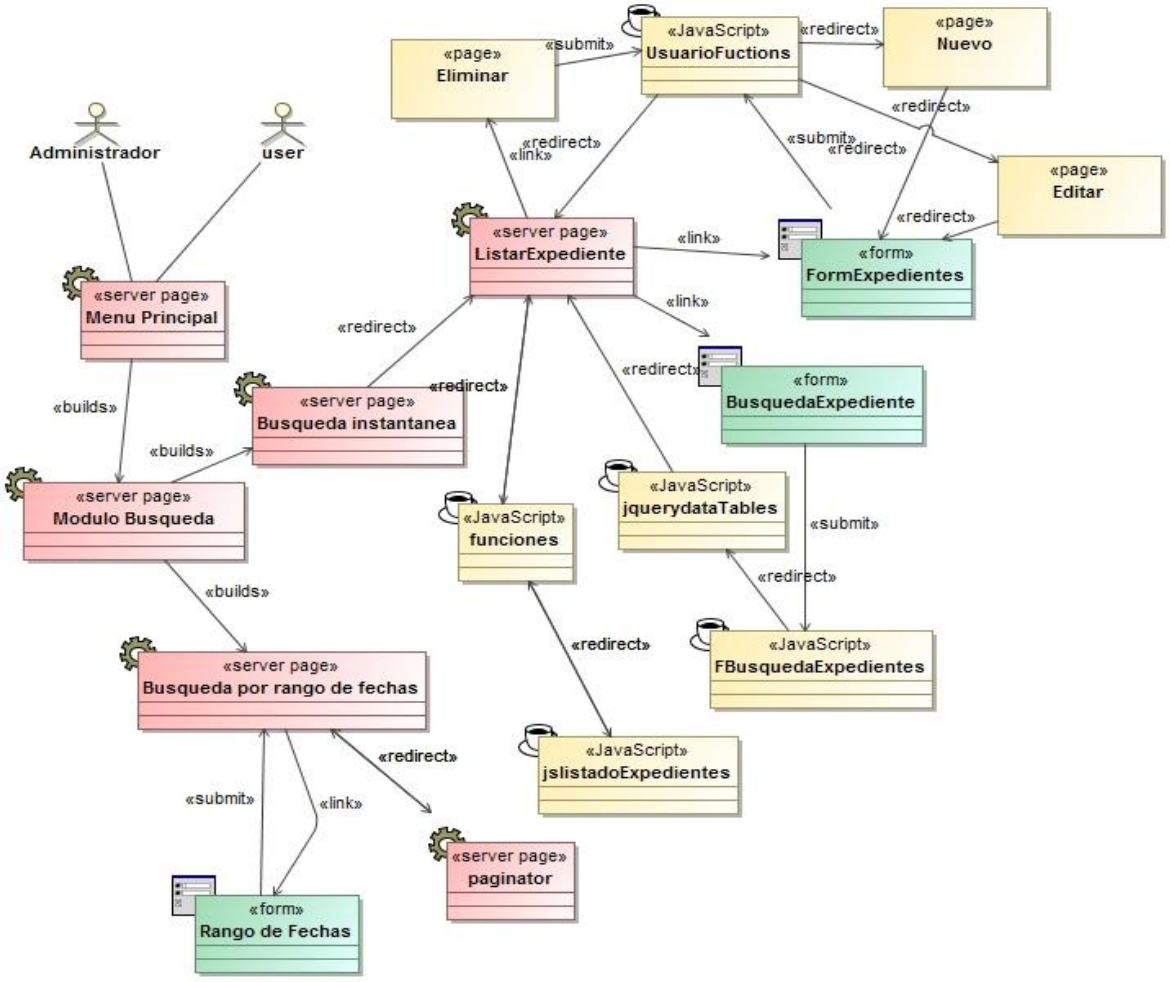


Figura 3.30 Diseño Navegacional – Módulo de Búsqueda
Fuente: Elaboración Propia

3.4.4.7 DISEÑO INTERFAZ ABSTRACTA – MÓDULO DE BÚSQUEDA

Una vez que las estructuras navegacionales son definidas, se pasa a la siguiente etapa de la metodología OOHDM donde se especifica los objetos navegacionales y cuál de estos activaran la navegación para este módulo se especifican los ADVs en las Figuras 3.31.

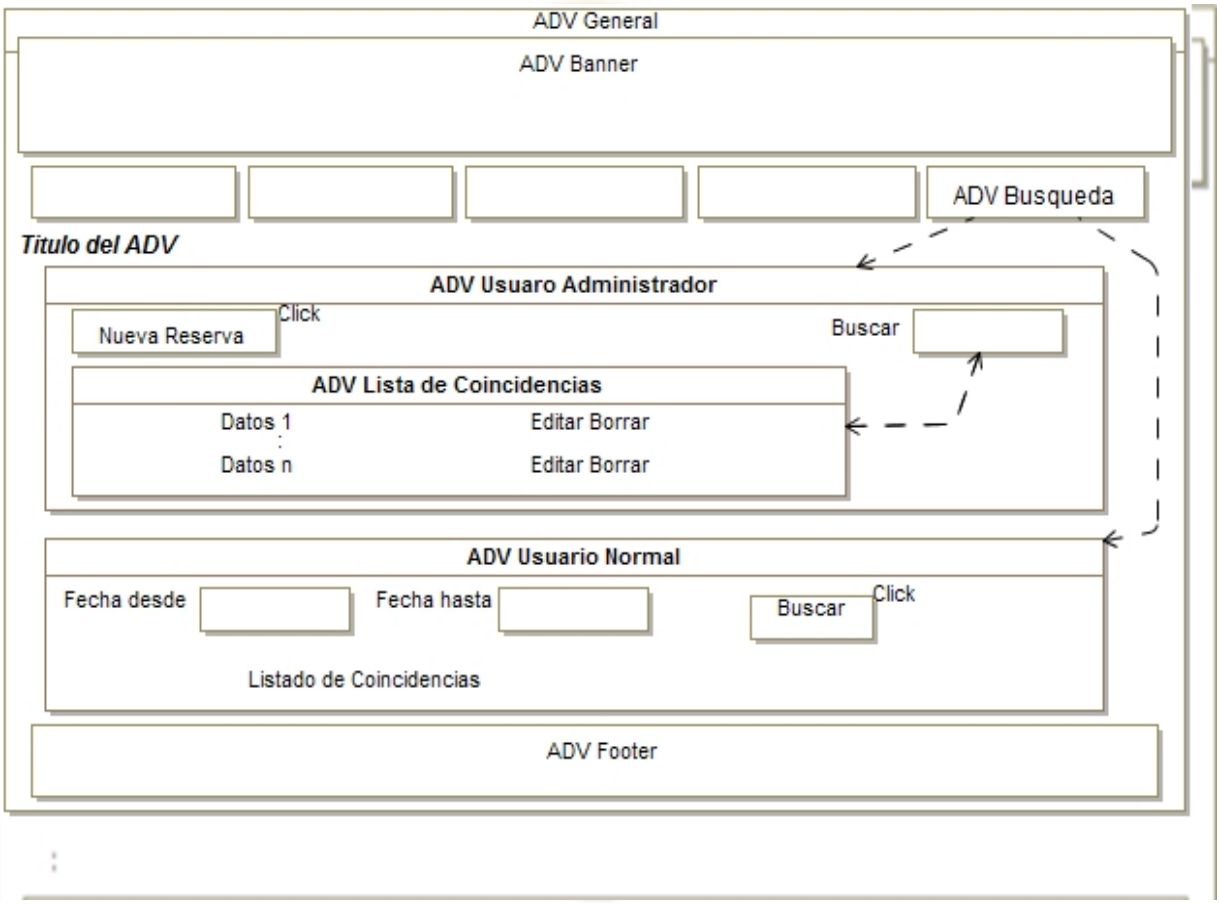


Figura 3.31 Diseño Navegacional – Módulo de Búsqueda
Fuente: Elaboración Propia

3.4.4.8 DISEÑO DE IMPLEMENTACIÓN – MÓDULO DE BÚSQUEDA

En esta etapa consiste en la aplicación de los diseños especificados anteriormente. La Figuras 3.32 y 3.33 muestran la interfaz de búsqueda para documentos resguardados fuera y dentro del Área de Archivo Central.

Figura 3.32 Implementación – Módulo de Búsqueda - Fechas
Fuente: Elaboración Propia

Figura 3.33 Implementación – Módulo de Búsqueda - Dependencias
Fuente: Elaboración Propia

CAPITULO IV

CALIDAD DE SOFTWARE

4.1 CALIDAD DEL SOFTWARE

La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario
(IEEE, Std. 610-1990).

4.2 CALIDAD ISO/IEC 9126

La calidad del software se evalúa teniendo en cuenta la etapa del desarrollo, se deben fijar la metas de la calidad tanto para el software final como para desarrollos incompletos y tener en cuenta que es imposible que las metas y criterios sean iguales para un software pequeño y un gran software empresarial.

Se establecen categorías para las cualidades de la calidad externa e interna y calidad en uso del software, teniendo en cuenta estos 7 indicadores: funcionalidad, confiabilidad, utilidad, eficiencia, capacidad de mantenimiento, portabilidad y calidad en uso, que se subdividen a su vez en varios indicadores estas se pueden medir por métrica interna o externa.

4.2.1 FUNCIONALIDAD

Funcionalidad es la capacidad del software de cumplir y proveer las funciones para satisfacer las necesidades explícitas e implícitas cuando es utilizado en condiciones específicas. Ejemplo, lo que hace el software para satisfacer las necesidades sin tener en cuenta de cómo lo hace y cuando.
La funcionalidad se divide en 5 criterios:

- **Adecuación:** La capacidad del software para proveer un adecuado conjunto de funciones que cumplan las tareas y objetivos especificados por el usuario.
- **Exactitud:** La capacidad del software para hacer procesos y entregar los resultados solicitados con precisión o de forma esperada.
- **Interoperabilidad:** La capacidad del software de interactuar con uno o más sistemas específicos.

- **Seguridad:** La capacidad del software para proteger la información y los datos de manera que los usuarios o los sistemas no autorizados no puedan acceder a ellos para realizar operaciones, y la capacidad de aceptar el acceso a los datos de los usuarios o sistemas autorizados
- **Conformidad de la funcionalidad:** La capacidad del software de cumplir los estándares referentes a la funcionalidad

La funcionalidad de un sistema no puede ser medido directamente, entonces corresponde derivar mediante otras medidas directas como el punto función, para esto se tiene la siguiente relación:

$$\mathbf{PF=Cuenta_Total \times [0{,}65+0{,}01 \times \sum (Fi)]}$$

La tabla 4.1 nos permitirá hacer el cálculo del punto función.

Parámetros de Medida	Cuenta	Factor de Peso			Total
		Simple	Medio	Complejo	
Número de Entradas del Usuario	11	3	4	6	44
Número de Salidas del usuario	13	4	5	7	65
Número de Peticiones de Usuario	10	3	4	6	40
Número de Archivos	11	7	10	14	109
Número de Interfaces	2	5	7	10	14
Total	47				272

Tabla 4.1 Cálculo de Puntos Función
Fuente: Elaboración Propia

∑ (Fi) puede ser de 1 a 14 los valores de ajuste de la complejidad según las respuestas a las siguientes preguntas:

1. ¿Requiere el sistema de copias de seguridad y recuperación fiables?

2. ¿Se requiere comunicación de datos?

3. ¿Existen funciones de procesamiento distribuido?

4. ¿Es crítico el rendimiento?

5. ¿Se ejecutará el sistema en un entorno operativo existente y fuertemente utilizado?

6. ¿Requiere el sistema entrada de datos interactiva?

7. ¿Requiere la entrada de datos interactiva que las transacciones de datos se lleven a cabo sobre múltiples pantallas u operaciones?

8. ¿Se actualizarán los archivos maestros de forma interactiva?

9. ¿Son complejas las entradas, las salidas, los archivos o las peticiones?

10. ¿Es complejo el procesamiento interno?

11. ¿Se ha diseñado código para que sea reutilizable?

12. ¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?

13. ¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente usada por el usuario?

Cada una de las preguntas anteriores es respondida usando la escala de rangos desde 0 (No importante o aplicable) hasta 5 (Absolutamente esencial).

Una vez que se han calculado los puntos de función se usa de forma analógica a las LDC como forma de normar a la productividad, calidad y otros atributos del software.

0	1	2	3	4	5
Sin influencia	Incidental	Moderado	Medio	Significativo	Esencial

∑ (Fi): (i=1 hasta 14) son los valores de ajuste de complejidad.

1=5, 2=5, 3=4, 4=3, 5=5, 6=4, 7=4, 8=5, 9=3, 10=4, 11=3, 12=3, 13=3, 14=5

∑ (Fi) = 56

PF= 240*(0.65+ (0.01*56))

PF = 240.4 valor en punto función.

Si consideramos el máximo valor de complejidad como ∑ (Fi) =70, se tiene:

PF= 240*(0.65+ (0.01*70))

PF= 324

Entonces si ∑ (Fi) es considerada como el 100%, la relación obtenida entre los puntos será:

PF = 290.4/324=0.9

PF = 90%

Por lo tanto la funcionalidad que tiene el sistema es del 90%, tomando en cuenta el punto de función máximo.

4.2.2 CONFIABILIDAD

La confiabilidad es la capacidad del software para asegurar un nivel de funcionamiento adecuado cuando es utilizando en condiciones específicas. La definición de confiabilidad en la **ISO/IEC 2382-14:1997** es "la habilidad de la unidad funcional de realizar una función requerida...". En este caso la confiabilidad se amplía a sostener un nivel especificado de funcionamiento y no una función requerida.

La confiabilidad se divide en 4 criterios:

- **Madurez:** La capacidad que tiene el software para evitar fallas cuando encuentra errores. Ejemplo, la forma como el software advierte al usuario cuando no encuentra espacio suficiente el disco duro donde esta almacenando los datos.
- **Tolerancia a errores:** La capacidad que tiene el software para mantener un nivel de funcionamiento en caso de errores.
- **Recuperabilidad:** La capacidad que tiene el software para restablecer su funcionamiento adecuado y recuperar los datos afectados en el caso de una falla.
- **Conformidad de la fiabilidad:** La capacidad del software de cumplir a los estándares o normas relacionadas a la fiabilidad.

La confiabilidad del sistema es directamente proporcional a la calidad de sus componentes

$$R(t) = e^{-\lambda t}$$

Dónde:

R (t): Función de confiabilidad de un componente en el tiempo t.

λ: Taza constante de fallo.

t: Periodo de operación de tiempo.

- Si n componentes funcionan independientemente conectados en serie y el i-esimo componente tiene confiabilidad Ri(t), entonces la confiabilidad total es dada por:

$$R(t)=R1(t)*R2(t)*R3(t)\ldots Rn(t)$$

- Si n componentes funcionan independientemente y actúan en paralelo y el i-esimo componente tiene confiabilidad Ri(t), entonces la confiabilidad total es dada por:

$$R(t)= 1-[1-R_1(t)]* [1-R_2(t)]* [1-R_3(t)]*\ldots* [1-R_n(t)]$$

Si se tiene el siguiente diagrama de transferencia:

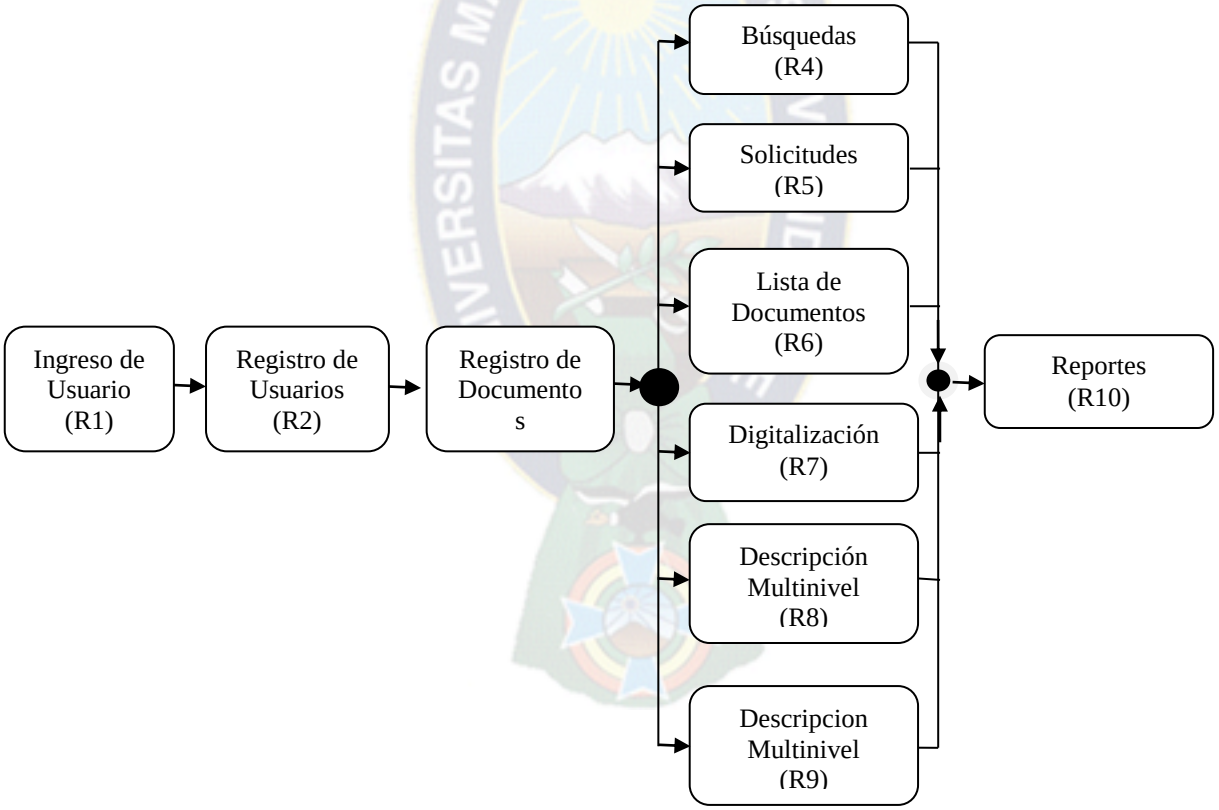


Figura 4.1 Diagrama de Transferencia
Fuente: Elaboración Propia

Luego obtenemos la siguiente tabla 4.2:

Ri()	λ	t(semanas)	$e^{\lambda*t}$
R1()	0.005	6	0.97
R2()	0.005	6	0.97
R3()	0.005	6	0.97
R4()	0.01	6	0.94
R5()	0.01	6	0.94
R6()	0.01	6	0.94
R7()	0.01	6	0.94
R8()	0.01	6	0.94
R9()	0.01	6	0.94
R10()	0.01	6	0.94

Tabla 4.2 Cálculo de Confiabilidad
Fuente: Elaboración Propia

Si todos tienen la misma confiabilidad para cada uno de los componentes es $R(t)= e^{\lambda(-0.01t)}$ la confiabilidad dentro de seis semanas será:

$R_T(t) = \{1-[1-R_4(t)]* [1-R_5(t)]* [1-R_6(t)]* [1-R_7(t)]* [1-R_8(t)]* [1-R_9(t)]\}*[R_1(t)] [R_2(t)] [R_3(t)] [R_{10}(t)]$

$= 0.999*0.887$

$=0.887$

=88.7% de confiabilidad del sistema.

4.2.3 USABILIDAD

La usabilidad es la capacidad del software de ser entendido, aprendido, y usado en forma fácil y atractiva. Algunos criterios de funcionalidad, fiabilidad y eficiencia afectan la usabilidad, pero para los propósitos de la ISO/IEC 9126 ellos no clasifican como usabilidad. La usabilidad está determinada por los usuarios finales y los usuarios indirectos del software, dirigidos a todos los ambientes a la preparación del uso y el resultado obtenido.

Las métricas de usabilidad son:

- **Entendimiento:** La capacidad que tiene el software para permitir al usuario entender si es adecuado, y de una manera fácil como ser utilizado para las tareas y las condiciones particulares de la aplicación. En este criterio se debe tener en cuenta la documentación y de las ayudas que el software entrega.
- **Aprendizaje:** La forma como el software permite al usuario aprender su uso. También es importante considerar la documentación.
- **Operabilidad:** La manera como el software permite al usuario operarlo y controlarlo.
- **Atracción:** La presentación del software debe ser atractivo al usuario. Esto se refiere a las cualidades del software para hacer más agradable al usuario, ejemplo, el diseño gráfico.
- **Conformidad de uso:** La capacidad del software de cumplir los estándares o normas relacionadas a su usabilidad.

La tabla 4.3 nos muestra el cálculo de confiabilidad del sistema.

Usuarios	Facilidad de Comprensión	Facilidad de Aprendizaje	Facilidad de Operación
Usuario 1	90%	95%	100%
Usuario 2	90%	94%	95%
Usuario 3	98%	94%	90%
Usuario 4	94%	95%	96%
Promedio	93%	94.50%	95.25%

Tabla 4.3 Cálculo de Usabilidad
Fuente: Elaboración Propia

En conclusión la facilidad de uso es del 94%.

4.2.4 CAPACIDAD DE MANTENIMIENTO

La capacidad de mantenimiento es la cualidad que tiene el software para ser modificado. Incluyendo correcciones o mejoras del software, a cambios en el entorno, y especificaciones de requerimientos funcionales.

- **Capacidad de ser analizado:** La forma como el software permite diagnósticos de deficiencias o causas de fallas, o la identificación de partes modificadas.
- **Cambiabilidad:** La capacidad del software para que la implementación de una modificación se pueda realizar, incluye también codificación, diseño y documentación de cambios.

- **Estabilidad:** La forma como el software evita efectos inesperados para modificaciones del mismo.
- **Facilidad de prueba:** La forma como el software permite realizar pruebas a las modificaciones sin poner el riesgo los datos.
- **Conformidad de facilidad de mantenimiento:** La capacidad que tiene el software para cumplir con los estándares de facilidad de mantenimiento.

Para calcular la mantenibilidad del sistema utilizaremos las medidas directas proporcionadas por la IEEE 982.1 – 1998 el cual sugiere un índice de madurez del sistema, que consiste en los cambios que producen en cada versión del producto para lo cual se tiene la siguiente relación.

$$\textbf{(1)IMS= } [M_T - (F_a + F_b + F_c)]/M_T$$

Dónde:

M_T= Número de módulos en la versión actual.

F_a= Número de módulos en la versión actual que se han cambiado.

F_b= Número de módulos en la versión actual que se han añadido.

F_c= Número de módulos en la versión anterior que se han borrado en la versión actual.

Si el valor de IMS se aproxima a 1, el sistema empieza a estabilizarse.

Entonces los valores encontrados son:

$$M_{T=6}$$

$$F_a=1$$

$$F_b=0$$

$$F_c=0$$

Reemplazando estos valores en (1)

$$IMS= [6-(1+0+0)]/6$$

$$IMS=0.83 * 100$$

$$IMS=83\%$$

Por tanto puede indicarse que el sistema tiende a estabilizarse en un 83% y el 17% indica que aún no es estable debido a los cambios.

4.2.5 PORTABILIDAD

Es la capacidad que tiene el software para ser trasladado de un entorno a otro, cuenta con las siguientes métricas de calidad:

- **Adaptabilidad:** Es como el software se adapta a diferentes entornos especificados (hardware o sistemas operativos) sin que implique reacciones negativas ante el cambio. Incluye la escalabilidad de capacidad interna (Ejemplo: Campos en pantalla, tablas, volúmenes de transacciones, formatos de reporte, etc.).
- **Facilidad de instalación:** La facilidad del software para ser instalado en un entorno específico o por el usuario final.
- **Coexistencia:** La capacidad que tiene el software para coexistir con otro o varios software, la forma de compartir recursos comunes con otro software o dispositivo.
- **Reemplazabilidad:** La capacidad que tiene el software para ser remplazado por otro software del mismo tipo, y para el mismo objetivo. Ejemplo, la remplazabilidad de una nueva versión es importante para el usuario, la propiedad de poder migrar los datos a otro software de diferente proveedor.
- **Conformidad de portabilidad:** La capacidad que tiene el software para cumplir con los estándares relacionados a la portabilidad.

Además de estos aspectos también podemos considerar la facilidad de instalación, ajuste y adaptación al cambio dado por la siguiente formula:

$$(1)GP=1-[CT/CRD]$$

Dónde:

GP= Grado de Portabilidad

CT= Costo de Transportar

CRD= Costo de Re- desarrollo

Si GP >0, la portabilidad es más rentable que el re-desarrollo

Si GP=1, la portabilidad es perfecta

Si GP<0, el re-desarrollo es más rentable que la portabilidad.

Reemplazando en (1) tenemos:

$$GP=1-[75/1000]$$

$$GP= 0.9625$$

Por lo tanto el sistema puede transportarse de un entorno a otro en un 96% más rentable que el re-desarrollo.

4.2.6 EFICIENCIA

La eficiencia es el conjunto de atributos relacionados con la relación entre el nivel de desempeño del software y la calidad de recursos necesitados bajo condiciones establecidas. Las métricas de calidad son:

- **Comportamiento en el tiempo:** Atributos del software relativos a los tiempos de respuesta y de procesamiento de los datos.
- **Comportamiento de recursos:** Atributos del software relativos a la cantidad de recursos usados y la duración de su uso en la realización de sus funciones.

Bajo estos atributos evaluamos la eficiencia mostrada en la tabla 4.4 de forma global.

Atributo	CrE(Xi)	IEi%
Eficiencia	CNV	78.5
Desempeño	CNV	80
Páginas de Acceso Rápido	CDP	80
Accesibilidad	CNV	77.5
Accesibilidad de Información	CNV	75
Soporte a Versión sólo Texto	CMN	60
Legibilidad al desactivar la Propiedad Imagen del Browser	CNV	90
Imagen con Titulo	CB	100
Legibilidad Global	CDP	80
Accesibilidad de Ventanas	CNV	80
Número de Vistas considerando Marcos (frames)	CMN	60
Versión sin Marcos	CB	100

Tabla 4.4 Cálculo de la Eficiencia
Fuente: Elaboración Propia

4.3 SEGURIDAD

4.3.1 SEGURIDAD DE LA BASE DE DATOS

La seguridad del sistema se divide en tres partes Encriptación de datos, Autenticación de Usuarios, Manejo de Sesiones que se describirán a continuación.

4.3.1.1 ENCRIPTACIÓN DE DATOS

Consiste en transformar un valor legible en uno ilegible mediante un algoritmo lo cual permite que cierta información sea inútil para aquellas personas que por alguna razón tengan acceso al sistema o a la base de datos. En PHP se utiliza la función MD5 (MessageDigest 5), que es una función hash irreversible (de un sólo sentido), es decir, encripta el password tecleado por el usuario y es imposible que partiendo desde la cadena encriptada se vuelva a la contraseña origen. Por esto mismo no hay problema de que alguien pueda acceder al campo encriptado de la base de datos.

El Sistema Web de Gestion de Archivo esta implementado en lenguaje PHP y gestor de base de datos MySQL, hace uso del algoritmo MD5. La base de datos tiene que tener algún tipo de protección. Es por ello que algunos campos se guardan encriptados en la base de datos, principalmente cuando una página requiere el nombre de usuario y contraseña, esta última se encripta y se guarda en la Base de datos.

4.3.1.2 AUTENTICACIÓN DE USUARIOS

Un elemento importante para la seguridad del sistema y que sirve como requisito para ingresar al sistema, la autenticación de usuarios consta de un nombre de usuario y una contraseña variables definidas en el momento de registro del sistema

4.3.1.3 MANEJO DE SESIONES

Las sesiones son un tipo de variable que tiene como tarea guardar el estado de un usuario en particular. Además el uso de sesiones permite restringir el acceso a un url, puesto de no existir una sesión, el sistema re direccionará a la ventana de Formulario para autenticación de usuarios. En el momento del ingreso se le asigna una sesión al usuario mediante su nombre de usuario y su

contraseña, esta sesión guarda los datos del usuario identificando el tipo de usuario para determinar si el usuario es administrador o usuario normal.

4.4 ANÁLISIS DE COSTOS

El análisis de costos implica una estimación del software mediante el método de estimación paramétrico basado en puntos función descrito en el segundo capítulo.

PF Lite es un método de estimación paramétrico derivado del método de análisis de punto función FPA definido por el Grupo Internacional de Usuarios de Puntos Función IFPUGen esta versión se considera que todos los elementos tiene una complejidad media. Al calcular los PF Lite se obtiene una estimación del tamaño funcional del proyecto de desarrollo por lo que se aconseja considerar un rango del valor obtenido PF±20%.

Para la estimación del esfuerzo, duración, equipo de trabajo, costo del desarrollo del proyecto software se utilizara como base a los PF calculados en la Tabla 4.5.

Elemento	Cantidad		Peso		Total
Datos-ILF: Ficheros Lógicos Internos	7	x	10	=	70
Interfaces-ELF: Ficheros Lógicos Externos	2	x	7	=	14
Entradas-EI: Entradas Externas	20	x	4	=	80
Salidas-EO: Salidas Externas	15	x	5	=	75
Consultas-EQ: Consultas Externas	13	x	4	=	52
			PF Suma de Totales		291
				+20%	349.2
				-20%	232.8

Tabla 4.5 Puntos función no ajustados
Fuente: Elaboración Propia

La cuenta final de los PF de sistema desarrollado con PF Lite es:

$$\boldsymbol{PF = 291 \pm 58.2}$$

4.4.1 ESTIMACIÓN DEL ESFUERZO

Para la estimación del esfuerzo se utilizara la siguiente relación:

$$Esfuerzo = C * PF^{E}$$

Dónde:

$$C = 58.16 \; y \; E = 0.664$$

$$Esfuerzo(pro) = 58.16 * 291^{0.664}$$

$$Esfuerzo(pro) = 2515 \; horas$$

$$Esfuerzo(min) = 2169 \; Horas$$

$$Esfuerzo(max) = 2839 \; Horas$$

El desarrollar el proyecto requerirá un esfuerzo aproximado de 2515 horas.

4.4.2 ESTIMACIÓN DE LA DURACIÓN

$$\text{Duración} = C * (\text{Esfuerzo})^{E}$$

Dónde:

$$C = 0.411 \text{ y } E = 0.328$$

$$\text{Duración} = 0.411*(2515)^{0.328}$$

$$\text{Duración} = 5.36 \text{ [Meses]}$$

El tiempo estimado de desarrollo del sistema es de aproximadamente de 5 meses.

4.4.3 ESTIMACIÓN DEL EQUIPO DE TRABAJO

Cantidad de personas = Esfuerzo_total / (Duracion_total * 20 * 8)

Cantidad de personas = 2515/ (5.36 * 20 * 8)

Cantidad de personas = 2.93 [Personas]

El equipo de desarrollo deberá estar conformado por 3 personas.

4.4.4 ESTIMACIÓN DEL COSTO DE DESARROLLO

Para este cálculo se toma como el valor del costo promedio hora igual a 25 Bs, se obtiene este valor al considerar que un profesional en el área de tecnologías de la información percibe en promedio un sueldo de 4000 Bs mensuales constituido de 20 jornadas laborales de 8 horas.

Costo = Esfuerzo_total * Costo_promedio_hora

Costo = 1954 * 25, Costo = 62875 [Bs]

El desarrollo del sistema tendrá un costo aproximado de 62875 Bs.

4.5 ANÁLISIS DE BENEFICIOS

Los beneficios del presente proyecto son aquellos tienen un impacto importante para la institución.

- Mejora en la atención a solicitudes.
- Facilita la consulta y el acceso a la documentación de la Institución.
- Reducción respecto al tiempo en el tratamiento archivístico y en el servicio respecto a la atención de solicitudes.
- Aumenta la transparencia organizativa de la documentación resguardada a través de la navegación multinivel de las agrupaciones documentales.
- Contempla la incorporación de los recursos digitales asociados a documentos y descripción archivística en distintos niveles de descripción.
- Ahorra tiempo y esfuerzo en el registro de la documentación que ingresa en el Archivo en al menos 3 veces más rápido que con el sistema anterior.

Al contar con un administrador del sistema que en este caso es el un miembro del personal o el responsable del Área de Archivo Central se da mayor seguridad a todos los usuarios en lo que se refiere a la confidencialidad, la integridad y la disponibilidad creando roles y permisos para navegar por el sistema.

CAPÍTULO V
CONCLUSIONES Y RECOMENDACIONES

A la conclusión del desarrollo del presente proyecto y habiendo logrado la implementación del Sistema de Información de Gestión y Control Documental para el Instituto Nacional De Laboratorios De Salud "Néstor Morales Villazón" - INLASA, se llega a las siguientes conclusiones y recomendaciones.

5.1 CONCLUSIONES

- La implementación del sistema servirá como herramienta de gestión que ayudará a mejorar el tratamiento documental.
- La aplicación de la norma ISAD (G) permite la realización de la descripción en forma estructurada con elementos bien definidos, tratando con esto que la descripción documental se realice con un modelo ya definido, facilitando el trabajo y en cierta medida permitiendo la elaboración de descripciones más homogéneas.
- El módulo de búsqueda por criterios múltiples redujo notablemente el tiempo invertido en la consulta de documentación para de esta manera brindar una atención más eficiente sin pérdida de tiempo en las solicitudes.
- Se incorporó imágenes y/o recursos digitales asociados a expedientes ampliando el acceso y la preservación del mismo

5.2 RECOMENDACIONES

Las siguientes recomendaciones contribuirían en la ampliación del nivel de alcance del presente proyecto.

- Se recomienda la incorporación de un módulo de control para notas internas y externas que forman parte de la documentación en su fase administrativa.
- Se recomienda crear normas y políticas de uso del sistema enfocándose en darle un buen uso de manera que su uso no solo este orientado al tratamiento archivístico sino que también este orientado al servicio y difusión del patrimonio documental resguardado en el INLASA.

- Se recomienda la incorporación de las normas de descripción archivística EAD que se define también como una forma de comunicación que permite a los archivos distribuir electrónicamente instrumentos de descripción a usuarios sin importar la distancia y la norma ISAAR (CPF), Norma Internacional sobre los Registros de Autoridad de Archivos relativos a Instituciones, Personas y Familias que en conjunción con la norma ISAD(G) representan un sistema de comunicación entre creadores de documentos, usuarios en general y gestores de la información.
- Se recomienda incorporar un método de encriptación para el resguardo de documentos en cuanto a su dirección lógica.

BIBLIOGRAFÍA.

Alberch, I. & Fugueras, R. (2003). *Los archivos entre la memoria histórica y la sociedad del conocimiento*. Editorial MOC. Barcelona.

Aramayo, S. (2013). *Programa Institucional de Gestión de Bioriesgo- INLASA*. Bolivia: La Paz: Publicaciones Ministerio de Salud.

Beck, K. (2000). "*Extreme Programming Explained. Embrace Change".* UK: Ediciones Pearson Education .(Traducido al español como:"Una explicación de la programación extrema. Aceptar el cambio". Addison Wesley).

Cabezas, G. L. (2004). *Manual Imprescindible de PHP5*. Madrid: Ediciones Anaya Multimedia (Grupo Anaya, S.A.).

Calero, M. (2003). *Apolo Software. Obtenido de Una explicacion de Programación Extrema*.Recuperado de: http://www.apolosoftware.com/

De Concetta. (2011). *Curso Básico de Correspondencia*. México. Editorial. Trillas.

Giorgis, Muños & Horra (2009) Desarrollo de aplicaciones basadas en *OOHDM.*

Gómez, López, Migani & Otazú (2009). Un Modelo de Estimación de Proyectos de Software.

ISAD(G). (2000). *Norma Internacional General de Descripción Archivística 2ª ed.*

Madrid, España: Consejo Internacional de Archivos.

Laudon. (1996). *Administración de los sistemas de Información*: Editorial Prentice Hispanoamericana S.A. 3ra Edición.

Oporto, L. & Campos, L. (2009). *Guía Práctica Para La Organización De Archivos Administrativos.* Bolivia: Ediciones Consultora BCG Y M SRL.

Oporto, L. & Ramírez, F. (2009). *Legislación Archivística Boliviana*. Bolivia: Ediciones Consultora BCG Y M SRL.

Palacio, J. (2006). Flexibilidad con Scrum. Edición Octubre 2008.

Pérez J. & Gardey A. (2009). Definición de ingeniería de software. Recuperado de: http://definicion.de/ingenieria-de-software.html.

Platini, (2003). *Calidad en el Desarrollo y Mantenimiento del Software.*

Pressman, (2006). *Ingeniería de Software*, Edición Concepción Fernández.

Reglamento de Archivos Administrativos del Ministerio de Salud y Previsión Social/ (2001).*Resolución Ministerial Nro. 0386.*

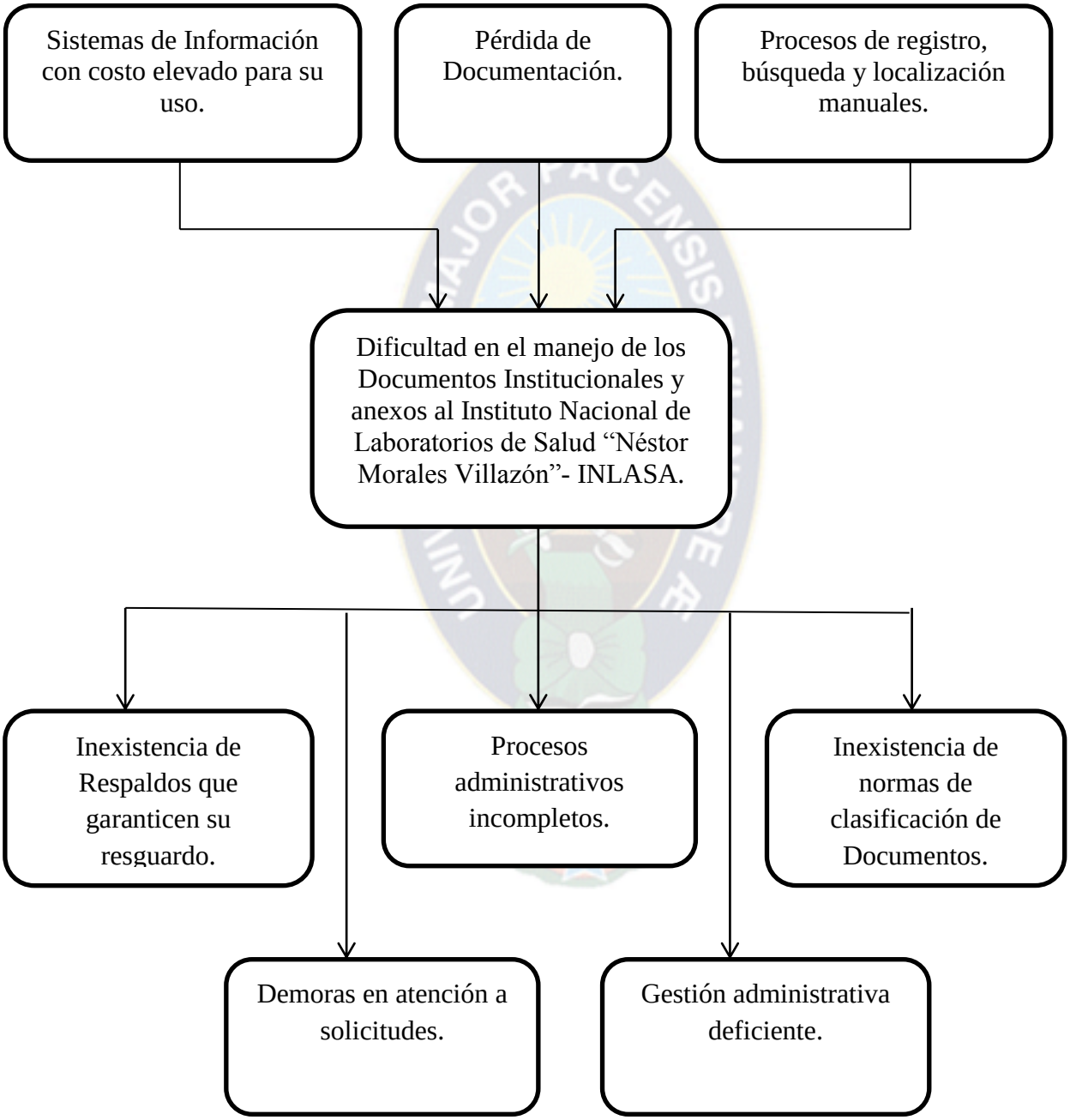
Sección 1.02 Rossi, G. (1996) *El Modelo hipermedia de diseño orientado a objetos* (OOHDM)
Schmuller, (2007). *Aprendiendo UML*. México: Editorial Prentie Hall Naucalpan de Juárez.

Sección 1.03 Schwabe, D. (1996). *El Modelo hipermedia de diseño orientado a objetos* (OOHDM)
Vega, R.L. & Fernández, J.C. (2009), *Coordenadas paradigmáticas, históricas y epistemológicas de la Ciencia de la información*: una sistematización, (UK), Vol. 14, No. 2. Recuperado de: http://InformationR.net/ir/14-1/paperxxx.html.

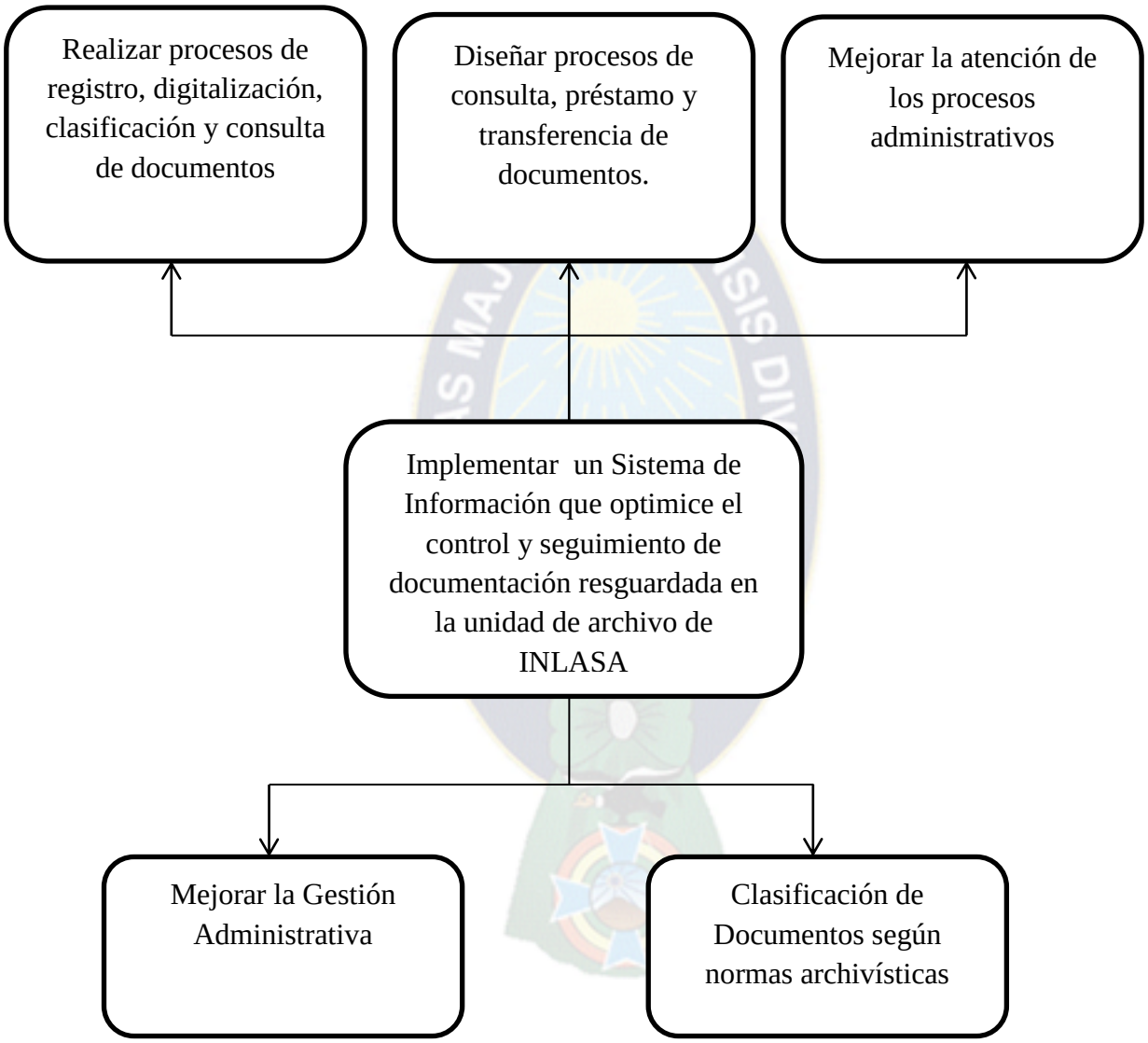
Schwaber, K. (2004) *Gestión de proyectos ágil con Scrum.*

ANEXOS

ANEXO A ÁRBOL DE PROBLEMAS



ANEXO B ÁRBOL DE OBJETIVOS



ANEXO C - ENCUESTAS

ENTREVISTA DIRIGIDA A AUTORIDADES Y FUNCIONARIOS DEL INSTITUTO NACIONAL DE LABORATORIOS DE SALUD "NÉSTOR MORALES VILLAZÓN" - INLASA.

FORMACIÓN PROFESIONAL : _____

CARGO QUE DESEMPEÑA : _____

RESPONDA A LAS SIGUIENTES PREGUNTAS O MARQUE CON UNA "X" DONDE CORRESPONDA

1. ¿Ud. Conoce las normas internas existentes en el Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón" - INLASA. ?

a) Si

b) No

c) No Sabe

2.¿Existen en el Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón". quejas por perdida de documentación?

a) Si

b) No

c) No Sabe

3.¿Existen quejas sobre los plazos al responder a solicitudes?

a) Si

b) No

c) No sabe

4. ¿Exigen internamente el cumplimiento de plazos para una respuesta oportuna?

a) Si

b) No

c) No sabe

5. ¿Existen o han existido despidos a funcionarios por la pérdida de documentación?

a) Si

b) No

c) No sabe

6. ¿Existe el interés por parte de la Entidad en mantener y capacitar a los funcionarios para una buena gestión documental?

a) Si

b) No

c) No sabe

7. ¿Existe normativa para el control óptimo de la documentación?

a) Si

b) No

c) No sabe

8. ¿Existe preocupación del Director(a) por una buena gestión administrativa?

a) Si

b) No

c) No sabe

9. ¿Existe la necesidad de un Reglamento Interno para el Control y Gestión de los documentos existentes en el Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón".?

a) Si

b) No

c) No sabe

10. ¿Existe la necesidad de incorporar tecnologías de la información y comunicación para el control y gestión de los documentos existentes en el Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón".?

a) Si

b) No

c) No sabe

¡GRACIAS POR SU COLABORACIÓN!

ANEXO D – HISTORIAS DE USUARIO

Historia de Usuario	**Número:**1	**SIGCD**
Nombre Historia de Usuario: Administración de la Gestión Documental		**Prioridad:** Alta (Alta / Media / Baja)
Usuario: Lic. Miriam Rivas P.	**Iteración Asignada: PRIMERA ITERACIÓN**	
Historia: En la entrevista que se tuvo con la encargada de archivo se pudo evidenciar que cada documento que resguarda en el Archivo Central se puede identificar mediante su Número de hoja de Ruta que es un código único utilizado para derivar a las diferentes unidades que conforman el Instituto, el número de ruta significa el número de carta que entre por ventanilla en general, también se lo puede identificar por el número de Cite que es el lugar de donde pertenece el documento, de esta manera a cada documento se le asigna un número de identificación y fecha para luego archivarlo. Existen distintos tipos de documentos y se considera como documento principal la unidad documental a la que se asigna el número de identificación y la fecha en la que se recepciona el documento.		
Objetivo: Desarrollar un módulo de descripción que permita la integración de las descripciones en un sistema unificado de información basado en la norma ISAD(G) y la navegación por el cuadro de clasificación documental.		
Desarrollador: Richard Paucara Machaca		
Estimación: 1 semana		

Historia de Usuario	**Número:**2	**SIGCD**
Nombre Historia de Usuario: Recursos Digitales y Registro de Documentos		**Prioridad:** Media (Alta / Media / Baja)
Usuario: Lic. Miriam Rivas P.	**Iteración Asignada: SEGUNDA ITERACION**	
Historia: Para realizar un mejor registro de la documentación esta debe contar con un recurso digital para una mejor difusión y rápido acceso. Cabe señalar que la documentación resguardada forma parte de un archivador que luego se convierte en un empastado al año siguiente de su recepción dicho empastado contiene un número determinado y homogéneo de expedientes o documentos. A la vez se adjunta todos los documentos que hayan sido registrados en el día, de tal forma que se genere un código con las claves o iniciales de cada unidad a la que corresponda, seguida de la fecha de recepción		
Objetivo: Asociar un recurso digital a cada documento que se registre, los recursos digitales disponibles pueden ser documentos pdf, documentos Word, hojas de cálculo Excel, imágenes escaneadas en el momento de la recepción del documento físico. Mediante una interfaz amigable mediante el cual se pueda ingresar y visualizar información del estado de los empastados resguardados por año.		
Desarrollador: Richard Paucara Machaca		
Estimación: 1 semana		

Historia de Usuario	**Número:**3	**SIGCD**
Nombre Historia de Usuario: Administración y Control de Usuarios		**Prioridad:** Alta (Alta / Media / Baja)
Usuario: Lic. Miriam Rivas P.	**Iteración Asignada: TERCERA ITERACION**	
Historia: Las actividades del Área de Archivo Central se resume en conservación de los documentos y servicio, cada una de ellas requiere una serie de actividades que permitan la correcta manipulación de los documentos. El registro de la documentacion anterioriormente se la realizo en documentos excel y word, por ende no se cuenta con la tecnología necesaria para su correcto uso. Para acceder al sistema cada uno de los usuarios debera contar con una cuenta de usuario que consta de un nombre de usuario y contraseña. Cada usuario debera ser registrado por el Administrador.		
Objetivo: Desarrollar un módulo de administración donde se registre, actualice o elimine todos los documentos resguardados en un sistema unificado. Así también este módulo servirá para proporcionar nuevas cuentas de usuario, para actualizarlas o eliminarlas por parte del Administrador. Los datos de gestiones pasadas se las importaran a mysql después de adaptarlas en un solo formato unificado y así poder contar con toda la documentación necesaria.		
Estimación: 1 semana		

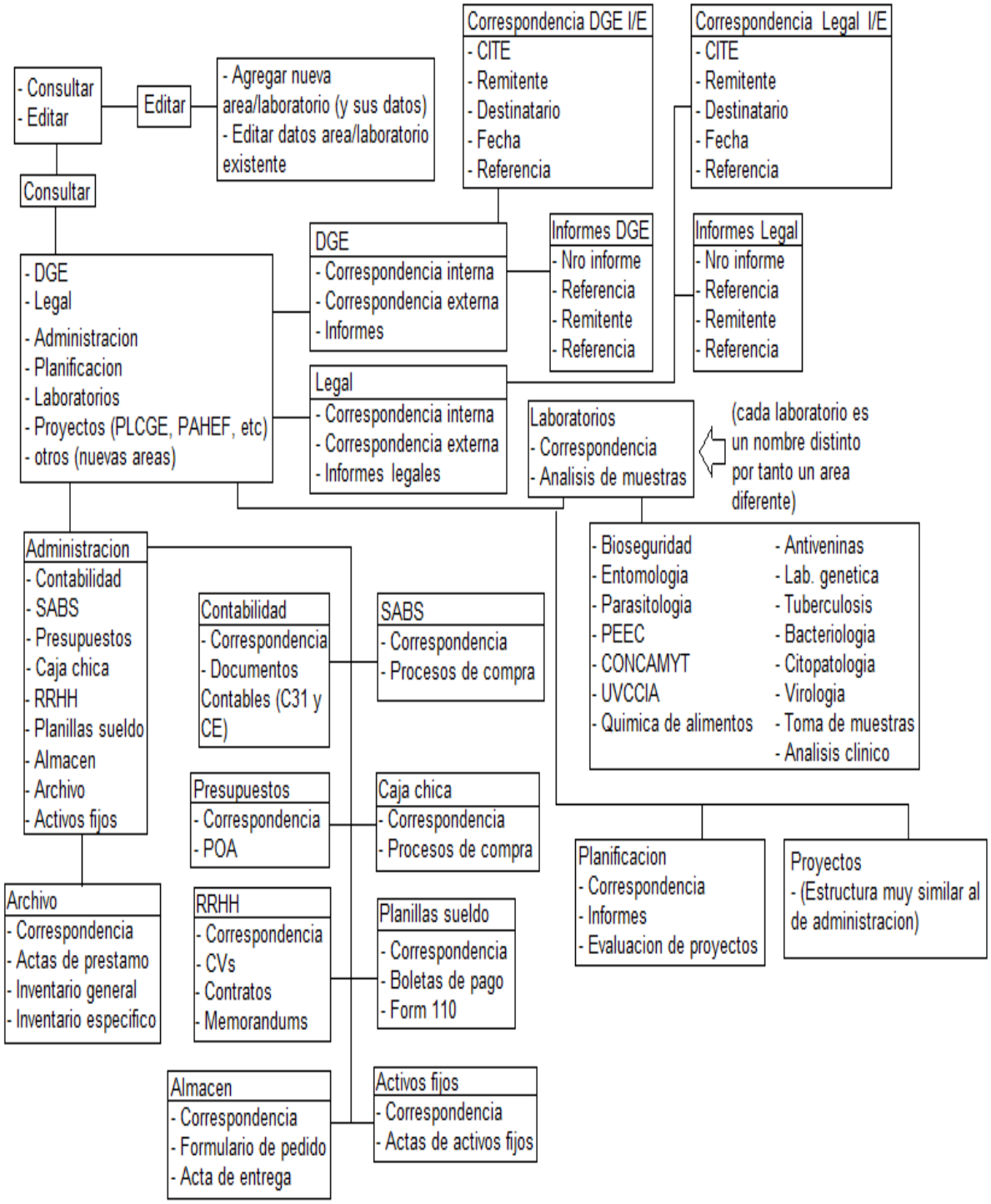
Historia de Usuario	**Número:**4	**SIGCD**
Nombre Historia de Usuario: Búsquedas por Fechas o dependencias		**Prioridad:** Media (Alta / Media / Baja)
Usuario: Lic. Miriam Rivas P.	**Iteración Asignada: CUARTA ITERACION**	
Historia: La búsqueda de expedientes se la debe poder realizar por medio de criterios según sea por fechas o por dependencias, se debe generar un listado de los registros que contengas el criterio de búsqueda.		

Objetivo: Desarrollar el módulo de búsquedas que permita la gestión y control de las búsquedas por parte de los usuarios del sistema.
Estimación: 1 semana

Historia de Usuario	**Número:**4	**SIGCD**
Nombre Historia de Usuario: Diseño de Base de datos		**Prioridad:** Alta (Alta / Media / Baja)
Usuario: Lic. Miriam Rivas P.	**Iteración Asignada: PRIMERA ITERACION**	
Historia: La Base de Datos que tendrá el nombre de SGA debe tener fácil acceso a la documentación y al tratamiento archivístico que se realiza en el Área de Archivo Central		
Objetivo: Diseñar una base de datos que facilite el acceso a la documentación resguardada, y que pueda brindar soporte en la práctica archivística.		
Estimación: 1 semana		

Historia de Usuario	**Número:**5	**SIGCD**
Nombre Historia de Usuario: Gestión y control de solicitudes.		**Prioridad:** Media (Alta / Media / Baja)
Usuario: Lic. Miriam Rivas P.	**Iteración Asignada: PRIMERA ITERACION**	
Historia: El registro de esta solicitud se la efectúa por medio de una carta de solicitud formal de documentación, caso de que el documento solicitado no se encuentre en el Área de Archivo Central, o solicitudes específicas por parte de funcionarios del instituto.		
Objetivo: Desarrollar el módulo de solicitudes que permita la gestión y control de las solicitudes por parte de los usuarios del sistema.		
Estimación: 1 semana		

ANEXO E – ORGANIGRAMA GENERAL INLASA



ANEXO F – GLOSARIO DE TÉRMINOS

- **Administración de documentos** Es el conjunto de métodos y prácticas destinadas a planificar, dirigir y controlar la producción, circulación, organización, conservación, uso seleccionado de los documentos y su destino final del documento.
- **Archivo activo** Es el que conserva la documentación activa
- **Archivo Administrativo** Es el archivo que reúne los documentos, es aquel al que se transferirá la documentación que deba conservarse permanentemente, por no haber sido de dictamen de eliminación en el archivo intermedio.
- **Archivo Central** Dentro del ciclo vital de los documentos, es el que coordina y controla el funcionamiento de los distintos archivos activos y reúne los documentos transferidos por los mismos, una vez finalizado su trámite y cuando su consulta no es constante. Concentra toda la documentación generada por las distintas unidades de la institución .Debe estar ubicado en el nivel operativo en relación a sus funciones técnicas, y a nivel directivo en relación a sus funciones directrices de información y administración documental de toda la Institución, dependiendo de Secretaría Ejecutiva
- **Archivo de trámite o gestión** Unidad responsable de la administración de uso cotidiano y necesario para la consulta de la oficina u unidad administrativa.
- **Archivo histórico** Unidad responsable de organizar, conservar, administrar, describir y divulgar la memoria institucional
- **Archivo inactivo** Conjunto ordenado y clasificado de expedientes de documentación, inactiva que registran y conservan la historia administrativa y funcional de la entidad.
- **Archivo Intermedio** Dentro del ciclo vital de los documentos, es aquel al que se transfieren los documentos de llamados archivos centrales, inactivos o memorias institucionales de las entidades públicas, cuando su consulta por los organismos productores esporádica, y en el que permanecen hasta su destino final, que puede ser su conservación permanente o su eliminación.
- **Archivo** Un archivo, es el conjunto de documentos que se genera o produce en una unidad administrativa, área u oficina como consecuencia natural y jurídica de los asuntos o trámites de la sociedad civil en cualquier soporte.
- **Asunto** Contenido específico de los documentos de archivo y expedientes de una serie que permite la individualización dentro del conjunto de características homogéneas al que están

integrados. Es el tema, motivo, argumento, etc. De qué trata un expediente o documento de archivo, que se genera como consecuencia de una acción administrativa.

- **Baja documental o expurgo** Es la eliminación de documentos que hayan perdido sus valores administrativos, legales, fiscales, contables o técnicos y que no tenga valor histórico.
- **Calendario de caducidad de documentos** El responsable del archivo central debe programar el calendario de documentos con fichas de conclusión de los expedientes transferibles, con la finalidad de dar baja o transferencia secundaria al documento.
- **Ciclo vital de los documentos** Etapas por las que sucesivamente atraviesan los documentos desde que se producen en el archivo activo y pasan por el archivo central y/o intermedio hasta que se eliminan y se conservan en un archivo permanente**.**
- **Ciclo vital de los documentos** Se denomina ciclo vital de los documentos a la trayectoria que tienen desde su inicio o producción administrativa hasta su conclusión.
- **Clasificación de archivo** Proceso de identificación y agrupación de expedientes homogéneos, sobre la base de la estructura de clasificación determinado por el Archivo Central.
- **Clasificación** Operación intelectual que consiste en el establecimiento de las categorías y grupos que reflejan la estructura orgánica y/o funcional del fondo documental
- **Correspondencia** Medio de comunicación entre funcionarios o autoridades de una rama administrativa y comprende también a las comunicaciones que estas envían a personas particulares.
- **Custodia** Responsabilidad sobre el cuidado de los documentos que se basa en su posesión física y que no implica la propiedad jurídica ni el derecho a restringir el acceso a los documentos
- **Documento de archivo** Documentos, o soporte documental, creado, recibido, utilizado en las actividades de la Unidad Administrativa
- **Documentos activos** Documento de uso frecuente, necesario para el ejercicio de las actividades de las Unidades Administrativas. Se conserva en los Archivos de Gestión.
- **Documentos semiactivos** Documentos de uso esporádico que debe conservarse en el Archivo Central.
- **Expediente** Unidad documental formada por un conjunto ordenado de documentos generado orgánica y funcionalmente por un sujeto productor en la resolución de un mismo asunto.
- **Fondo** Conjunto de documentos o de series generadas por cada uno de los sujetos productores conforman la estructura de un organismo en el ejercicio de sus competencias

- **Legalización** Procedimiento de consulta que describe las unidades de instalación y/o las unidades de una o series respetando su estructura.
- **Ordenación** Operación archivística realizada dentro del proceso de organización, que consiste en establecer secuencia dentro de las categorías y grupos, de acuerdo con las series naturales cronológicas y/o alfabéticas
- **Organización documental** Conjunto de actividades destinadas a ordenar, clasificar información documental.
- **Portada** Primera página del expediente donde se indican los elementos que describen su contenido.
- **Principio de Procedencia** Principio fundamental de los archivos que establece que los documentos producidos por institución u organismo no deben mezclarse con el de los otros.
- **Principio de respeto de orden original** Principio que establece que no se debe alterar la organización dada al fondo documental por la unidad productora, cualquiera que esta sea. Deriva del principio de procedencia.
- **Serie** Conjunto de documentos producidos por un sujeto (entidad) en el desarrollo de una misma actividad administrativa y regulada por la misma norma jurídica y/o de procedimiento.
- **Tipo documental** Unidad documental productora por un organismo en el desarrollo de una competencia concreta, regulada por una norma de procedimiento y cuyo formato. Contenido informativo y soporte es homogéneo.
- **Transferencia** Procedimiento habitual de ingreso de fondos en un archivo mediante traslado de las fracciones de series documentales, una vez que estas han cumplido el plazo de permanencia fijado por las normas establecidas en la valoración para cada una de las etapas del ciclo vital de los documentos.
- **Unidad de instalación** Es la formada por una o varias unidades documentales homogéneas reunidas para su conservación y colocación en los archivadores verticales, cajas, carpetas legajos, etc.
- **Unidad documental** Elemento indivisible de una serie documental, documento o varios que forman un expediente.
- **Valor documental** Valor de los documentos que tienen características administrativas, legales, fiscales o contables en los archivos de trámite o concentración.

- **Valoración** Actividad que consiste en el análisis de los valores documentales para establecer criterios de disposición y acciones de transferencia. FONDESIF Manual del Archivo Central.
- **Valoración** Fase del tratamiento archivístico que consiste en analizar y determinar los valores **primarios y secundarios** de las series documentales, fijando los plazos de transferencia, acceso y conservación o eliminación total o parcial.

La Paz, Noviembre 11 de 2016

Señor:

M. Sc. Edgar Clavijo Cárdenas

DIRECTOR CARRERA DE INFORMÁTICA
FAC. CIENCIAS PURAS Y NATURALES
UNIVERSIDAD MAYOR DE SAN ANDRÉS
Presente.

REF: AVAL PARA LA DEFENSA DE PROYECTO DE GRADO

De mi mayor consideración:

Mediante la presente, me dirijo a su Autoridad, en calidad de **Tutora Metodológica** para Informar que luego de haber realizado el seguimiento del Proyecto de Grado titulado: **"Sistema de Información de Gestión y Control Documental para el Instituto Nacional De Laboratorios de Salud "Néstor Morales Villazón – INLASA"** presentado por el Univ. Richard Paucara Machaca con C.I. 5966206 L.P., para optar al título de **LICENCIATURA EN INFORMÁTICA**: **MENCIÓN INGENIERÍA DE SISTEMAS INFORMÁTICOS.**

En este sentido, presento mi **conformidad y aval** respectivo para la defensa pública del Proyecto de Grado de acuerdo a Reglamento vigente en la Universidad Mayor de San Andrés.

Sin otro particular, me suscribo con las atenciones más distinguidas.

Lic. Bhylenia Yhasmyna Rios Miranda

TUTORA METODOLÓGICA

La Paz, Noviembre 11 de 2016

Señora:

Lic. Bhylenia Yhasmyna Rios Miranda

TUTORA METODOLÓGICA

Presente.-

REF: CONFORMIDAD Y AVAL DE PROYECTO DE GRADO

De mi mayor consideración:

Tengo a bien dirigirme a su persona, para darle a conocer que luego de efectuar el seguimiento a la estructura y contenido del Proyecto de Grado titulado: **"Sistema de Información de Gestión y Control Documental para El Instituto Nacional de Laboratorios de Salud "Néstor Morales Villazón – INLASA"**, elaborado por el Universitario: Richard Paucara Machaca con C.I. 5966206 L.P., en calidad de **Asesor** expreso mi conformidad con el contenido y la forma de trabajo, dando **mi Aval,** para que el postulante pueda realizar la defensa del Proyecto de Grado, para optar al título de **LICENCIADO EN INFORMÁTICA MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS**, de acuerdo a normas y reglamento vigentes.

Sin otro particular, me suscribo de su persona con las consideraciones más distinguidas.

M.Sc. Carlos Mullisaca Choque

ASESOR