



UNIVERSIDAD TÉCNICA DEL NORTE

**FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS
CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN SISTEMAS COMPUTACIONALES**

TEMA:

“Sistema Web de Gestión de Procesos para una Junta de Agua Potable utilizando las tecnologías de software libre, JSF.”

APLICATIVO:

“Modulo Facturación y Registro de Información (SISFAC).”

AUTOR: Franklin Andrés Cheza Luna

DIRECTOR: Ing. Mauricio Rea.

IBARRA – ECUADOR

2014



UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA

**AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD
TÉCNICA DEL NORTE**

1 IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital institucional determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual ponemos a disposición la siguiente investigación:

DATOS DE CONTACTO	
CEDULA DE IDENTIDAD	100282019-7
APELLIDOS Y NOMBRES	FRANKLIN ANDRES CHEZA LUNA
DIRECCIÓN	El Arcángel – Mirador de Ibarra
EMAIL	andrexischeza@gmail.com
TELÉFONO FIJO :	TELÉFONO MOVIL: 0989192607

DATOS DE LA OBRA	
TITULO:	“Sistema Web de Gestión de Procesos para una Junta de Agua Potable utilizando las tecnologías de software libre, JSF.”
AUTOR:	FRANKLIN ANDRES CHEZA LUNA
FECHA: AAAAMMDD	2014-01-09
PROGRAMA	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSTGRADO
TITULO POR EL QUE OPTA:	INGENIERO EN SISTEMAS COMPUTACIONALES
DIRECTOR:	ING. MAURICIO REA.

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Franklin Andrés Cheza Luna, con cédula de identidad Nro. 100282019-7, en calidad de autor y titular de los derechos patrimoniales del trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en formato digital y autorizo a la Universidad Técnica del Norte, la publicación del trabajo en el Repositorio Digital Institucional y uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión; en concordancia con la Ley de Educación Superior Artículo 144.



Firma

Nombre: FRANKLIN ANDRES CHEZA LUNA

Cédula: 100282019-7

Ibarra a los 9 días del mes de Enero del 2014



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

**CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE INVESTIGACIÓN A FAVOR DE LA
UNIVERSIDAD TÉCNICA DEL NORTE**

Yo, CHEZA LUNA FRANKLIN ANDRES, con cédula de identidad Nro. 100282019-7, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, artículo 4, 5 y 6, en calidad de autor del trabajo de grado denominado: **“Sistema Web de Gestión y Control de Procesos para una Junta de Agua Potable utilizando las tecnologías de software libre, JSF.”**, que ha sido desarrollado para optar por el título de Ingeniero en Sistemas Computacionales, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En mi condición de autor me reservo los derechos morales de la obra antes mencionada, aclarando que el trabajo aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la biblioteca de la Universidad Técnica del Norte

Firma

Nombre: FRANKLIN ANDRES CHEZA LUNA

Cédula: 100282019-7

Ibarra a los 9 días del mes de Enero del 2014



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICADO DE ASESOR

El señor Franklin Andrés Cheza Luna ha trabajado en su totalidad en el desarrollo del proyecto de tesis "**Sistema Web de Gestión de Procesos para una Junta de Agua Potable utilizando las tecnologías de software libre, JSF.**", previo a la obtención del Título de Ingeniero en Sistemas Computacionales, trabajo que lo realizo con interés profesional y responsabilidad, es todo cuanto puedo certificar en honor a la verdad.

Ing. Mauricio Rea
DIRECTOR DE TRABAJO DE GRADO



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CONSTANCIAS

El autor manifiesta que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asumo la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

En la ciudad de Ibarra, Enero del 2014

EL AUTOR:

FRANKLIN ANDRES CHEZA LUNA
C.I.: 100282019-7



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DECLARACIÓN

Yo, Franklin Andrés Cheza Luna, declaro bajo juramento que el trabajo aquí escrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Universidad Técnica del Norte- Ibarra, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

FRANKLIN ANDRES CHEZA LUNA
C.I.: 100282019-7



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

El presente trabajo va dedicado a Dios y toda mi familia, de una manera muy especial a mi mamá ELVIA por ser el mejor regalo que me ha dado la vida, mi mayor alegría, mis ojos, mi gran virtud, eres la mayor muestra de amor, cariño, comprensión y bondad que Dios ha guardado para mí. Tú lo que más amo, por haberme aguantado todas esas locuras que cometía pero tú siempre me comprendiste y me diste perennemente el amor, apoyo y la fuerza para no dejarme vencer y siempre ser perseverante en los momentos más difíciles que la vida me presentaba.

A mi hermano Fernando que con su forma de pensar y ser me dio una gran lección y ejemplo de vida para lograr corregir mis propios errores y mejorar como ser humano. Cuantos momentos inolvidables que pasamos juntos, siempre los recordare.

A mi abuelito Leonidas que es una persona maravillosa, que con su ejemplo pude descubrir que con esfuerzo y constancia se logra alcanzar los objetivos soñados. Ojala y yo logre llegar a alcanzar una gran parte de lo que tú eres.

A la memoria de mi padre Luis que me hubiese gustado mucho darle el placer de verme hecho ingeniero y donde quiera que esté sabe que en estos últimos tiempos he pensado mucho en él.

A mi prima por esas palabras sinceras que me hicieron ver las cosas que yo por egoísmo no podía ver y saber que si ella pudo lograrlo pues yo también puedo.



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTOS

Un agradecimiento especial a Dios por haberme dado todos estos talentos y capacidades, haberme hecho descubrir todo el potencial que ha puesto en mí. Sin la ayuda de él nada de esto hubiese podido lograr.

Quiero agradecer de una manera muy especial a mi mamá ELVIA por haberme hecho un hombre de bien y guiado para ser un excelente profesional, que ha vivido para mí y daría con los ojos cerrados una vida llena de felicidad por evitarme un instante de sufrimiento. Por apoyarme en este viaje de la vida llamado universidad y estar cada día cuidándome, comprendiéndome y sobre todo aguantándome. Ni toda mi vida entera será suficiente para pagarte todo el esfuerzo y sacrificio que has hecho y sigues asiendo por mí, eres la persona más importante en mi vida te adoro.

A mi hermano que a pesar que a veces, no nos comprendíamos y discrepábamos en varias cosas siempre estuviste ahí apoyándome ya sea moralmente e intelectualmente, por haberme ayudado en los primeros semestres a seguir adelante y no dejarme vencer por las adversidades de la vida.

A mi abuelito que con sus consejos y ejemplo lograron que yo sea un hombre de bien, honesto, respetuoso y siempre alegre.

A mis amigos, amigas, compañeros y compañeras de viaje universitario por haberlos conocido y sobre todo por haber compartido inolvidables momentos en las aulas universitarias. Por los amigos y amigas que tomaron caminos diferentes y que por caprichos de la vida ya no están, cada uno me enseñó algo que me faltaba para formarme como mejor ser humano. Cuantos momentos inolvidables que pasamos juntos dentro y fuera de las aulas universitarias, todas esas locuras que vivimos y sobre todo el apoyo que nos dábamos cada uno cuando sacábamos una mala nota. Gracias por haber sido parte de mi vida y dejarme ser parte de la suya siempre los recordare.

Quiero agradecer a la Universidad Técnica del Norte, nuestra carrera CISIC y de una manera muy especial a los excelentes profesionales que tuve como mentores, quienes me han dado el camino y la guía para lograr ser un buen profesional.

También un agradecimiento a todas esas personas que formaron parte de este viaje y que de una u otra forma contribuyeron con un granito de arena para que pudiera llegar a conseguir el objetivo planteado, fueron muchísimas. A las personas que supieron encontrar más pronto el camino a la plenitud y nos dejaron una enseñanza maravillosa, a los que nos mostraron que estamos hechos de la misma materia que los sueños.

RESUMEN

El Sistema Web de Gestión de Procesos para una Junta de Agua Potable utilizando las tecnologías de software libre es una aplicación que da una solución integral para el manejo de las actividades que desempeña la JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO, de una forma más óptima y con herramientas informáticas adaptadas a los requerimientos de dicha junta de agua.

El presente proyecto de tesis utiliza la arquitectura de desarrollo de software MODELO VISTA CONTROLADOR (MVC), además hace uso de la metodología RUP, como proceso de desarrollo de software.

El desarrollo del presente aplicativo consta de siete capítulos los mismos que se basan en diferentes análisis de los requerimientos indispensables de la JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO, para el cumplimiento de las funciones, optimizando el tiempo de atención y sobre todo llevando un control más eficiente de la información de cada cliente.

En el primer capítulo se muestran las diferentes características y funcionalidades con que cuenta el aplicativo, realizando una representación de las necesidades que tiene la junta de agua y que serán mejoradas, se tiene en cuenta un análisis de los proceso que lleva actualmente la junta de agua para posteriormente buscar una óptima mejora.

En el segundo capítulo se muestra el tratamiento de la estandarización de las normas y políticas que permiten una óptima comprensión de los correspondientes documentos, como son el código de programación de la base de datos y sus respectivos recursos inmersos en su implementación, entre las personas involucradas en el desarrollo.

En el tercer capítulo muestra el análisis de los interesados, la designación de los responsables, las características del producto, además cuenta con el desarrollo del artefacto plan de desarrollo de software, en esta parte se determina los documentos entregables del proyecto, conlleva un análisis de cómo se encuentra organizado el proyecto y su gestión.

El cuarto capítulo, contiene el desarrollo de los casos de uso, se realiza un análisis de cuáles son los casos de uso principales que intervienen en el sistema, continuando con un análisis detallado de cada caso de uso, además cuenta con una descripción de cada caso de uso, su flujo básico y alternativo.

Continuando con el capítulo quinto se tiene un estudio de la arquitectura de software que se utilizará en el desarrollo del sistema, realizando una investigación del patrón de diseño MVC, diseño de capas, para posteriormente implementarlo en el desarrollo del aplicativo, además tenemos los diagramas de secuencia en donde se muestra la interacción entre los diferentes objetos del sistema, al final de este capítulo se realiza un análisis de los riesgos que pueden presentarse y el posible plan de contingencia.

Al iniciar el capítulo sexto se tiene la implementación del sistema el mismo que se desarrolla con la creación de subsistemas con su correspondiente análisis, creamos un mapa de navegación que tendrá nuestro aplicativo simultáneamente con los prototipos de la interfaz de usuario, finalizando con el desarrollo de prueba para realizar los correspondientes ensayos para de esta manera poder probar el correcto funcionamiento del sistema.

Finalmente en el séptimo capítulo tenemos las conclusiones y recomendaciones a las que hemos llegado después de culminar el desarrollo del sistema.

SUMMARY

The Web Management and Process Control for Water Board technology using free software is an application that provides a comprehensive solution for the management of activities plays BOARD ADMINISTRATOR WATER AND SEWER OLIVE VIEWPOINT, of a more optimal and tools adapted to the requirements of the water board form.

This thesis project uses architecture development MODEL VIEW CONTROLLER (MVC) software also makes use of the RUP methodology, such as software development process.

The development of this application consists of seven chapters thereof based on different analyzes of the essential requirements of the BOARD ADMINISTRATOR WATER AND SEWER OLIVE VIEWPOINT, to fulfill the functions, optimizing time and especially taking care more efficient control of customer information.

The first chapter is the different features and functionality available to the application, performing a representation of the needs that the water board and will be improved, taking into account an analysis of the process currently takes the water board and searches for optimal improvement.

In the second chapter the treatment of standardization of rules and policies that allow optimal understanding of the relevant documents shows, such as the programming code of the database and their embedded resources in its implementation, between the people involved in development.

The third chapter shows the stakeholder analysis, the designation of responsibility, product characteristics, also has development device software development plan, in this part of the project deliverables is determined, involves an analysis of how the project and its management is organized.

The fourth chapter contains the development of use cases, an analysis of what the main use cases involved in the system, continuing with a detailed each use case analysis is performed, also has a description of each case of use, basic and alternative flow.

Continuing the fifth chapter is a study of the software architecture to be used in system development, conducting research MVC design pattern, design layers, then implement them in developing the application, we also have diagrams sequence which shows the interaction between different objects in the system at the end of this chapter an analysis of the risks that may arise and possible contingency plan is made.

When you start the sixth chapter is the implementation of the same system that is developed with the creation of subsystems with their analysis, we create a navigation map that will our application simultaneously with the prototypes of the user interface, ending with the development of test to the tests for this way to test the proper operation of the system.

Finally in the seventh chapter we have the conclusions and recommendations that we have reached after completion of system development.

**JUNTA ADMINISTRADORA DE AGUA POTABLE Y
ALCANTARILLADO "MIRADOR EL OLIVO"
Ibarra - Ecuador**



Ibarra, 10 de Enero del 2014

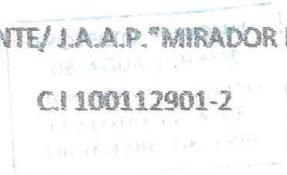
CERTIFICA:

Que: El señor FRANKLIN ANDRES CHEZA LUNA, portador de de la cédula de ciudadanía número 100282019-7, implementó el "Sistema Web de Gestión de Procesos para nuestra Junta de Agua Potable utilizando las Tecnologías se Software Libre JSF", Con el Aplicativo "MODULO FACTURACION Y REGISTRO DE INFORMACION (SISFAC)", y una vez culminado Certifico que se cumplió con las expectativas y requerimientos solicitados por la institución.

Es todo cuanto puedo certificar en honor a la verdad pudiendo el interesado hacer uso del presente documento en lo que a bien crea conveniente.

Ing. Fabián Almeida

PRESIDENTE/ J.A.A.P. "MIRADOR DEL OLIVO"



Dirección: Calle Empedrada Auxiliar a Yahuarcocha y San Miguel Arcángel

Teléfono: 2640-294 - Celular: 0999574683

ÍNDICE GENERAL

IDENTIFICACIÓN DE LA OBRA.....	ii
AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD	¡Error! Marcador no definido.
CERTIFICADO DE ASESOR.....	¡Error! Marcador no definido.
CONSTANCIAS	¡Error! Marcador no definido.
DECLARACIÓN	¡Error! Marcador no definido.
DEDICATORIA	viii
AGRADECIMIENTOS.....	ix
1. CAPÍTULO I.....	1
1.1 INTRODUCCIÓN.....	1
1.2. ALCANCE	2
1.3. Descripción del producto.....	3
1.4. Contexto empresarial	4
1.5. Objetivos del producto.....	4
1.5.1 Objetivo Principal.....	4
1.5.2 Objetivos específicos:.....	4
2.- CAPÍTULO II.- MARCO TEÓRICO	5
2.1 Estándares de Programación.....	5
2.1.1 Introducción	5
2.1.2 Propósito.....	5
2.1.3 Descripción.....	5
2.2 Estandarización de la aplicación o sistema.....	5
2.3 Proceso Unificado de Rational (RUP).....	6
2.3.1 Adaptar el proceso	6
2.3.2 Equilibrar prioridades.....	6
2.3.3 Demostrar valor iterativamente	6
2.3.4 Colaboración entre equipos	6
2.3.5 Elevar el nivel de abstracción	7
2.3.6 Enfocarse en la calidad	7
2.4 Estandarización de la base del lenguaje de programación.....	7
2.4.1 Introducción	7
2.4.2 Organización de ficheros.....	7
2.4.3 Ficheros Fuente Java (.java).....	8
2.4.4 Comentarios de inicio.....	8
2.4.5 Sentencia de paquete.....	9
2.4.6 Sentencias de importación.....	9
2.4.7 Declaraciones de clases e interfaces.....	10
2.4.8 Sangría	10
2.4.9 Longitud de línea	11
2.4.10 División de Líneas	11
2.4.11 Comentarios	11
2.4.12 Comentarios de Implementación	11
2.4.13 Comentarios de Documentación.....	12
2.4.14 Declaraciones	12
2.4.15 Inicialización	12
2.4.16 Localización	12
2.4.17 Declaración de clases / interfaces	12
2.4.18 Sentencias	13
2.4.19 Espacios en Blanco.....	13
2.4.20 Nomenclatura de identificadores.....	13
2.4.21 Paquetes.....	14

2.4.22 Clases e Interfaces	14
2.4.23 Métodos.....	14
2.4.24 Variables	14
2.4.25 Constantes	14
2.4.26 Visibilidad de atributos de instancia y de clase.....	15
2.4.27 Referencias a miembros de una clase.....	15
2.5 Estandarización de la Base de Datos.....	15
2.5.1 Normas a Aplicarse.....	15
2.5.2 Base de Datos, Tablas, Vistas, Procedimientos Almacenados y Triggers.....	16
2.5.2.1 TABLAS.....	16
2.5.2.2 VISTAS.....	16
2.5.2.3 PROCEDIMIENTOS ALMACENADOS O FUNCIONES.....	16
2.5.2.4 TRIGGERS.	16
2.5.2.5 Campos.....	16
3.- CAPÍTULO III.- FASE DE INICIO	17
3.1.- VISIÓN	17
3.1.1 Introducción.....	17
3.1.1.1 Propósito	17
3.1.1.2 Alcance.....	17
3.1.1.3 Definiciones, Acrónimos y abreviaturas	17
3.1.1.4 Referencias	17
3.1.2 Posicionamiento.....	18
3.1.2.1 Oportunidad de Negocios	18
3.1.2.2 Definición del Problema.....	19
3.1.2.3 Sentencias que define la Posición del producto.	20
3.1.3 Descripción de los Interesados y Usuarios	20
3.1.3.1 Población Objetivo	21
3.1.3.2 Resumen de los Interesados	21
3.1.3.3 Resumen de los Usuarios.....	22
3.1.3.4 Entorno del Usuario	22
3.1.3.5 Perfiles de los Usuarios	23
3.1.3.6 Perfiles de Usuario.....	25
3.1.3.7 Necesidades de Interesados y Usuarios.....	26
3.1.3.8 Alternativas y Competencia	27
3.1.4 Vista general del producto	27
3.1.4.1 Resumen del Producto.	27
3.1.4.2 Perspectiva del Producto.....	27
3.1.4.3 Resumen de Características	27
3.1.4.4 Supuestos y Dependencias.....	28
3.1.4.5 Características del Producto.....	28
3.1.4.6 Restricciones.....	29
3.1.4.7 Niveles de Calidad	29
3.2.- PLAN DE DESARROLLO DE SOFTWARE.....	29
3.2.1 Introducción.....	29
3.2.2 Propósito	30
3.2.3 Alcance.....	30
3.2.4 Definiciones, Acrónimos y Abreviaturas	30
3.2.5 Descripción General	30
3.2.6 Propósito del Proyecto, Alcance, y Objetivos.	31
3.2.7 Detalle del Proceso.....	31
3.2.7.1 Facturación.	31
3.2.7.2 Registro de la información.	31
3.2.8 Entregables del proyecto.	32
3.2.9 Organización del Proyecto.....	35

3.2.9.1	Participantes del Proyecto.	35
3.2.9.2	Roles y Responsabilidades.	36
3.2.10	Administración de los Procesos.	37
3.2.10.1	Plan del Proyecto.	37
3.2.10.2	Plan de las Fases.	37
3.2.10.3	Calendario del Proyecto.	40
3.2.10.4	Seguimiento del Proyecto.	45
4.	CAPÍTULO IV.- FASE DE ELABORACIÓN.	46
4.1	ESPECIFICACIÓN DE CASOS DE USO.	46
4.1.1	Introducción.	46
4.1.2	Propósito.	46
4.1.3	Alcance.	46
4.1.4	Definiciones, Acrónimos y Abreviaturas.	46
4.1.5	Referencias.	46
4.1.6	Panorama General.	46
4.1.7	Definición de Roles.	47
4.1.7.1	Administrador.	47
4.1.7.2	Operador.	47
4.1.7.3	Cliente.	47
4.1.8	Diagramas de Casos de Uso.	47
4.1.9	Caso de uso ingreso al sistema.	47
4.1.10	Caso de uso creación de un contrato para un cliente.	48
4.1.11	Caso de uso Registro de las Lecturas de Cada Mes.	50
4.1.12	Caso de uso Registro del Valor del Fondo Rojo.	51
4.1.13	Caso de uso Pago del Consumo de Agua Potable.	52
4.1.14	Caso de Uso Suspensión del Servicio de Agua Potable.	55
4.1.15	Caso de Uso Reconexión del Servicio de Agua Potable.	57
4.1.16	Caso de Uso Convocatoria.	59
4.1.17	Caso de Uso Solicitud de un Préstamo.	61
4.1.18	Caso de Uso Pago de un Préstamo Realizado.	63
4.1.19	Caso de Uso Pago de los Intereses de un Préstamo Realizado.	65
4.1.20	Caso de Uso Pago de Multas.	67
4.2	Especificación de Caso de Uso.	68
4.2.1	Introducción.	68
4.2.2	Especificación de Caso de Uso: Ingresar al Sistema.	70
4.2.2.1	Descripción.	70
4.2.2.2	Actores.	70
4.2.2.3	Precondiciones.	70
4.2.2.4	Flujo de Eventos.	71
4.2.2.5	Pos condiciones.	71
4.2.3	Especificación de Caso de Uso: Validar Usuario.	71
4.2.3.1	Descripción.	71
4.2.3.2	Actores.	71
4.2.3.3	Precondiciones.	71
4.2.3.4	Flujo de Eventos.	71
4.2.3.5	Pos condiciones.	72
4.2.4	Especificación de Caso de Uso: Mostrar Menú Principal del Sistema.	72
4.2.4.1	Descripción.	72
4.2.4.2	Actores.	72
4.2.4.3	Precondiciones.	72
4.2.4.4	Flujo de Eventos.	72
4.2.4.5	Pos condiciones.	72
4.2.5	Especificación de Caso de Uso: Solicitar el Servicio de Agua Potable.	72
4.2.5.1	Descripción.	72

4.2.5.2 Actores.....	73
4.2.5.3 Precondiciones.....	73
4.2.5.4 Flujo de Eventos	73
4.2.5.5 Pos condiciones.....	73
4.2.6 Especificación de Caso de Uso: Crear Contrato.....	73
4.2.6.1 Descripción.....	73
4.2.6.2 Actores.....	73
4.2.6.3 Precondiciones.....	74
4.2.6.4 Flujo de Eventos	74
4.2.6.5 Pos condiciones.....	74
4.2.7 Especificación de Caso de Uso: Registrar Cobro por El Servicio de Agua Potable.....	74
4.2.7.1 Descripción.....	74
4.2.7.2 Actores.....	74
4.2.7.3 Precondiciones.....	74
4.2.7.4 Flujo de Eventos	75
4.2.7.5 Pos condiciones.....	75
4.2.8 Especificación de Caso de Uso: Generar Factura por el Servicio de Agua Potable.....	75
4.2.8.1 Descripción.....	75
4.2.8.2 Actores.....	75
4.2.8.3 Precondiciones.....	76
4.2.8.4 Flujo de Eventos	76
4.2.8.5 Pos condiciones.....	76
4.2.9 Especificación de Caso de Uso: Imprimir Factura por el Servicio de Agua Potable.....	76
4.2.9.1 Descripción.....	76
4.2.9.2 Actores.....	76
4.2.9.3 Precondiciones.....	77
4.2.9.4 Flujo de Eventos	77
4.2.10 Especificación de Caso de Uso: Actualizar Contrato.....	77
4.2.10.1 Descripción.....	77
4.2.10.2 Actores.....	77
4.2.10.3 Precondiciones.....	77
4.2.10.4 Flujo de Eventos	78
4.2.10.5 Pos condiciones.....	78
4.2.11 Especificación de Caso de Uso: Eliminar Contrato.....	78
4.2.11.1 Descripción.....	78
4.2.11.2 Actores.....	78
4.2.11.3 Precondiciones.....	78
4.2.11.4 Flujo de Eventos	78
4.2.11.5 Pos condiciones.....	79
4.2.12 Especificación de Caso de Uso: Generar Reporte de Contratos	79
4.2.12.1 Descripción.....	79
4.2.12.2 Actores.....	79
4.2.12.3 Precondiciones.....	79
4.2.12.4 Flujo de Eventos	79
4.2.12.5 Pos condiciones.....	80
4.2.13 Especificación de Caso de Uso: Registrar Lecturas.....	80
4.2.13.1 Descripción.....	80
4.2.13.2 Actores.....	80
4.2.13.3 Precondiciones.....	80
4.2.13.4 Flujo de Eventos	80
4.2.13.5 Pos condiciones.....	81

4.2.14 Especificación de Caso de Uso: Actualizar Lectura	81
4.2.14.1 Descripción.....	81
4.2.14.2 Actores.....	81
4.2.14.3 Precondiciones.....	81
4.2.14.4 Flujo de Eventos	81
4.2.14.5 Pos condiciones.....	82
4.2.15 Especificación de Caso de Uso: Eliminar Lectura	82
4.2.15.1 Descripción.....	82
4.2.15.2 Actores.....	82
4.2.15.3 Precondiciones.....	82
4.2.15.4 Flujo de Eventos	83
4.2.15.5 Pos condiciones.....	83
4.2.16 Especificación de Caso de Uso: Generar Reporte Lecturas.....	83
4.2.16.1 Descripción.....	83
4.2.16.2 Actores.....	83
4.2.16.3 Precondiciones.....	83
4.2.16.4 Flujo de Eventos	83
4.2.16.5 Pos condiciones.....	84
4.2.17 Especificación de Caso de Uso: Registrar Fondo Rojo	84
4.2.17.1 Descripción.....	84
4.2.17.2 Actores.....	84
4.2.17.3 Precondiciones.....	84
4.2.17.4 Flujo de Eventos	84
4.2.17.5 Pos condiciones.....	85
4.2.18 Especificación de Caso de Uso: Actualizar Fondo Rojo	85
4.2.18.1 Descripción.....	85
4.2.18.2 Actores.....	85
4.2.18.3 Precondiciones.....	85
4.2.18.4 Flujo de Eventos	85
4.2.18.5 Pos condiciones.....	86
4.2.19 Especificación de Caso de Uso: Eliminar Fondo Rojo	86
4.2.19.1 Descripción.....	86
4.2.19.2 Actores.....	86
4.2.19.3 Precondiciones.....	86
4.2.19.4 Flujo de Eventos	86
4.2.19.5 Pos condiciones.....	87
4.2.20 Especificación de Caso de Uso: Generar Reporte Fondo Rojo	87
4.2.20.1 Descripción.....	87
4.2.20.2 Actores.....	87
4.2.20.3 Precondiciones.....	87
4.2.20.4 Flujo de Eventos	87
4.2.20.5 Pos condiciones.....	87
5. CAPÍTULO V.- FASE DE CONSTRUCCIÓN.....	88
5.1 Documento de la arquitectura del software.....	88
5.1.1 Introducción	88
5.1.2 Propósito.....	88
5.1.3 Alcance.....	88
5.1.4 Definiciones, Acrónimos y Abreviaturas.....	88
5.1.5 Referencias.....	88
5.1.6 Resumen	89
5.2 Representación Arquitectónica	89
5.3 Objetivos y Restricciones de Arquitectura	89
5.3.1 Objetivos.....	89
5.3.2 Restricciones.....	89

5.4 Vista dinámica.....	90
5.4.2 Diagramas de Secuencia.....	90
5.5 Vista Lógica.....	92
5.5.1 Introducción.....	92
5.5.2 Descomposición de subsistemas	92
5.5.6 Vista de Procesos	93
5.7 Vista de Despliegue	95
5.7.1 Servidos de Aplicaciones Web.....	96
5.7.1 Computador Interno	96
5.8 Diseño en Capas.....	96
5.8.1 Capa Interfaz de Usuario	97
5.8.2 Capa Lógica del Negocio.....	97
5.8.3 Capa de Persistencia.....	97
5.9 Vista de Datos	97
5.9.1 Modelo Relacional.....	97
5.9.2 Modelo Físico.....	97
5.10 Tamaño y Performance	98
5.10.1 Tiempo de respuesta en el acceso a la Base de Datos.....	98
5.10.2 Tiempo de respuesta de transacciones	98
5.10.3 Espacio en disco para el cliente	98
5.10.4 Espacio en disco para el servidor de base de datos	98
5.11 Calidad	98
5.11.1 Usabilidad.....	98
5.11.2 Eficiencia	99
5.11.3 Seguridad.....	99
5.11.4 Confiabilidad	99
5.11.5 Mantenimiento.....	99
5.12 Arquitectura del Proyecto.....	99
5.12.1 Introducción.....	99
5.12.2 Paquetes de Análisis	99
5.12.3 Vista Dinámica.....	102
5.12.4 Diagramas de Secuencia.....	103
5.12.4.1 Diagrama de Secuencia Caso de Uso: Ingresar al Sistema.....	103
5.12.4.2 Diagrama de Secuencia Caso de Uso: Validar Usuario	104
5.12.4.3 Diagrama de Secuencia Caso de Uso: Mostrar Menú Principal del Sistema.....	105
5.12.4.4 Diagrama de Secuencia Caso de Uso: Solicitar el Servicio de Agua Potable	107
5.12.4.5 Diagrama de Secuencia Caso de Uso: Crear Contrato	108
5.12.4.6 Diagrama de Secuencia Caso de Uso: Registrar Cobro por El Servicio de Agua Potable	109
5.12.4.7 Diagrama de Secuencia Caso de Uso: Generar Factura por el Servicio de Agua Potable	110
5.12.4.8 Diagrama de Secuencia Caso de Uso: Imprimir Factura por el Servicio de Agua Potable	111
5.12.4.9 Diagrama de Secuencia Caso de Uso: Actualizar Contrato	112
5.12.4.10 Diagrama de Secuencia Caso de Uso: Eliminar Contrato	113
5.12.4.11 Diagrama de Secuencia Caso de Uso: Generar Reporte de Contratos.....	114
5.12.4.12 Diagrama de Secuencia Caso de Uso: Registrar Lectura	115
5.12.4.13 Diagrama de Secuencia Caso de Uso: Actualizar Lectura.....	116
5.12.4.14 Diagrama de Secuencia Caso de Uso: Eliminar Lectura.....	117
5.12.4.15 Diagrama de Secuencia Caso de Uso: Generar Reporte Lecturas.....	118
5.12.4.16 Diagrama de Secuencia Caso de Uso: Registrar Fondo Rojo	119
5.12.4.17 Diagrama de Secuencia Caso de Uso: Actualizar Fondo Rojo	120

5.12.4.18 Diagrama de Secuencia Caso de Uso: Eliminar Fondo Rojo	121
5.12.4.19 Diagrama de Secuencia Caso de Uso: Generar Reporte Fondo Rojo	122
5.12.4.20 Diagrama de Secuencia Caso de Uso: Pagar Consumo Agua Potable	123
5.12.5 Vista Lógica	124
5.12.5.1 Descomposición den subsistemas.....	124
5.12.6 Vista de Procesos.....	125
5.12.6.1 Diagrama de Clase: clase Subsistema Usuarios	125
5.12.6.2 Diagrama de Clase: clase Subsistema Contratos.....	126
5.12.6.3 Diagrama de Clase: clase Subsistema Lecturas.....	127
5.12.6.4 Diagrama de Clase: clase Subsistema Fondo Rojo.....	128
5.12.6.5 Diagrama de Clase: clase Subsistema Carta de Pago	129
5.12.6.6 Diagrama de Clase: clase Subsistema Suspensión del Servicio.....	130
5.12.6.7 Diagrama de Clase: clase Subsistema Reconexión del Servicio.....	131
5.12.6.8 Diagrama de Clase: clase Subsistema Convocatorias	132
5.12.7 Vista de Datos	132
5.12.8 Modelo Relacional	132
5.12.9 Modelo Físico	133
5.13 Lista de Riesgos	133
5.13.1 Introducción	133
5.13.1.1 Propósito.....	133
5.13.1.2 Alcance	133
5.13.1.3 Definición, acrónimos y abreviaturas	133
5.13.1.4 Revisión General	134
5.13.2 Riesgos	134
5.13.2.1 Deficiencia en la recolección de información.....	134
5.13.2.2 no llenar las expectativas de la Junta de Agua Potable	134
5.13.2.3 El MVC no se acople a la metodología RUP	135
5.13.2.4 La base de datos desarrollada en PostgreSQL no se acople al modelo del patrón MVC.....	136
5.14 Prototipo de Interfaz de Usuario	136
5.14.1 Introducción	136
5.14.1.1 Propósito.....	136
5.14.1.2 Descripción.....	137
5.14.2 Archivos de Configuración	137
5.14.13 Mapa de navegación	138
5.14.4 Estructura de las Páginas	139
6. CAPÍTULO VI.- FASE DE TRANSICIÓN.	143
6.1 Implementación y Pruebas.....	143
6.1.1 Introducción	143
6.2 Definición de Subsistemas e Implementación.	143
6.3 Modulo Web.....	143
6.3.1 Subsistema Usuarios.....	143
6.3.2 Subsistema Contratos	144
6.3.3 Subsistema Lecturas.....	145
6.3.4 Subsistema Carta de Pago.	146
6.3.5 Subsistema Préstamos.	147
6.3.6 Subsistema Pagos.....	148
6.4 Caso de Prueba.....	149
6.4.1 Introducción	149
6.4.1.1 Propósito.....	149
6.4.1.2 Alcance	150
6.4.2 Casos de Prueba.....	150
6.4.2.1 Caso de Prueba para el Caso de Uso: Crear Contrato	150

6.4.2.2 Caso de Prueba para el Caso de Uso: Generar Factura por el Servicio de Agua Potable	150
6.4.2.3 Caso de Prueba para el Caso de Uso: Registrar Lectura	151
6.4.2.4 Caso de Prueba para el Caso de Uso: Pagar Consumo Agua Potable	151
6.4.2.5 Caso de Prueba para el Caso de Uso: Verificar Lectura	152
6.4.2.6 Caso de Prueba para el Caso de Uso: Generar Carta de Pago	152
6.4.2.7 Caso de Prueba para el Caso de Uso: Generar Citaciones	153
6.4.2.8 Caso de Prueba para el Caso de Uso: Registrar Suspensión.....	153
6.4.2.9 Caso de Prueba para el Caso de Uso: Registrar Convocatoria	154
6.4.2.10 Caso de Prueba para el Caso de Uso: Solicitar Préstamo.....	154
6.4.2.11 Caso de Prueba para el Caso de Uso: Verificar Contrato	154
6.4.2.12 Caso de Prueba para el Caso de Uso: Registrar Préstamo	155
6.4.2.13 Caso de Prueba para el Caso de Uso: Pagar Préstamo	155
6.4.2.14 Caso de Prueba para el Caso de Uso: Verificar Préstamo	155
6.4.2.15 Caso de Prueba para el Caso de Uso: Registrar Pago Préstamo	156
6.4.2.16 Caso de Prueba para el Caso de Uso: Pagar Intereses.....	156
6.4.2.17 Caso de Prueba para el Caso de Uso: Registrar Pago Intereses	156
7. CAPÍTULO VII.- CONCLUSIONES Y RECOMENDACIONES	157
7.1 CONCLUSIONES.	157
7.2 RECOMENDACIONES.....	157
8. GLOSARIO.....	160
9. REFERENCIAS	165
9.1 SITIOS WEB	165
9.2 TESIS	166
9.3 LIBROS	166
10. ANEXOS	168
10.1 Diccionario de datos.	168
10.1.1 Esquema de la base de datos.	168
10.1.2 Tablas.....	168
10.2 Modelo Entidad – Relación.	170
10.3 Modelo Físico.....	171

ÍNDICE DE ILUSTRACIONES

Ilustración 1.- Fases y Flujo de Trabajo.	40
Ilustración 2.- Caso de uso ingreso al sistema.	48
Ilustración 3.- Caso de uso creación de un contrato para un cliente.	49
Ilustración 4.- Caso de uso Registro de las Lecturas de Cada Mes.	50
Ilustración 5.- Caso de uso Registro del Valor del Fondo Rojo.	51
Ilustración 6.- Caso de uso Pago del Consumo de Agua Potable.	52
Ilustración 7.- Caso de Uso Suspensión del Servicio de Agua Potable.	55
Ilustración 8.- Caso de Uso Reconexión del Servicio de Agua Potable.	57
Ilustración 9.- Caso de Uso Convocatoria.	59
Ilustración 10.- Caso de Uso Solicitud de un Préstamo.	61
Ilustración 11.- Caso de Uso Pago de un Préstamo Realizado.	63
Ilustración 12.- Caso de Uso Pago de los Intereses de un Préstamo Realizado.	65
Ilustración 13.- Caso de Uso Pago de Multas.	67
Ilustración 14.- Diagrama de Secuencia de la Arquitectura MVC	91
Ilustración 15.- Estructura MVC.	92
Ilustración 16.- Diagrama Hibernate.	93
Ilustración 17.- Diagrama ICEfaces.	94
Ilustración 18.- Diagrama PRIMFACES	95
Ilustración 19.- Diagrama de Despliegue.	95
Ilustración 20.- Diagrama de Distribución de Capas del Sistema	96
Ilustración 21.- Dependencia entre Paquetes del Sistema Web	102
Ilustración 22.- Diagrama de Secuencia Caso de Uso: Ingresar al Sistema	103
Ilustración 23.- Diagrama de Secuencia Caso de Uso: Validar Usuario	104
Ilustración 24.- Diagrama de Secuencia Caso de Uso: Mostrar Menú Principal del Sistema	105
Ilustración 25.- Diagrama de Secuencia Caso de Uso: Solicitar el Servicio de Agua Potable	107
Ilustración 26.- Diagrama de Secuencia Caso de Uso: Crear Contrato.	108
Ilustración 27.- Diagrama de Secuencia Caso de Uso: Registrar Cobro por El Servicio de Agua Potable	109
Ilustración 28.- Diagrama de Secuencia Caso de Uso: Generar Factura por el Servicio de Agua Potable	110
Ilustración 29.- Diagrama de Secuencia Caso de Uso: Imprimir Factura por el Servicio de Agua Potable	111
Ilustración 30.- Diagrama de Secuencia Caso de Uso: Actualizar Contrato.	112
Ilustración 31.- Diagrama de Secuencia Caso de Uso: Eliminar Contrato.	113
Ilustración 32.- Diagrama de Secuencia Caso de Uso: Generar Reporte de Contratos	114
Ilustración 33.- Diagrama de Secuencia Caso de Uso: Registrar Lectura.	115
Ilustración 34.- Diagrama de Secuencia Caso de Uso: Actualizar Lectura.	116
Ilustración 35.- Diagrama de Secuencia Caso de Uso: Eliminar Lectura	117
Ilustración 36.- Diagrama de Secuencia Caso de Uso: Generar Reporte Lecturas. ..	118
Ilustración 37.- Diagrama de Secuencia Caso de Uso: Registrar Fondo Rojo.	119
Ilustración 38.- Diagrama de Secuencia Caso de Uso: Actualizar Fondo Rojo.	120
Ilustración 39.- Diagrama de Secuencia Caso de Uso: Eliminar Fondo Rojo.	121
Ilustración 40.- Diagrama de Secuencia Caso de Uso: Generar Reporte Fondo	122
Ilustración 41.- Diagrama de Secuencia Caso de Uso: Pagar Consumo Agua.	123

Ilustración 42.- Identificación de subsistemas de diseño a partir de paquetes de análisis.	124
Ilustración 43.- Dependencia entre subsistema de diseño.	125
Ilustración 44.- Diagrama de Clase: clase Subsistema Usuarios.	126
Ilustración 45.- Diagrama de Clase: clase Subsistema Contratos.....	126
Ilustración 46.- Diagrama de Clase: clase Subsistema Lecturas.	127
Ilustración 47.- Diagrama de Clase: clase Subsistema Fondo Rojo.....	128
Ilustración 48.- Diagrama de Clase: clase Subsistema Carta de Pago.	129
Ilustración 49.- Diagrama de Clase: clase Subsistema Suspensión del Servicio.	130
Ilustración 50.- Diagrama de Clase: clase Subsistema Reconexión del Servicio.	131
Ilustración 51.- Diagrama de Clase: clase Subsistema Convocatorias.....	132
Ilustración 54.- Mapa de Navegación.	138
Ilustración 55.- Estructura de la pagina layout.xhtml	139
Ilustración 56.- Interfaz de Usuario. Validar Usuario.....	140
Ilustración 57.- Interfaz de usuario. Pagina moduloA.xhtml.....	141
Ilustración 58.- Interfaz de Usuario. Página moduloB.xhtmlFuente: Andrés Cheza ...	142
Ilustración 59.- Subsistemas. Módulo Diseño	143
Ilustración 60.- Diagrama de componentes, Subsistema Usuarios.....	144
Ilustración 61.- Diagrama de componentes, Subsistema Contratos.....	145
Ilustración 62.- Diagrama de componentes, Subsistema Lecturas	146
Ilustración 63.- Diagrama de componentes, Subsistema Carta de Pago.....	146
Ilustración 64.- Diagrama de componentes, Subsistema Préstamos.....	147
Ilustración 65.- Diagrama de componentes, Subsistema Pagos.....	149

ÍNDICE DE TABLAS

Tabla 1.- Declaraciones de clases e interfaces.....	10
Tabla 2.- Definición del problema.....	20
Tabla 3.- Sentencias que define la Posición del Problema	20
Tabla 4.- Resumen de los interesados.	22
Tabla 5.- Resumen de los Usuarios	22
Tabla 6.- Responsable del proyecto.	23
Tabla 7.- Coordinador del Proyecto.	24
Tabla 8.-Desarrollador del Proyecto.	24
Tabla 9.- Administrador del sistema.....	25
Tabla 10.- Administrador funcional del sistema.....	25
Tabla 11.- Perfil Usuario Externo.	25
Tabla 12.- Necesidades de Interesados y Usuarios.	26
Tabla 13.- Sistemas de Apoyo al Cliente.	27
Tabla 14.- Roles y Responsabilidad.	37
Tabla 15.- Plan de las Fases.....	37
Tabla 16.- Descripción Hitos.	38
Tabla 17.- Objetivos de la Iteración.	38
Tabla 18.- Entregas.	39
Tabla 19.- Calendario del Proyecto Fase Inicio.	41
Tabla 20.- Calendario del Proyecto. Fase de Elaboración.	42
Tabla 21.- Calendario del Proyecto. Fase de Construcción Iteración 1.	43
Tabla 22.- Calendario del Proyecto. Fase de Construcción Iteración 2.	44
Tabla 23.- Calendario del Proyecto. Fase de Construcción Iteración 3.	45
Tabla 24.- Caso de uso ingreso al sistema.....	48
Tabla 25.- Caso de uso creación de un contrato para un cliente.	49
Tabla 26.- Caso de uso Registro de las Lecturas de Cada Mes.....	51
Tabla 27.- Caso de uso Registro del Valor del Fondo Rojo.	52
Tabla 28.- Caso de uso Pago del Consumo de Agua Potable.	54
Tabla 29.- Caso de Uso Suspensión del Servicio de Agua Potable.	56
Tabla 30.- Caso de Uso Re conexión del Servicio de Agua Potable.	58
Tabla 31.- Caso de Uso Convocatoria.	60
Tabla 32.- Caso de Uso Solicitud de un Préstamo.....	62
Tabla 33.- Pago de un Préstamo Realizado.	64
Tabla 34.- Caso de Uso Pago de los Intereses de un Préstamo Realizado.....	66
Tabla 35.- Caso de Uso Pago de Multas.	68
Tabla 36.- Resumen de Casos de Uso.....	70
Tabla 37.- Descripción de los paquetes de análisis.	102
Tabla 380.- Estructura del Prototipo de Interfaz de Usuario. Página moduloA.xhtml.....	141
Tabla 39.- Caso de Prueba para el Caso de Uso: Crear Contrato.....	150
Tabla 40.- Caso de Prueba para el Caso de Uso: Generar Factura por el Servicio de Agua Potable.....	150
Tabla 41.- Caso de Prueba para el Caso de Uso: Registrar Lectura.....	151
Tabla 42.- Caso de Prueba para el Caso de Uso: Pagar Consumo Agua Potable.	151
Tabla 43.- Caso de Prueba para el Caso de Uso: Verificar Lectura.	152
Tabla 44.- Caso de Prueba para el Caso de Uso: Generar Carta de Pago.....	152

Tabla 45.- Caso de Prueba para el Caso de Uso: Generar Citaciones.....153
Tabla 46.- Caso de Prueba para el Caso de Uso: Registrar Suspensión.153
Tabla 47.- Caso de Prueba para el Caso de Uso: Registrar Convocatoria.154
Tabla 48.- Caso de Prueba para el Caso de Uso: Solicitar Préstamo.....154
Tabla 49.- Caso de Prueba para el Caso de Uso: Verificar Contrato.....154
Tabla 50.- Caso de Prueba para el Caso de Uso: Registrar Préstamo.....155
Tabla 51.- Caso de Prueba para el Caso de Uso: Pagar Préstamo.....155
Tabla 52.- Caso de Prueba para el Caso de Uso: Verificar Préstamo.155
Tabla 53.- Caso de Prueba para el Caso de Uso: Registrar Pago Préstamo.....156
Tabla 54.- Caso de Prueba para el Caso de Uso: Pagar Intereses.156
Tabla 55.- Caso de Prueba para el Caso de Uso: Registrar Pago Intereses.....156

1. CAPÍTULO I.

1.1 INTRODUCCIÓN.

La automatización de procesos es una ventaja prioritaria y necesaria dentro de las empresas, tomando en cuenta los grandes beneficios que brinda se ha tornado en un tema de interés por parte de las empresas que no hacen uso de las tecnologías disponibles en la actualidad.

La “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” es una institución autónoma creada el 28 de febrero de 1990 con la colaboración del Banco Ecuatoriano de la Vivienda (BEV), el Instituto de Obras Sanitarias (IEOS) y la Unidad de Apoyo a los Damnificados del Terremoto (UADT). Dicho proyecto maneja y administra el servicio de agua potable para cuatro comunidades del sector rural de la ciudad de Ibarra. Las comunidades están ubicadas en la parte Este de la ciudad y son Añaspamba, Yuracucito, El Arcángel y El Mirador de Ibarra.

En la actualidad la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” registra la información referente al cobro del consumo de agua, las mingas realizadas, sesiones o asambleas organizadas en papel que se anexan a una carpeta de manera manual lo cual dificulta un desempeño ágil y adecuado, además esta propensa a que con el paso del tiempo sufra un deterioro o se pierde y de esta manera no presta la seguridad para la custodia de la información valiosa, asimismo como también es muy tedioso tener que registrar en varios lugares la información correspondiente, estando expuesta a errores en la escritura o incorporación de datos causando así duplicidad, inconsistencia de la misma y confusión al momento de requerir dicha información. Por otra parte vale mencionar que no cuentan con un sistema de contabilidad en el cual se registre los ingresos y gastos que efectúa la junta que son muy necesarios para transparentar.

Este proyecto pretende dar respuesta a la necesidad que tiene la Junta Administradora de Agua Potable y Alcantarillado Mirador del Olivo, de disponer de una herramienta e instrumento adecuado para el servicio a los clientes se optimo y sobre todo que la información de cada uno de los clientes no se pierda.

El aplicativo tiene como fin dar una visión amplia de todo lo que supone la conformación del proceso de facturación y registro de la información, desde una óptica particularmente sistemática que se presenta ya que tiene la disposición de abarcar la gran mayoría de los elementos tecnológicos que actualmente se están utilizando específicamente de software y las actividades que el personal de la institución realiza sobre los servicios a sus clientes.

La metodología manejada está basada en la Orientación del Marco Lógico, que proporciona las herramientas apropiadas para la edificación del aplicativo, indicando la óptima aplicación de los mecanismos que está la conformarán iniciando con el patrón de diseño MVC^[1], el lenguaje de programación JAVA, el framework JSF^[2] para la parte web, la utilización de los Frameworks ICEface y PrimeFace para la parte de interfaz de usuario, el framework Hibernate para la persistencia de datos, el sistema gestor de base de datos PostgreSQL.

Al final del desarrollo de este proyecto, se otorgará un aplicativo que cumpla con las necesidades y las expectativas que la Junta Administradora de Agua Potable y Alcantarillado Mirador del Olivo tiene sobre él, la utilización debe ser para varios sistemas operativos y en diferentes versiones, además de ser implementado con herramientas de software libre, disminuyendo en un gran porcentaje los costes de desarrollo.

1.2. ALCANCE

El sistema beneficiará de forma directa a la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” en especial al personal encargado del proceso de facturación y registro de la información logrando así mejor desempeño del trabajo, como también mejora en gran escala en el servicio al cliente, por medio del registro automático de toda la información pertinente como también la disponibilidad total de la misma alcanzando eficiencia y rapidez, evitando el deterioro y la pérdida de la información. En un futuro se tendrá la posibilidad de implementar el servicio de consultas a través de la web del consumo de agua específicamente para abonados de la junta de agua, información relacionada con el servicio que brinda dicha institución, requisitos para ser cliente del servicio que brinda dicha junta y otra información relacionada con la empresa. Además se tendrá la posibilidad de administrar el sistema, esto se realizará solo por usuarios del sistema con privilegios.

El sistema tendrá la siguiente funcionalidad:

Emisión de Factura/Carta de Pago.

- ❖ Encabezado de documento comercial.
 - Tipo de comprobante.
 - Número del documento.
 - Fecha de emisión.
 - Fecha de pago.
 - Información específica del cliente.

¹ Modelo Vista Controlador (Wikipedia, 2012)

² JavaServer Faces (Wikipedia, 2012)

- Forma de pago.
- ❖ Detalle del cobro.
 - Lecturas actual y anterior.
 - Cantidad de metros cúbicos de agua consumidos.
 - Descripción general del consumo de agua.
 - Precio por metro cubico de agua, en base a las obligaciones del cliente.
- ❖ Totales del documento.
- ❖ Cambios e impresión de los documentos.

Control y seguridad del sistema.

Registro/Actualización/Eliminación

- ❖ Facturas
- ❖ Carta de pago
- ❖ Tarifas.
- ❖ Abonados
- ❖ Empleados
- ❖ Usuarios

Reportes

- ❖ Facturas
- ❖ Carta de pago
- ❖ Cobros
- ❖ Cobros por un período determinado.
- ❖ Cobros por un determinado cliente.
- ❖ Tarifas.
- ❖ Lista de usuarios.
- ❖ Abonados
- ❖ Empleados

1.3. Descripción del producto

El sistema que se desarrollará es un proyecto que se asienta en las necesidades de un control y manejo más conveniente de la información que maneja la institución, lleva por nombre SISTEMA WEB DE GESTIÓN DE PROCESOS PARA UNA JUNTA DE AGUA POTABLE UTILIZANDO LAS TECNOLOGÍAS DE SOFTWARE LIBRE y se ha quedado de acuerdo que será representado por la siglas SISFAC^[3].

³ Sistema de Facturación y Registro de la Información

Trata de enfrentar los problemas que tiene la institución en actualizar y automatizar los procesos internos, específicamente el proceso de facturación y registro de la información, con la finalidad de tener los datos organizados, eficiencia en la emisión de facturas y un control total de los cobros que se realizan dentro de la misma lo que se traduce en brindar un mejor servicio a todos sus clientes.

1.4. Contexto empresarial

La realización de este sistema está orientado para instituciones públicas y autónomas que pretenden manejar de manera más eficiente sus procesos administrativos, abarcando diversos campos como son facturación, registro de información y la contabilidad.

1.5. Objetivos del producto.

1.5.1 Objetivo Principal.

- Desarrollar un sistema informático de facturación y registro de la información.

1.5.2 Objetivos específicos:

- Precisar, delimitar y analizar el proceso de facturación y registro de la información en la junta de agua.
- Plantear y estudiar las posibles soluciones informáticas para la solución del problema.
- Seleccionar e investigar las herramientas informáticas a utilizarse en el desarrollo de la solución informática.
- Realizar el diseño minucioso de la arquitectura y funcionamiento de la solución planteada.
- Desarrollar y probar el sistema hasta lograr los resultados esperados.
- Capacitar a los usuarios finales del sistema de facturación y registro de la información.

Elaboración de la documentación pertinente relacionada con el uso y buen rendimiento del sistema.

2.- CAPÍTULO II.- MARCO TEÓRICO

2.1 Estándares de Programación.

2.1.1 Introducción

Entre los aspectos más importantes al iniciar un proyecto de desarrollo de software es el tratamiento de la estandarización de las normas y políticas que permiten una óptima comprensión de los correspondientes documentos, como son el código de programación de la base de datos y sus respectivos recursos inmersos en su implementación, entre las personas involucradas en el desarrollo.

Al intentar dilucidar bloques de código escrito en diferentes proyectos, las personas encargadas del mantenimiento del sistema llegan a desperdiciar un tiempo elevado y además en ocasiones son propensos a cometer errores por lo cual este documento es una ayuda que facilitar el entendimiento de la codificación realizada en el sistema de facturación y registro de la información.

2.1.2 Propósito.

Este artefacto tiene como propósito dar a conocer a los diferentes interesados los estándares de programación que presidirán el desarrollo y mantenimiento de la aplicación que se va a realizar.

2.1.3 Descripción

El presente documento exhibe al los interesados las reglas y políticas que permitan estandarizar y normar el desarrollo del sistema “Facturación y Registro de la Información”, utilizando el gestor de base de datos PostgreSQL para creación de la base de datos, y el lenguaje de programación JAVA y JSF para la parte web, por ende debemos de tener en cuenta los siguientes aspectos:

- Aplicación o sistema.
- Base de Datos.
- Lenguaje de Programación.

2.2 Estandarización de la aplicación o sistema.

Se llamará aplicación al desarrollo de software funcionalmente independiente que, tenga la facilidad de interconectarse a terceros desarrollos si así lo requiera. La aplicación se desarrollará con el sistema gestor de base de datos (SGDBDD) PostgreSQL 9.0.3 o su superior, el lenguaje de programación será JAVA con el framework JSF 2.1 para lo

referente a la web y para la parte de la vista se utilizará los frameworks ICEfaces y PRIMFACES.

2.3 Proceso Unificado de Rational (RUP)

Es un proceso de desarrollo de software desarrollado por la empresa Rational Software, actualmente propiedad de IBM. Junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos.

El RUP no es un sistema con pasos firmemente establecidos, sino un conjunto de metodologías adaptables al contexto y necesidades de cada organización.

El RUP está basado en 6 principios clave que son los siguientes:

2.3.1 Adaptar el proceso

El proceso deberá adaptarse a las necesidades del cliente ya que es muy importante interactuar con él. Las características propias del proyecto u organización, el tamaño del mismo, así como su tipo o las regulaciones que lo condicionen, influirán en su diseño específico. También se deberá tener en cuenta el alcance del proyecto en un área subformal para hacer un proceso de satisfacción del software.

2.3.2 Equilibrar prioridades

Los requisitos de los diversos participantes pueden ser diferentes, contradictorios o disputarse recursos limitados. Debe encontrarse un equilibrio que satisfaga los deseos de todos. Gracias a este equilibrio se podrán corregir desacuerdos que surjan en el futuro.

2.3.3 Demostrar valor iterativamente

Los proyectos se entregan, aunque sea de un modo interno, en etapas iteradas. En cada iteración se analiza la opinión de los inversores, la estabilidad y calidad del producto, y se refina la dirección del proyecto así como también los riesgos involucrados.

2.3.4 Colaboración entre equipos

El desarrollo de software no lo hace una única persona sino múltiples equipos. Debe haber una comunicación fluida para coordinar requisitos, desarrollo, evaluaciones, planes, resultados, etc.

2.3.5 Elevar el nivel de abstracción

Este principio dominante motiva el uso de conceptos reutilizables tales como patrón del software, lenguajes 4GL o marcos de referencia (frameworks) por nombrar algunos. Esto evita que los ingenieros de software vayan directamente de los requisitos a la codificación de software a la medida del cliente, sin saber con certeza qué codificar para satisfacer de la mejor manera los requisitos y sin comenzar desde un principio pensando en la reutilización del código. Un alto nivel de abstracción también permite discusiones sobre diversos niveles y soluciones arquitectónicas. Éstas se pueden acompañar por las representaciones visuales de la arquitectura, por ejemplo con el lenguaje UML.

2.3.6 Enfocarse en la calidad

El control de calidad no debe realizarse al final de cada iteración, sino en todos los aspectos de la producción. El aseguramiento de la calidad forma parte del proceso de desarrollo y no de un grupo independiente.

2.4 Estandarización de la base del lenguaje de programación.

2.4.1 Introducción

El objeto del presente documento es el establecimiento de los estándares o convenciones de programación empleados en el desarrollo de software sobre la plataforma Java. Este modelo de programación está basado en los estándares recomendados por Sun Microsystems, que han sido difundidos y aceptados ampliamente por toda la comunidad Java, y que han terminado por consolidarse como un modelo estándar de programación de facto.

Estas normas son muy útiles por muchas razones, entre las que destacan:

- Facilitan el mantenimiento de una aplicación. Dicho mantenimiento constituye el 80% del coste del ciclo de vida de la aplicación.
- Permite que cualquier programador entienda y pueda mantener la aplicación. En muy raras ocasiones una misma aplicación es mantenida por su autor original.
- Los estándares de programación mejoran la legibilidad del código, al mismo tiempo que permiten su comprensión rápida.

2.4.2 Organización de ficheros

Las clases en Java se agrupan en paquetes. Estos paquetes se deben organizar de manera jerárquica, de forma que todo código desarrollado para el sistema tendrá que estar incluido dentro del paquete "org.sistema".

Dentro del paquete principal las clases se organizarán en subpaquetes en función del área, organismo o sección del sistema al que pertenezca el código desarrollado. Por ejemplo, para el acceso a los objetos de transferencia POJOS se tendría el paquete "org.sistema.model.dto".

Un fichero consta de secciones que deben estar separadas por líneas en blanco y comentarios opcionales que identifiquen cada sección.

Deben evitarse los ficheros de gran tamaño que contengan más de 1000 líneas. En ocasiones, este tamaño excesivo provoca que la clase no encapsule un comportamiento claramente definido, albergando una gran cantidad de métodos que realizan tareas funcional o conceptualmente heterogéneas.

2.4.3 Ficheros Fuente Java (.java)

Cada fichero fuente Java debe contener una única clase o interfaz pública. El nombre del fichero tiene que coincidir con el nombre de la clase. Cuando existan varias clases privadas asociadas funcionalmente a una clase pública, podrán colocarse en el mismo fichero fuente que la clase pública. La clase pública debe estar situada en primer lugar dentro del fichero fuente. ^[4]

En todo fichero fuente Java distinguimos las siguientes secciones:

- ❖ Comentarios de inicio.
- ❖ Sentencia de paquete.
- ❖ Sentencias de importación.
- ❖ Declaraciones de clases e interfaces.

2.4.4 Comentarios de inicio.

Todo fichero fuente debe comenzar con un comentario que incluya el nombre de la clase, información sobre la versión del código, la fecha y el copyright. El copyright indica la propiedad legal del código, el ámbito de distribución, el uso para el que fue desarrollado y su modificación.

Dentro de estos comentarios iniciales podrían incluirse adicionalmente comentarios sobre los cambios efectuados sobre dicho fichero (mejora, incidencia, error, etc.). Estos comentarios son opcionales si los ficheros están bajo un sistema de control de versiones bien documentado, en caso contrario se recomienda su uso. Estos comentarios constituyen el historial de cambios del fichero.

A continuación se muestra un comentario de inicio para la clase "JceSecurity.java".⁴

⁴ Javafoundations. [Online]. [Diciembre 21, 2012] <http://javafoundations.blogspot.com/2010/07/java-estandares-de-programacion.html>

```

/*
 * @(#)JceSecurity.java 1.50 04/04/14
 *
 * Copyright 2004 Sun Microsystems, Inc. All rights reserved.
 * SUN PROPRIETARY/CONFIDENTIAL. Use is subject to license terms.
 */
/*
 * This class instantiates implementations of JCE engine classes from
 * providers registered with the java.security.Security object.
 *
 * @author Jan Luehe
 * @author Sharon Liu
 * @version 1.50, 04/14/04
 * @since 1.4
 */

```

2.4.5 Sentencia de paquete.

La primera línea no comentada de un fichero fuente debe ser la sentencia de paquete, que indica el paquete al que pertenece(n) la(s) clase(s) incluida(s) en el fichero fuente. Por ejemplo:

```
package javax.crypto;
```

2.4.6 Sentencias de importación.

Tras la declaración del paquete se incluirán las sentencias de importación de los paquetes necesarios. Esta importación de paquetes obligatorios seguirá el siguiente orden:

- Paquetes del JDK de java.
- Paquetes de utilidades no pertenecientes al JDK de Java, de frameworks de desarrollo o de proyectos opensource tales como apache, hibernate, springframework, etc.
- Paquetes desarrollados para del sistema.
- Paquetes de la aplicación.

Se recomienda minimizar en la medida de lo posible el uso de importaciones del tipo "package.*", pues dificultan la comprensión de las dependencias existentes entre las clases utilizadas por la aplicación. En caso contrario, se recomienda utilizar comentarios de línea tras la importación.

```

import java.io.*; // BufferedReader, PrintWriter, FileInputStream, File
import java.util.ArrayList;
import org.apache.log4j.Logger;
import org.apache.lucene.analysis.Analyzer;
import es.provincia.organismo.corporativas.atlas.vo.AgendaVO;
import es.provincia.organismo.atlas.vo.AnuncioVO;
import es.provincia.organismo.atlas.vo.OrganigramaVO;

```

2.4.7 Declaraciones de clases e interfaces.

La siguiente tabla describe los elementos que componen la declaración de una clase o interfaz, así como el orden en el que deben estar situados.

Elementos de declaración de una clase / interfaz	Descripción
Comentario de documentación de la clase/interfaz <code>/** ... */</code>	Permite describir la clase/interfaz desarrollada. Necesario para generar la documentación de la api mediante javadoc
Sentencia <code>class / interface</code>	
Comentario de implementación de la clase/interfaz, si es necesario <code>/* ... */</code>	Este comentario incluye cualquier información que no pueda incluirse en el comentario de documentación de la clase/interfaz.
Variables de clase (estáticas)	En primer lugar las variables de clase públicas (<code>public</code>), después las protegidas (<code>protected</code>), posteriormente las de nivel de paquete (sin modificador), y por último las privadas (<code>private</code>).
Variables de instancia	Primero las públicas (<code>public</code>), después las protegidas (<code>protected</code>), luego las de nivel de paquete (sin modificador), y finalmente las privadas (<code>private</code>).
Constructores	
Métodos	Deben agruparse por funcionalidad en lugar de agruparse por ámbito o accesibilidad. Por ejemplo, un método privado puede estar situado entre dos métodos públicos. El objetivo es desarrollar código fácil de leer y comprender.

Tabla 1.- Declaraciones de clases e interfaces.
Fuente: <http://javafoundations.blogspot.com/2010/07/>

2.4.8 Sangría

Como norma general se establecen 4 caracteres como unidad de sangría. Los entornos de desarrollo integrado (IDE) más populares, tales como Eclipse o NetBeans, incluyen facilidades para formatear código Java.

2.4.9 Longitud de línea

La longitud de línea no debe superar los 80 caracteres por motivos de visualización e impresión.

2.4.10 División de Líneas

Cuando una expresión ocupe más de una línea, esta se podrá romper o dividir en función de los siguientes criterios:

- ✓ Tras una coma.
- ✓ Antes de un operador.
- ✓ Se recomienda las rupturas de nivel superior a las de nivel inferior.
- ✓ Alinear la nueva línea con el inicio de la expresión al mismo nivel que la línea anterior.
- ✓ Si las reglas anteriores generan código poco comprensible, entonces estableceremos tabulaciones de 8 espacios.

Ejemplos:

```
unMetodo(expresionLarga1, expresionLarga 2, expresionLarga 3,  
          expresionLarga 4, expresionLarga 5);
```

```
if ((condicion1 && condicion2)  
    || (condicion3 && condicion4)  
    ||!(condicion5 && condicion6)) {  
    unMetodo();  
}
```

2.4.11 Comentarios

Distinguimos dos tipos de comentarios: los comentarios de implementación y los de documentación.

2.4.12 Comentarios de Implementación

Estos comentarios se utilizan para describir el código ("el cómo"), y en ellos se incluye información relacionada con la implementación, tales como descripción de la función de variables locales, fases lógicas de ejecución de un método, captura de excepciones, etc.

Distinguimos tres tipos de comentarios de implementación:

- Comentarios de bloque: Permiten la descripción de ficheros, clases, bloques, estructuras de datos y algoritmos.

- Comentarios de línea: Son comentarios cortos localizados en una sola línea y tabulados al mismo nivel que el código que describen. Si ocupa más de una línea se utilizará un comentario de bloque. Deben estar precedidos por una línea en blanco.
- Comentario a final de línea: Comentario situado al final de una sentencia de código y en la misma línea.

2.4.13 Comentarios de Documentación

Los comentarios de documentación, también denominados "comentarios javadoc", se utilizan para describir la especificación del código, desde un punto de vista independiente de la implementación, de forma que pueda ser consultada por desarrolladores que probablemente no tengan acceso al código fuente.

2.4.14 Declaraciones

Se recomienda el uso de una declaración por línea, promoviendo así el uso de comentarios.

2.4.15 Inicialización

Toda variable local tendrá que ser inicializada en el momento de su declaración, salvo que su valor inicial dependa de algún valor que tenga que ser calculado previamente.

2.4.16 Localización

Las declaraciones deben situarse al principio de cada bloque principal en el que se utilicen, y nunca en el momento de su uso.

La única excepción a esta regla son los índices de los bucles "for", ya que, en Java, pueden incluirse dentro de la propia sentencia "for".

Se debe evitar el uso de declaraciones que oculten a otras declaraciones de ámbito superior.

2.4.17 Declaración de clases / interfaces

Durante el desarrollo de clases / interfaces se deben seguir las siguientes reglas de formateo:^[5]

- No incluir ningún espacio entre el nombre del método y el paréntesis inicial del listado de parámetros.

⁵ Javafoundations. [Online]. [Diciembre 21, 2012] <http://javafoundations.blogspot.com/2010/07/java-estandares-de-programacion.html>

- El carácter inicio de bloque ("{"") debe aparecer al final de la línea que contiene la sentencia de declaración.
- El carácter fin de bloque ("}") se sitúa en una nueva línea tabulada al mismo nivel que su correspondiente sentencia de inicio de bloque, excepto cuando la sentencia sea nula, en tal caso se situará detrás de "{".
- Los métodos se separarán entre sí mediante una línea en blanco.

2.4.18 Sentencias

Cada línea debe contener como máximo una sentencia. Las sentencias pertenecientes a un bloque de código estarán tabuladas un nivel más a la derecha con respecto a la sentencia que las contiene.

Todas las sentencias de un bloque deben encerrarse entre llaves "{ ... }", aunque el bloque conste de una única sentencia. Esta práctica permite añadir código sin cometer errores accidentalmente al olvidar añadir las llaves.

2.4.19 Espacios en Blanco

Las líneas y espacios en blanco mejoran la legibilidad del código permitiendo identificar las secciones de código relacionadas lógicamente.

Se utilizarán espacios en blanco en los siguientes casos:

- Entre una palabra clave y un paréntesis.
- Esto permite que se distingan las llamadas a métodos de las palabras clave.
- Tras cada coma en un listado de argumentos.
- Para separar un operador binario de sus operandos, excepto en el caso del operador (".").
- Nunca se utilizarán espacios entre los operadores unarios (p.e., "++" o "--") y sus operandos.
- Para separar las expresiones incluidas en la sentencia "for". Al realizar el moldeado o "casting" de clases.

2.4.20 Nomenclatura de identificadores

Las convenciones de nombres de identificadores permiten que los programas sean más fáciles de leer y por tanto más comprensibles. También proporcionan información sobre la función que desempeña el identificador dentro del código, es decir, si es una constante, una variable, una clase o un paquete, entre otros.

2.4.21 Paquetes

Se escribirán siempre en letras minúsculas para evitar que entren en conflicto con los nombres de clases o interfaces. El prefijo del paquete siempre corresponderá a un nombre de dominio de primer nivel, tal como: es, eu, org, com, net, etc.

2.4.22 Clases e Interfaces

Los nombres de clases deben ser sustantivos y deben tener la primera letra en mayúsculas. Si el nombre es compuesto, cada palabra componente deberá comenzar con **mayúsculas**.

Los nombres serán simples y descriptivos. Debe evitarse el uso de acrónimos o abreviaturas, salvo en aquellos casos en los que dicha abreviatura sea más utilizada que la palabra que representa (URL, HTTP, etc.).

Las interfaces se nombrarán siguiendo los mismos criterios que los indicados para las clases. Como norma general toda interfaz se nombrará con el prefijo "I" para diferenciarla de la clase que la implementa (que tendrá el mismo nombre sin el prefijo "I").

2.4.23 Métodos

Los métodos deben ser verbos escritos en minúsculas. Cuando el método esté compuesto por varias palabras cada una de ellas tendrá la primera letra en mayúsculas.

2.4.24 Variables

Las variables se escribirán siempre en minúsculas. Las variables compuestas tendrán la primera letra de cada palabra componente en mayúsculas.

Las variables nunca podrán comenzar con el carácter "_" o "\$". Los nombres de variables deben ser cortos y sus significados tienen que expresar con suficiente claridad la función que desempeñan en el código. Debe evitarse el uso de nombres de variables con un sólo carácter, excepto para variables temporales.^[6]

2.4.25 Constantes

Todos los nombres de constantes tendrán que escribirse en mayúsculas. Cuando los nombres de constantes sean compuestos las palabras se separarán entre sí mediante el carácter de subrayado "_".

⁶ Javafoundations [Online]. [Diciembre 21, 2012] <http://javafoundations.blogspot.com/2010/07/java-estandares-de-programacion.html>

2.4.26 Visibilidad de atributos de instancia y de clase

Los atributos de instancia y de clase serán siempre privados, excepto cuando tengan que ser visibles en subclases heredadas, en tales casos serán declarados como protegidos.

El acceso a los atributos de una clase se realizará por medio de los métodos "get" y "set" correspondientes, incluso cuando el acceso a dichos atributos se realice en los métodos miembros de la clase.

2.4.27 Referencias a miembros de una clase

Evitar el uso de objetos para acceder a los miembros de una clase (atributos y métodos estáticos). Utilizaremos en su lugar el nombre de la clase.

2.5 Estandarización de la Base de Datos.

2.5.1 Normas a Aplicarse.

- a. Los nombres de Bases de Datos, tablas, vistas, procedimientos almacenados, triggers deben de ser escritas en letras minúsculas.
- b. Los nombres de variables de la base de datos deben ir en singular y deben ser representativos, se recomienda no ser demasiado largos. En lo posible se debe evitar utilizar nombres que no tengan nada que ver con lo que se está desarrollando, esto podría causar una confusión en el desarrollo del esquema y sería difícil de recordar.
- c. Se recomienda utilizar abreviaturas, sobre todo si el nombre más característico es sumamente extenso.
- d. Los nombres que identifican las claves primarias deben ir precedidos de id guion bajo y el nombre del campo representativo con letras minúsculas, por ejemplo id_fact.
- e. En lo referente a la creación de índices idx_ seguido de las siglas del nombre de la tabla, luego el nombre del campo.
- f. Para la creación de secuencias seq_ seguido de las siglas del nombre de la tabla, a continuación el nombre del campo.
- g. Estilo de las consultas SQL.

La simbolización de cualquier sentencia (SELECT, UPDATE, DELETE, INSERT O INNER JOIN) debe emplear la indexación para su clarificación, facilitando la búsqueda de las tablas implicadas (FROM) y de las condiciones impuestas (WHERE).

A continuación mostramos un ejemplo de la utilización de SELECT. Si se necesitara recuperar varios campos, podría agruparse varios por línea:

```
SELECT c.cedula, c.nombres, c.dirección FROM clientes c INNER JOIN contratos cc
```

ON c.cedula=cc.cedula;

La mayoría de sentencias llevarán el estilo descrito anteriormente en el ejemplo y otras algo más simples dependiendo de las necesidades.

2.5.2 Base de Datos, Tablas, Vistas, Procedimientos Almacenados y Triggers.

La base de datos que se utilizará en el desarrollo se llamará sisfac.

Las tablas, vistas se utilizarán letras minúsculas para los correspondientes nombres, además no se utilizarán tildes ni caracteres especiales.

2.5.2.1 TABLAS.

Tabla de contratos cabecera contratos_cabecera.

2.5.2.2 VISTAS.

Las vistas se nombrarán con un identificador v_ seguido del nombre de la vista.

2.5.2.3 PROCEDIMIENTOS ALMACENADOS O FUNCIONES.

Los procedimientos almacenados se nombrarán con un identificador pa_ luego seguirá el nombre del procedimiento.

2.5.2.4 TRIGGERS.

Para los triggers se nombrarán anteponiendo las siglas tr_ luego seguirá el nombre del trigger.

2.5.2.5 Campos.

En lo referente a los nombres de los campos pertenecientes a una tabla se nombrarán anteponiendo el identificador id en caso de ser clave primaria seguida con una combinación de nombres siempre y cuando empiecen con letras minúsculas.

Los campos deberán contener su respectivo comentario dentro de la base de datos.

A continuación se muestra un ejemplo de una tabla con sus campos.

Cientes

id_cliente
cedula_cliente
nombres_cliente
apellidos_cliente
direccion_cliente
telefono_cliente
foto_cliente

3.- CAPÍTULO III.- FASE DE INICIO

3.1.- VISIÓN

3.1.1 Introducción

La producción del presente documento se emprendió en el período de inicio del proyecto en la disciplina de requisitos, realizando y detallando las numerosas partes con que cuenta este documento desde su propósito y alcance posteriormente considerando el posicionamiento del sistema, la descripción de los interesados y usuarios para finalmente indicar los requerimientos adecuados para desarrollo del sistema (SISFAC)^[7].

3.1.1.1 Propósito

El propósito del presente documento es coleccionar, analizar y delimitar las necesidades de alto nivel y las particularidades del Sistema Web de Gestión de Procesos (SISFAC). Se concentra en las capacidades requeridas por las partes interesadas y los usuarios finales, y por qué estas necesidades se encuentran presentes. Los pormenores de cómo el Sistema Web de Gestión de Procesos (SISFAC), plasma con estas necesidades se puntualizan en los documentos de caso de uso y las especificaciones complementarias.

3.1.1.2 Alcance

Este documento de visión se aplica al Sistema Web de Gestión de Procesos. Este sistema está diseñado para beneficiar de forma directa a la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” en especial al personal encargado del proceso de facturación y registro de la información logrando así mejor desempeño del trabajo, como también mejora en gran escala en el servicio al cliente, por medio del registro automático de toda la información pertinente como también la disponibilidad total de la misma alcanzando eficiencia y rapidez, evitando el deterioro y la pérdida de la información.

3.1.1.3 Definiciones, Acrónimos y abreviaturas

Ver Glosario de términos.

3.1.1.4 Referencias

- Glosario.
- Plan de desarrollo de Software.
- Metodología RUP (Rational Unified Process)
- Diagrama de casos de uso.

⁷ Sistema de Facturación y Registro de la Información

3.1.2 Posicionamiento

3.1.2.1 Oportunidad de Negocios

Actualmente no existen sistemas de facturación y registro de la información que puedan ser utilizados libremente y con las características requeridas por las instituciones. Al desarrollar una aplicación que va a gestionar diversos procesos como: (1) Facturación, (2) Registro de la información. Estos procesos permitirán tener un mejor manejo de la información de clientes, usuarios, empleados y demás actividades que realiza la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”. En la actualidad la institución registra la información referente al cobro del consumo de agua, las mingas realizadas, sesiones o asambleas organizadas en papel que se anexan a una carpeta de manera manual lo cual dificulta un desempeño ágil y adecuado, además esta propensa a que con el paso del tiempo sufra un deterioro o se pierde y de esta manera no presta la seguridad para la custodia de la información valiosa, asimismo como también es muy tedioso tener que registrar en varios lugares la información correspondiente, estando expuesta a errores en la escritura o incorporación de datos causando así duplicidad, inconsistencia de la misma y confusión al momento de requerir dicha información.

Esta aplicación puede ayudar a esta institución autónoma a tener la información referente a los clientes debidamente ordenada y fácil de manejar de una manera más rápida y eficaz sin necesidad de importunar a los clientes permitiendo saber cuánto adeuda un determinado cliente, en que sectores se creó un nuevo ramal, si de pronto existe la necesidad de algún cambio de materiales, cuales fueron la decisiones que adoptaron en las pasadas asambleas realizadas ya que en la actualidad existe varias maneras de registrar la información tanto en documentos que se anexan a carpetas y otra parte en un sistema manejado en Excel.

Además el sistema en un futuro será accesible a través de internet, para que los usuarios puedan realizar alguna consulta, información acerca de las gestiones que está realizando la institución.

Dado que este sistema se aplicara en su mayoría en formato web, este sistema está desarrollado principalmente para el uso informativo y de gestión institucional. El producto terminado no sería más probable que se venderá con fines de lucro, sin embargo se puede mencionar que tiene la facilidad de ser modificados pocos componentes y que funcione en varias instituciones autónomas, según sus necesidades.

3.1.2.2 Definición del Problema

<p>El problema de</p>	<p>Pérdida y deterioro de la información que maneja la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.</p> <p>No poseer un sistema que automatice los procesos de la institución. Existen procesos que se hacen manualmente. Registros de nuevos clientes, cobros del consumo de agua, información relacionada con las asambleas realizadas, registro de mingas, asambleas.</p> <p>Es difícil hacer un seguimiento de actividades tanto de clientes como empleados de la junta de agua.</p> <p>Desactualización de la información de las actividades de la institución.</p>
<p>Que afecta a</p>	<p>Clientes de la institución.</p> <p>Empleados de la Junta de Agua.</p>
<p>El impacto de ellos es</p>	<p>Pérdida de tiempo en recolección de la información y almacenamiento de la misma.</p> <p>Pérdida de información con el paso del tiempo.</p> <p>Deterioro de la información guardada en documentos.</p> <p>La información guardada en documentos puede estar propensa a errores.</p> <p>Consumo de recursos innecesarios tratando de encontrar información relacionada con un determinado cliente.</p> <p>Pérdida de tiempo en dar un rápido y eficiente servicio a los clientes.</p> <p>Poca información de las decisiones tomadas en pasadas asambleas realizadas.</p>
<p>Una solución exitosa debería</p>	<p>Implementar una solución informática de calidad soportada por una metodología eficaz de desarrollo de software.</p> <p>Obtener información automática de los clientes de la institución.</p> <p>Mostrar información detallada de las asambleas, mingas realizadas.</p> <p>Almacenamiento seguro y ordenado de la información que maneja la institución.</p> <p>Obtención de la información rápida y eficaz para dar un mejor servicio a los clientes.</p> <p>Cada cliente pueda observar las actividades que realiza la institución de una manera fácil y cómoda al abrir una página web.</p>

Tabla 2.- Definición del problema

Fuente: Andrés Cheza.

3.1.2.3 Sentencias que define la Posición del producto.

De	Resolver el problema que tiene con un software adecuado a las necesidades de la "JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO".
Quienes	Desarrollado para el usuario de la institución e información para otros usuarios.
El nombre del producto	Sistema de Facturación y Registro de Información (SISFAC).
Que	Las personas autorizadas tengan acceso sin demasiadas complicaciones. Permita a la "JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO" tener una adecuada organización y actualización de la información tanto de los clientes como también de los empleados.
A diferencia de	Realizar procesos manuales de recolección, organización e integración de la información final, de actividades, cobros de la institución, facturación.
Nuestro producto	Es un sistema automatizado completo a las necesidades principales de la "JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO".

Tabla 3.- Sentencias que define la Posición del Problema

Fuente: Andrés Cheza.

3.1.3 Descripción de los Interesados y Usuarios

Para suministrar de una forma efectiva productos y servicios que integren sus grupos de interés y necesidades de los usuarios reales, es imperioso identificar e implicar a todas las partes interesadas como parte del proceso de modelado de requerimientos. Además

se debe identificar a los usuarios del sistema y asegurar que la comunidad de interesados adecuadamente los representa.

En la presente sección se brinda un perfil de las partes interesadas y los usuarios involucrados en el proyecto, los primordiales problemas que ellos reciben que se abordarán en la solución propuesta. En esta parte no se describe sus solicitudes o requisitos específicos ya que estos son obtenidos en un actor independiente. En su lugar se personifican los antecedentes y la justificación de por qué las exigencias son necesarias.

3.1.3.1 Población Objetivo

En esta parte del proyecto de documento de Visión se considera la posibilidad futura del mercado. ¿Qué camino tiene el mercado y cómo se desarrolla actualmente? A través del uso de estos datos demográficos clave del mercado, nuestra decisión de producto puede ser más ocasionada para ir hacia la toma de decisiones más adecuadas.

Realizar un estimado del tamaño del mercado y dirección de crecimiento para de esta manera predecir lo más real posible el comportamiento del usuario potencial y la cantidad de dinero que el cliente potencial está dispuesto a invertir. Al obtener esa información, podemos construir este sistema de acuerdo a las características que desea el usuario potencial. La tendencia de la tecnología y la situación del mercado actualmente son los factores clave para elegir tal decisión.

El presente sistema va hacer desarrollado como un propósito de tesis, que debe quedar funcional en la institución autónoma, por lo tanto, la notoriedad en el mercado de software no será utilizado con fines comerciales, sino de uso institucional.

3.1.3.2 Resumen de los Interesados

En el presente documento Visión contiene una lista de los interesados y que están directamente involucrados en la definición y alcance del proyecto.

Nombre	Descripción	Responsabilidades
Responsable del Proyecto	Presidente de la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.	Responsable a nivel directivo de los diferentes requerimientos y verifica que cumplan las normas que rigen a la institución.
Coordinador del Proyecto	Director del Proyecto	Realiza una coordinación, análisis del desarrollo de la aplicación.

Tabla 4.- Resumen de los interesados.

Fuente: Andrés Cheza.

3.1.3.3 Resumen de los Usuarios

Los usuarios son todas aquellas personas involucradas directamente en el uso del sistema. A continuación se presenta una lista de los usuarios:

Nombre	Descripción	Responsabilidad
Administrador del sistema	Persona de la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” que administra el control de los sistemas de la JUNTA.	Administrar eficazmente el sistema, gestionar acceso a usuarios, facilitar mantenimiento al sistema frente a nuevos requerimientos.
Administrador funcional del sistema	Persona de la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” que administra el sistema de información.	Administrar funcionalmente el sistema: creación de nuevos usuarios, actualizaciones y modificaciones
Usuario Externo	Abonados de la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.	Información de referente a la junta de agua potable.

Tabla 5.- Resumen de los Usuarios

Fuente: Andrés Cheza.

3.1.3.4 Entorno del Usuario

En la presente sección se puntualiza, la información sobre el medio que este sistema brinda para el usuario objetivo. Este sistema será gratuito para uso y de código abierto. Este sistema se centra en mostrar la información de la institución autónoma, ordenada y de fácil manipulación para la los usuarios de la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.

El proyecto ha sido implementado mediante un patrón de desarrollo MVC ^[8]. Utilizando herramientas de software libre como son Netbeans como IDE de desarrollo, frameworks ICEface, PrimeFace para la vista, hibérnate para la capa de persistencia todo esto corriendo sobre un servidor apache tomcat, el lenguaje de programación que se usó es

⁸ Modelo Vista Controlador (Wikipedia, 2012)

JAVA y JSF ^[9] para la parte web y el sistema gestor de base de datos PostgreSQL donde fue desarrollada la base de datos.

3.1.3.5 Perfiles de los Usuarios

En la presente sección se encuentra la descripción de cada actor del sistema.

3.1.3.5.1 Responsable del proyecto.

Representante	Ing. Mauricio Rea.
Descripción	Director del proyecto de tesis.
Tipo	Director.
Responsabilidades	Mantiene la infraestructura de desarrollo de software. Esto incluye instalación, configuración, copia de seguridad, etc.
Criterio de Éxito	Proyecto de tesis entregado.
Implicación	Revisor de gestión.
Entregables	Informes.
Comentarios	Mantiene una relación constante con el desarrollador del proyecto. Brinda apoyo para guiar al desarrollador cuando sea necesario.

Tabla 6.- Responsable del proyecto.
Fuente: Andrés Cheza.

⁹ JavaServer Faces (Wikipedia, 2012)

3.1.3.5.2 Coordinador del Proyecto.

Representante	Ing. Fabian Almeida
Descripción	Presidente de la "JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO".
Tipo	Gerencial.
Responsabilidades	Planea, gestiona y asigna recursos, forma prioridades, coordina interacciones con clientes, usuarios y mantiene centrado al equipo del proyecto. Establece un conjunto de prácticas que garantiza la integridad y la calidad del servicio que brinda la junta.
Criterio de Éxito	Sistema en Funcionamiento.
Implicación	Gestor de proyectos.
Entregables	Informes.
Comentarios	Coordinar el desarrollo del proyecto. Mantener en funcionamiento el sistema por parte de los usuarios.

Tabla 7.- Coordinador del Proyecto.
Fuente: Andrés Cheza.

3.1.3.5.3 Desarrollador del Proyecto

Representante	Egresado Andrés Cheza
Descripción	N/A
Tipo	Usuario Experto.
Responsabilidades	Diseño de la estructura de almacenamiento de datos persistentes que se utiliza en el sistema. Diseño de la interfaz de usuario. Esto incluye recopilar los requisitos de utilización y los diseños de interfaz de usuario candidata a la creación de prototipos para cumplir estos requisitos. Desarrolla los componentes de software y efectúa las pruebas necesarias para su correcto funcionamiento.
Criterio de Éxito	Sistema en funcionamiento.
Implicación	Diseñador. Desarrollador. Diseñador de base de datos.
Entregables	Proyecto, sistema y documentación.
Comentarios	N/A

Tabla 8.- Desarrollador del Proyecto.
Fuente: Andrés Cheza.

3.1.3.6 Perfiles de Usuario

Esta sección implementa la descripción de cada usuario del sistema.

3.1.3.6.1 Administrador del sistema

Representante	Ing. Fabián Almeida.
Descripción	Presidente de la Junta de Agua.
Tipo	Usuario
Responsabilidades	Coordina y administra la junta de agua.
Criterio de Éxito	Sistema en funcionamiento.
Implicación	Administrador del sistema.
Entregables	Informes.
Comentarios.	N/A.

Tabla 9.- Administrador del sistema.
Fuente: Andrés Cheza.

3.1.3.6.2 Administrador funcional del sistema

Representante	Persona de la Junta de Agua.
Descripción	Secretaria.
Tipo	Usuario.
Responsabilidades	Mantener funcional el sistema, nuevas modificaciones, nuevos reportes, actualizaciones, etc.
Criterio de Éxito	Sistema en funcionamiento.
Implicación	Administrador del sistema.
Entregables	Informes.
Comentarios	N/A.

Tabla 10.- Administrador funcional del sistema.
Fuente: Andrés Cheza.

3.1.3.6.3 Usuario Externo

Representante	Abonado de la Junta de Agua.
Descripción	Usuario final.
Tipo	Usuario casual.
Responsabilidades	Revisar Informes.
Criterio de Éxito	Información correcta.
Implicación	Usuario Final.
Entregables	N/A.
Comentarios	N/A.

Tabla 11.- Perfil Usuario Externo.
Fuente: Andrés Cheza.

3.1.3.7 Necesidades de Interesados y Usuarios.

Necesidad	Prioridad	Intereses	Situación Actual	Soluciones Propuestas
Poseer información organizada y de rápido acceso	ALTA	Poder tener organizada la información para evitar pérdida de tiempo.	Sistema de facturación en Excel y cuaderno.	Desarrollar sistema de almacenamiento de la información para tener un óptimo manejo de los procesos de facturación y registro de la información.
Tener un sistema de facturación acorde a las necesidades de la institución.	ALTA	Poder realizar un mejor proceso de facturación.	Facturación en hojas de Excel.	Implementar un sistema de facturación que brinde un manejo adecuado del cobro del consumo de agua.
Salvaguardar la información que maneja la institución.	ALTA	Evitar la pérdida de información.	Se almacena la información de forma manual en documentos que luego se anexan a una carpeta.	Desarrollar un sistema de almacenamiento de información que permita resguardar la información.
Obtener información detallada de los clientes, asambleas y mingas realizadas, empleados.	ALTA	Poder tener estadísticas de los procesos que se han realizado en la junta de agua.	La información es almacenada en varios lugares y documentos.	Obtener información rápida de los procesos que realiza la junta de agua.
Realización del sistema con herramientas de software libre.	MEDIA	Utilizar políticas gubernamentales sobre la utilización de software libre y verificar el funcionamiento para futuras aplicaciones.	Excel.	Utilizar herramientas como: IDE de desarrollo Netbeans o Eclipse. Lenguaje: JAVA y JSF. Frameworks: IceFace, PrimeFace Hibernate.

Tabla 12.- Necesidades de Interesados y Usuarios.
Fuente: Andrés Cheza.

3.1.3.8 Alternativas y Competencia

Al ser la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” una institución autónoma creada con el fin de dar un servicio a las comunidades rurales Añaspamba, Yuracucito, El Arcangel y el Mirador del Olivo no cuenta con un sistema informático y hace uso de la herramienta Excel para realizar los procesos de facturación y parte del registro de la información sin lograr un manejo adecuado de la información.

3.1.4 Vista general del producto

3.1.4.1 Resumen del Producto.

El sistema de facturación y registro de la información, es una aplicación web desarrollada para automatizar y mejorar el manejo de la información que realiza la junta de agua, tener ordenada la información para de esta manera realizar una mejor atención al cliente o abonado evitando la pérdida de información o duplicación de la misma, optimizando el tiempo de atención del usuario.

3.1.4.2 Perspectiva del Producto.

Esta es una perspectiva de los procesos que se anhela que la aplicación manipulara.

3.1.4.3 Resumen de Características

A continuación se expondrá un listado con los beneficios que logrará el cliente a partir del producto: Resumen de características.

Beneficios para el Cliente	Características que lo Apoyan
Obtener una información detallada de algunos procesos que lleva adelante la junta de agua.	Obtener la información detallada de los servicios que brinda la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.
Disminución del tiempo de espera para ser atendido.	Obtener agilidad y rapidez en los procesos de atención al cliente.
Mejor manejo de la información, evitando pérdidas o duplicidad.	Manejo optimizado de la información, salvaguardando la información y evitando se pueda duplicar.

Tabla 13.- Sistemas de Apoyo al Cliente.
Fuente: Andrés Cheza.

3.1.4.4 Supuestos y Dependencias

En el sistema podemos establecer las siguientes conjeturas:

- Manejo correcto de la información de manera periódica, para presentar informes de acuerdo al trabajo que realiza “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.
- Reconocer las deficiencias tenidas anteriormente a la elaboración del proyecto y de esta manera mejorar el servicio que brinda esta institución a sus clientes.
- Registrar la información concerniente a las asambleas y mingas realizadas por la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.
- Ingreso y actualización de los nuevos clientes.

Como dependencias tenemos las siguientes:

El sistema dependerá del funcionamiento del equipo destinado para alojar la aplicación, se instalará en un servidor de aplicaciones web que está ubicado en la institución.

3.1.4.5 Características del Producto.

En la presente sección se detalla las características con que cuenta el sistema:

- **Desarrollo modular.** Gracias a la implementación con el patrón de arquitectura de software MVC podemos tener un diseño modular en el código ya que se encuentran separadas los distintos procesos según su tipo. ^[10]
- **Flexibilidad.** Al contar con una estructura modular conocemos que la aplicación, estará sujeta a permutas según las necesidades y a los tipos de procesos que debemos cambiar.
- **Fácil gestión y uso.** La presente aplicación cuenta con una manera sencilla para la misión por parte de los usuarios ya que cuenta con un control gráfico de las diferentes opciones con que cuenta el sistema, reportes, vistas ingresos, actualizaciones y más.
- **Herramienta de código abierto.** El sistema será un aplicativo que servirá para ejecutar, distribuir, estudiar, cambiar y optimizar el software, permitiendo que se convierta en una herramienta no solo utilizada en la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”, sino para otras instituciones similares a esta.
- **Multi-plataforma.** En la actualidad debido al uso de varias plataformas tanto de hardware como de software, el presente aplicativo se ha planteado para que

¹⁰ Bucheli, Gabriel. Sistema de Información y Administración de Bienes Informáticos. Ibarra. UTN. 2011

funcione en varias arquitecturas como para procesadores más comunes como son x86, x86-64 y además en diferentes sistemas operativos como GNU/Linux, Windows y Mac OS X.

3.1.4.6 Restricciones

El presente aplicativo posee las siguientes restricciones:

- Aplicable para varios módulos solo para la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” debido a los requerimientos específicos de la institución.
- La facturación no será la función principal ya que también implementará un registro de información manejada por la institución.
- El funcionamiento por el momento se lo realizara internamente, hasta poseer los requerimientos necesarios para que el funcionamiento se lo haga a través de internet.

3.1.4.7 Niveles de Calidad

El desarrollo del sistema facturación y registro de la información se ajusta a la metodología de desarrollo de software RUP observando los parámetros que la metodología define.

3.2.- PLAN DE DESARROLLO DE SOFTWARE

3.2.1 Introducción.

El presente plan de desarrollo del Software es una versión elaborada para ser mostrada como parte del desarrollo del SISTEMA WEB DE GESTION DE PROCESOS PARA UNA JUNTA DE AGUA POTABLE UTILIZANDO LAS TECNOLOGIAS DE SOFTWARE LIBRE, para una institución autónoma como es la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”, como un proyecto previo la obtención del título de Ingeniero en Sistemas Computacionales de la carrera de Ingeniería en Sistemas Computacionales, Facultad de Ingeniería en Ciencias Aplicadas en la Universidad Técnica del Norte.

El proyecto ha sido ofertado por Franklin Andrés Cheza Luna, cimentado en una metodología de Rational Unified Process (RUP) en la que se procederá a cumplir con las cuatro fases que marca la metodología, constando en la primera y segunda fase con una iteración, la tercera con tres iteraciones y la cuarta con dos iteraciones. Es importante destacar esto puesto que utilizamos la terminología RUP en este documento. Se incluirá

el detalle para las fases de Inicio y Elaboración, Construcción y Transacción para dar una visión global de todo proceso.

3.2.2 Propósito

El propósito fundamental del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto. En él se puntualiza la guía de desarrollo del software.

Los usuarios del Plan de Desarrollo del Software son:

- El Responsable, Coordinador y el jefe del proyecto lo utiliza para organizar la agenda y necesidades de recursos, y para realizar su seguimiento.

3.2.3 Alcance

El presente plan de Desarrollo del Software tiene como alcance orientar toda una guía usada para el desarrollo del “SISTEMA WEB DE GESTION DE PROCESOS PARA UNA JUNTA DE AGUA POTABLE UTILIZANDO LAS TECNOLOGIAS DE SOFTWARE LIBRE”. Se puntualiza las iteraciones individuales sea los planes de cada iteración, documentos que se aportan en forma separada. En el proceso de desarrollo del artefacto “Visión” se define las características del producto a desarrollar, lo cual el seguimiento en cada una de las iteraciones ocasionará el ajuste de este documento produciendo nuevas versiones actualizadas.

3.2.4 Definiciones, Acrónimos y Abreviaturas

Ver referencia al artefacto Glosario.

3.2.5 Descripción General

Después de revisar esta introducción, la continuación del documento está organizado en las siguientes secciones:

Descripción general del Proyecto. Muestra una representación del propósito, alcance y objetivos del proyecto, instaurando los artefactos que serán producidos y utilizados durante el desarrollo del proyecto.

Organización del Proyecto. Detalla la Organización del talento Humano que conforma el equipo de desarrollo.

Administración del Proceso. Muestra una explicación de los costos y planificación considerada, define las fases y hitos del proyecto y muestra cómo se realizará su seguimiento.

Planes y Guías de aplicación. Suministra una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas durante el proceso.

3.2.6 Propósito del Proyecto, Alcance, y Objetivos.

En la presente sección se realizara un análisis de los motivos del sistema que se propone desarrollar en que puede ayudar esta aplicación a las labores de los procesos que maneja la junta de agua.

El proyecto se desarrolla específicamente con el propósito de realizar un control y manejo adecuado de la información que lleva la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”, evitar la pérdida y deterioro de la información, optimizando recursos y agilizando las labores administrativas de la institución.

El aplicativo debe ser desarrollado para una plataforma web, que podrá ser visto dentro de la institución autónoma y con el tiempo atreves de internet.

3.2.7 Detalle del Proceso

3.2.7.1 Facturación.

La facturación los analizamos en los siguientes componentes:

- Facturación del consumo de agua potable. Permitirá facturar el consumo mensual de agua potable de un cliente, establecer los metros cúbicos que el cliente ha consumido durante el mes.
- Facturación de multas, cobros. Se podrá llevar un control de los cobros que se realizan por concepto de multas y otros.

3.2.7.2 Registro de la información.

- Registro de la información de las asambleas. Permitirá registrar la información relacionada a las asambleas realizadas por la junta de agua potable, para de esta manera evitar que se pierda o deteriore dicha información.
- Registro de la información de las mingas. Registrará la información de las mingas, así también de las asistencias a la as mismas.
- Manejo de la información de los préstamos realizados. Tiene como objetivo registrar la información de los préstamos realizados a los clientes, sus pagos mensuales de los préstamos.

3.2.8 Entregables del proyecto.

Seguiremos indicando cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que proponemos para el desarrollo de este proyecto.

Debemos recalcar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos, son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, la consecuencia de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y firmeza de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración. ^[11]

- **Caso de negocio.**

Este artefacto muestra la información necesaria desde un punto de vista empresarial para determinar si es viable invertir en este proyecto o no.

- **Plan de Desarrollo de Software.**

El plan de desarrollo de software corresponde al presente documento.

- **Modelo de caso de negocio**

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Solicitantes finales, etc.) permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representa con un diagrama de Caso de Uso usando estereotipos específicos para este modelo.

- **Modelo de Objetos del Negocio.**

Es un modelo que describe la realización de cada caso de uso del negocio, constituyendo los actores internos, la información que en términos generales manipulan y los flujos de trabajo (workflows) asociados al caso de uso del negocio.

- **Glosario.**

Es un documento que define los principales términos usados en el proyecto. Permite establecer una terminología consensuada.

- **Modelo de Casos de Uso.**

- El modelo de Casos de Uso presenta las funciones del sistema y los actores que hacen uso de ellas. Se representa mediante Diagramas de Casos de Uso.

¹¹Bucheli Gabriel. Sistema de Información y Administración de Bienes Informáticos. Ibarra. UTN. 2011.

- **Visión.**
Este documento define la visión del producto desde la perspectiva del cliente, especificando las necesidades y características del producto. Constituye una base de acuerdo en cuanto a los requisitos del sistema.
- **Especificaciones de Casos de Uso.**
Para los casos de uso que lo requieran (cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa) se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, post-condiciones, flujo de eventos, requisitos no-funcionales asociados. Además, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.
- **Especificaciones Adicionales.**
Este documento capturará todos los requisitos que no han sido incluidos como parte de los casos de uso y se refieren a requisitos no-funcionales globales.
- **Prototipos de Interfaces de Usuario.**
Se trata de prototipos que permiten al usuario hacerse una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación de su parte respecto a los requisitos del sistema.
- **Modelo de Análisis y Diseño.**
El presente modelo establece la realización de los casos de uso en clases y pasando desde una representación en términos de análisis (sin incluir aspectos de implementación), de acuerdo al avance del proyecto.
- **Modelo de Datos.**
Previendo que la persistencia de la información del sistema será soportada por una base de datos relacional, este modelo describe la representación lógica de los datos persistentes, de acuerdo con el enfoque para modelo relacional de datos. Para expresar este modelo se utiliza un Diagrama de Clases (donde se utiliza un profile UML para Modelado de Datos, para conseguir la representación de tablas, claves, etc.).
- **Modelo de Implementación.**
Este modelo es una colección de componentes y los subsistemas que los contienen. Estos componentes incluyen: ficheros ejecutables, ficheros de código fuente y, todo otro tipo de ficheros necesarios para la implementación y despliegue del sistema. Este modelo es sólo una visión preliminar al final de la fase de Elaboración, posteriormente tiene bastante refinamiento.
- **Modelo Despliegue.**

En el presente modelo se muestra el despliegue, la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

- **Casos de Prueba.**

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba, y los resultados esperados. Estos casos de prueba son aplicados como pruebas de regresión en cada iteración.

- **Solicitud de Cambio.**

Los cambios propuestos para los artefactos se formalizan mediante este documento. Mediante este documento se hace un seguimiento de los defectos detectados, solicitud de mejoras o cambios en los requisitos del producto. En nuestro caso al final de cada iteración se establecerá base-line.

- **Plan de Iteración.**

Es un conjunto de actividades y tareas ordenadas temporalmente, con recursos asignados, dependencias entre ellas. Se realiza para cada iteración, y para todas las fases.

- **Evaluación de Iteración.**

Este documento incluye la evaluación de los resultados de cada iteración, el grado en el cual se han conseguido los objetivos de la iteración, las lecciones aprendidas y los cambios a ser realizados.

- **Lista de Riesgos.**

- Este documento incluye una lista de riesgos conocidos y vigentes en el proyecto, ordenados en orden decreciente de importancia y con acciones específicas de contingencia o para su mitigación.

- **Manual de Instalación.**

En el presente documento incluye las instrucciones para realizar la instalación del software necesario para poder ejecutar la aplicación.

- **Material de Apoyo para el Usuario Final.**

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Manual del Usuario, Manual de Operación, Manual de Mantenimiento y Sistema de Ayuda.

- **Producto.**

Los ficheros del producto empaquetados y almacenados en un CD con los mecanismos apropiados para facilitar su instalación. El producto, a partir de la

primera iteración de la fase de Construcción es desarrollado incremental e iterativamente, obteniéndose una nueva release al final de cada iteración.

- **Documento de Arquitectura de Software.**

Este producto de trabajo proporciona una visión general arquitectónica completa del sistema, mediante una serie de vistas arquitectónicas diferentes para representar aspectos del sistema.

- **Mapa de Navegación.**

El presente artefacto describe la estructura de los elementos de interfaz de usuario del sistema, junto con las vías de navegación potenciales.

El mapa de navegación sirve de telón de fondo y de enlace entre los guiones gráficos individuales. Los guiones gráficos describen cómo navega el usuario a través de los elementos de la interfaz de usuario para llevar a cabo las funciones del sistema y el mapa de navegación define cuales son las vías de navegación válidas.

- **Plan de Compilación de Integración.**

Este artefacto proporciona un plan detallado para su integración dentro de una iteración.

- **Depósito de Proyecto.**

Este producto de trabajo almacena todas las versiones de archivos de proyecto y directorios. También almacena todos los datos derivados y los metadatos asociados con los archivos y directorios.

3.2.9 Organización del Proyecto.

3.2.9.1 Participantes del Proyecto.

Responsable del proyecto. Debido a que este es un proyecto de tesis el responsable de que el sistema tenga su correcto funcionamiento es el Estudiante Franklin Andrés Cheza Luna.

Responsables de control y seguimiento. Se designa al Director de Tesis Ingeniero Mauricio Rea conjuntamente con el Ingeniero Fabián Almeida Presidente de la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.

Programador. Será el estudiante Franklin Andrés Cheza de sistemas quien desarrollará el proyecto.

Asesor Legal. Una persona especializada en los aspectos legales relacionados con el sistema y su puesta en funcionamiento.

Pruebas e Ingreso de Información. Estará encargada la señora secretaria de la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.

3.2.9.2 Roles y Responsabilidades.

Identifica el rol que tendrá cada puesto y las responsabilidades que se le asignarán.

Puesto	Responsabilidad
Responsable del Proyecto	El responsable del proyecto asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto orientado en los objetivos. Además, el jefe del proyecto se encargará de supervisar el establecimiento de la arquitectura del sistema. Gestión de riesgos, planificación del proyecto.
Responsables del Control y Seguimiento	Serán los encargados de utilizar las revisiones del proyecto de los artefactos de aprobar el proyecto si cumple los objetivos y requisitos, conservándose en desarrollo utilizando la metodología RUP.
Analista de Sistemas	Se encarga de la especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante entrevistas.
Programador	Realiza la construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario. Gestión de requisitos, gestión de configuración y cambios, elaboración del modelo de datos, preparación de las pruebas funcionales, elaboración de la documentación. Elaborar modelos de implementación y despliegue.
Asesor Legal	Será el encargado de especificar las leyes, decretos que determinen el funcionamiento legal del sistema y su aplicación en la institución autónoma.

Pruebas e ingreso de Información	Se realizará la verificación, el funcionamiento final del sistema y son quienes ingresaran la información y si se debe realizar cambios al sistema.
---	---

Tabla 14.- Roles y Responsabilidad.

Fuente: Andrés Cheza.

3.2.10 Administración de los Procesos.

Identificar el Documento de Caso de Negocio o documento de Visión.

3.2.10.1 Plan del Proyecto.

En la presente sección se presenta la organización en fases e iteraciones y el calendario del proyecto.

3.2.10.2 Plan de las Fases.

El desarrollo se llevará a cabo en base a fases con una o más iteraciones en cada una de ellas. En la presente tabla se muestra la distribución de tiempos y el número de iteraciones de cada fase.

Fase	Nro. Iteraciones	Duración
Fase de Inicio	1	4 semanas
Fase de Elaboración	1	8 semanas
Fase de Construcción	3	17 semanas
Fase de Transición	1	4 semanas

Tabla 15.- Plan de las Fases.

Fuente: Andrés Cheza.

Los hitos que definen el final de cada fase se describen en la siguiente tabla:

Descripción	Hito
Fase de Inicio	En esta fase se desarrollará las exigencias del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se hará un refinamiento del Plan de Desarrollo del Proyecto. Realización del Glosario, caso de negocio, identificación inicial de riesgos. La aceptación del cliente-usuario del artefacto Visión y el plan de Desarrollo marcan el final de esta fase.
Fase de Elaboración	En la presente fase se analizan los requisitos y se desarrolla un prototipo de arquitectura. Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán realizados en la primera release de la fase de construcción deben estar analizados y diseñados. La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase. Lista examinada de riesgos y del caso de negocio.

Fase de Construcción	A medida que se vaya desarrollando la fase de construcción se terminan de examinar y diseñar todos los casos de uso, refinando el Modelo de Análisis/Diseño. El producto se edifica en base a 3 iteraciones, cada una produciendo una release a la cual se le aplican las pruebas y se valida con el cliente/usuario. Se comienza la producción de material de apoyo al usuario. El fin de esta fase es la versión de la release 2.0, con toda la capacidad operacional del producto, lista para que los usuarios realicen las pruebas.
Fase de Transición	Esta fase prepara dos releases para distribución, asegurando una implementación y cambio del sistema previo de manera adecuada, incluyendo el entrenamiento de los usuarios. El hito que marca la conclusión de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario.

Tabla 16.- Descripción Hitos.

Fuente: Andrés Cheza.

3.2.10.2.1 Objetivos de la Iteración.

A continuación establecemos los objetivos de cada iteración.

Fase (n* iteraciones)	Objetivos de cada iteración
Inicio (1)	Instaurar los requisitos, casos de uso relativos a todos los procesos del sistema, además de los artefactos iniciales.
Elaboración (1)	Culminar el análisis y diseño de todos los diagramas de flujo, prototipo de interfaz de usuario. Diagramas de asistencia y actividades. Prototipo de la arquitectura que se va a utilizar. Desarrollo de la base de datos. Diagramas de clases.
Construcción (3)	Desarrollo y pruebas de la versión beta.
	Desarrollo y pruebas de la versión R1.0 del sistema. Desarrollo y experimentación de la versión 2.0 full del sistema. Manuales de usuario.
Transición (1)	Instalación de versión 2.0 y pruebas del sistema. Carga de Base de Datos.

Tabla 17.- Objetivos de la Iteración.

Fuente: André Cheza.

3.2.10.2.2 Entregas.

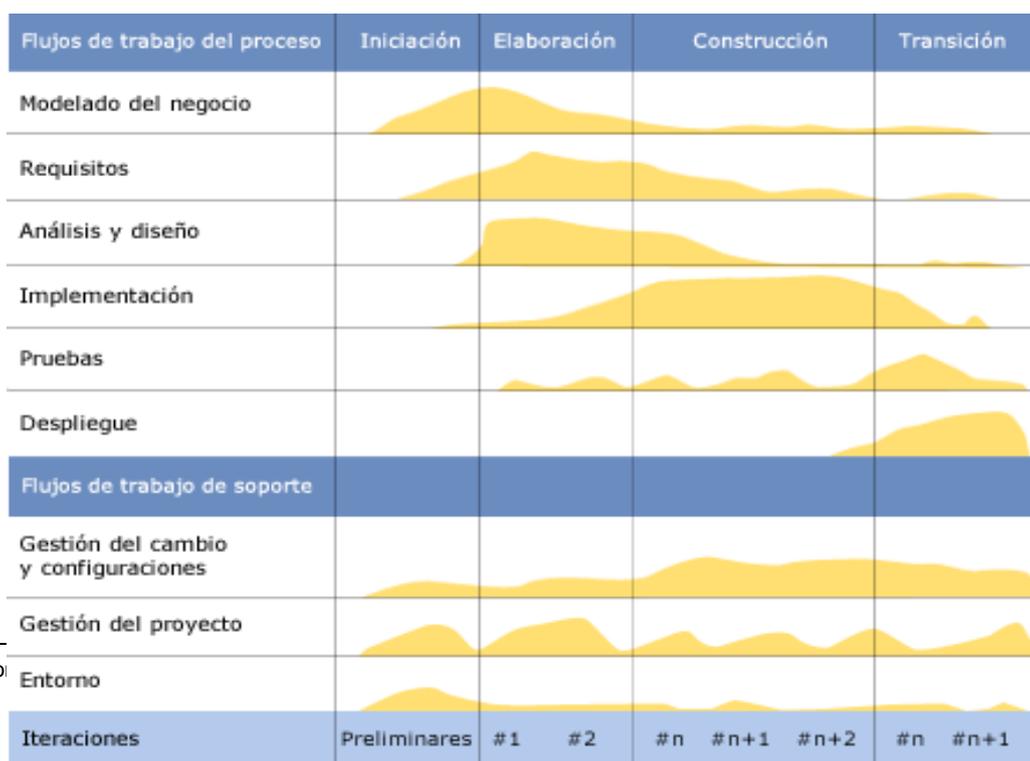
Se entregarán en la fase de construcción las siguientes versiones ejecutables del sistema:

Versión Beta	Será verificado en la primera interacción que será utilizado para verificar la obtención automática de la información.
Versión 1.0	Se ejecutará en la segunda interacción de la fase de construcción se comprobara si existen errores en la versión beta para corregirlos y mejorar el sistema.
Versión 2.0 Full	Esta versión se entregará en la tercera interacción de la fase de construcción integrará todos los módulos que conforman el sistema.

Tabla 18.- Entregas.
Fuente: Andrés Cheza.

3.2.10.2.3 Cronograma del Proyecto

En la presente sección se presenta un calendario de las primordiales tareas del proyecto incluyendo solo las fases de Inicio y Elaboración. Como se ha mencionado, el proceso iterativo e incremental de RUP ^[12] está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la totalidad de los artefactos son generados muy tempranamente en el proyecto pero van desarrollándose en mayor o menor grado de acuerdo a la fase de iteración del proyecto. La siguiente figura enseña este enfoque, en ella lo ensombrecido marca el énfasis de cada disciplina en un instante determinado del desarrollo.



¹² Ratio

Ilustración 1.- Fases y Flujo de Trabajo.

Fuente: JACOBSON, Ivar BOOCH, James. El proceso unificado de desarrollo de software. Adison Wesley: Madrid 2000.

3.2.10.3 Calendario del Proyecto.

Para el presente proyecto se ha determinado en el siguiente calendario. La fecha de aprobación indica cuando el artefacto en cuestión tiene un estado de completitud suficiente para someterse a revisión y aprobación, pero esto no evita la posibilidad de su posterior refinamiento y cambios.

Disciplina / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
REQUISITOS		
Glosario	Semana 1: 02/04/2012	Siguiente fase.
Documento de visión	Semana 1: 02/04/2012	Semana 4: 23/04/2012
Modelos de casos e uso (a nivel de requerimientos).	Semana 4: 16/04/2012	Siguiente fase.
Modelo de casos de negocio (diagramas de contexto)	Semana 4: 23/04/2012	Siguiente fase.
ANALISIS		
Modelo de Objetos de Negocio.	Semana 4: 23/04/2012	Siguiente fase.
Especificaciones de casos de uso.	Semana 6: 07/05/2012	Siguiente fase.
Especificaciones suplementarias.	Semana 8: 21/05/2012	Siguiente fase.
Modelo de análisis y diseño (modelo entidad-relación).	Semana 9: 28/05/2012	Siguiente fase.
Documento de arquitectura de software.	Semana 11: 04/06/2012	Siguiente fase.
DISEÑO		
Mapa de navegación.	Semana 10: 21/06/2012	Siguiente fase.
Prototipos de interfaces de usuario (plantillas).	Semana 12: 11/06/2012	Siguiente fase.
Modelo de diseño de clase.	Semana 8: 21/05/2012	Siguiente fase.
Modelo de datos (modelo relacional).	Semana 13: 18/06/2012	Siguiente fase.
IMPLEMENTACION		
Modelo de implementación.	Semana14: 25/06/2012	Siguiente fase.
Modelo de despliegue.	Semana 15: 02/07/2012	Siguiente fase.
Planificar la integración del sistema.	Semana 16: 02/07/2012	Siguiente fase.
Manual de instalación.	Semana 17: 16/07/2012	Siguiente fase.
Manual de usuario.	Semana 18: 23/07/2012	Siguiente fase.
PRUEBAS		
Casos de prueba.	Semana 20: 06/08/2012	Siguiente fase.
Solicitud de cambios.	Semana 21: 13/08/2012	Siguiente fase.

GESTION DE CAMBIOS Y CONFIGURACION		
Depósito de proyecto.	Semana 22: 20/08/2012	Semana 6: 07/05/2012
Solicitud de cambios.	Semana 22: 20/08/2012	Siguiente fase.
GESTION DE PROYECTOS		
Lista de riesgos.	Semana 23: 27/08/2012	Aprobado.
Desarrollar el caso de negocio.		Aprobado.
Plan de desarrollo de software.	Semana 23: 27/08/2012	Semana 4: 23/08/2012
Plan de iteración.	Semana 23: 27/08/2012	Aprobado.
Evaluación de iteración.	Semana 24: 03/09/2012	Siguiente fase.
Guías de programación.	Semana 24: 03/09/2012	Siguiente fase.
ENTORNO		
Personalizar el proceso de desarrollo.	Semana 25: 10/09/2012	Siguiente fase.
Guía de artefactos.	Semana 25: 10/09/2012	Siguiente fase.

Tabla 19.- Calendario del Proyecto Fase Inicio.

Fuente: Andrés Cheza.

Disciplina / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
REQUISITOS		
Glosario	Semana 1: 02/04/2012	Semana 8: 21/05/2012
Documento de visión	Semana 1: 02/04/2012	Aprobado.
Modelos de casos e uso (a nivel de requerimientos).	Semana 4: 16/04/2012	Semana 9: 28/05/2012
Modelo de casos de negocio (diagramas de contexto)	Semana 4: 23/04/2012	Semana 9: 28/05/2012
ANALISIS		
Modelo de Objetos de Negocio.	Semana 4: 23/04/2012	Siguiente fase.
Especificaciones de casos de uso.	Semana 6: 07/05/2012	Siguiente fase.
Especificaciones suplementarias.	Semana 8: 21/05/2012	Siguiente fase.
Modelo de análisis y diseño (modelo entidad-relación).	Semana 9: 28/05/2012	Semana 15: 09/07/2012
Documento de arquitectura de software.	Semana 11: 04/06/2012	Semana 15: 09/07/2012
DISEÑO		
Mapa de navegación.	Semana 10: 21/06/2012	Semana 16: 16/07/2012
Prototipos de interfaces de usuario (plantillas).	Semana 12: 11/06/2012	Semana 16: 16/07/2012
Modelo de diseño de clase.	Semana 8: 21/05/2012	Semana 16: 16/07/2012
Modelo de datos (modelo relacional).	Semana 13: 18/06/2012	Semana 16: 16/07/2012
IMPLEMENTACION		
Modelo de implementación.	Semana 14: 25/06/2012	Siguiente fase.
Modelo de despliegue.	Semana 15: 02/07/2012	Siguiente fase.
Planificar la integración del sistema.	Semana 16: 02/07/2012	Siguiente fase.
Manual de instalación.	Semana 17: 16/07/2012	Siguiente fase.
Manual de usuario.	Semana 18: 23/07/2012	Siguiente fase.
PRUEBAS		
Casos de prueba.	Semana 20: 06/08/2012	Siguiente fase.
Solicitud de cambios.	Semana 21: 13/08/2012	Siguiente fase.

GESTION DE CAMBIOS Y CONFIGURACION		
Depósito de proyecto.	Semana 22: 20/08/2012	Semana 23: 27/08/2012
Solicitud de cambios.	Semana 22: 20/08/2012	Siguiente fase.
GESTION DE PROYECTOS		
Lista de riesgos.	Semana 23: 27/08/2012	Aprobado.
Desarrollar el caso de negocio.		Aprobado.
Plan de desarrollo de software.	Semana 23: 27/08/2012	Semana 24: 03/09/2012
Plan de iteración.	Semana 23: 27/08/2012	Aprobado.
Evaluación de iteración.	Semana 24: 03/09/2012	Siguiente fase.
Guías de programación.	Semana 24: 03/09/2012	Semana 25: 10/09/2012
ENTORNO		
Personalizar el proceso de desarrollo.	Semana 25: 10/09/2012	Siguiente fase.
Guía de artefactos.	Semana 25: 10/09/2012	Siguiente fase.

Tabla 20.- Calendario del Proyecto. Fase de Elaboración.

Fuente: Andrés Cheza.

Disciplina / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
REQUISITOS		
Glosario	Semana 1: 02/04/2012	Aprobado.
Documento de visión	Semana 1: 02/04/2012	Aprobado.
Modelos de casos e uso (a nivel de requerimientos).	Semana 4: 16/04/2012	Aprobado.
Modelo de casos de negocio (diagramas de contexto)	Semana 4: 23/04/2012	Aprobado.
ANALISIS		
Modelo de Objetos de Negocio.	Semana 4: 23/04/2012	Semana 14: 02/07/2012
Especificaciones de casos de uso.	Semana 6: 07/05/2012	Semana 14: 02/07/2012
Especificaciones suplementarias.	Semana 8: 21/05/2012	Semana 15: 09/07/2012
Modelo de análisis y diseño (modelo entidad-relación).	Semana 9: 28/05/2012	Aprobado.
Documento de arquitectura de software.	Semana 11: 04/06/2012	Aprobado.
DISEÑO		
Mapa de navegación.	Semana 10: 21/06/2012	Aprobado.
Prototipos de interfaces de usuario (plantillas).	Semana 12: 11/06/2012	Aprobado.
Modelo de diseño de clase.	Semana 8: 21/05/2012	Aprobado.
Modelo de datos (modelo relacional).	Semana 13: 18/06/2012	Aprobado.
IMPLEMENTACION		
Modelo de implementación.	Semana 14: 25/06/2012	Semana 20: 13/08/2012
Modelo de despliegue.	Semana 15: 02/07/2012	Semana 20: 13/08/2012
Planificar la integración del sistema.	Semana 16: 02/07/2012	Semana 20: 13/08/2012
Manual de instalación.	Semana 17: 16/07/2012	Siguiente fase.
Manual de usuario.	Semana 18: 23/07/2012	Siguiente fase.
PRUEBAS		
Casos de prueba.	Semana 20: 06/08/2012	Siguiente fase.
Solicitud de cambios.	Semana 21: 13/08/2012	Siguiente fase.
GESTION DE CAMBIOS Y		

CONFIGURACION		
Depósito de proyecto.	Semana 22: 20/08/2012	Aprobado.
Solicitud de cambios.	Semana 22: 20/08/2012	Siguiente fase.
GESTION DE PROYECTOS		
Lista de riesgos.	Semana 23: 27/08/2012	Aprobado.
Desarrollar el caso de negocio.		Aprobado.
Plan de desarrollo de software.	Semana 23: 27/08/2012	Aprobado.
Plan de iteración.	Semana 23: 27/08/2012	Aprobado.
Evaluación de iteración.	Semana 24: 03/09/2012	Siguiente fase.
Guías de programación.	Semana 24: 03/09/2012	Aprobado.
ENTORNO		
Personalizar el proceso de desarrollo.	Semana 25: 10/09/2012	Siguiente fase.
Guía de artefactos.	Semana 25: 10/09/2012	Siguiente fase.

Tabla 21.- Calendario del Proyecto. Fase de Construcción Iteración 1.

Fuente: Andrés Cheza.

Disciplina / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
REQUISITOS		
Glosario	Semana 1: 02/04/2012	Aprobado.
Documento de visión	Semana 1: 02/04/2012	Aprobado.
Modelos de casos e uso (a nivel de requerimientos).	Semana 4: 16/04/2012	Aprobado.
Modelo de casos de negocio (diagramas de contexto)	Semana 4: 23/04/2012	Aprobado.
ANALISIS		
Modelo de Objetos de Negocio.	Semana 4: 23/04/2012	Aprobado.
Especificaciones de casos de uso.	Semana 6: 07/05/2012	Aprobado.
Especificaciones suplementarias.	Semana 8: 21/05/2012	Aprobado.
Modelo de análisis y diseño (modelo entidad-relación).	Semana 9: 28/05/2012	Aprobado.
Documento de arquitectura de software.	Semana 11: 04/06/2012	Aprobado.
DISEÑO		
Mapa de navegación.	Semana 10: 21/06/2012	Aprobado.
Prototipos de interfaces de usuario (plantillas).	Semana 12: 11/06/2012	Aprobado.
Modelo de diseño de clase.	Semana 8: 21/05/2012	Aprobado.
Modelo de datos (modelo relacional).	Semana 13: 18/06/2012	Aprobado.
IMPLEMENTACION		
Modelo de implementación.	Semana 14: 25/06/2012	Aprobado.
Modelo de despliegue.	Semana 15: 02/07/2012	Aprobado.
Planificar la integración del sistema.	Semana 16: 02/07/2012	Semana 20: 13/08/2012
Manual de instalación.	Semana 17: 16/07/2012	Siguiente fase.
Manual de usuario.	Semana 18: 23/07/2012	Siguiente fase.
PRUEBAS		
Casos de prueba.	Semana 20: 06/08/2012	Siguiente fase.
Solicitud de cambios.	Semana 21: 13/08/2012	Siguiente fase.
GESTION DE CAMBIOS Y CONFIGURACION		

Depósito de proyecto.	Semana 22: 20/08/2012	Aprobado.
Solicitud de cambios.	Semana 22: 20/08/2012	Siguiente fase.
GESTION DE PROYECTOS		
Lista de riesgos.	Semana 23: 27/08/2012	Aprobado.
Desarrollar el caso de negocio.		Aprobado.
Plan de desarrollo de software.	Semana 23: 27/08/2012	Aprobado.
Plan de iteración.	Semana 23: 27/08/2012	Aprobado.
Evaluación de iteración.	Semana 24: 03/09/2012	Siguiente fase.
Guías de programación.	Semana 24: 03/09/2012	Aprobado.
ENTORNO		
Personalizar el proceso de desarrollo.	Semana 25: 10/09/2012	Siguiente fase.
Guía de artefactos.	Semana 25: 10/09/2012	Siguiente fase.

Tabla 22.- Calendario del Proyecto. Fase de Construcción Iteración 2.

Fuente: Andrés Cheza.

Disciplina / Artefactos generados o modificados durante la Fase de Inicio	Comienzo	Aprobación
REQUISITOS		
Glosario	Semana 1: 02/04/2012	Aprobado.
Documento de visión	Semana 1: 02/04/2012	Aprobado.
Modelos de casos e uso (a nivel de requerimientos).	Semana 4: 16/04/2012	Aprobado.
Modelo de casos de negocio (diagramas de contexto)	Semana 4: 23/04/2012	Aprobado.
ANALISIS		
Modelo de Objetos de Negocio.	Semana 4: 23/04/2012	Aprobado.
Especificaciones de casos de uso.	Semana 6: 07/05/2012	Aprobado.
Especificaciones suplementarias.	Semana 8: 21/05/2012	Aprobado.
Modelo de análisis y diseño (modelo entidad-relación).	Semana 9: 28/05/2012	Aprobado.
Documento de arquitectura de software.	Semana 11: 04/06/2012	Aprobado.
DISEÑO		
Mapa de navegación.	Semana 10: 21/06/2012	Aprobado.
Prototipos de interfaces de usuario (plantillas).	Semana 12: 11/06/2012	Aprobado.
Modelo de diseño de clase.	Semana 8: 21/05/2012	Aprobado.
Modelo de datos (modelo relacional).	Semana 13: 18/06/2012	Aprobado.
IMPLEMENTACION		
Modelo de implementación.	Semana14: 25/06/2012	Aprobado.
Modelo de despliegue.	Semana 15: 02/07/2012	Aprobado.
Planificar la integración del sistema.	Semana 16: 02/07/2012	Aprobado.
Manual de instalación.	Semana 17: 16/07/2012	Siguiente fase.
Manual de usuario.	Semana 18: 23/07/2012	Siguiente fase.
PRUEBAS		
Casos de prueba.	Semana 20: 06/08/2012	Siguiente fase.
Solicitud de cambios.	Semana 21: 13/08/2012	Semana 25: 10/09/2012
GESTION DE CAMBIOS Y CONFIGURACION		
Depósito de proyecto.	Semana 22: 20/08/2012	Aprobado.
Solicitud de cambios.	Semana 22: 20/08/2012	Semana 25: 10/09/2012

GESTION DE PROYECTOS		
Lista de riesgos.	Semana 23: 27/08/2012	Aprobado.
Desarrollar el caso de negocio.		Aprobado.
Plan de desarrollo de software.	Semana 23: 27/08/2012	Aprobado.
Plan de iteración.	Semana 23: 27/08/2012	Aprobado.
Evaluación de iteración.	Semana 24: 03/09/2012	Siguiente fase.
Guías de programación.	Semana 24: 03/09/2012	Aprobado.
ENTORNO		
Personalizar el proceso de desarrollo.	Semana 25: 10/09/2012	Siguiente fase.
Guía de artefactos.	Semana 25: 10/09/2012	Aprobado.

Tabla 23.- Calendario del Proyecto. Fase de Construcción Iteración 3.
Fuente: Andrés Cheza.

3.2.10.4 Seguimiento del Proyecto.

3.2.10.4.1 Control de Plazos.

El calendario del proyecto tendrá un seguimiento y valoración mensual por el jefe de proyecto y por el director del proyecto.

3.2.10.4.2 Control de Calidad.

Los defectos detectados en las revisiones y formalizados también en una solicitud de permuta tendrán un seguimiento para asegurar la conformidad respecto de la solicitud de dichas deficiencias. Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las guías de revisión y listas de verificación incluidas en la metodología RUP.

3.2.10.4.3 Gestión de Riesgos.

A partir de la fase de Inicio se mantendrá una lista de riesgos asociados al proyecto y de las acciones establecidas como estrategia para aminorar o acciones de contingencia. Esta lista será evaluada al menos una vez en cada iteración.

3.2.10.4.4 Gestión de Configuración.

Se realizará un trámite de configuración para llevar un registro de los artefactos generados y sus versiones. Además se incluirá la gestión de las solicitudes de cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles a todos los participantes en el proyecto. Al culminar cada iteración se establecerá un registro del estado de cada artefacto, estableciendo una versión, la cual podrá ser modificada sólo por una solicitud de cambio aprobada.

4. CAPÍTULO IV.- FASE DE ELABORACIÓN

4.1 ESPECIFICACIÓN DE CASOS DE USO.

4.1.1 Introducción.

El presente artefacto es un guía de las funciones esperadas para el sistema y su entorno. En los diagramas de caso de uso consta de casos de uso y actores que están relacionados con el sistema. Muestra a detalle las labores que el sistema va a desempeñar como una herramienta de soporte para la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.

4.1.2 Propósito.

Mostrarnos paso a paso el modo en que el sistema interactúa con los actores y lo que el sistema realiza en el caso de uso.

- El propósito del presente documento es mostrar el comportamiento del sistema necesario para lograr uno o más objetivos esperados.

4.1.3 Alcance.

En un modelo de caso de uso tiene varias características que describirán ya sea en forma textual y grafica la funcionalidad del caso de uso de las actividades principales que desarrollara el sistema.

4.1.4 Definiciones, Acrónimos y Abreviaturas.

Ver artefacto Glosario.

4.1.5 Referencias.

- Documento Visión.
- Documento plan de desarrollo.

4.1.6 Panorama General.

A continuación se realiza un completo detalle de los roles que tendrá el sistema y una descripción de cada uno de los mismos, para luego pasar a mostrar cada uno de los casos de uso iniciamos con el nombre de CASO DE USO, una breve descripción mostrando qué función desempeña.

4.1.7 Definición de Roles

En esta sección se define los principales roles que tienen los actores que interactúan de alguna forma con el sistema. En el posterior desarrollo de este documento, se indicará como se realiza esta interacción y como el sistema deberá manejarlas.

4.1.7.1 Administrador

Es la persona que se encarga del mantenimiento de los datos en la aplicación. Será la persona encargada de las operaciones CRUD de usuarios, empleados, contratos y las diferentes constantes del sistema. Tendrá contacto directo con la aplicación.

4.1.7.2 Operador

Es la persona encargada de la atención de los clientes de “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”. Será la persona encargada del cobro de consumo de agua así también de emitir facturas por cobros de multas, pago de préstamos, pago de intereses. Tendrá contacto directo con la aplicación.

4.1.7.3 Cliente

Es la persona encargada de realizar solicitudes de servicios que ofrece la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO” a las personas que interactúan directamente con el sistema para estos ejecuten alguna de las funcionalidades del sistema. Solo tendrá contacto directo con la aplicación para las consultas que estén vinculadas directamente con la persona.

4.1.8 Diagramas de Casos de Uso

En esta sección se presenta los correspondientes diagramas de casos de uso para la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”, esto permitirá identificar más claramente las diferentes relaciones entre los actores y los casos de uso.

4.1.9 Caso de uso ingreso al sistema

En el presente caso de uso se visualiza la interacción entre el sistema y los actores que acceden al mismo.

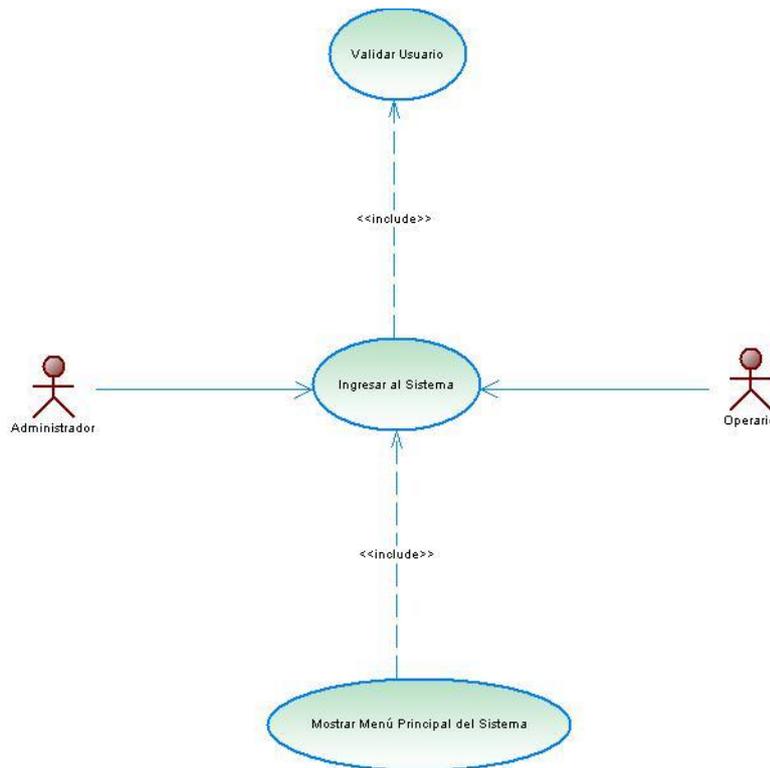


Ilustración 2.- Caso de uso ingreso al sistema.
Fuente: Andrés Cheza

Caso de Uso	Descripción Caso de Uso
Ingresar al Sistema	Es la parte donde se realiza el ingreso al sistema por los usuarios determinados.
Validar Usuario	Permite realizar una validación de los datos ingresados por el usuario.
Mostrar Menú Principal del Sistema	Se encarga de mostrar el menú principal a los usuarios que ingresan

Tabla 24.- Caso de uso ingreso al sistema.
Fuente: Andrés Cheza

4.1.10 Caso de uso creación de un contrato para un cliente

En este caso de uso muestra la manera en que una persona realiza la solicitud del servicio de agua potable y como interactúa con los demás actores que intervienen en el sistema.

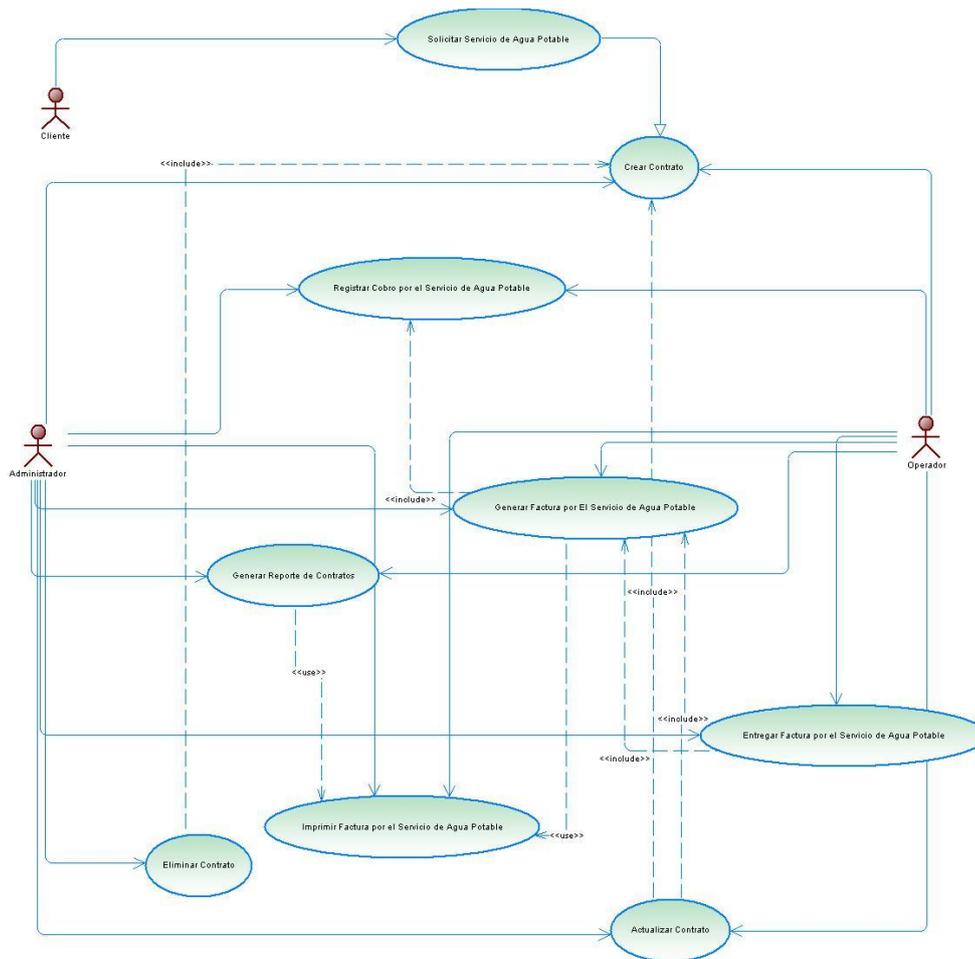


Ilustración 3.- Caso de uso creación de un contrato para un cliente.
Fuente: Andrés Cheza

Caso de Uso	Descripción Caso de uso
Solicitar Servicio de Agua Potable	Es la parte donde el cliente realiza una solicitud para el servicio de agua.
Crear Contrato	Se encarga de la creación de un contrato para un cliente.
Registrar Cobro por El Servicio de Agua Potable	Realiza el correspondiente registro del cobro por el servicio.
Generar Factura por el Servicio de Agua Potable	Permite crear una factura con los respectivos cobros que se realiza.
Imprimir Factura por el Servicio de Agua Potable	Permite realizar las impresiones de los correspondientes documentos para ser entregados.
Actualizar Contrato	Permite realizar la actualización de un contrato o también una factura del cobro por el servicio solicitado.
Eliminar Contrato	Permite realizar la eliminación de un contrato o también una factura del cobro por el servicio solicitado.
Generar Reporte de Contratos	Se encarga de permitir realizar un reporte de los contratos creados.

Tabla 25.- Caso de uso creación de un contrato para un cliente.
Fuente: Andrés Cheza

4.1.11 Caso de uso Registro de las Lecturas de Cada Mes.

En el presente caso de uso se especifica el proceso que se lleva a cabo para el registro de las lecturas tomadas de cada mes.

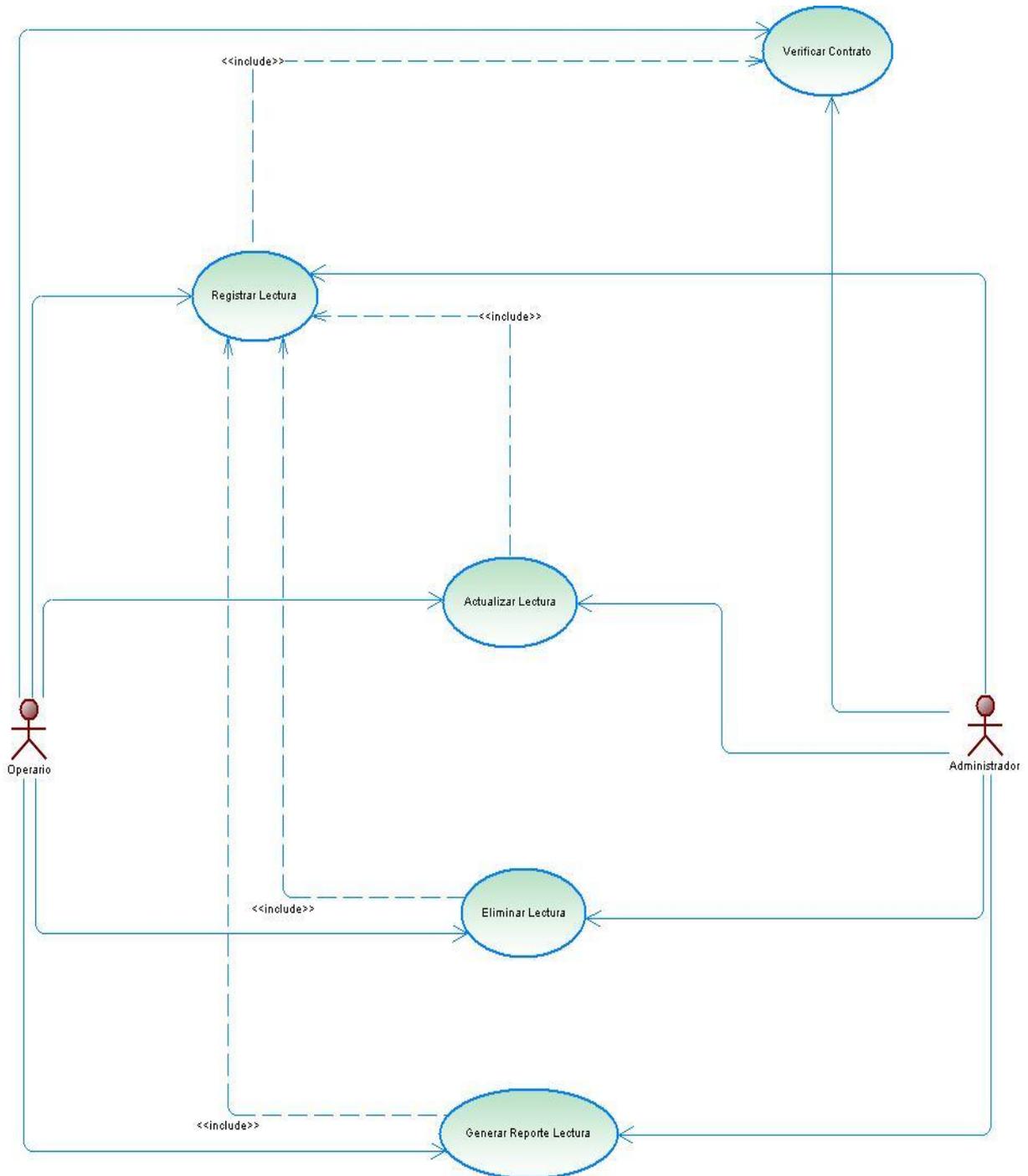


Ilustración 4.- Caso de uso Registro de las Lecturas de Cada Mes.
Fuente: Andrés Cheza.

Caso de Uso	Descripción Caso de Uso
Verificar Contrato	Se encarga de verificar que un cliente tenga creado un contrato en el sistema.
Registrar Lectura	Permite realizar el correspondiente registro de las lecturas tomadas mes a mes.
Actualizar Lectura	Tiene la función de actualizar los datos registrados de una lectura.
Eliminar Lectura	Realiza una actualización de los datos registrados de una lectura.
Generar Reporte Lectura	Se encarga de realizar un reporte diario de las lecturas registradas en el sistema.

Tabla 26.- Caso de uso Registro de las Lecturas de Cada Mes.
Fuente: Andrés Cheza.

4.1.12 Caso de uso Registro del Valor del Fondo Rojo.

En este caso de uso especificamos la manera en que se realiza el registro del valor correspondiente al fondo rojo.

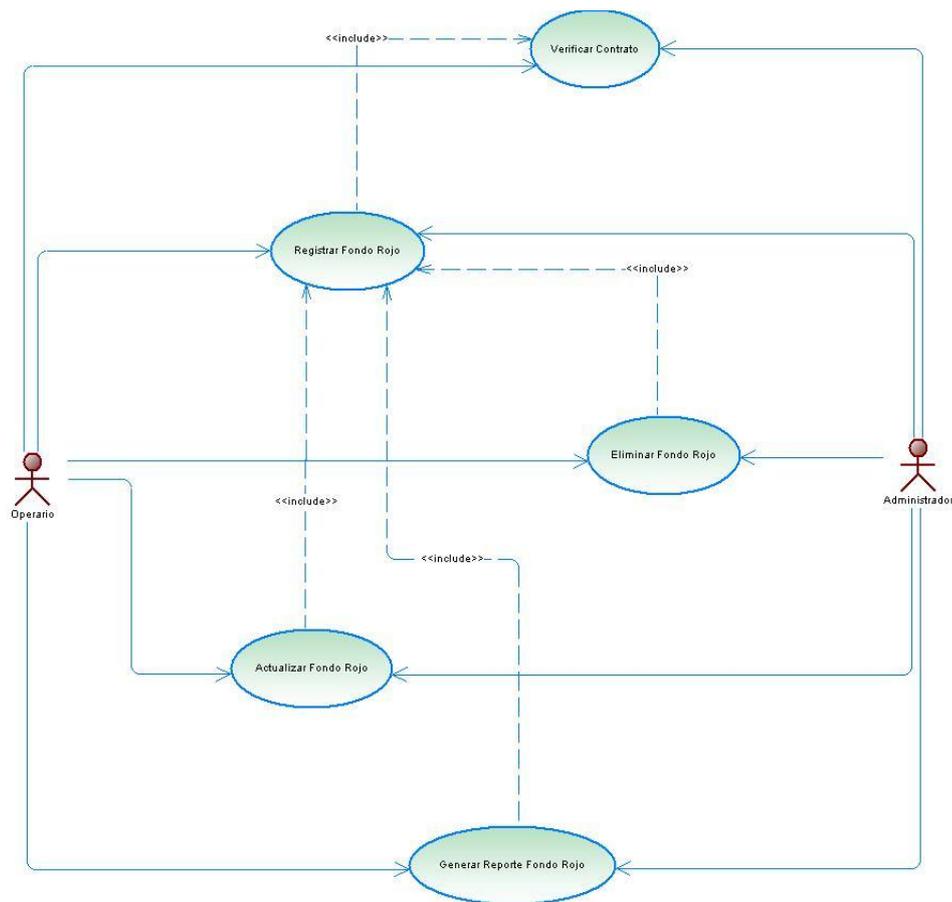


Ilustración 5.- Caso de uso Registro del Valor del Fondo Rojo.
Fuente: Andrés Cheza.

Caso de Uso	Descripción Caso de Uso
Verificar Contrato	Tiene como objetivo verificar que un usuario tenga un contrato registrado en el sistema.
Registrar Fondo Rojo	Permite registrar el valor correspondiente al fondo rojo.
Actualizar Fondo Rojo	Se encarga de realizar la actualización de los datos del fondo rojo registrado en el sistema.
Eliminar Fondo Rojo	El objetivo es realizar la eliminación de un fondo rojo registrado previamente en el sistema.
Generar Reporte Fondo Rojo	Permite realizar un reporte diario de los datos del fondo rojo registrados en el sistema.

Tabla 27.- Caso de uso Registro del Valor del Fondo Rojo.

Fuente: Andrés Cheza.

4.1.13 Caso de uso Pago del Consumo de Agua Potable

En este caso de uso se refiere a la manera que un cliente realiza el correspondiente pago por el consumo de agua potable y cómo interactúan los correspondientes actores del sistema.

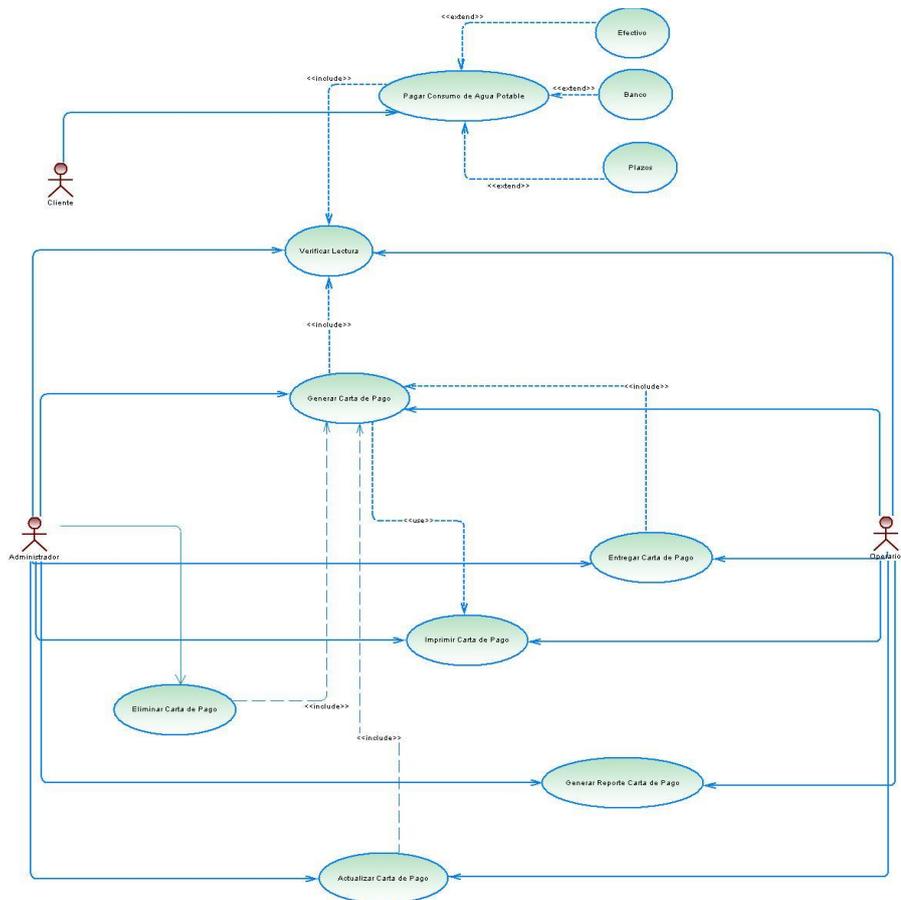


Ilustración 6.- Caso de uso Pago del Consumo de Agua Potable.

Fuente: Andrés Cheza

Caso de Uso	Descripción de Caso de Uso
Pagar Consumo Agua Potable	Permite realizar el pago del consumo del agua potable por un cliente.
Efectivo	El valor a para puede realizárselo en efectivo.
Banco	El pago también puede ser realizado a través del banco.
Plazos	Nos permite establecer el pago a través de un plazo determinado.
Validar Lectura	Se encarga de realizar la correspondiente validación de la lectura que tiene un determinado usuario.
Registrar Cobro	Permite realizar el registro de los valores referentes al cobro.
Generar Carta de Pago	Tiene como función realizar la generación de la carta de pago para el cliente.
Imprimir Carta de Pago	Nos permite realizar la impresión de la carta de pago así como también de un reporte si fuese necesario.
Entregar Carta de Pago	Se realiza la correspondiente entrega de la carta de pago al cliente.
Actualizar Carta de Pago	Tiene la función de actualizar un pago realizado por un cliente.
Eliminar Carta de Pago	Se encarga de eliminar un pago realizado por un cliente.
Generar Reporte Carta de Pago	Se encarga de realizar un reporte de los correspondientes pagos realizados por los clientes.

Tabla 28.- Caso de uso Pago del Consumo de Agua Potable.
Fuente: Andrés Cheza

4.1.14 Caso de Uso Suspensión del Servicio de Agua Potable

Este diagrama de caso de uso tiene como objetivo mostrar el proceso que el sistema realiza para la suspensión del servicio de agua potable a un cliente.

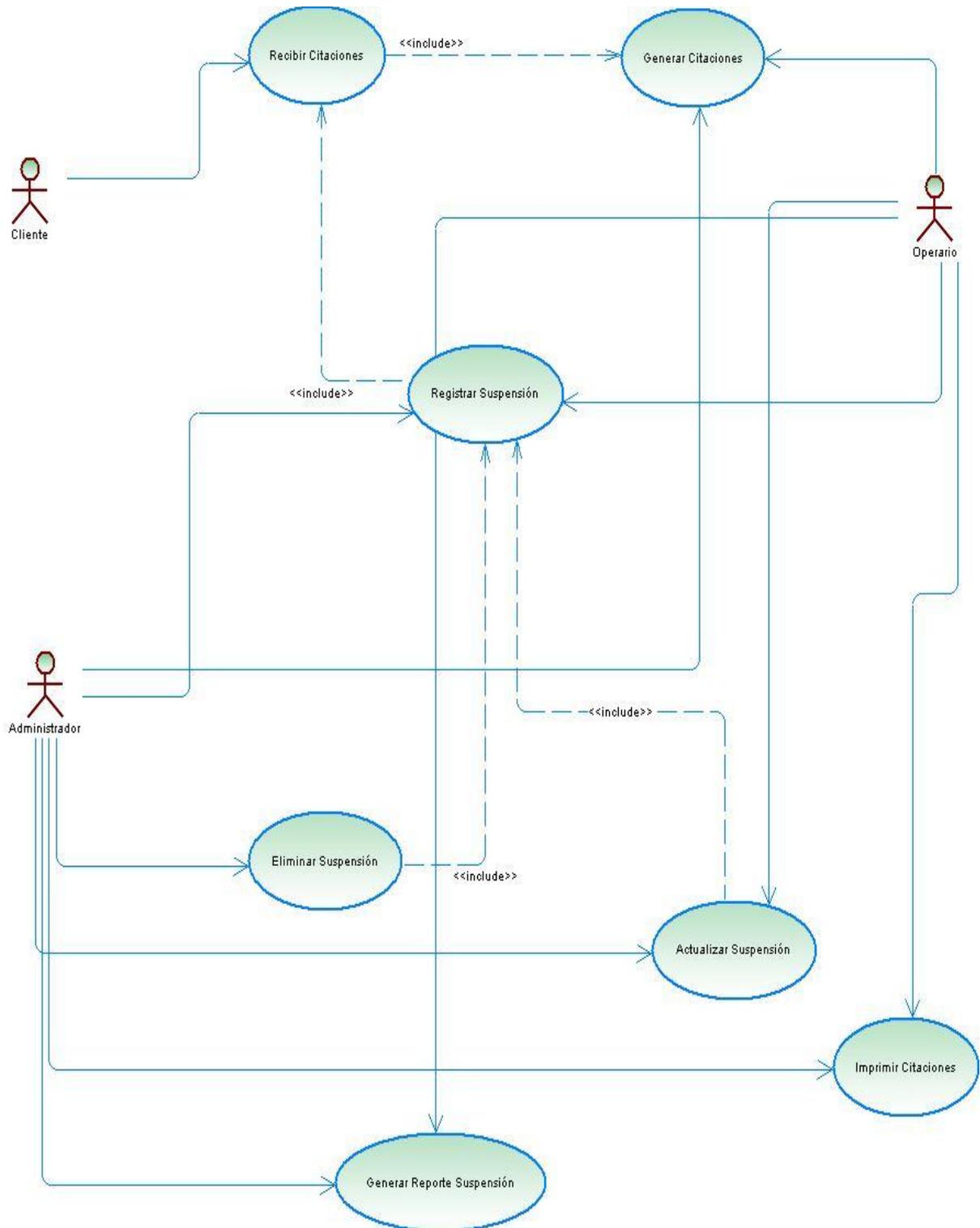


Ilustración 7.- Caso de Uso Suspensión del Servicio de Agua Potable.
Fuente: Andrés Cheza

Caso de Uso	Descripción de Caso de Uso
Generar Citaciones	Permite generar citaciones de aviso para un cliente.
Imprimir	Realiza la impresión de las citaciones que se emitirán a un cliente de acuerdo a las condiciones en que se encuentre respecto al pago del servicio de agua potable.
Recibir Citaciones	Tiene la función de que un cliente reciba de una a tres citaciones.
Registrar Suspensión	Permite realizar el registro de la suspensión realizada a un cliente.
Actualizar Suspensión	Se encarga de realizar la actualización de los datos de la suspensión registrada de un cliente.
Eliminar Suspensión	Permite realizar la eliminación de los datos de la suspensión registrada de un cliente.
Generar Reporte Suspencion	Tiene la función de permitir realizar un reporte de las suspensiones generadas a los clientes.

Tabla 29.- Caso de Uso Suspensión del Servicio de Agua Potable.
Fuente: Andrés Cheza.

4.1.15 Caso de Uso Reconexión del Servicio de Agua Potable

En este diagrama de casos de uso permite ver el proceso que el sistema realiza para la re conexión del servicio de agua potable a un cliente y como interactúa con los actores del sistema.

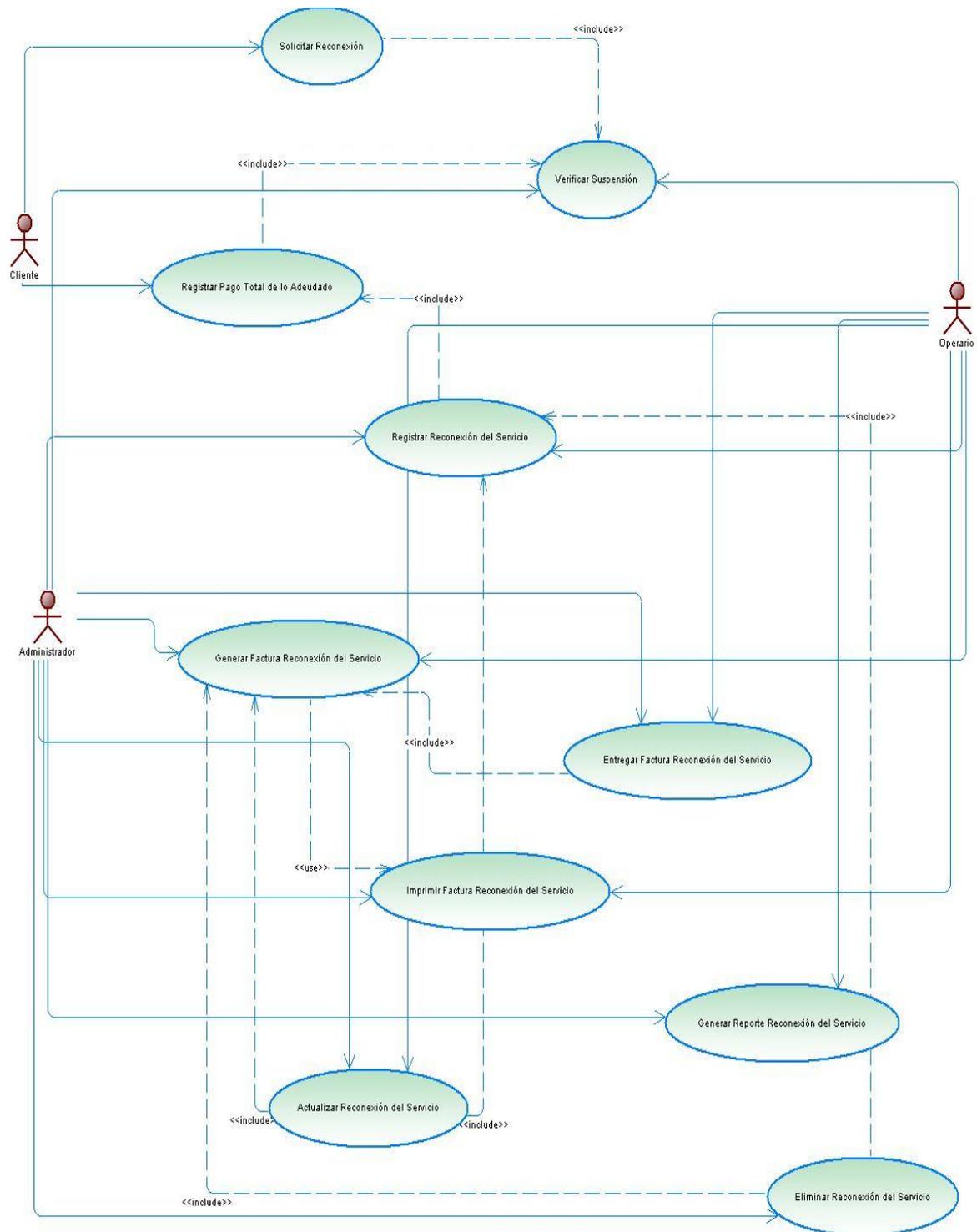


Ilustración 8.- Caso de Uso Reconexión del Servicio de Agua Potable.
Fuente: Andrés Cheza

Caso de Uso	Descripción de Caso de Uso
Solicitar Reconexión	Permite realizar una solicitud para la re conexión del servicio de agua potable.
Verificar Suspensión	Se encarga de realizar la correspondiente verificación de que exista la suspensión del servicio para poder realizar el respectivo trámite.
Registrar Pago Total Adeudado	Se encarga de realizar el pago total de la deuda que tiene el cliente para continuar con el proceso de re conexión.
Registrar Re Conexión del Servicio	Realiza el registro de la re conexión del servicio para el cliente que lo solicito.
Generar Factura Reconexión Servicio	Tiene la función de generar una factura con los correspondientes datos del cliente y valores que este cancelo.
Imprimir Factura Reconexión Servicio	Permite realizar la impresión de la factura y también de los reportes referentes al trámite realizado por el cliente si fuese necesario.
Entregar Factura Reconexión Servicio	Se encarga de entregar la factura al cliente que solicito el correspondiente tramite de re conexión del servicio de agua potable.
Actualizar Reconexión del Servicio	Tiene la función de realizar una actualización de los datos tanto de la re conexión como también de la factura que se genera con la constancia de pago de un cliente.
Eliminar Reconexión del Servicio	Se encarga de eliminar los datos tanto de la re conexión como también de la factura que se genera con la constancia de pago de un cliente.
Generar Reporte Reconexión del Servicio	Permite generar un reporte del proceso de re conexión del servicio de agua potable a un cliente.

Tabla 30.- Caso de Uso Re conexión del Servicio de Agua Potable.

Fuente: Andrés Cheza.

4.1.16 Caso de Uso Convocatoria

Este diagrama de casos de uso indica la interacción que tiene el proceso de convocatorias a diferentes asuntos que realiza la junta con los clientes.

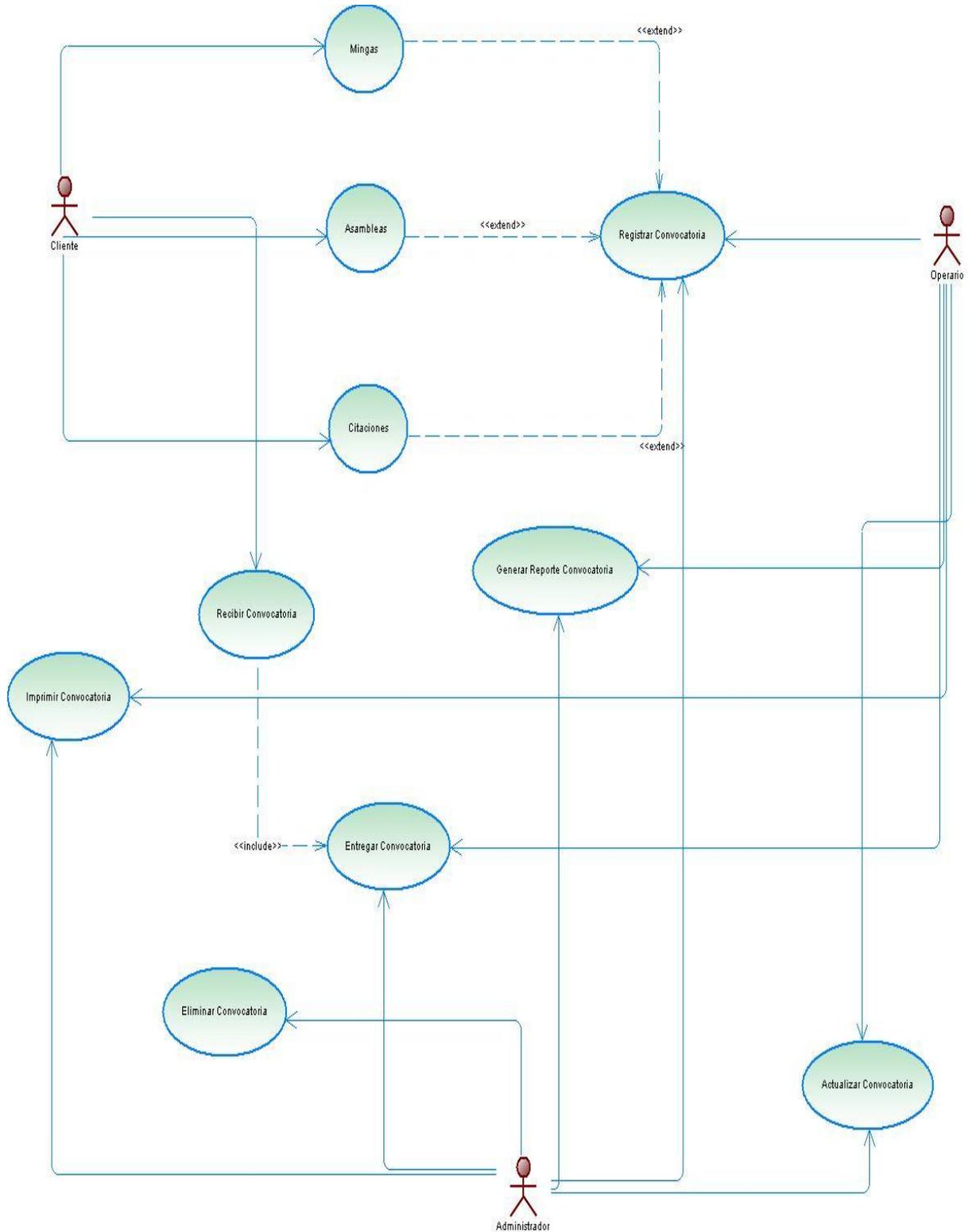


Ilustración 9.- Caso de Uso Convocatoria.
Fuente: Andrés Cheza.

Caso de Uso	Descripción de Caso de Uso
Registrar Convocatoria	Se encarga de registrar la descripción de la convocatoria.
Mingas	Permite establecer uno de los motivos por los cuales se realiza la convocatoria como en este caso puede referirse a mingas.
Asambleas	Tiene la función de indicar uno de los motivos por los cuales se realiza la convocatoria de acuerdo a este caso puede ser asambleas.
Citaciones	Permite establecer uno de los motivos por los cuales se realiza la convocatoria también puede indicar que se trata de citas.
Imprimir Convocatoria	Se encarga de realizar la impresión de la correspondiente convocatoria.
Entregar Convocatoria	Permite ser entregada la convocatoria al cliente.
Actualizar Convocatoria	Tiene como objetivo realizar una actualización de los datos de una convocatoria.
Eliminar Convocatoria	Se encarga de realizar la eliminación de una convocatoria.
Generar Reporte Convocatoria	La función de este caso de uso es realizar un reporte referente a las convocatorias.

Tabla 31.- Caso de Uso Convocatoria.
Fuente: Andrés Cheza.

4.1.17 Caso de Uso Solicitud de un Préstamo

Este diagrama de casos de uso demuestra los pasos que un cliente debe seguir para realizar un préstamo y como se realiza la interacción con los demás actores del sistema.

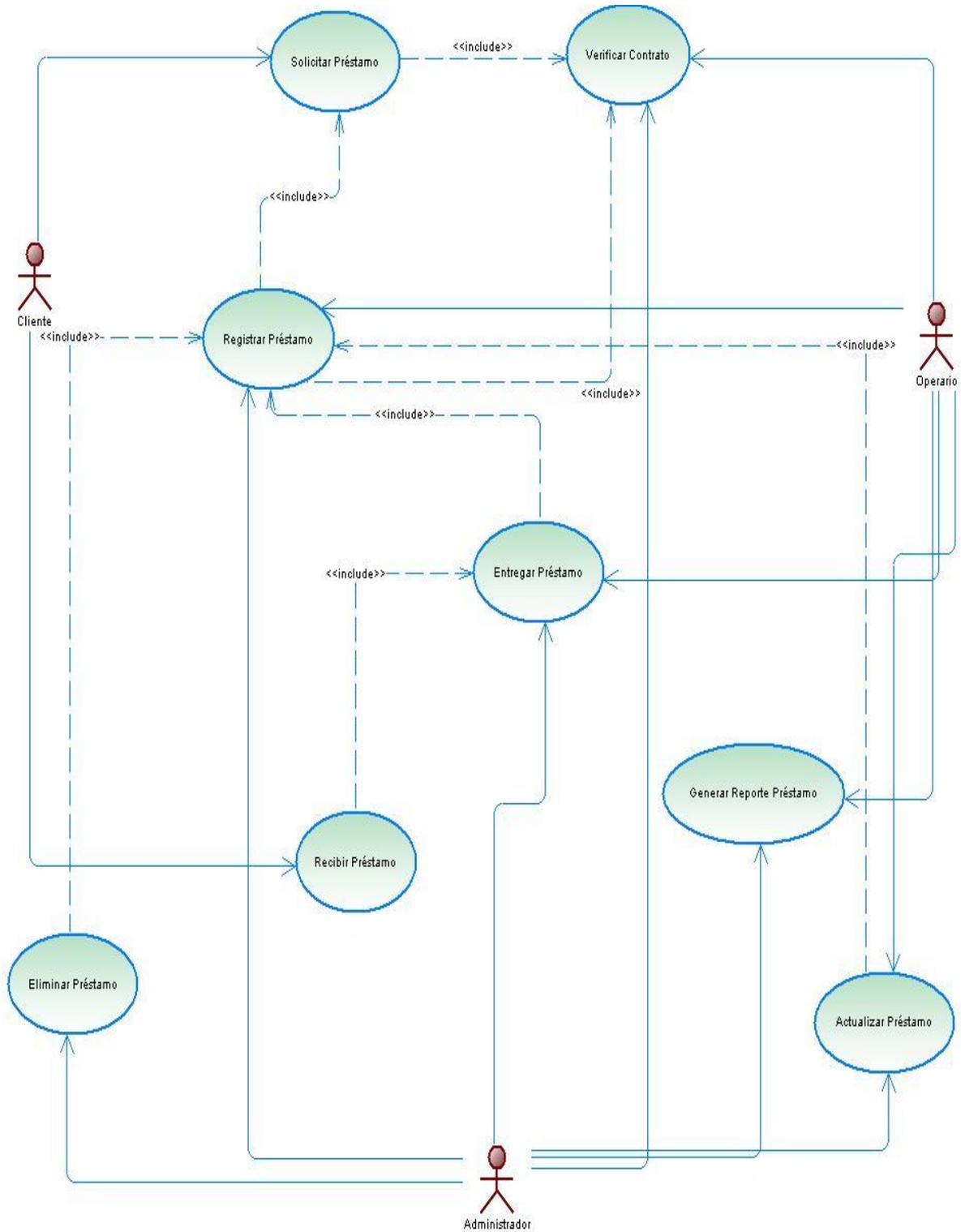


Ilustración 10.- Caso de Uso Solicitud de un Préstamo.
Fuente: Andrés Cheza.

Caso de Usos	Descripción de Caso de Usos
Solicitar Préstamo	Se encarga de realizar la solicitud de un préstamo por un cliente.
Verificar Contrato	Permite realizar la correspondiente verificación de la existencia del contrato del cliente que realiza la solicitud de un préstamo para hacer valida la solicitud.
Registrar Préstamo	Tiene como función realizar el registro de los datos de un préstamo realizado.
Entregar Préstamo	Después de haber sido registrado el préstamo se procede a la entrega del préstamo que se ha solicitado.
Actualizar Préstamo	Se encarga de realizar una actualización de datos referentes a un préstamo previamente registrado en el sistema.
Eliminar Préstamo	La función que corresponde es la de eliminar un préstamo que se haya registrado en el sistema.
Generar Reporte Préstamo	Permite realizar un reporte de los préstamos realizados.

Tabla 32.- Caso de Uso Solicitud de un Préstamo.
Fuente: Andrés Cheza.

4.1.18 Caso de Uso Pago de un Préstamo Realizado

En estos casos de uso si manifiesta el proceso que un cliente tiene que realizar para el pago de un préstamo, la cancelación del préstamo puede ser en varias cuotas o directamente el pago total.

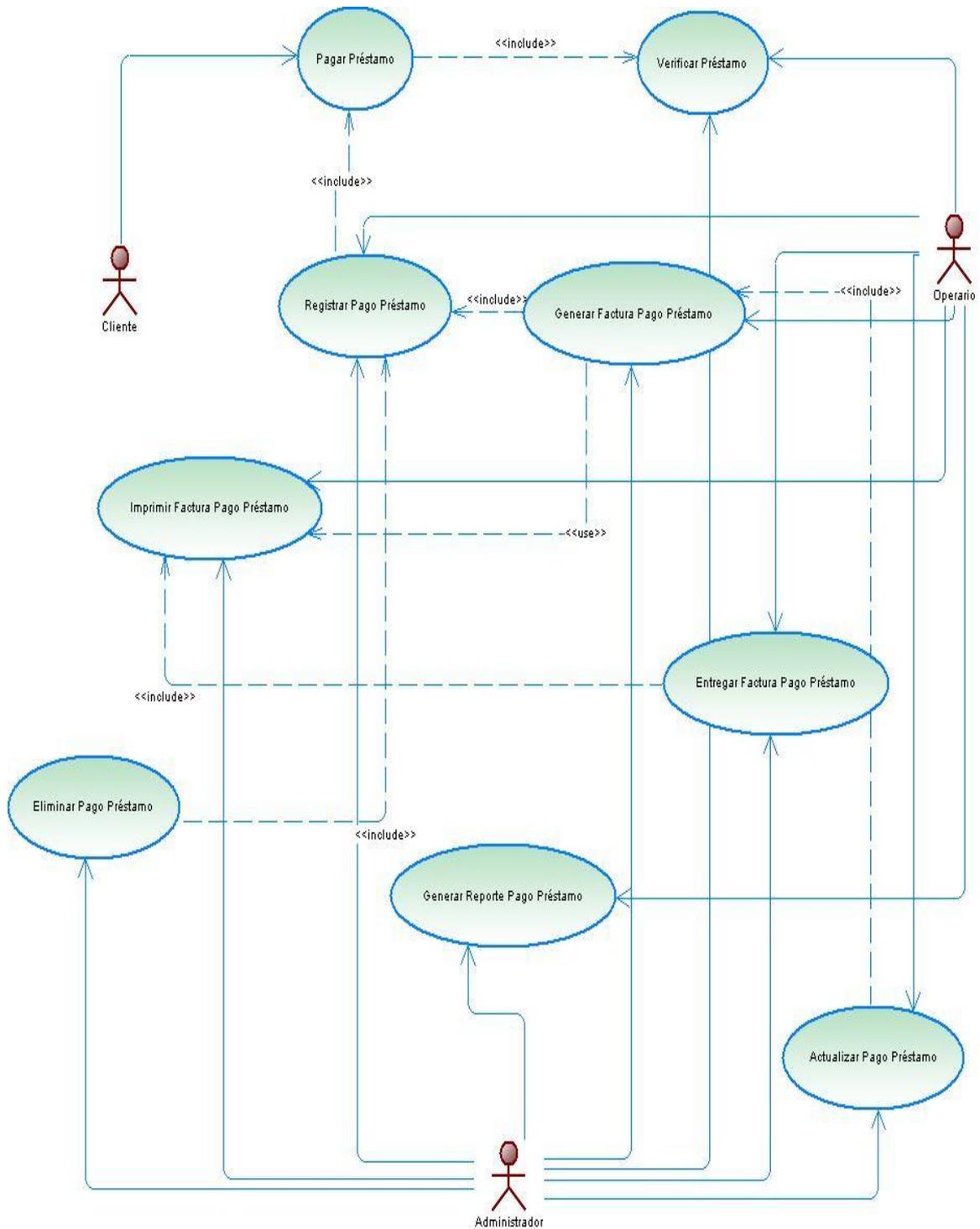


Ilustración 11.- Caso de Uso Pago de un Préstamo Realizado.
Fuente: Andrés Cheza.

Caso de Uso	Descripción de Caso de Uso
Pagar Préstamo	Permite realizar el pago del préstamo de acuerdo al acuerdo que se haya llegado al momento de su solicitud.
Verificar Préstamo	Tiene como objetivo la verificación de que exista el préstamo realizado por el determinado cliente para poder continuar con el proceso.
Registrar Pago Préstamo	Se encarga de registrar el pago que realiza un cliente.
Generar Factura Pago Préstamo	Permite realizar la generación de la factura referente al pago realizado por el cliente.
Imprimir Pago Préstamo	Después de la correspondiente generación de la factura se procede a la impresión de este documento.
Entregar Factura Pago Préstamo	Se encarga de realizar la entrega de la factura al cliente.
Actualizar Pago Préstamo	Tiene la función de realizar la actualización del pago del préstamo.
Eliminar Pago Préstamo	La función de este caso de uso es la de permitir eliminar el pago del préstamo realizado.
Generar Reporte Pago Préstamo	Permite generar el reporte de los pagos que se realizan.

Tabla 33.- Pago de un Préstamo Realizado.
Fuente: Andrés Cheza.

4.1.19 Caso de Uso Pago de los Intereses de un Préstamo Realizado

Este diagrama de caso de uso muestra la forma en que un cliente debe actuar para realizar el pago de los intereses del préstamo y la interacción con los demás actores que se relacionan con el sistema.

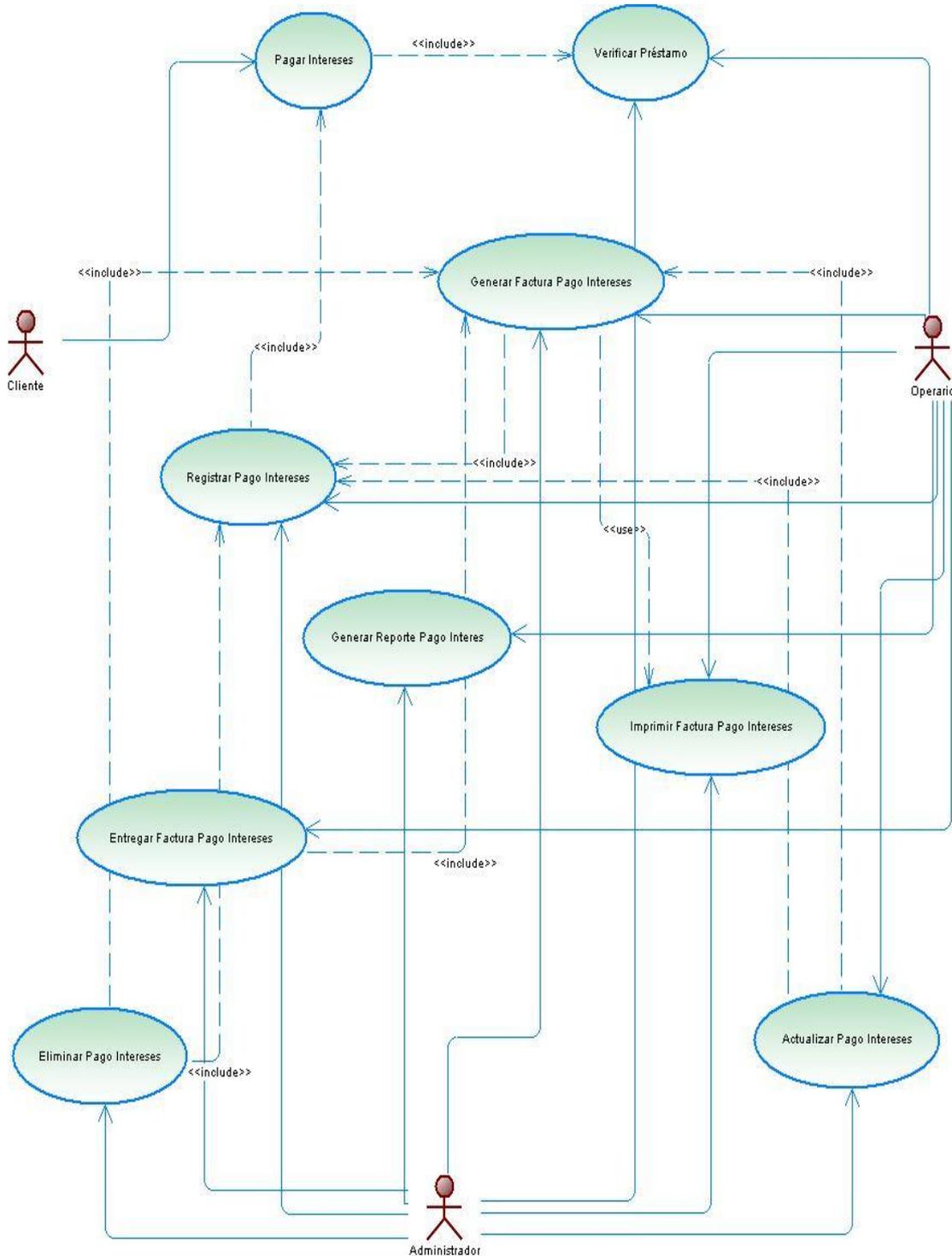


Ilustración 12.- Caso de Uso Pago de los Intereses de un Préstamo Realizado.
Fuente: Andrés Cheza.

Caso de Uso	Descripción de Caso de Uso
Pagar Intereses	Permite realizar el pago de los intereses de un préstamo por un cliente.
Registrar Pago Intereses	Se encarga de realizar el registro del pago de los intereses que realiza un cliente.
Generar Factura Pago Intereses	Después de haber sido registrado el pago de los intereses del préstamo se procede a la generación de la factura.
Imprimir Pago Intereses	Permite realizar la impresión de la factura y los reportes si así se lo deseara o fuese necesario.
Entregar Factura Pago Intereses	Luego de imprimirse la factura se procede a la entrega de la misma a un cliente.
Actualizar Pago Intereses	Se encarga de realizar la actualización de los datos registrados referentes al pago de los intereses del préstamo.
Eliminar Pago Intereses	Permite eliminar los pagos de los intereses registrados en el sistema.
Generar Reporte Pago Intereses	Tiene como objetivo realizar un reporte de los pagos de los intereses realizados.

Tabla 34.- Caso de Uso Pago de los Intereses de un Préstamo Realizado.
Fuente: Andrés Cheza.

4.1.20 Caso de Uso Pago de Multas

En el presente diagrama de casos de uso se define el proceso que un cliente debe seguir para el pago de las multas, y su relación con los actores inmersos en el sistema.

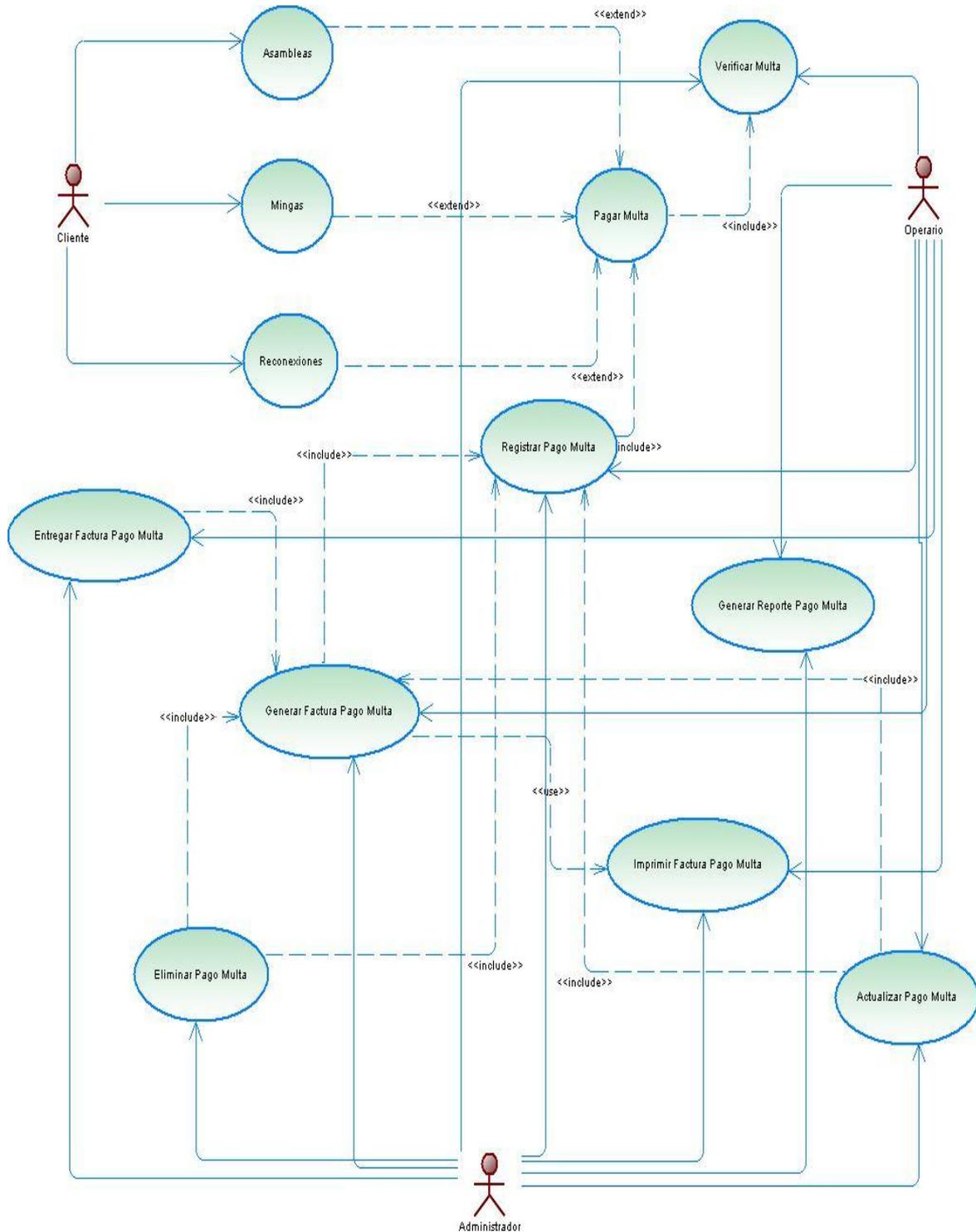


Ilustración 13.- Caso de Uso Pago de Multas.
Fuente: Andrés Cheza

Caso de Uso	Descripción de Caso de Uso
Pagar Multa	Permite realizar el pago de una multa referente a una determinada obligación.
Asambleas	Las multas pueden ser asambleas que no ha asistido el cliente.
Mingas	Una de las multas puede ser referente a las mingas que un cliente no asistió.
Re conexión	Podría ser también una re conexión del servicio de agua potable la razón de la multa.
Registrar Pago Multa	Se encarga de realizar el registro del pago correspondiente a la multa establecida para un cliente.
Generar Factura Pago Multa	Luego de haber sido registrado el pago de la multa se procede a la creación o generación de la factura.
Imprimir Factura Pago Multa	Permite realizar la impresión de la factura para luego continuar con el proceso.
Entregar Factura Pago Multa	Después de ser impresa la factura se procede a la entrega correspondiente al cliente.
Actualizar Pago Multa	Se encarga de realizar una actualización de los datos registrados referente al cobro de la multa.
Eliminar Pago Multa	Permite eliminar un pago referente a una multa previamente registrado en el sistema.
Generar Reporte Pago Multa	Podemos generar un reporte diario de los pagos realizados concernientes a las multas por los clientes.

Tabla 35.- Caso de Uso Pago de Multas.

Fuente: Andrés Cheza.

4.2 Especificación de Caso de Uso.

4.2.1 Introducción.

En esta sección se definirán de una manera formal los diferentes casos de uso de la aplicación, esto servirá para identificar de una manera más clara las diferentes funcionalidades que debe tener el sistema.

En la siguiente tabla realizamos un resumen de los casos de uso especificados en los anteriores diagramas.

Numero	Caso de Uso
1	Ingresar al Sistema
2	Validar Usuario
3	Mostrar Menú Principal del Sistema
4	Solicitar el Servicio de Agua Potable
5	Crear Contrato
6	Registrar Cobro por El Servicio de Agua Potable
7	Generar Factura por el Servicio de Agua Potable
8	Imprimir Factura por el Servicio de Agua Potable
9	Actualizar Contrato
10	Eliminar Contrato
11	Generar Reporte de Contratos
12	Registrar Lectura
13	Actualizar Lectura
14	Eliminar Lectura
15	Generar Reporte Lecturas
16	Registrar Fondo Rojo
17	Actualizar Fondo Rojo
18	Eliminar Fondo Rojo
19	Generar Reporte Fondo Rojo
20	Pagar Consumo Agua Potable
21	Verificar Lectura
22	Generar Carta de Pago
23	Imprimir Carta de Pago
24	Actualizar Carta de Pago
25	Eliminar Carta de Pago
26	Generar Reporte Carta de Pago
27	Generar Citaciones
28	Imprimir Citaciones
29	Registrar Suspensión
30	Actualizar Suspensión
31	Eliminar Suspensión
32	Generar Reporte Suspensión
33	Verificar Suspensión
34	Registrar Pago Total Adeudado
35	Registrar Re Conexión del Servicio
36	Generar Factura Re conexión Servicio
37	Imprimir Factura Re conexión Servicio
38	Entregar Factura Re conexión Servicio
39	Actualizar Re conexión del Servicio
40	Eliminar Re conexión del Servicio
41	Generar Reporte de Re conexión del Servicio
42	Registrar Convocatoria
43	Imprimir Convocatoria
44	Entregar Convocatoria
45	Actualizar Convocatoria
46	Eliminar Convocatoria
47	Generar Reporte Convocatoria
48	Solicitar Préstamo
49	Verificar Contrato
50	Registrar Préstamo
51	Entregar Préstamo

52	Actualizar Préstamo
53	Eliminar Préstamo
54	Generar Reporte Préstamo
55	Pagar Préstamo
56	Verificar Préstamo
57	Registrar Pago Préstamo
58	Generar Factura Pago Préstamo
59	Imprimir Factura Pago Préstamo
60	Entregar Factura Pago Préstamo
61	Actualizar Pago Préstamo
62	Eliminar Pago Préstamo
63	Generar Reporte Pago Préstamo
64	Pagar Intereses
65	Registrar Pago Intereses
66	Generar Factura Pago Intereses
67	Imprimir Factura Pago Intereses
68	Entregar Factura Pago Intereses
69	Actualizar Pago intereses
70	Eliminar Pago intereses
71	Generar Reporte Pago intereses
72	Pagar Multa
73	Verificar Multa
74	Registrar Pago Multa
75	Generar Factura Pago Multa
76	Imprimir Factura Pago Multa
77	Entregar Factura Pago Multa
78	Actualizar Pago Multa
79	Eliminar Pago Multa
80	Generar Reporte Pago Multa

Tabla 36.- Resumen de Casos de Uso.
Fuente: Andrés Cheza.

Nota.- Los veinte primeros casos de uso de prioridad, servirán para el análisis durante el presente proyecto los demás se encuentran especificados en el CD del proyecto, carpeta Anexos documentos diagramas y especificación casos de uso.

4.2.2 Especificación de Caso de Uso: Ingresar al Sistema

4.2.2.1 Descripción.

Permite realizar el ingreso de un usuario al sistema.

4.2.2.2 Actores.

Administrador / Operario.

4.2.2.3 Precondiciones.

El sistema debe estar ejecutándose correctamente.

4.2.2.4 Flujo de Eventos

Flujo Básico.

- a. El actor realiza un pre ingreso al sistema.
- b. El sistema muestra una ventana modal con un formulario para digitar el nombre de usuario y la contraseña.
- c. El actor digita su nombre de usuario y su contraseña correspondiente.
- d. El actor pulsa el botón ingresar.
- e. El sistema realiza una verificación de que los datos ingresados sean correctos.

Flujos Alternativos.

- a. El sistema comprueba la valides de los datos ingresados, si los datos no son correctos, se realiza un aviso al actor de ello permitiendo que pueda corregirlos.
- b. El sistema comprueba que todos los campos del formulario estén llenos, si existe algún casillero sin llenar, el sistema avisa al actor para que los pueda llenar.
- c. Si el actor presiona el botón cancelar, finaliza el caso de uso.

4.2.2.5 Pos condiciones

El sistema ha capturado los datos ingresados por el usuario.

4.2.3 Especificación de Caso de Uso: Validar Usuario

4.2.3.1 Descripción.

Realiza la validación correspondiente de un usuario.

4.2.3.2 Actores.

Administrador / Operario.

4.2.3.3 Precondiciones.

Especificación de Caso de Uso 4.2.2

4.2.3.4 Flujo de Eventos

Flujo Básico.

- a. El sistema realiza una búsqueda en la tabla usuarios para verificar si existe el usuario.
- b. El sistema muestra el éxito del ingreso.

Flujos Alternativos.

- a. El sistema realiza una búsqueda en la tabla usuarios para verificar si existe el usuario, si existe el usuario el ingreso es exitoso, caso contrario regresa a la ventana modal e informa al actor de que el usuario no existe.

4.2.3.5 Pos condiciones.

El sistema comprueba que los datos ingresados por el usuario han sido correctos. El sistema realiza la operación de re direccionar a la página correspondiente.

4.2.4 Especificación de Caso de Uso: Mostrar Menú Principal del Sistema.

4.2.4.1 Descripción.

Permite desplegar el menú principal del sistema.

4.2.4.2 Actores.

Administrador / Operario.

4.2.4.3Precondiciones.

Especificación de Caso de Uso 4.2.3

4.2.4.4 Flujo de Eventos

Flujo Básico.

- a. El sistema muestra el menú principal de acuerdo a los privilegios que el actor tenga.
- b. El actor tiene la posibilidad de realizar las operaciones que el sistema permite efectuar.

Flujos Alternativos.

N/A.

4.2.4.5 Pos condiciones.

El usuario ha ingresado satisfactoriamente al sistema y está listo para realizar las operaciones que necesite efectuar.

4.2.5 Especificación de Caso de Uso: Solicitar el Servicio de Agua Potable

4.2.5.1 Descripción.

Permite realizar la solicitud del servicio de agua potable por parte de una persona a la "JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO".

4.2.5.2 Actores.

Cliente y Administrador.

4.2.5.3 Precondiciones.

Especificación de Caso de Uso 4.2.4

4.2.5.4 Flujo de Eventos

Flujo Básico.

- a. El cliente realiza una solicitud del servicio de agua potable al administrador del sistema.
- b. El administrador del sistema hace click en el menú del sistema opción clientes.
- c. El sistema despliega las opciones con las que cuenta dicho menú.
- d. El administrador del sistema pulsa sobre la opción crear nuevo cliente.
- e. El sistema muestra el formulario con los campos necesarios para ingresar los datos del nuevo cliente.
- f. El administrador introduce los datos del nuevo cliente.
- g. El sistema verifica que los datos ingresados sean correctos y los almacena en el sistema.

Flujos Alternativos.

- a. El sistema comprueba la validez de los datos ingresados, si los datos no son correctos, se realiza un aviso al actor de ello permitiendo que pueda corregirlos.
- b. El sistema comprueba que todos los campos del formulario estén llenos, si existe algún casillero sin llenar, el sistema avisa al actor para que los pueda llenar.

4.2.5.5 Pos condiciones

Los datos del nuevo cliente fueron almacenados correctamente en el sistema.

4.2.6 Especificación de Caso de Uso: Crear Contrato

4.2.6.1 Descripción.

Permite crear un contrato en el sistema para un cliente previamente registrado.

4.2.6.2 Actores.

Administrador.

4.2.6.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.5

4.2.6.4 Flujo de Eventos

Flujo Básico.

- a. El administrador pulsa en el menú la opción contratos.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El administrador hace click en la opción crear nuevo contrato.
- d. El sistema muestra un formulario con los campos a ser llenados.
- e. El administrador llena el formulario con los datos necesarios.
- f. El sistema realiza una verificación de que los datos ingresados sean correctos y los almacena en el sistema.

Flujos Alternativos.

- a. El sistema comprueba que todos los campos estén llenos, si existe un casillero sin llenar, el sistema avisa al actor para que pueda llenar los campos que son necesarios.
- b. El sistema comprueba la validez de los datos, si los datos no son correctos, realiza un aviso al actor de ello permitiendo que los corrija.

4.2.6.5 Pos condiciones.

Los datos del contrato han sido almacenados correctamente en el sistema permitiendo que el nuevo cliente cuente con un contrato registrado en el sistema.

4.2.7 Especificación de Caso de Uso: Registrar Cobro por El Servicio de Agua Potable.

4.2.7.1 Descripción.

Permite realizar el registro del cobro por el servicio de agua potable.

4.2.7.2 Actores.

Cliente, Administrador.

4.2.7.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.6

4.2.7.4 Flujo de Eventos

Flujo Básico.

- a. El administrador pulsa en el menú la opción contratos.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El administrador hace click en la opción agregar detalle del contrato.
- d. El sistema realiza una búsqueda interna para ver que contratos se encuentran creados recientemente y verificar los que aún no han sido cancelados.
- e. El sistema muestra un menú desplegable y el formulario con los campos de texto para ingresar el detalle del contrato.
- f. El Administrador presiona sobre el menú desplegable y hace click en la opción que muestra el nombre del cliente.
- g. El sistema muestra los datos referentes al cliente y su respectivo contrato además activa los campos de texto para que puedan ser llenados con los datos solicitados.
- h. El administrador llena los cuadros de texto con los datos pertinentes.
- i. El sistema comprueba la valides de los datos ingresados previamente y los almacena.

Flujos Alternativos.

- a. El sistema comprueba que todos los campos estén llenos, si existe un casillero sin llenar, el sistema avisa al actor para que pueda llenar los campos que son necesarios.
- b. El sistema comprueba la valides de los datos, si los datos no son correctos, realiza un aviso al actor de ello permitiendo que los corrija.

4.2.7.5 Pos condiciones.

Los datos del detalle del contrato han sido almacenados correctamente en el sistema y el usuario realiza el correspondiente pago por el servicio de agua potable.

4.2.8 Especificación de Caso de Uso: Generar Factura por el Servicio de Agua Potable.

4.2.8.1 Descripción.

Permite generar la factura correspondiente al pago del servicio de agua.

4.2.8.2 Actores.

Cliente, Operador.

4.2.8.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.7

4.2.8.4 Flujo de Eventos

Flujo Básico.

- a. El operario pulsa en el menú la opción facturas
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El operario hace click en emitir factura.
- d. El sistema muestra el formulario con las diferentes opciones para generar la factura.
- e. El operador pulsa en el menú desplegable y hace click en la opción correspondiente al sector que pertenece el cliente.
- f. El sistema realiza una búsqueda de clientes por sectores.
- g. El operador pulsa en el menú desplegable y busca el nombre del cliente al cual se va a emitir la factura.
- h. El sistema muestra los datos del cliente y activa las opciones necesarias para generar la factura.
- i. El operador completa la información necesaria y pulsa en el botón generar factura.
- j. El sistema almacena los datos y muestra la factura.

Flujos Alternativos.

N/A

4.2.8.5 Pos condiciones.

El sistema almacena los datos de la factura emitida al cliente para su posterior utilización en algún otro proceso.

4.2.9 Especificación de Caso de Uso: Imprimir Factura por el Servicio de Agua Potable.

4.2.9.1 Descripción.

Permite generar la factura correspondiente al pago del servicio de agua.

4.2.9.2 Actores.

Cliente, Operador.

4.2.9.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.8

4.2.9.4 Flujo de Eventos

Flujo Básico.

- a. El operario pulsa en el menú la opción facturas
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El operario hace click en ver facturas.
- d. El sistema muestra las diferentes opciones que tiene.
- e. El operador pulsa en el menú desplegable y hace click en la opción correspondiente al sector que pertenece el cliente.
- f. El sistema realiza una búsqueda de clientes por sectores.
- g. El operador pulsa en el menú desplegable y busca el nombre del cliente al cual se va a imprimir la factura.
- h. El sistema muestra la factura con los correspondientes datos.
- i. El operador pulsa en la opción imprimir factura.
- j. El sistema realiza la impresión de la factura.

Flujos Alternativos.

N/A

4.2.9.5 Pos condiciones.

El sistema imprime la factura la cual es emitida al cliente.

4.2.10 Especificación de Caso de Uso: Actualizar Contrato.

4.2.10.1 Descripción.

Permite realizar la actualización de los datos de un contrato.

4.2.10.2 Actores.

Administrador.

4.2.10.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.6

4.2.10.4 Flujo de Eventos

Flujo Básico.

- a. El administrador pulsa en el menú la opción contratos.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El administrador hace click en actualizar contratos.
- d. El sistema muestra todos los contratos registrados en el sistema.
- e. El administrador verifica el contrato del cliente y pulsa en la opción actualizar.
- f. El sistema despliega una ventana modal con los datos del contrato que se desea actualizar.
- g. El administrador revisa que datos va a actualizar y luego presiona sobre el botón actualizar.
- h. El sistema verifica la valides de los datos, realiza la actualización de los datos y almacena en el sistema.
- i. El operador revisa la correcta actualización de los datos.

Flujos Alternativos.

- a. El sistema comprueba la valides de los datos, si los nuevos datos no son correctos avisa al administrador de ello permitiéndole corregir.
- b. El sistema realiza una validación de que los campos de texto requerido estén llenos, caso contrario muestra un mensaje al administrador de ello dándole la opción de que los revise y los llene.

4.2.10.5 Pos condiciones.

N/A

4.2.11 Especificación de Caso de Uso: Eliminar Contrato.

4.2.11.1 Descripción.

Permite eliminar un contrato del sistema.

4.2.11.2 Actores.

Administrador.

4.2.11.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.6

4.2.11.4 Flujo de Eventos

Flujo Básico.

- a. El administrador pulsa en el menú la opción contratos.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El administrador hace click en eliminar contratos.
- d. El sistema muestra todos los contratos registrados en el sistema.
- e. El administrador verifica el contrato del cliente y pulsa en la opción eliminar.
- f. El sistema despliega una ventana modal con los datos del contrato que se desea eliminar.
- g. El administrador revisa que los datos que va a eliminar son los correctos, luego presiona sobre el botón eliminar.
- h. El sistema elimina los datos del contrato.
- i. El operador revisa la correcta eliminación de los datos.

Flujos Alternativos.

- a. El sistema muestra una ventana modal con dos opciones la de eliminar y la de cancelar eliminación. Si el usuario presiona en la opción eliminar el sistema eliminara definitivamente el contrato, caso contrario si presiona en el botón cancelar el sistema cancela la eliminación y muestra todos los contratos registrados.

4.2.11.5 Pos condiciones.

N/A

4.2.12 Especificación de Caso de Uso: Generar Reporte de Contratos

4.2.12.1 Descripción.

Tiene como función realizar reporte de los contratos que se encuentran registrados en el sistema.

4.2.12.2 Actores.

Administrador, Operario.

4.2.12.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.6

4.2.12.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción reportes.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.

- c. El actor hace click en la opción contratos.
- d. El sistema muestra todos los contratos registrados en el sistema.
- e. El actor pulsa en la opción generar reporte en formato pdf.
- f. El sistema realiza la acción solicitada y muestra todos los contratos en formato pdf, permitiendo varias opciones.
- g. El actor pulsa en imprimir o en la opción guardar archivo.
- h. El sistema realiza la opción seleccionada.

Flujos Alternativos.

N/A

4.2.12.5 Pos condiciones.

N/A

4.2.13 Especificación de Caso de Uso: Registrar Lecturas

4.2.13.1 Descripción.

Permite realizar el registro de las diferentes lecturas tomadas del consumo de agua potable.

4.2.13.2 Actores.

Administrador/Operario.

4.2.13.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.6

4.2.13.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción lecturas.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El actor hace click en la opción registrar lectura.
- d. El sistema muestra un menú desplegable para realizar una búsqueda de usuarios por sector.
- e. El actor pulsa en el menú desplegable y elige un determinado sector a buscar.
- f. El sistema toma la opción seleccionada y realiza una búsqueda de los clientes por sectores, además activa el siguiente menú desplegable.

- g. El actor pulsa en el siguiente menú desplegable y busca el nombre del cliente, hace click en este.
- h. El sistema muestra los datos del cliente con su respectivo contrato y activa los cuadros de texto correspondientes para que pueda ingresar los datos.
- i. El actor introduce los datos en los casilleros correspondientes y presiona el botón almacenar lectura.
- j. El sistema realiza una comprobación de los datos ingresados previamente y los almacena.

Flujos Alternativos.

- a. El sistema verifica que los campos de texto estén llenos correctamente, si existe un casillero vacío, se genera un aviso al actor de ello permitiendo que verifique los campos de texto que se encuentran vacíos y los corrija.
- b. El sistema realiza una validación de los datos ingresados, si los datos no son correctos, se avisa inmediatamente al actor de ello dándole la posibilidad que los corrija.

4.2.13.5 Pos condiciones.

Los datos de la lectura han sido almacenados en el sistema, para su posterior uso en otro proceso.

4.2.14 Especificación de Caso de Uso: Actualizar Lectura

4.2.14.1 Descripción.

Permite realizar una actualización de los datos de las diferentes lecturas tomadas del consumo de agua potable.

4.2.14.2 Actores.

Administrador/Operario.

4.2.14.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.13

4.2.14.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción lecturas.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.

- c. El actor hace clic en la opción actualizar lecturas.
- d. El sistema muestra todas las lecturas de los clientes almacenadas en el sistema.
- e. El actor verifica el nombre del cliente y la fecha de la lectura que va a ser actualizada.
- f. El actor pulsa en la opción actualizar lectura.
- g. El sistema muestra una ventana modal con los datos del cliente y la lectura a ser actualizada.
- h. El actor realiza la actualización correspondiente y pulsa en el botón actualizar.
- i. El sistema comprueba la validez de los datos y los almacena.
- j. El actor verifica que la actualización se haya generado correctamente.

Flujos Alternativos.

- a. El sistema verifica que los campos de texto estén llenos correctamente, si existe un casillero vacío, se genera un aviso al actor de ello permitiendo que verifique los campos de texto que se encuentran vacíos y los corrija.
- b. El sistema realiza una validación de los datos ingresados, si los datos no son correctos, se avisa inmediatamente al actor de ello dándole la posibilidad que los corrija.
- c. Si el actor pulsa sobre el botón cancelar, el sistema no realiza ninguna actualización y muestra nuevamente la tabla con todas las lecturas pertenecientes a sus respectivos clientes.

4.2.14.5 Pos condiciones.

N/A

4.2.15 Especificación de Caso de Uso: Eliminar Lectura

4.2.15.1 Descripción.

Permite realizar la eliminación de los datos de las diferentes lecturas tomadas del consumo de agua potable.

4.2.15.2 Actores.

Administrador/Operario.

4.2.15.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.13

4.2.15.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción lecturas.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El actor hace click en la opción eliminar lecturas.
- d. El sistema muestra todas las lecturas de los clientes almacenadas en el sistema.
- e. El actor verifica el nombre del cliente y la fecha de la lectura que va a ser eliminada.
- f. El actor pulsa en la opción eliminar lectura.
- g. El sistema muestra una ventana modal con los datos del cliente y la lectura a ser eliminada.
- h. El actor pulsa en el botón eliminar.
- i. El sistema elimina los datos de esa lectura.
- j. El actor verifica que la eliminación se realizó correctamente.

Flujos Alternativos.

- a. Si el actor pulsa sobre el botón cancelar, el sistema no realiza ninguna eliminación de datos y muestra nuevamente la tabla con todas las lecturas pertenecientes a sus respectivos clientes.

4.2.15.5 Pos condiciones.

N/A

4.2.16 Especificación de Caso de Uso: Generar Reporte Lecturas

4.2.16.1 Descripción.

Permite generar un reporte diario de los datos de las diferentes lecturas tomadas del consumo de agua potable.

4.2.16.2 Actores.

Administrador/Operario.

4.2.16.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.13

4.2.16.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción reportes.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El actor hace click en la opción lecturas.
- d. El sistema muestra todas las lecturas de los clientes almacenadas en el sistema.

- e. El actor pulsa en la opción generar reporte en formato pdf.
- f. El sistema genera el archivo en formato pdf con varias opciones.
- g. El actor tiene las opciones de imprimir archivo o guardar en varios formatos.
- h. El sistema realiza la opción elegida por el actor.

Flujos Alternativos.

N/A

4.2.16.5 Pos condiciones.

N/A

4.2.17 Especificación de Caso de Uso: Registrar Fondo Rojo

4.2.17.1 Descripción.

Permite registrar el valor del fondo rojo de cada cliente.

4.2.17.2 Actores.

Administrador/Operario.

4.2.17.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.13

4.2.17.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción fondo rojo.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El actor hace click en la opción registrar fondo rojo.
- d. El sistema muestra un menú desplegable con los sectores establecidos.
- e. El actor pulsa en el menú desplegable y elige un sector al cual pertenezca un cliente.
- f. El sistema toma el parámetro seleccionado y realiza una búsqueda de clientes por sector, y activa el siguiente menú desplegable.
- g. El actor pulsa el siguiente menú desplegable y elige el nombre del cliente.
- h. El sistema muestra los datos pertenecientes al cliente seleccionado luego activa los cuadros de texto para el ingreso de los datos pertinentes.
- i. El actor introduce los datos en los correspondientes cuadros de texto y pulsa el botón ingresar fondo rojo.

- j. El sistema comprueba la validez de los datos ingresados previamente y los almacena.

Flujos Alternativos.

- a. El sistema verifica que los campos de texto no se encuentren vacíos, si se encuentran vacíos, se genera un aviso al actor de ello permitiéndole llenar los mismos.
- b. El sistema comprueba la validez de los datos ingresados, si los datos no son correctos, se avisa al actor de ello permitiendo que los corrija.

4.2.17.5 Pos condiciones.

Los datos del fondo rojo han sido almacenados en el sistema, para una utilización posteriormente.

4.2.18 Especificación de Caso de Uso: Actualizar Fondo Rojo

4.2.18.1 Descripción.

Permite realizar una actualización de los datos del fondo rojo de un cliente registrados en el sistema.

4.2.18.2 Actores.

Administrador/Operario.

4.2.18.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.17

4.2.18.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción fondo rojo.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El actor hace click en la opción actualizar fondo rojo.
- d. El sistema muestra los datos de todos los fondos rojos registrados en el sistema.
- e. El actor verifica los datos del fondo rojo que van a ser actualizados.
- f. El actor pulsa en la opción actualizar.
- g. El sistema despliega una ventana modal con los datos del cliente y el fondo rojo.
- h. El actor actualiza los datos pertinentes en la ventana modal y presiona el botón actualizar.

- i. El sistema comprueba la valides de los datos ingresados previamente y realiza la actualización correspondiente.
- j. El actor verifica la actualización realizada previamente en el sistema.

Flujos Alternativos.

- a. El sistema verifica que los campos de texto no se encuentren vacíos, si se encuentran vacíos, se genera un aviso al actor de ello permitiéndole llenar los mismos.
- b. El sistema comprueba la valides de los datos ingresados, si los datos no son correctos, se avisa al actor de ello permitiendo que los corrija.

4.2.18.5 Pos condiciones.

N/A

4.2.19 Especificación de Caso de Uso: Eliminar Fondo Rojo

4.2.19.1 Descripción.

Permite realizar la eliminación de los datos del fondo rojo de un cliente registrados en el sistema.

4.2.19.2 Actores.

Administrador/Operario.

4.2.19.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.17

4.2.19.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción fondo rojo.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El actor hace click en la opción eliminar fondo rojo.
- d. El sistema muestra los datos de todos los fondos rojos registrados en el sistema.
- e. El actor verifica los datos del fondo rojo que van a ser eliminados.
- f. El actor pulsa en la opción eliminar.
- g. El sistema despliega una ventana modal con los datos del cliente y el fondo rojo.
- h. El actor pulsa en el botón eliminar.
- i. El sistema realiza la eliminación de los datos del fondo rojo.

- j. El actor verifica la eliminación del fondo rojo realizada previamente en el sistema.

Flujos Alternativos.

El sistema muestra una ventana modal con los datos del cliente y el fondo rojo a ser eliminados. Si el actor presiona el botón cancelar el sistema no realiza la eliminación y regresa a la tabla correspondiente a los datos de los fondos rojos.

4.2.19.5 Pos condiciones.

N/A

4.2.20 Especificación de Caso de Uso: Generar Reporte Fondo Rojo

4.2.20.1 Descripción.

Permite realizar un reporte diario de los datos del fondo rojo de los clientes registrados en el sistema.

4.2.20.2 Actores.

Administrador/Operario.

4.2.20.3 Precondiciones.

- a. Especificación de Caso de Uso 4.2.4
- b. Especificación de Caso de Uso 4.2.17

4.2.20.4 Flujo de Eventos

Flujo Básico.

- a. El actor pulsa en el menú la opción reportes.
- b. El sistema muestra las diferentes opciones con que cuenta este menú.
- c. El actor hace click en la opción fondo rojo.
- d. El sistema muestra los datos de todos los fondos rojos registrados en el sistema.
- e. El actor pulsa en la opción reporte en formato de archivo pdf.
- f. El sistema genera un reporte en archivo pdf y muestra al actor.
- g. El actor verifica el reporte y realiza una de las operaciones pertinentes como son imprimir o guardar en un formato diferente.
- h. El sistema realiza la operación solicitada previamente.

Flujos Alternativos.

N/A

4.2.20.5 Pos condiciones.

N/A

5. CAPÍTULO V.- FASE DE CONSTRUCCIÓN

5.1 Documento de la arquitectura del software.

5.1.1 Introducción

Uno de los desarrollos más importantes dentro de la cimentación del software es el desarrollo de la arquitectura de software, que permite representar la estructura del sistema, sirviendo de comunicación entre las personas involucradas en el desarrollo y ayudando a realizar diversos análisis que orienten el proceso de toma de decisiones.

Este documento provee al usuario especializado una vista de la arquitectura del Sistema de GESTIÓN DE PROCESOS PARA UNA JUNTA DE AGUA POTABLE UTILIZANDO LAS TECNOLOGÍAS DE SOFTWARE LIBRE.

La plantilla del presente documento se basó en las especificaciones de RUP ^[13] para el documento de arquitectura de software.

5.1.2 Propósito

El presente documento proporciona una representación de la arquitectura del sistema, haciendo uso de diversas perspectivas arquitectónicas para representar varios aspectos del sistema. Se realiza con el fin de documentar las decisiones de arquitectura significativas que se han tomado en el sistema.

Además se muestra el funcionamiento de la arquitectura MVC ^[14] que se usará.

5.1.3 Alcance

Este documento se centra en el desarrollo de la vista lógica y la implementación correspondiente. Además también se realiza una identificación de los varios componentes que pertenecen a cada una de las vistas.

5.1.4 Definiciones, Acrónimos y Abreviaturas.

Se brindan definiciones y acrónimos de términos usados en el presente documento que necesiten de alguna explicación para su correcta interpretación.

Ver documento Glosario.

5.1.5 Referencias

- Artefacto Modelo de Casos de Uso.
- Artefacto Glosario.

¹³ Rational Unified Process (Wikipedia, 2012)

¹⁴ Modelo Vista Controlador (Wikipedia, 2012)

5.1.6 Resumen

En los presentes apartados de este documento se muestra la arquitectura del software que se va a desarrollar. Para tal motivo se presenta de una manera clara los diferentes casos de uso que más representa la arquitectura del sistema, empleando un lenguaje fácil de entender y directo, así como gráficos y vistas de acuerdo a la metodología que se está utilizando.

5.2 Representación Arquitectónica

Este documento muestra la arquitectura como una serie de vistas: vistas de los casos de uso, vistas de los procesos, vista de despliegue y vista de ejecución. Para el desarrollo de los modelos se ha utilizado el lenguaje UML¹⁵ y para el diseño de los diagramas se ha utilizado el software Visual Paradigm y Power Designer.

5.3 Objetivos y Restricciones de Arquitectura

5.3.1 Objetivos

- Implementar la arquitectura MVC en el desarrollo del sistema, por contar con un funcionamiento eficiente, provee de las funciones y propiedades adecuadas.
- Realizar un análisis del diseño para verificar que tiene un alto grado de rendimiento y bajo acoplamiento entre cada uno de los componentes, para de esta manera sea fácil la manipulación y remplazo de los mismos.
- Desarrollar la vista lógica de los componentes que conformaran la arquitectura del sistema para comprobar que los componentes sean altamente reutilizables.

5.3.2 Restricciones

- Los requerimientos de rendimiento convenidos en el documento de visión, deben de ser tomados en cuenta como parte de la arquitectura del sistema a implementar
- La aplicación será construida bajo el patrón de diseño MVC, por lo que la mayoría de su funcionalidad se desempeñará con este esquema.
- El sistema usará como motor de Base de Datos a PostgreSQL. Siendo necesario la elaboración de una copia de rescate de las tablas para no afectar la información almacenada antes de poner en total funcionamiento al sistema.
- Para su desarrollo se utilizará el framework JSF 2.1

¹⁵ Lenguaje Unificado de Modelado (Wikipedia, 2012)

- El sistema al estar desarrollado con la tecnología JSF, es necesario tener instalado el JRE 1.7 o superior para que los usuarios puedan hacer uso del sistema.

5.4 Vista dinámica

En la vista dinámica se tiende a analizarse como pequeñas fragmentos del sistema, esto se refiere a, contextos individuales de operaciones. Personifica las interacciones de los objetos de un sistema.

Podríamos decir que representara como el sistema responderá a las acciones de los usuarios, como los datos son enviados del almacenamiento a la vista del usuario y como los objetos son creados y manejados: lo cual los transforma en los diagramas más manejados en los proyectos, ya que los mismos son los que muestran directamente particularidades específicas requeridas en el código final.

5.4.2 Diagramas de Secuencia

Los diagramas de secuencia muestran la interacción entre los diferentes objetos y el orden secuencial en el que se desarrollan dichas interacciones, esto nos muestra como los objetos se comunican entre sí.

En el presente diagrama de secuencia se detalla la secuencia de interacción del mecanismo de implementación MVC.

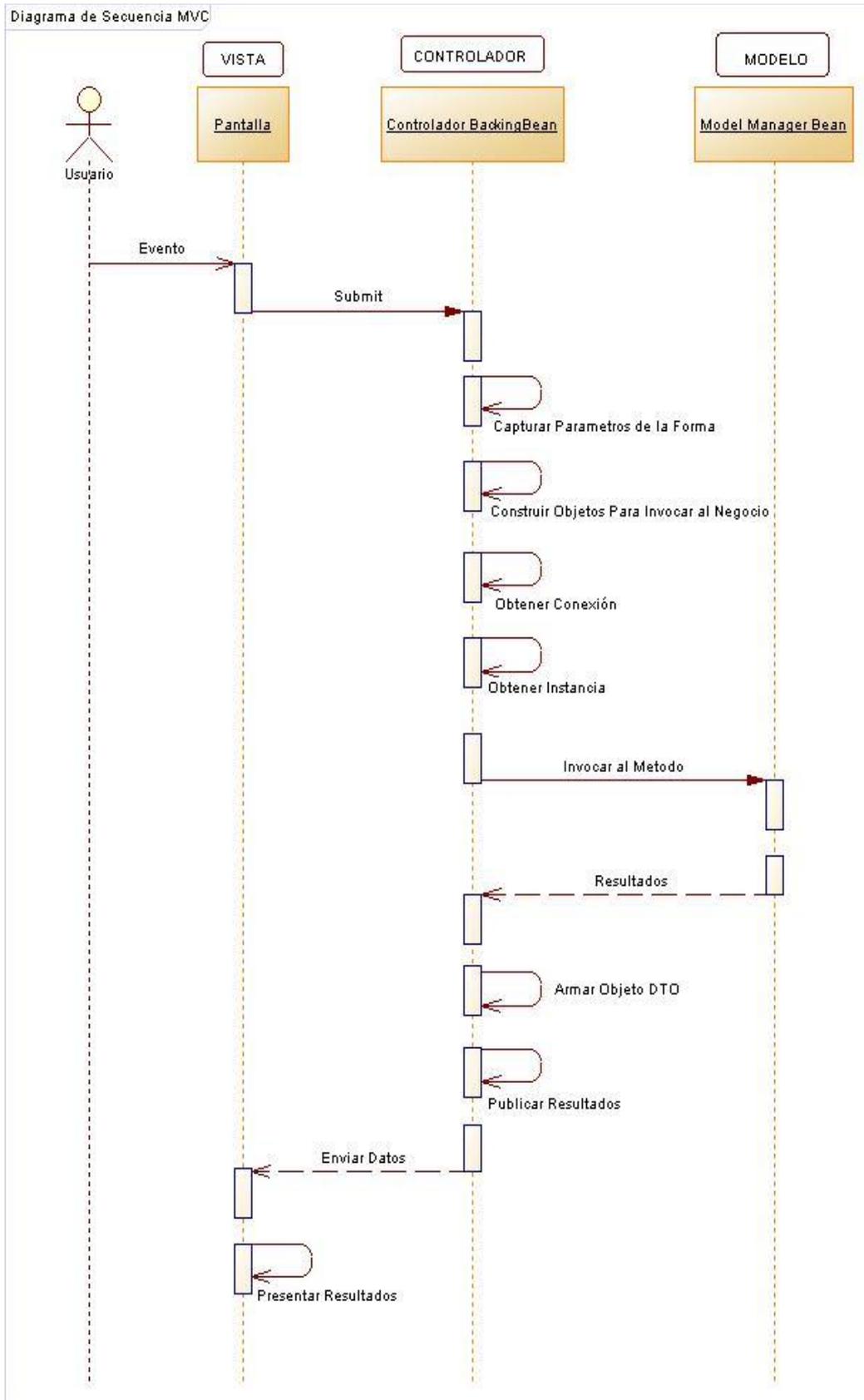


Ilustración 14.- Diagrama de Secuencia de la Arquitectura MVC
Fuente: Andrés Cheza

5.5 Vista Lógica

5.5.1 Introducción

La presente vista muestra las diferentes unidades lógicas que componen la arquitectura del sistema SISFAC, dicho sistema manejará un patrón de arquitectura de diseño de software llamado MVC.

Este es un patrón de diseño de software que tiene la facilidad de separar los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes diferentes.

En el primer refinamiento realizado reside en la descomposición en subsistemas. Los subsistemas personifican cortes verticales al diseño del sistema. Cada subsistema consiste en el agrupamiento de varias funcionalidades relacionadas entre si y posee la capacidad de desempeñarse como un sistema en sí mismo.

5.5.2 Descomposición de subsistemas

La presente explicación va relacionada con el framework JSF ^[16] y como interactúa con el patrón de arquitectura MVC.



Ilustración 15.- Estructura MVC.
Fuente: <http://blog.cubenube.com/>

- La capa del modelo es la parte encargada de la obtención, procesamiento, y almacenamiento de los datos según la acción transmitida desde el controlador. Una vez procesados esos datos, devuelve la información de respuesta al controlador en caso de ser necesario. Dichos datos pueden tener diferentes fuentes, ya sea una base de datos, ficheros de texto, ficheros XML, o cualquier

¹⁶ JavaServer Faces (Wikipedia, 2012)

otro sistema y/o combinación de los mismos. Es la única capa que puede tener interacción con los sistemas de almacenamiento.^[17]

- La vista recibe por parte del controlador los nuevos datos a mostrar, y los representa de forma gráfica para mejor entendimiento del usuario y pueda seguir interactuando con la aplicación. En el caso del "cloud computing" y páginas web, es el que genera el código XHTML, CSS, javascript, y cualquier otro lenguaje necesario, para mostrar dichos datos de una forma entendible y en su medida, atractiva al usuario.
- El controlador que en este caso serían los manager beans o backing beans son los encargados de la aplicación, deciden que hacer según interactúe el usuario con la aplicación. Es el encargado de gestionar la seguridad de la aplicación, control de errores, responder a las acciones solicitadas por el usuario invocando a los diferentes modelos y transmitir los datos devueltos a la vista para que los presente al usuario.

5.5.6 Vista de Procesos

En la capa del modelo tenemos la capa de mapeo objeto – relacional. Esta capa surge de la necesidad de tener un nivel de mayor abstracción del manejo de los datos. Teniendo esta capa, se podrá encapsular al usuario del diseño, implementación y uso directo de la base de datos, lo que hace que se puedan simplificar muchas etapas del proceso de desarrollo.

Además al utilizar una herramienta que nos permita realizar mapeos de este tipo, se podrá establecer una clara y concisa separación del paradigma relacional (bases de datos) y el paradigma orientado a objetos (clases), lo cual nos permitirá tener un mayor grado de definición para cada una de las capas de la arquitectura.

Para nuestra aplicación, hemos decidido utilizar el framework Hibernate.



Ilustración 16.- Diagrama Hibernate.

Fuente: <http://upload.wikimedia.org/wikipedia/commons/thumb>

¹⁷ Cubenube. [MVC]. (Octubre 17, 2012) <http://blog.cubenube.com/2011/11/la-arquitectura-modelo-vista.html>

Hibernate es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones.¹⁸

A continuación se presentaran algunas de las razones por las que se escogió Hibernate como framework para el mapeo objeto – relacional:

- ❖ Es open source
- ❖ Es un framework maduro, ya que es uno de los más utilizados actualmente con excelentes resultados.
- ❖ Hibérnate da un completo soporte al modelo de programación orientada a objetos, lo cual es una ventaja en el desarrollo de este proyecto ya que este se hará sobre JAVA.
- ❖ Ofrece un lenguaje natural para las búsquedas en las bases de datos (HSQL) que es muy similar al que hemos manejado (SQL).
- ❖ Maneja XML para los mapeos, lo que hace que estos sean de fácil entendimiento por la estructura que este maneja.

La capa de la vista o presentación, es la encargada de interactuar con el usuario, por ello se debe tener especial cuidado para desarrollarla.

Para nuestro proyecto, esta interfaz va a estar desarrollada sobre JSF con los frameworks para interfaces graficas como son ICEfaces y PRIMEFACES.



Ilustración 17.- Diagrama ICEfaces.

Fuente: <http://www.primefaces.org/showcase/images>

Es un framework de código abierto para construir aplicaciones web con AJAX tipo RIA (Rich Internet Application). Permite al programador incluir una serie de Ajax-tags en sus

¹⁸ Wikipedia. [Online]. [Octubre 17, 2012] <http://es.wikipedia.org/wiki/Hibernate>

JSP o xhtml de tal manera que el código Ajax es generado por el propio framework automáticamente.^[19]



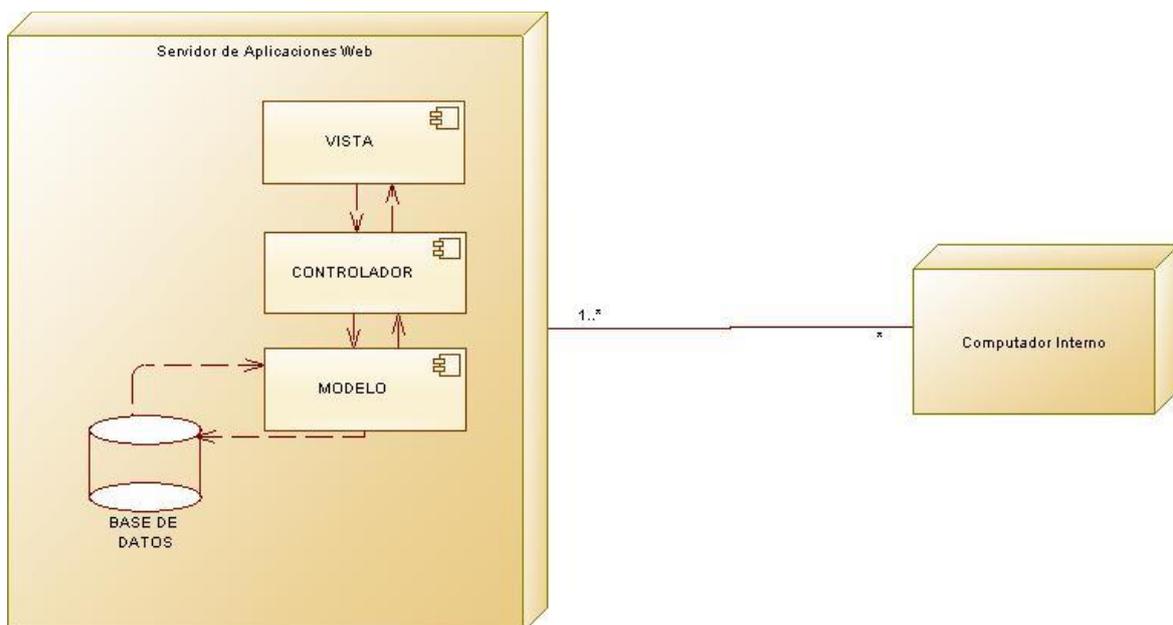
Ilustración 18.- Diagrama PRIMFACES
Fuente: <http://primefaces.org/>

PrimeFaces es un componente para JavaServer Faces (JSF) de código abierto que cuenta con un conjunto de componentes ricos que facilitan la creación de las aplicaciones web. Primefaces está bajo la licencia de Apache License V2. Una de las ventajas de utilizar Primefaces, es que permite la integración con otros componentes como por ejemplo RichFaces.^[20]

La capa del controlador está compuesta por los Backing beans, los cuales contienen una serie de acciones que permiten generar peticiones al modelo y dar respuestas a la capa de vista o presentación.

5.7 Vista de Despliegue

De acuerdo al diagrama de despliegue que presentamos a continuación muestra de una manera gráfica los nodos que conforman el sistema, con su respectiva descripción indicando la localización de las tareas en los nodos físicos.



¹⁹ Ecuared. [Online]. [Octubre 17, 2012] <http://www.ecured.cu/index.php/ICEfaces>

²⁰ Wikipedia. [Online]. [Noviembre 24, 2012] <http://es.wikipedia.org/wiki/Primefaces>

Ilustración 19.- Diagrama de Despliegue.
Fuente: Andrés Cheza

5.7.1 Servidos de Aplicaciones Web

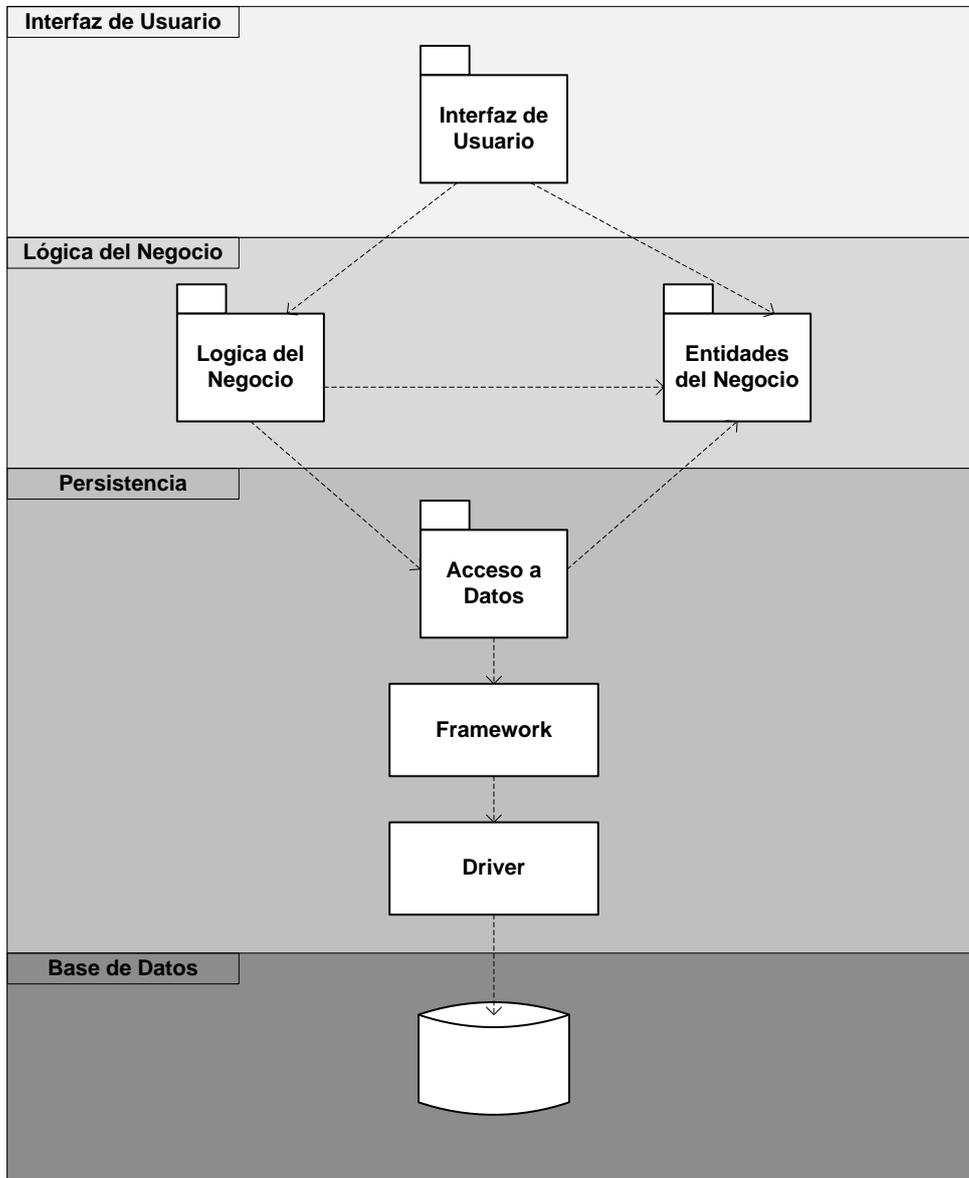
En este servidor se encuentra instalado el gestor de base de datos que nos permitirá crear la base de datos para el sistema, además contendrá el software necesario para el buen funcionamiento del sistema de Facturación y Registro de la Información (SISFAC).

5.7.1 Computador Interno

Es la computadora que utilizarán los usuarios registrados en el sistema.

5.8 Diseño en Capas

En la presente ilustración podemos apreciar la distribución de los diferentes paquetes que contiene cada una de las capas del sistema.



5.8.1 Capa Interfaz de Usuario

En esta capa se encuentra el paquete de interfaz de usuario, el mismo que contiene todos los archivos necesarios: como son ventanas, formularios, etc. para que el usuario pueda interactuar con el sistema.

5.8.2 Capa Lógica del Negocio

Esta capa contiene los diferentes paquetes de servicio de negocio y entidades de negocio. Contiene la lógica para el manejo y procesamiento de las operaciones del negocio.

5.8.3 Capa de Persistencia

En esta capa se encuentra el paquete de objetos de acceso de datos (DAO), los DTO y los archivos XML necesarios, que nos genera una interfaz transparente para la interacción con el framework el cual gestiona la interfaz con el driver enviando sentencias para interactuar con la base de datos.

5.9 Vista de Datos

Para la vista de datos se procederá a realizar una estructura de los datos desarrollando los siguientes modelos de datos:

5.9.1 Modelo Relacional

En el modelo relacional todos los datos son almacenados en relaciones, como cada relación es un conjunto de datos, el orden de almacenen no tiene mucha importancia. De acuerdo a lo descrito se puede apreciar que tiene la ventaja de que es más fácil de poder entender y de utilizar por los usuarios con poca experiencia.

En este modelo se considera la base de datos como una recopilación de relaciones. De una manera sencilla, una relación constituye una tabla que es un conjunto de filas, cada fila es un conjunto de campos y cada campo representa un valor que descifrado representa el mundo real. También cada fila se puede denominar una tupla o un registro y cada columna se le puede llamar también campo o atributo.

5.9.2 Modelo Físico

Un modelo físico de los datos es una muestra de un diseño de los datos que considere las instalaciones y los apremios de un sistema de gerencia dado de base de datos. En un ciclo de vida de un proyecto se deriva normalmente de un modelo lógico de los datos,

aunque puede ser encaminado de una puesta en práctica dada de la base de datos. En un modelo físico completo de los datos incluirá todos los artefactos de la base de datos solicitados para crear relaciones entre las tablas o para alcanzar metas del trabajo.

5.10 Tamaño y Performance

5.10.1 Tiempo de respuesta en el acceso a la Base de Datos

El presente sistema suministrará accesos a la base de datos con un tiempo de respuesta menor e igual a 5 segundos.

5.10.2 Tiempo de respuesta de transacciones

El programa no demorará más de 7 minutos en generar una distribución óptima para los cortes haciendo uso del algoritmo elegido y guardándolo en la base de datos.

5.10.3 Espacio en disco para el cliente

El espacio en disco obligatorio para la parte del cliente al menos deberá contar como mínimo 700 MB de espacio libre para tener un óptimo funcionamiento, esto incluye tanto el tamaño del software como el JRE 1.7.

5.10.4 Espacio en disco para el servidor de base de datos

En lo referente al servidor de base de datos el espacio en disco obligatorio deberá contar como mínimo 1 GB libres en disco para su adecuado funcionamiento.

La arquitectura elegida apoya las exigencias de latencia y capacidad en disco en la puesta en práctica de una arquitectura cliente servidor. La fracción del cliente solo se pone en ejecución en las PC locales de los distintos ambientes dentro de la junta de agua. Los componentes se han delineado para asegurarse de obtener requisitos mínimos de disco y memoria en el lado de las PC del cliente.

5.11 Calidad

Para un excelente aprovechamiento de la arquitectura de software se dan las siguientes exigencias de calidad:

5.11.1 Usabilidad

El presente sistema admitirá un manejo intuitivo por parte de los usuarios.

5.11.2 Eficiencia

El programa diferirá menos de 5 minutos en crear una distribución óptima para los cortes haciendo uso del algoritmo elegido.

5.11.3 Seguridad

El sistema admitirá el acceso a funcionalidades dependiendo del perfil con que cuente el usuario que ingresa al sistema, realizando una validación de su ingreso a través de una clave.

5.11.4 Confiabilidad

El sistema tomará en cuenta que la información ingresada en él sea válida, para tal motivo mostrará mensajes que expliquen al usuario acerca de los errores que éste pudiera generarse y de aquellos que pueda generar el mismo sistema.

5.11.5 Mantenimiento

El sistema tendrá la flexibilidad, de en un futuro ser mantenido.

5.12 Arquitectura del Proyecto

5.12.1 Introducción

De acuerdo a lo que se ha visto en el desarrollo del documento, tenemos que la base arquitectónica se presenta en la siguiente manera: a continuación se describe la arquitectura de la aplicación del sistema de GESTIÓN DE PROCESOS PARA UNA JUNTA DE AGUA POTABLE UTILIZANDO LAS TECNOLOGÍAS DE SOFTWARE LIBRE. El desarrollo de un sistema con una gran cantidad de software demanda que este sea visto desde otras perspectivas, diferentes usuarios tanto como del equipo de desarrollo como también del usuario final.

Asimismo se debe tomar en cuenta que una arquitectura no solo debe centrarse específicamente en la estructura y el comportamiento, sino que englobe temas como el uso, funcionalidad, rendimiento, capacidad de adaptación, reutilización, capacidad para ser comprendida, restricciones, compromisos entre alternativas, así también como aspectos estéticos.

5.12.2 Paquetes de Análisis

Los paquetes de análisis nos permiten tener una organización adecuada del modelo de análisis en grupos que sean adaptables. En la presente tabla se especifican los paquetes que se va analizar:

Módulo	Paquete de análisis	Caso de Uso
Sistema web	Usuario	Ingresar al Sistema
		Validar Usuario
		Mostrar Menú Principal del Sistema
	Contratos	Solicitar el Servicio de Agua Potable
		Crear Contrato
		Registrar Cobro por El Servicio de Agua Potable
		Generar Factura por el Servicio de Agua Potable
		Imprimir Factura por el Servicio de Agua Potable
		Actualizar Contrato
		Eliminar Contrato
		Generar Reporte de Contratos
	Lecturas	Registrar Lectura
		Actualizar Lectura
		Eliminar Lectura
		Generar Reporte Lecturas
	Fondo Rojo	Registrar Fondo Rojo
		Actualizar Fondo Rojo
		Eliminar Fondo Rojo
		Generar Reporte Fondo Rojo
	Carta de Pago	Pagar Consumo Agua Potable
		Verificar Lectura
		Generar Carta de Pago
		Imprimir Carta de Pago
		Actualizar Carta de Pago
		Eliminar Carta de Pago
		Generar Reporte Carta de Pago
	Suspensión del Servicio	Generar Citaciones
		Imprimir Citaciones
		Registrar Suspensión
		Actualizar Suspensión
		Eliminar Suspensión
		Generar Reporte Suspensión
	Reconexión del Servicio	Verificar Suspensión
		Registrar Pago Total Adeudado
		Registrar Re Conexión del Servicio
		Generar Factura Reconexión Servicio

		Imprimir Factura Reconexión Servicio
		Entregar Factura Reconexión Servicio
		Actualizar Reconexión del Servicio
		Eliminar Reconexión del Servicio
		Generar Reporte de Reconexión del Servicio
	Convocatorias	Registrar Convocatoria
		Imprimir Convocatoria
		Entregar Convocatoria
		Actualizar Convocatoria
		Eliminar Convocatoria
		Generar Reporte Convocatoria
	Prestamos	Solicitar Préstamo
		Verificar Contrato
		Registrar Préstamo
		Entregar Préstamo
		Actualizar Préstamo
		Eliminar Préstamo
		Generar Reporte Préstamo
	Pagos	Pagar Préstamo
		Verificar Préstamo
		Registrar Pago Préstamo
		Generar Factura Pago Préstamo
		Imprimir Factura Pago Préstamo
		Entregar Factura Pago Préstamo
		Actualizar Pago Préstamo
		Eliminar Pago Préstamo
		Generar Reporte Pago Préstamo
		Pagar Intereses
		Registrar Pago Intereses
		Generar Factura Pago Intereses
		Imprimir Factura Pago Intereses
		Entregar Factura Pago Intereses
		Actualizar Pago intereses
		Eliminar Pago intereses
		Generar Reporte Pago intereses
		Pagar Multa
		Verificar Multa
		Registrar Pago Multa
		Generar Factura Pago Multa
		Imprimir Factura Pago Multa
		Entregar Factura Pago Multa
		Actualizar Pago Multa
		Eliminar Pago Multa

Tabla 37.- Descripción de los paquetes de análisis.

Fuente: Andrés Cheza

En el presente diagrama se muestra las relaciones de dependencias entre los paquetes de análisis.

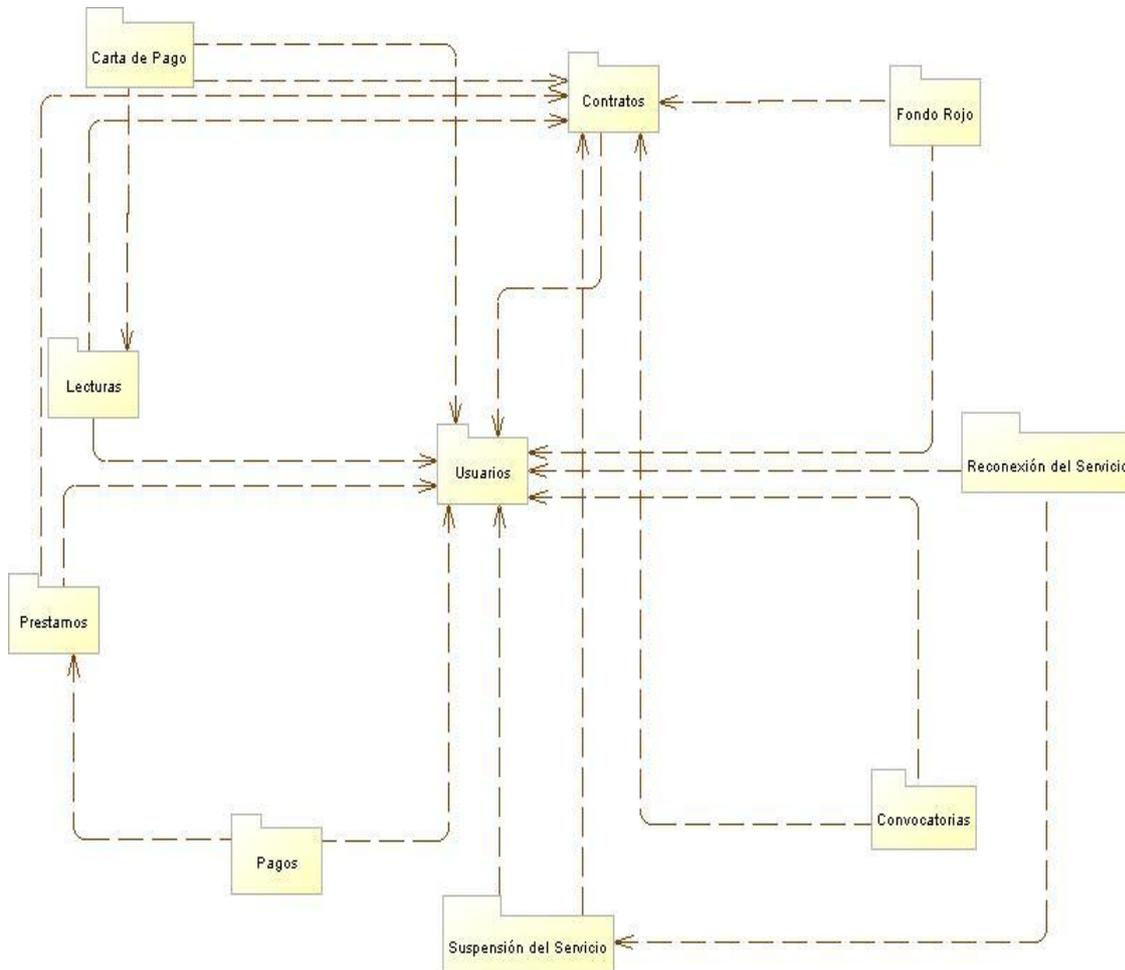


Ilustración 21.- Dependencia entre Paquetes del Sistema Web

Fuente: Andrés Cheza

5.12.3 Vista Dinámica

Los diagrama de secuencia instruyen la interacción entre los objetos y el orden secuencial en el que se desarrolla dichas interacciones, esto es como se comunican los objetos entre sí.

En los diferentes casos de uso se modelan las particularidades del sistema y se desarrollan escenarios, los diagramas de secuencias nos suministran un camino a partir de los escenarios para describir las operaciones en una forma más detallada. A continuación se tiene un detalle de los diagramas de secuencias considerados para los casos de uso de la anterior sección.

NOTA.- Los primeros veinte diagramas de secuencia de uso de prioridad, servirán para el correspondiente análisis durante el presente proyecto, los demás se encuentran especificados en el CD del proyecto, carpeta Anexos documento diagrama de secuencia.

5.12.4 Diagramas de Secuencia

5.12.4.1 Diagrama de Secuencia Caso de Uso: Ingresar al Sistema

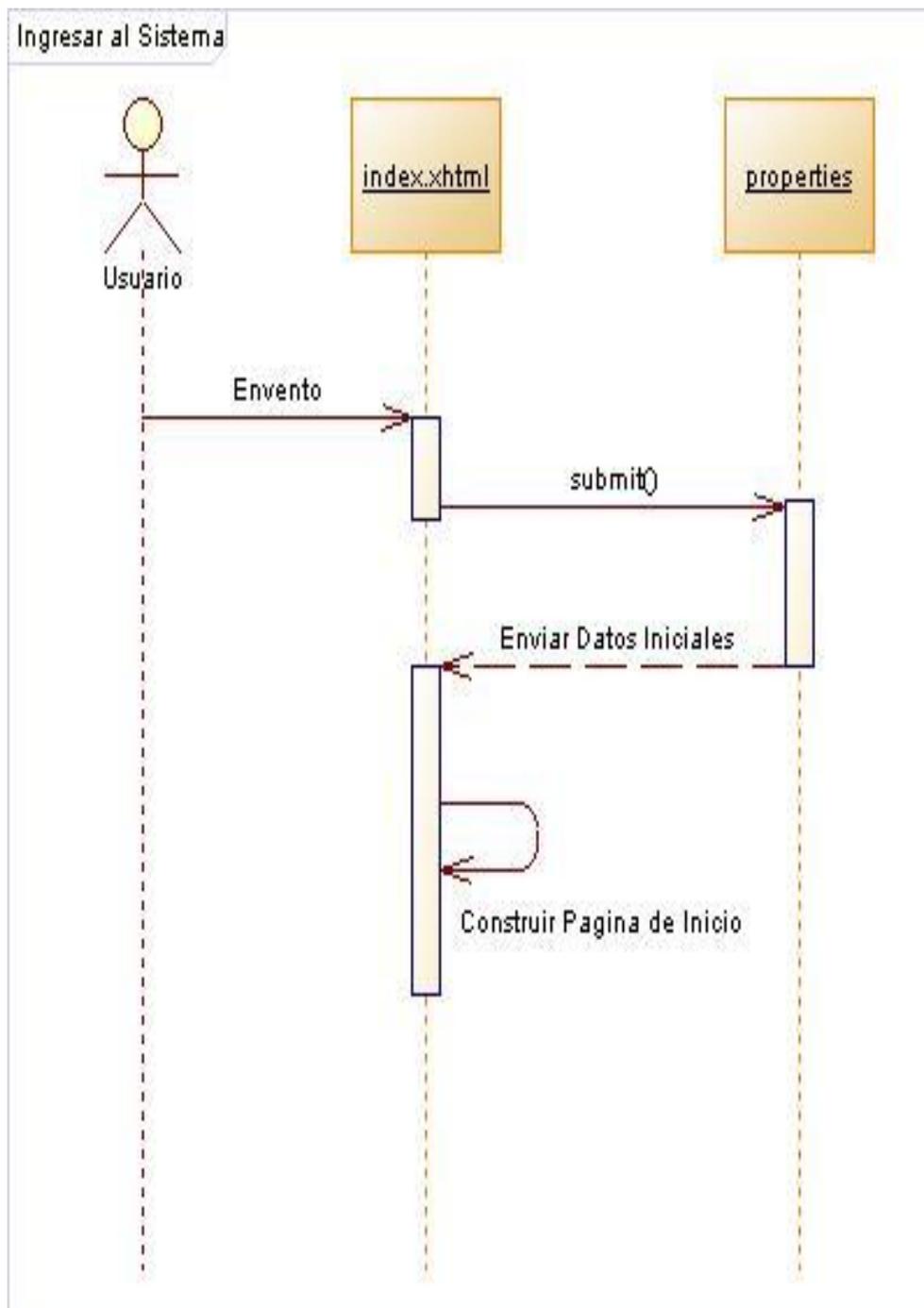


Ilustración 22.- Diagrama de Secuencia Caso de Uso: Ingresar al Sistema
Fuente: Andrés Cheza

En este diagrama analizamos como se interactúa en el sistema al momento de ingresar un usuario al sistema.

5.12.4.2 Diagrama de Secuencia Caso de Uso: Validar Usuario

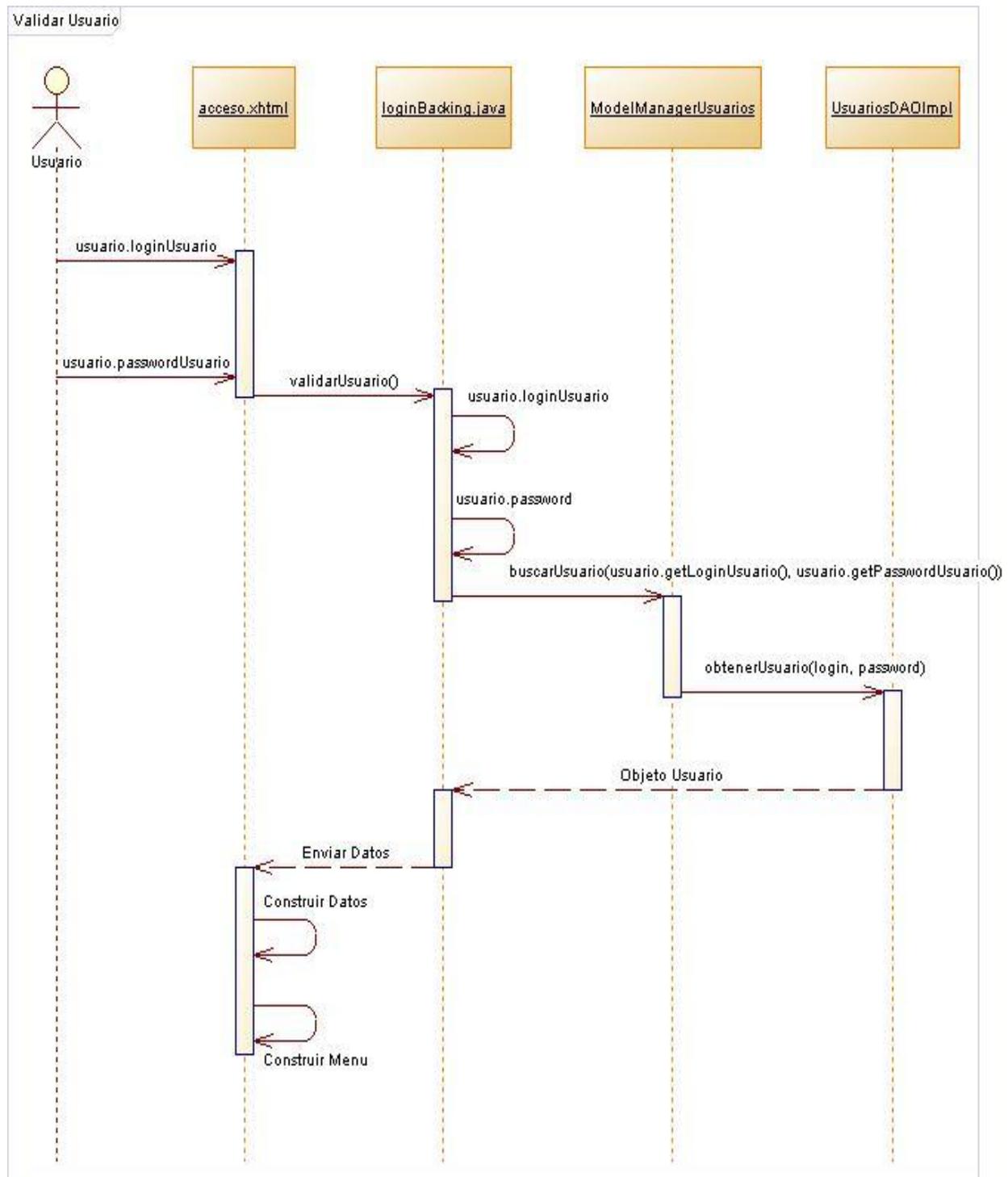


Ilustración 23.- Diagrama de Secuencia Caso de Uso: Validar Usuario
Fuente: Andrés Cheza

En la presente ilustración vemos cómo funciona la validación de un usuario al ingresar los datos en el formulario y enviarlos al sistema.

5.12.4.3 Diagrama de Secuencia Caso de Uso: Mostrar Menú Principal del Sistema

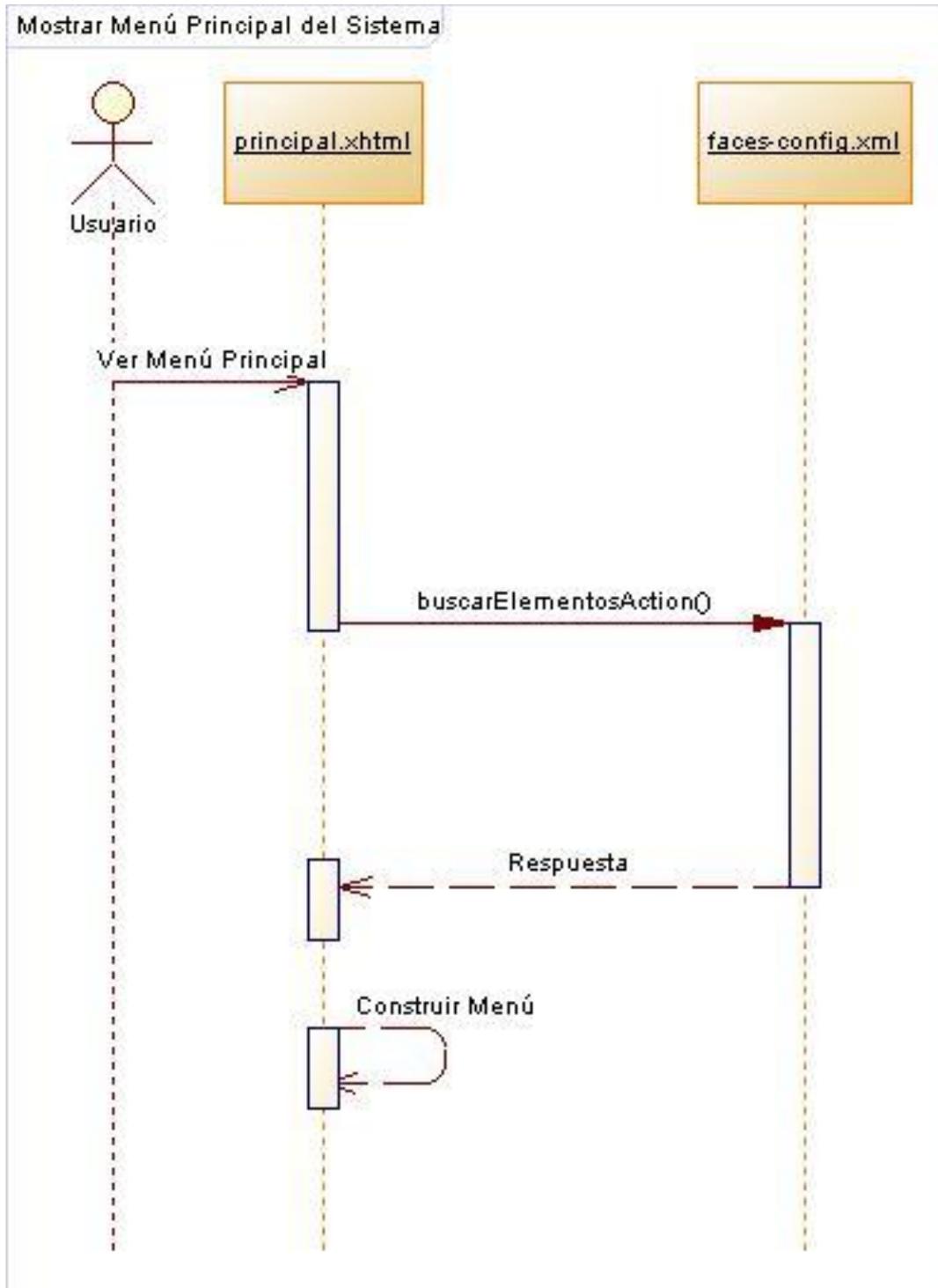


Ilustración 24.- Diagrama de Secuencia Caso de Uso: Mostrar Menú Principal del Sistema
Fuente: Andrés Cheza

En el presente diagrama vemos la interacción que realiza el sistema para poder visualizar el menú principal del sistema.

5.12.4.4 Diagrama de Secuencia Caso de Uso: Solicitar el Servicio de Agua Potable

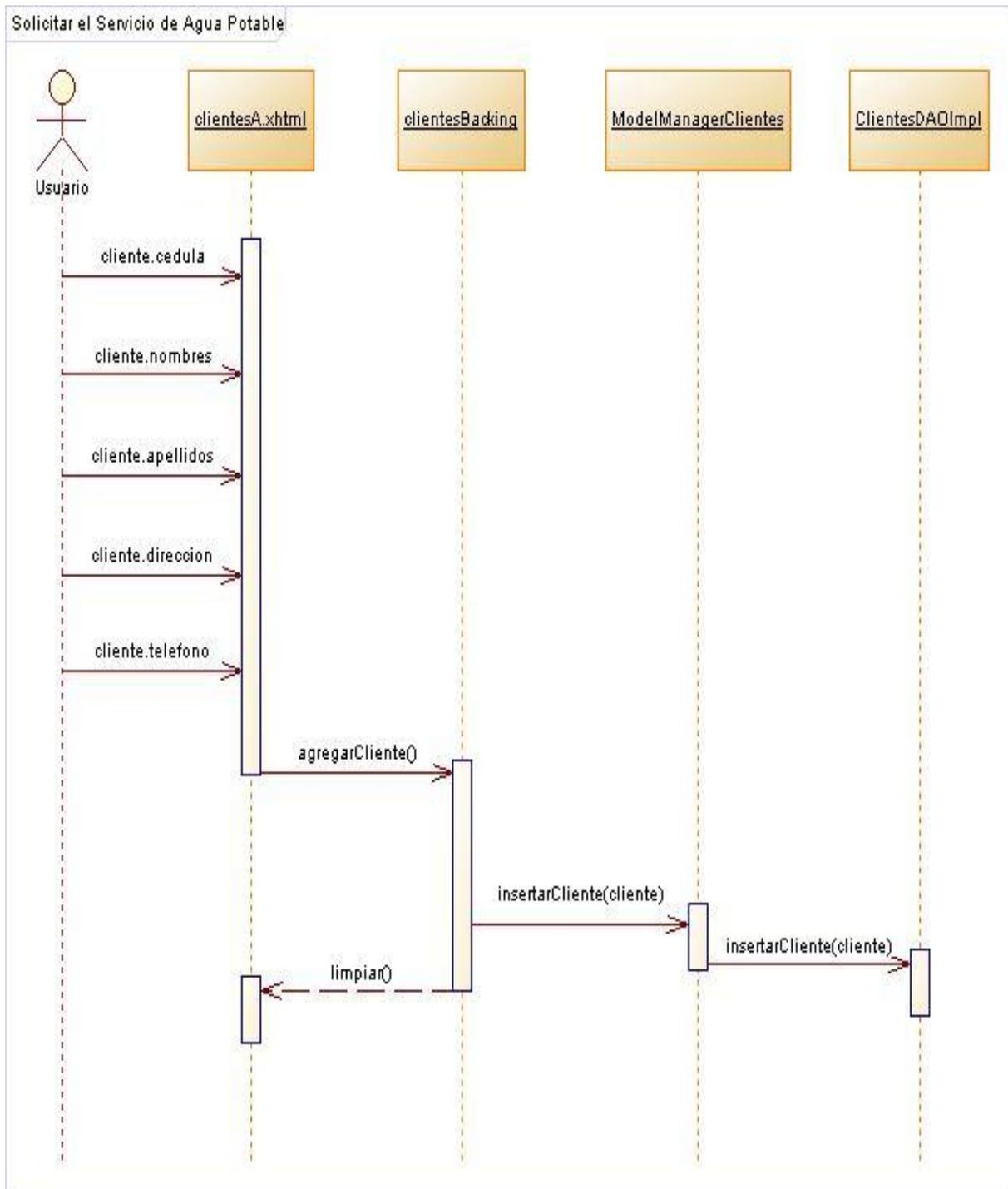


Ilustración 25.- Diagrama de Secuencia Caso de Uso: Solicitar el Servicio de Agua Potable
Fuente: Andrés Cheza

En la ilustración podemos apreciar como el usuario interactúa con el sistema para registrar los datos de un nuevo cliente.

5.12.4.5 Diagrama de Secuencia Caso de Uso: Crear Contrato

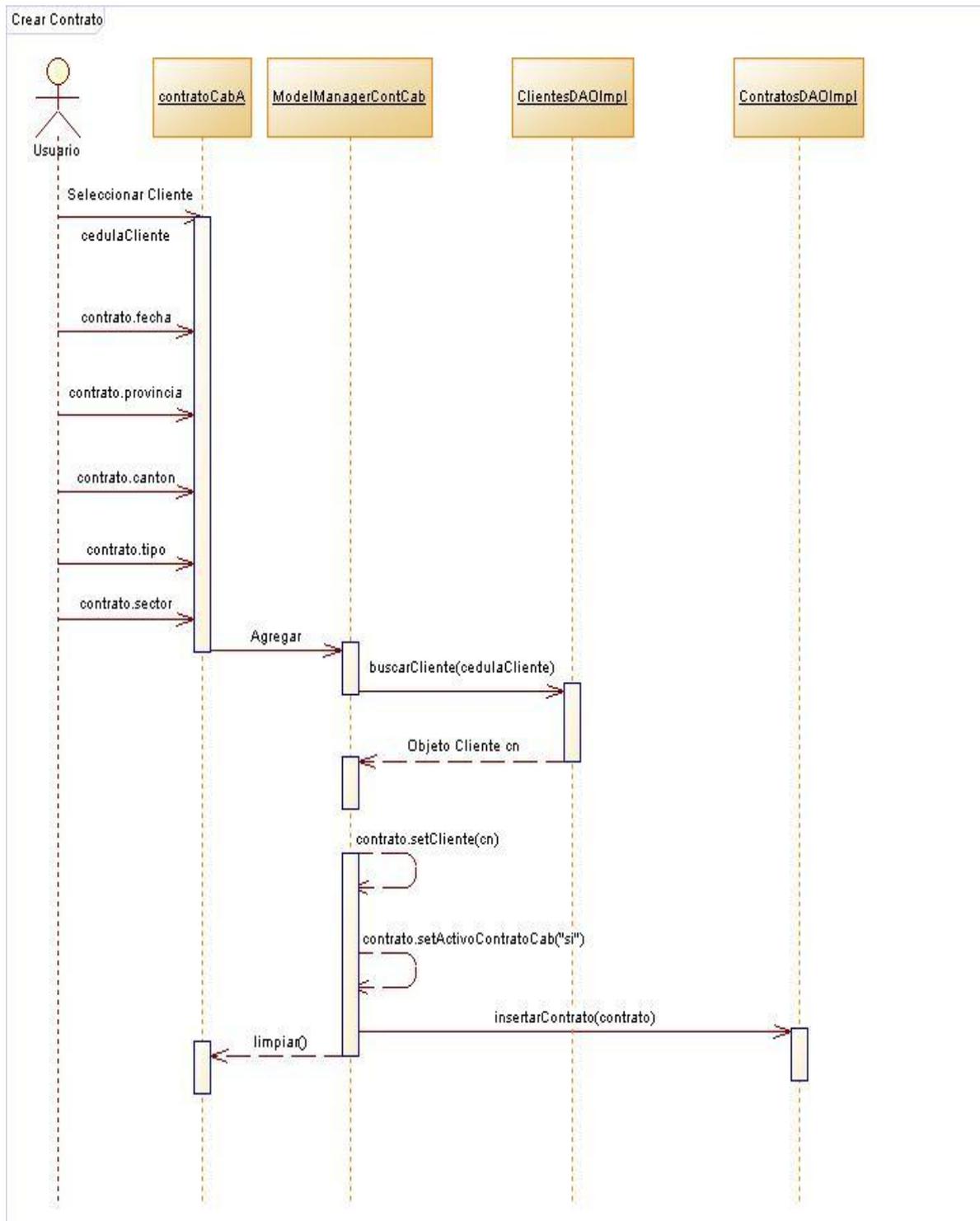


Ilustración 26.- Diagrama de Secuencia Caso de Uso: Crear Contrato
Fuente: Andrés Cheza

En el diagrama se puede ver como interactúa el usuario con el sistema para agregar un nuevo contrato.

5.12.4.6 Diagrama de Secuencia Caso de Uso: Registrar Cobro por El Servicio de Agua Potable

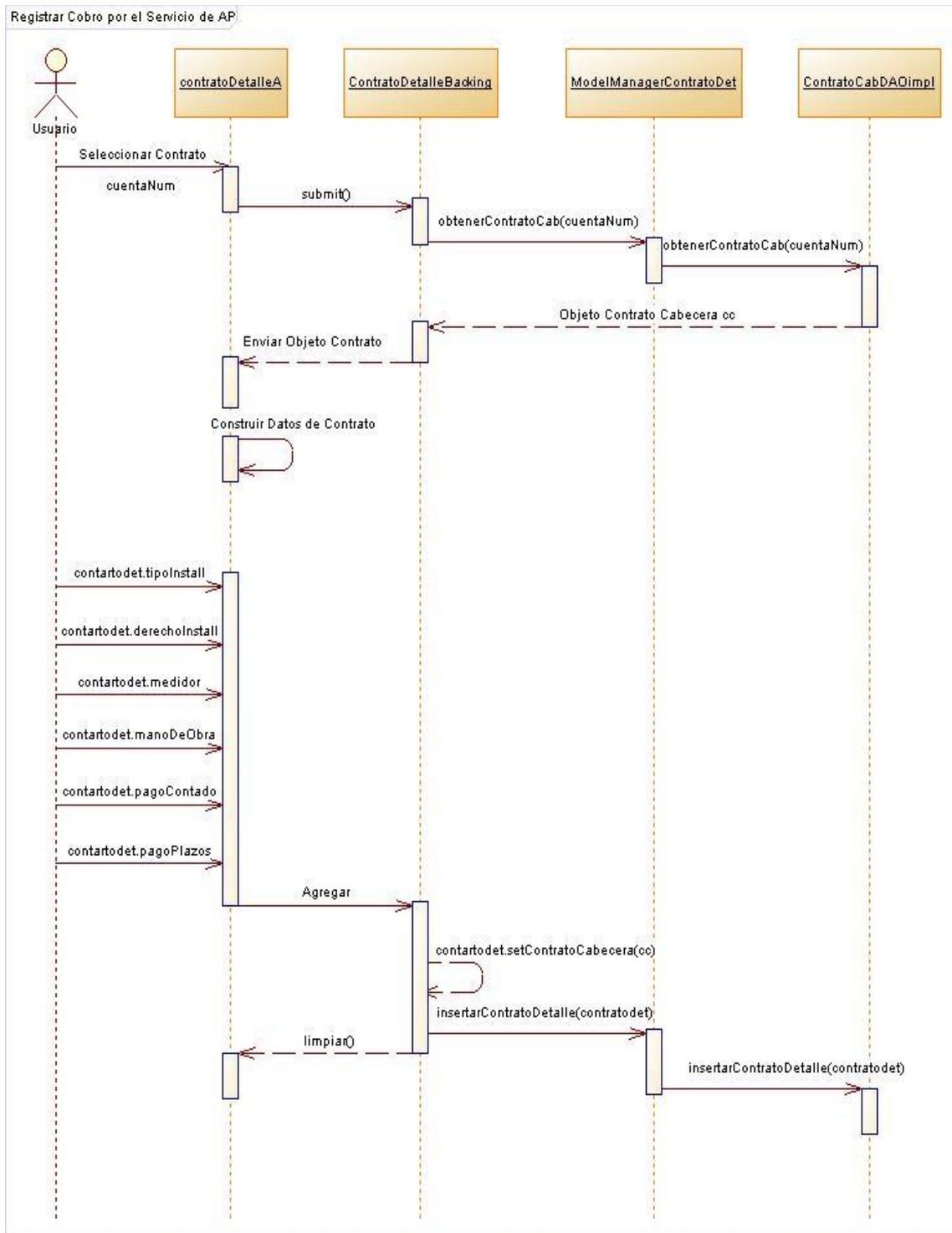


Ilustración 27.- Diagrama de Secuencia Caso de Uso: Registrar Cobro por El Servicio de Agua Potable
Fuente: Andrés Cheza

En la presente ilustración podemos ver cómo funciona gráficamente un proceso para el cobro por el servicio de agua potable.

5.12.4.7 Diagrama de Secuencia Caso de Uso: Generar Factura por el Servicio de Agua Potable

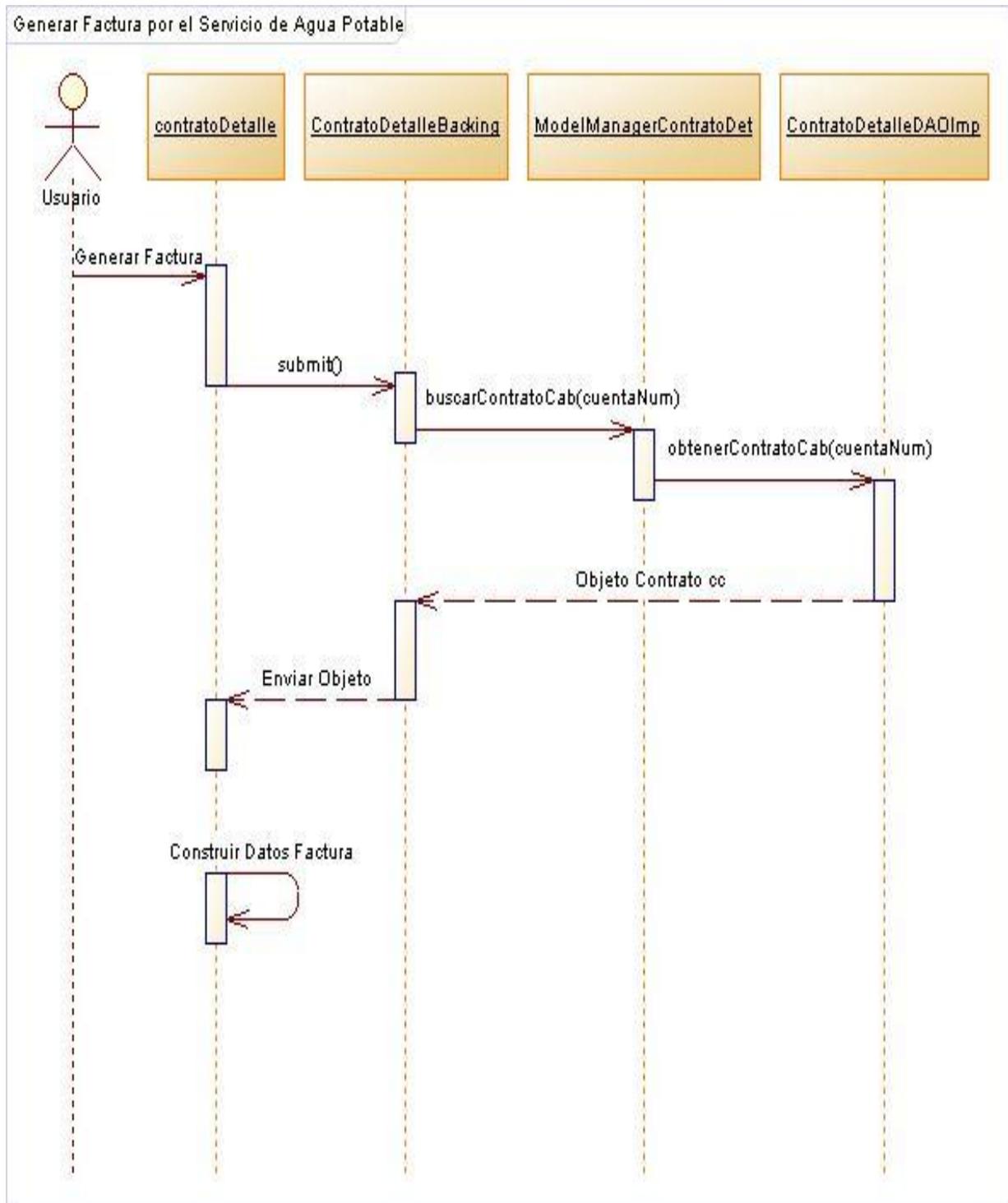


Ilustración 28.- Diagrama de Secuencia Caso de Uso: Generar Factura por el Servicio de Agua Potable
Fuente: Andrés Cheza

En el diagrama presentado se puede apreciar cómo funciona el proceso para generar la factura de coro por el servicio de agua potable.

5.12.4.8 Diagrama de Secuencia Caso de Uso: Imprimir Factura por el Servicio de Agua Potable

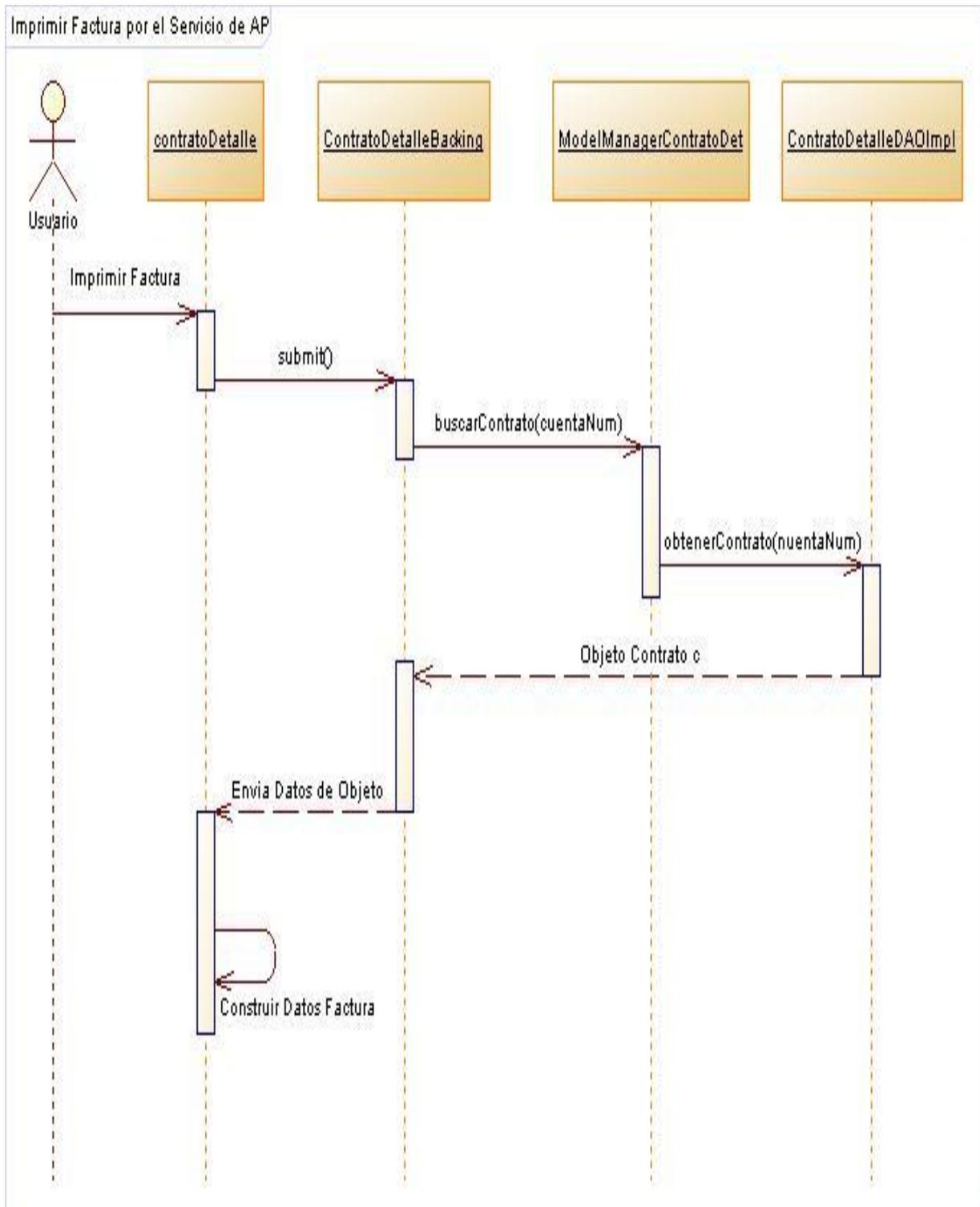


Ilustración 29.- Diagrama de Secuencia Caso de Uso: Imprimir Factura por el Servicio de Agua Potable
Fuente: Andrés Cheza

El diagrama muestra cómo se realiza el proceso de impresión de la factura por concepto del servicio de agua potable.

5.12.4.9 Diagrama de Secuencia Caso de Uso: Actualizar Contrato

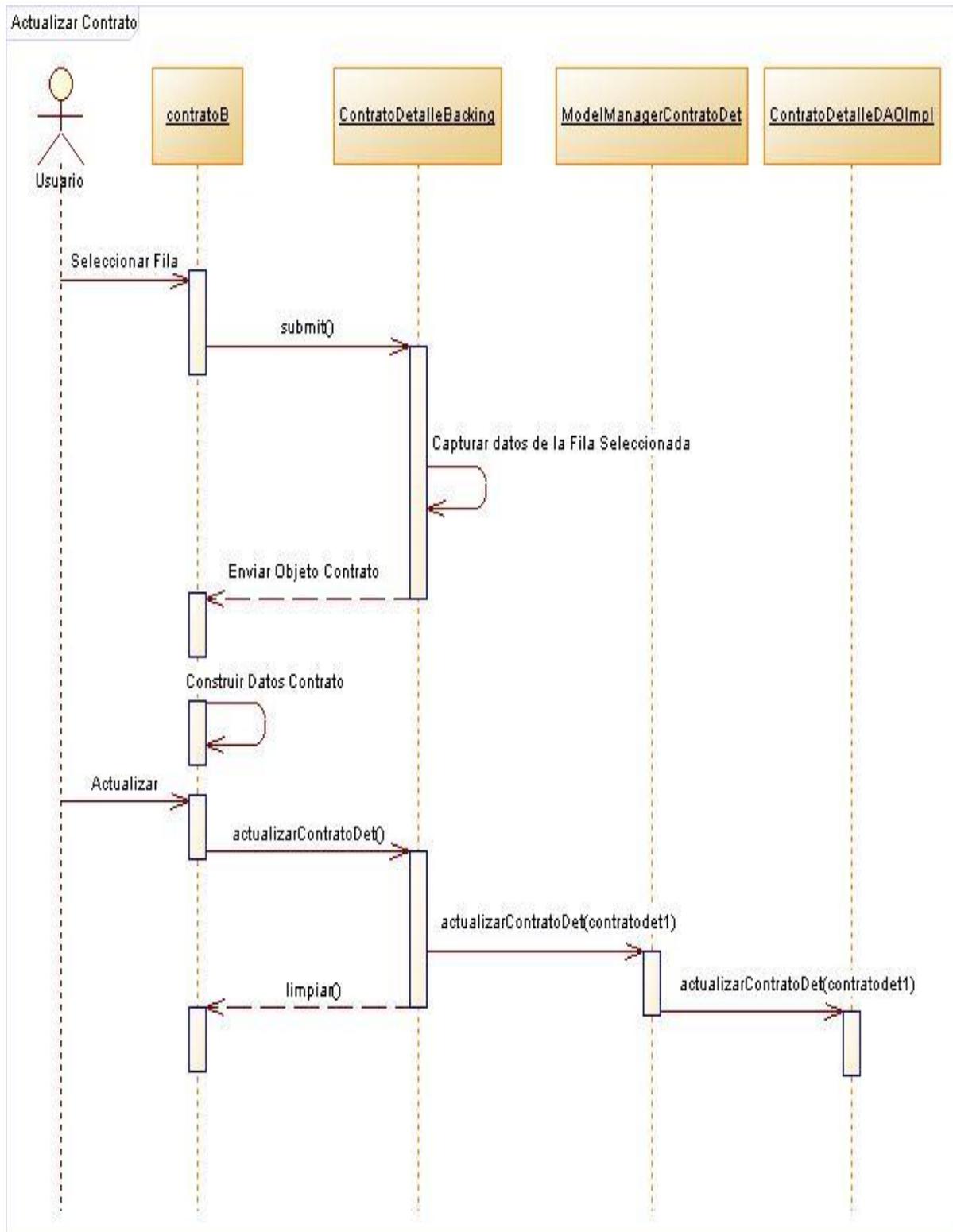


Ilustración 30.- Diagrama de Secuencia Caso de Uso: Actualizar Contrato.
Fuente: Andrés Cheza

En la ilustración presente vemos la interacción que realiza el usuario con el sistema para realizar la actualización de un contrato.

5.12.4.10 Diagrama de Secuencia Caso de Uso: Eliminar Contrato

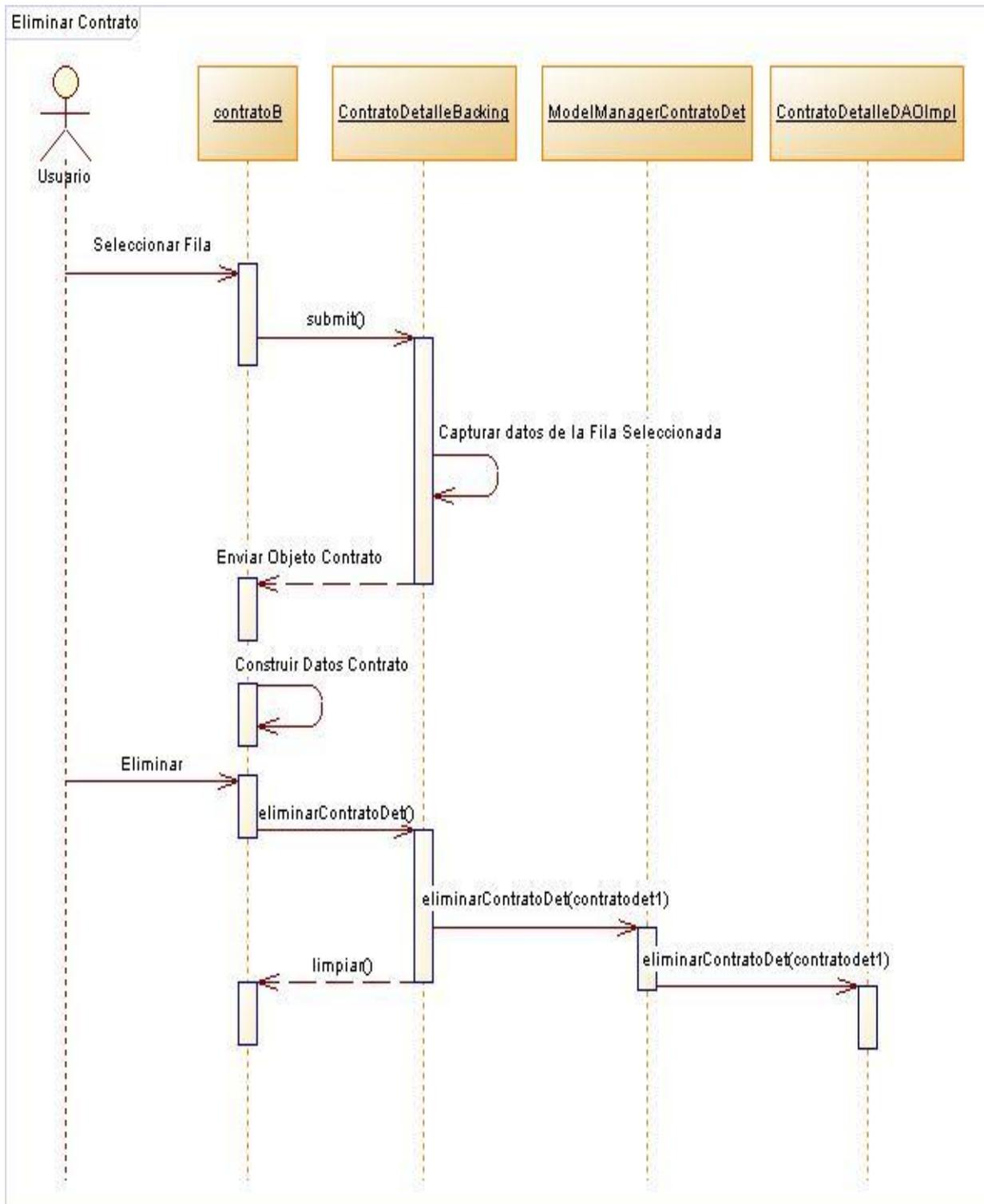


Ilustración 31.- Diagrama de Secuencia Caso de Uso: Eliminar Contrato

Fuente: Andrés Cheza

La ilustración muestra cómo funciona gráficamente el proceso de eliminación de un contrato registrado en el sistema por un usuario.

5.12.4.11 Diagrama de Secuencia Caso de Uso: Generar Reporte de Contratos

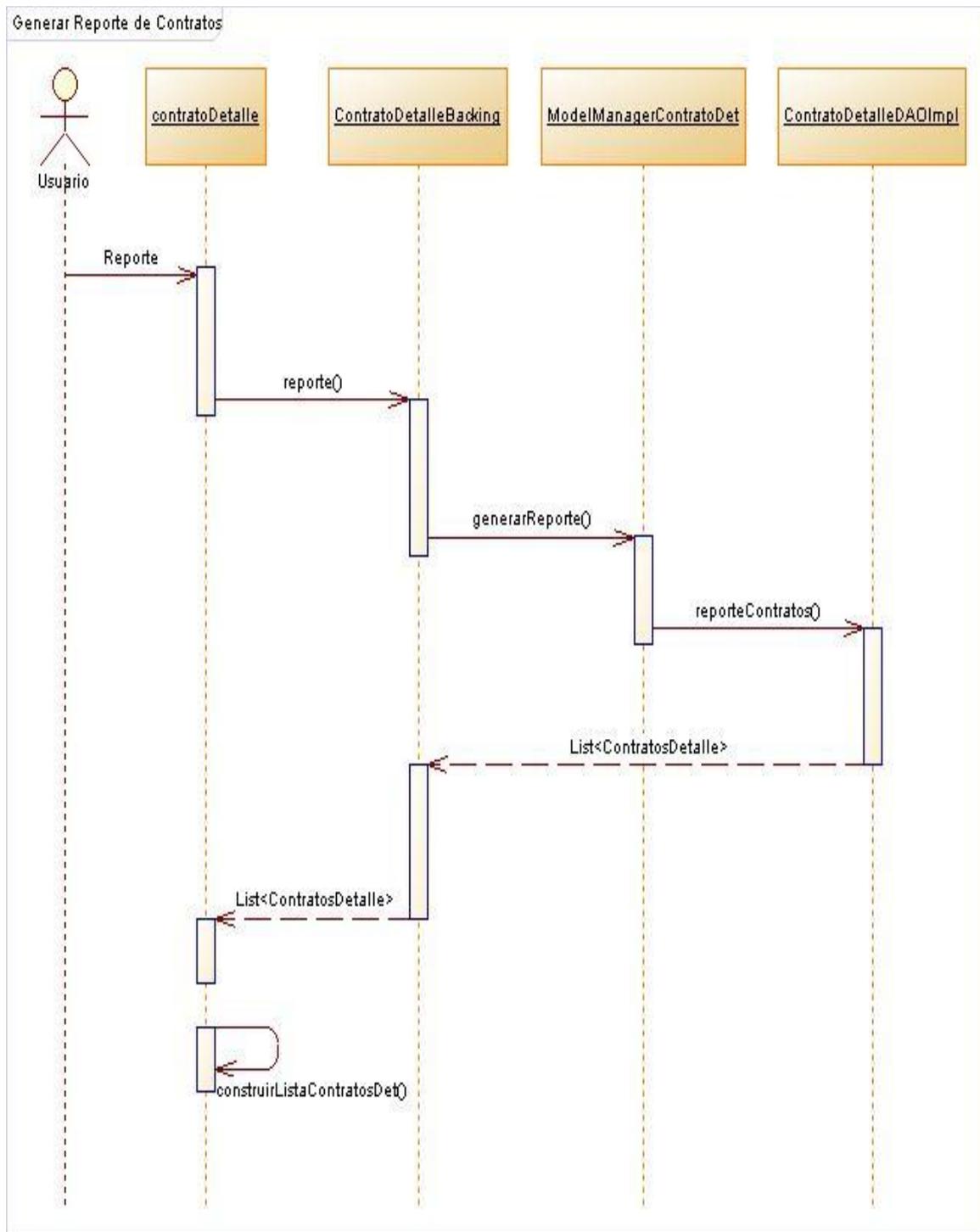


Ilustración 32.- Diagrama de Secuencia Caso de Uso: Generar Reporte de Contratos
Fuente: Andrés Cheza

En el presente diagrama se puede apreciar cómo interactúa el usuario con el sistema para realizar la generación del reporte de los contratos registrados en el sistema.

5.12.4.12 Diagrama de Secuencia Caso de Uso: Registrar Lectura

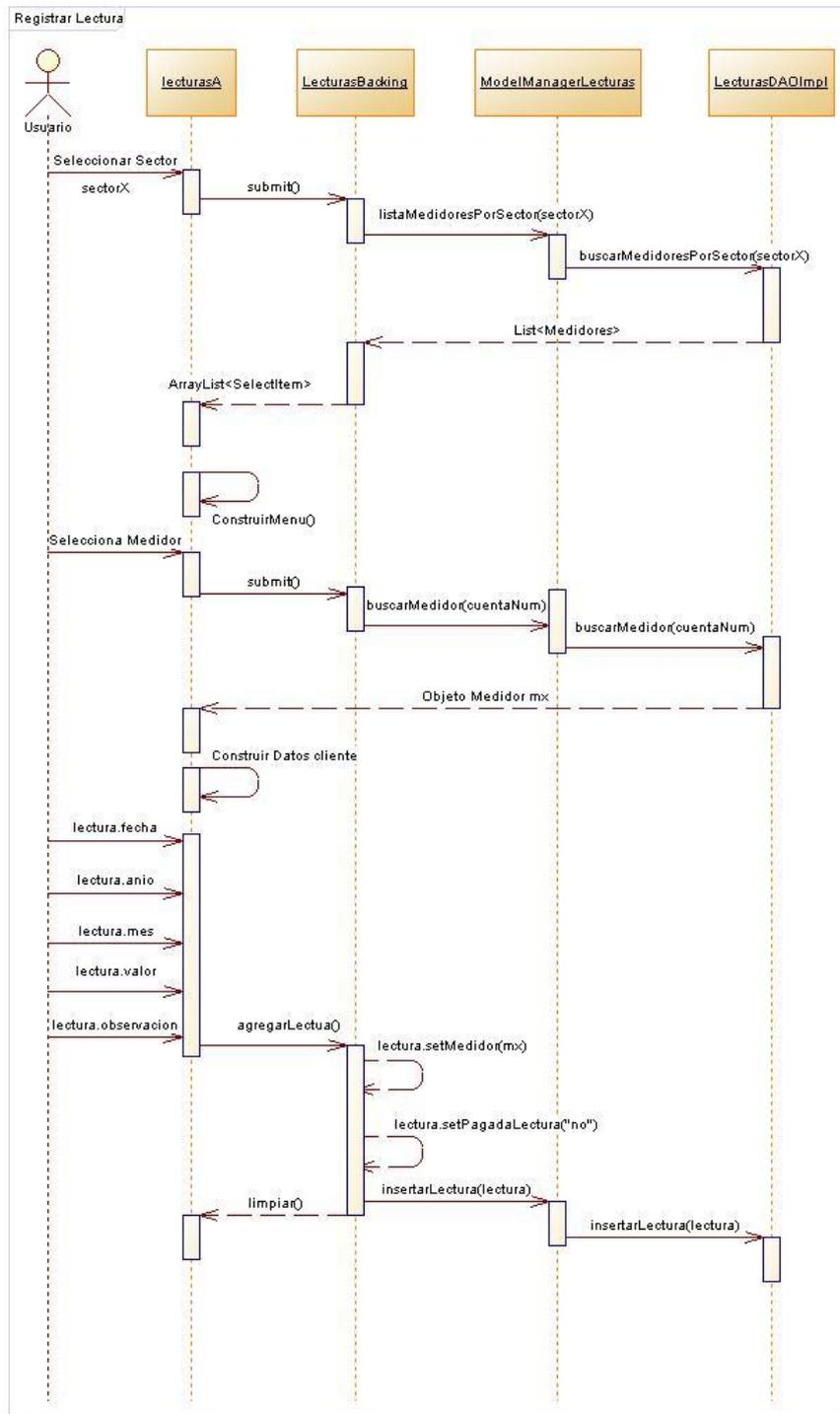


Ilustración 33.- Diagrama de Secuencia Caso de Uso: Registrar Lectura.
Fuente: Andrés Cheza

En el diagrama se muestra como se realiza el proceso de registrar una lectura en el sistema por parte del usuario.

5.12.4.13 Diagrama de Secuencia Caso de Uso: Actualizar Lectura

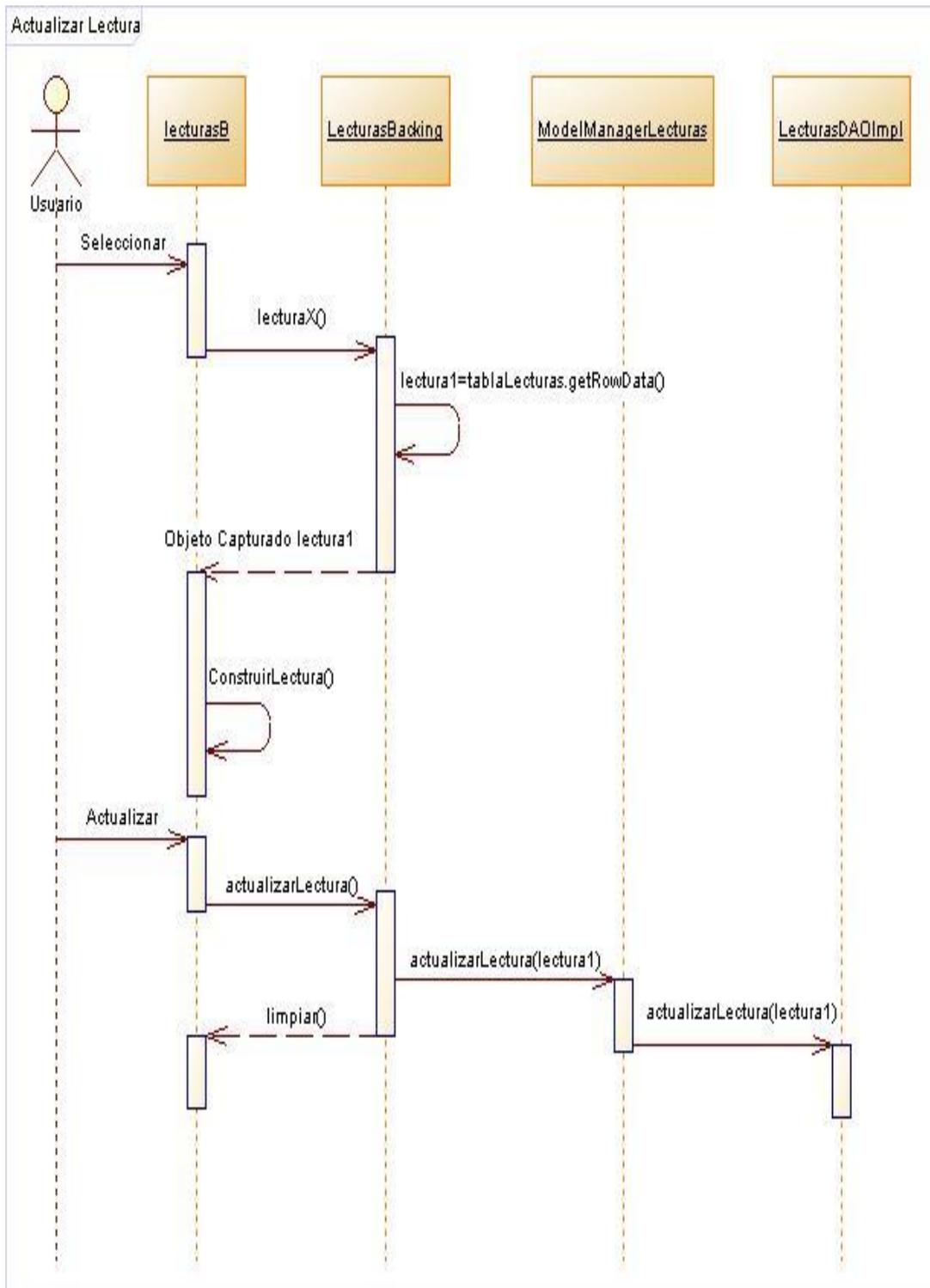


Ilustración 34.- Diagrama de Secuencia Caso de Uso: Actualizar Lectura.
Fuente: Andrés Cheza

En la presente ilustración se puede ver como se realiza la actualización de una lectura registrada en el sistema por el usuario.

5.12.4.14 Diagrama de Secuencia Caso de Uso: Eliminar Lectura

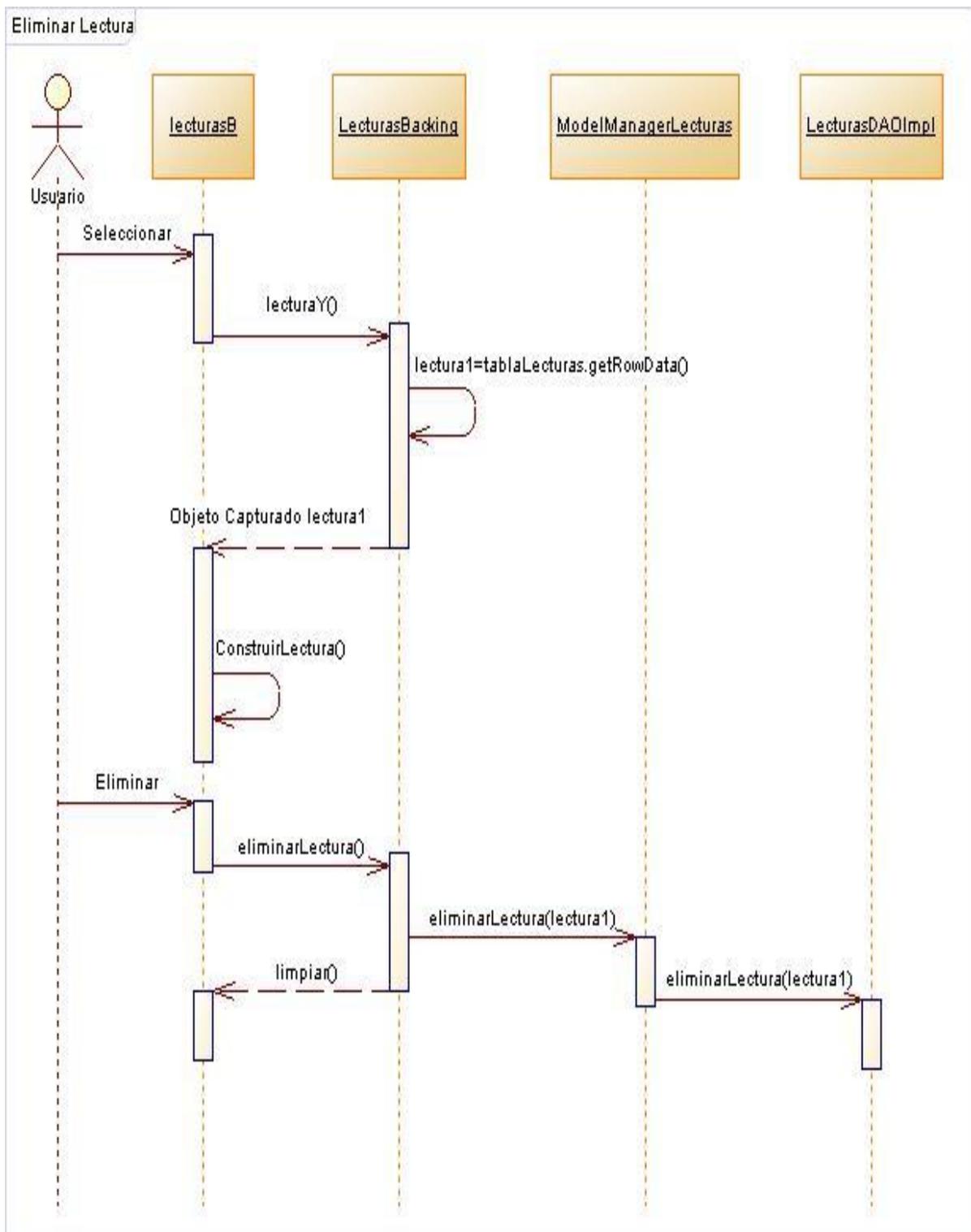


Ilustración 35.- Diagrama de Secuencia Caso de Uso: Eliminar Lectura
Fuente: Andrés Cheza

Se puede analizar en la ilustración como se realiza la eliminación de una lectura previamente registrada en el sistema por el usuario.

5.12.4.15 Diagrama de Secuencia Caso de Uso: Generar Reporte Lecturas

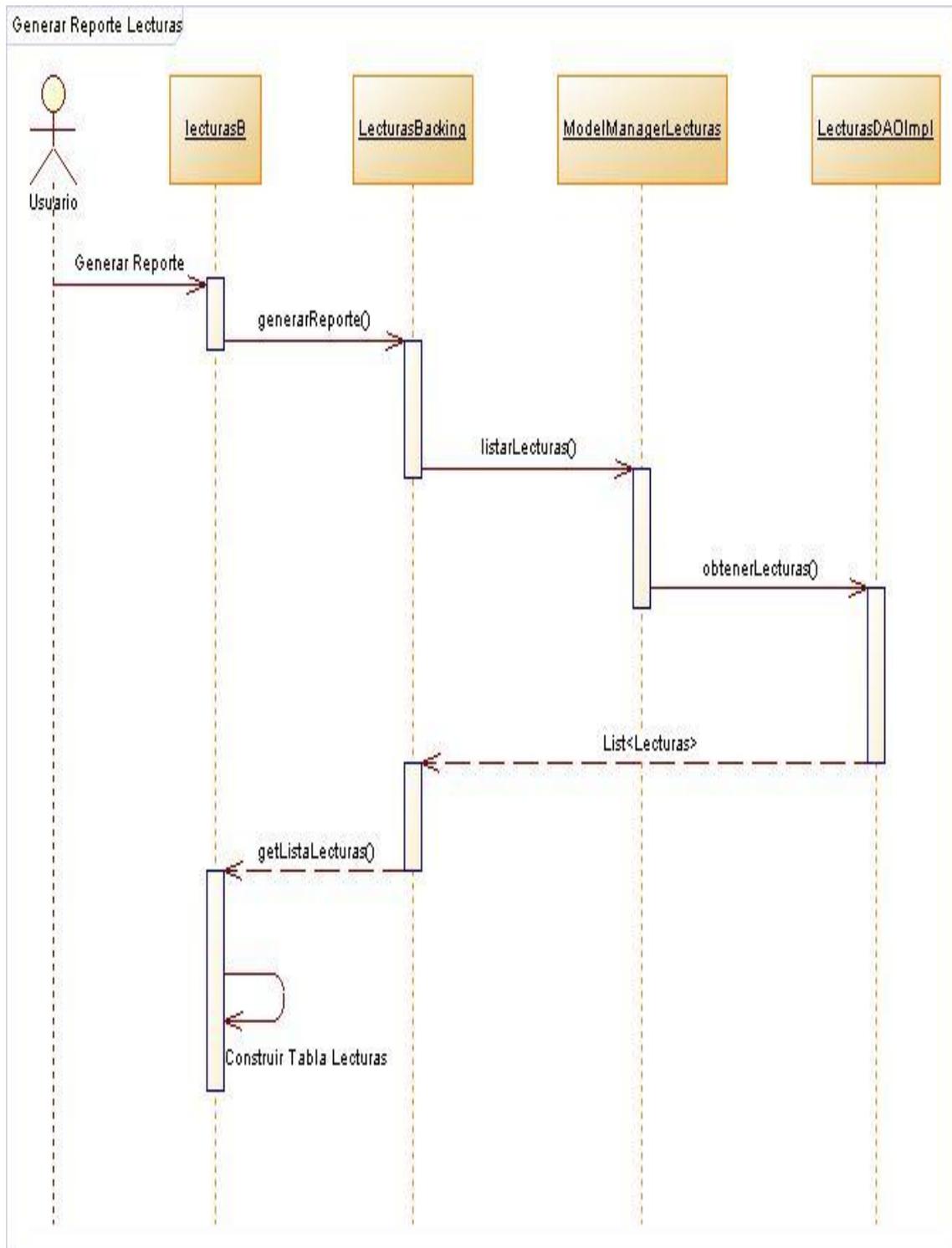


Ilustración 36.- Diagrama de Secuencia Caso de Uso: Generar Reporte Lecturas.
Fuente: Andrés Cheza

En el presente diagrama vemos como se realiza la interacción entre el usuario y el sistema para generar el reporte de las lecturas registradas en el sistema.

5.12.4.16 Diagrama de Secuencia Caso de Uso: Registrar Fondo Rojo

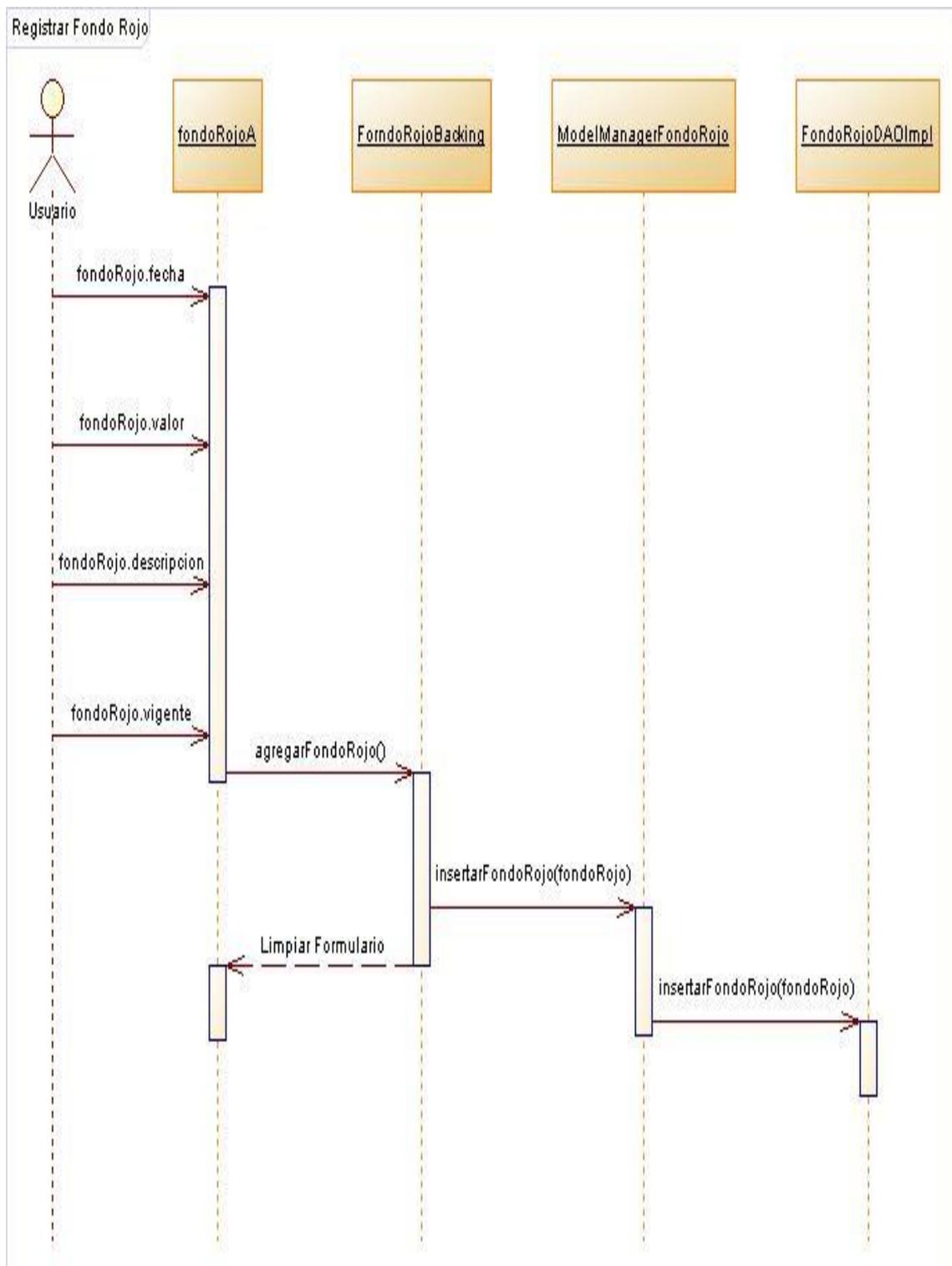


Ilustración 37.- Diagrama de Secuencia Caso de Uso: Registrar Fondo Rojo.
Fuente: Andrés Cheza

En el diagrama se puede apreciar cómo interactúan los diferentes objetos de análisis en el sistema para realizar el registro de los datos de un nuevo fondo rojo.

5.12.4.17 Diagrama de Secuencia Caso de Uso: Actualizar Fondo Rojo

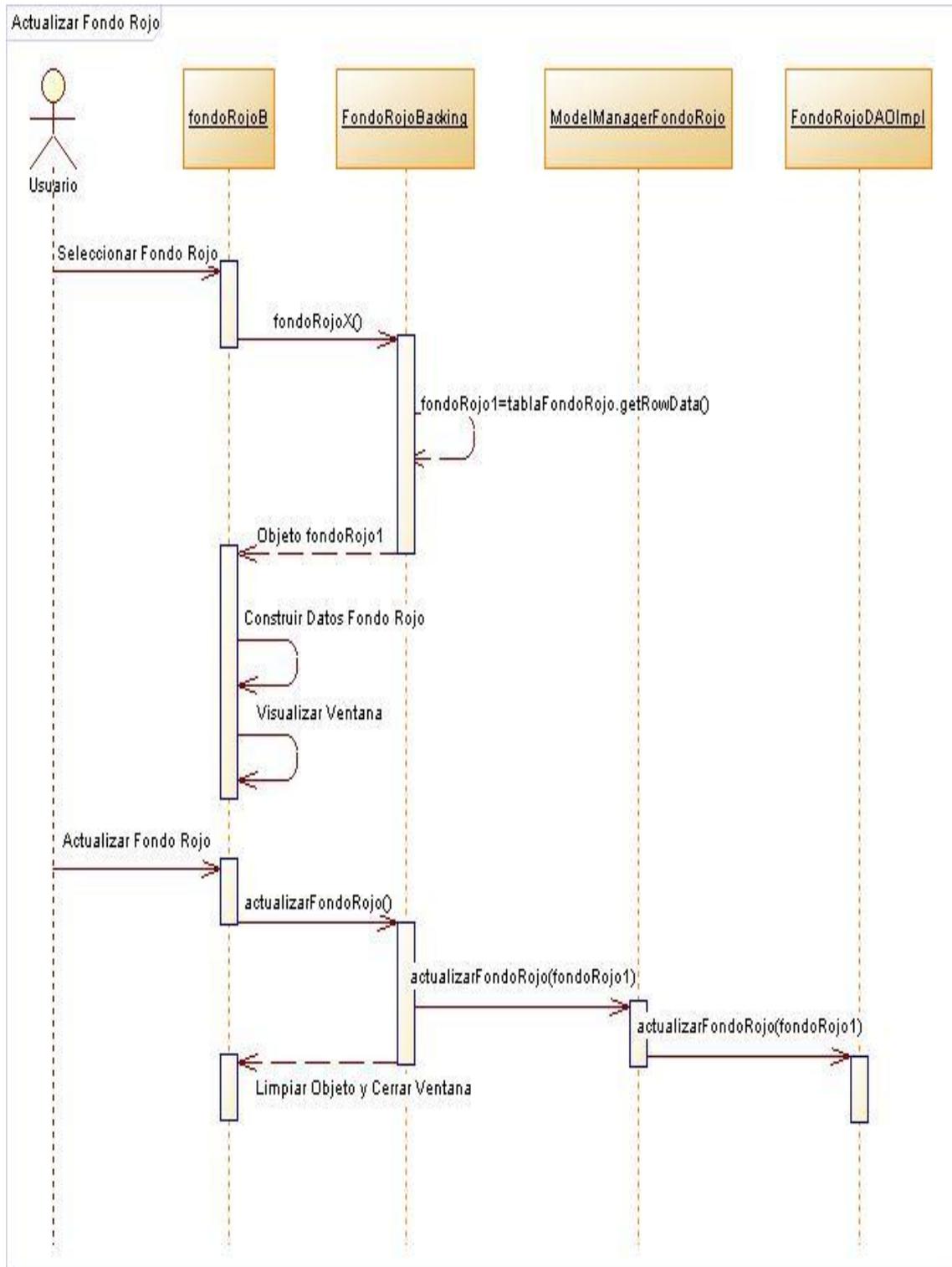


Ilustración 38.- Diagrama de Secuencia Caso de Uso: Actualizar Fondo Rojo.
Fuente: Andrés Cheza

La ilustración muestra gráficamente como se realiza la interacción entre el sistema y el usuario para realizar la actualización de los datos de un fondo rojo.

5.12.4.18 Diagrama de Secuencia Caso de Uso: Eliminar Fondo Rojo

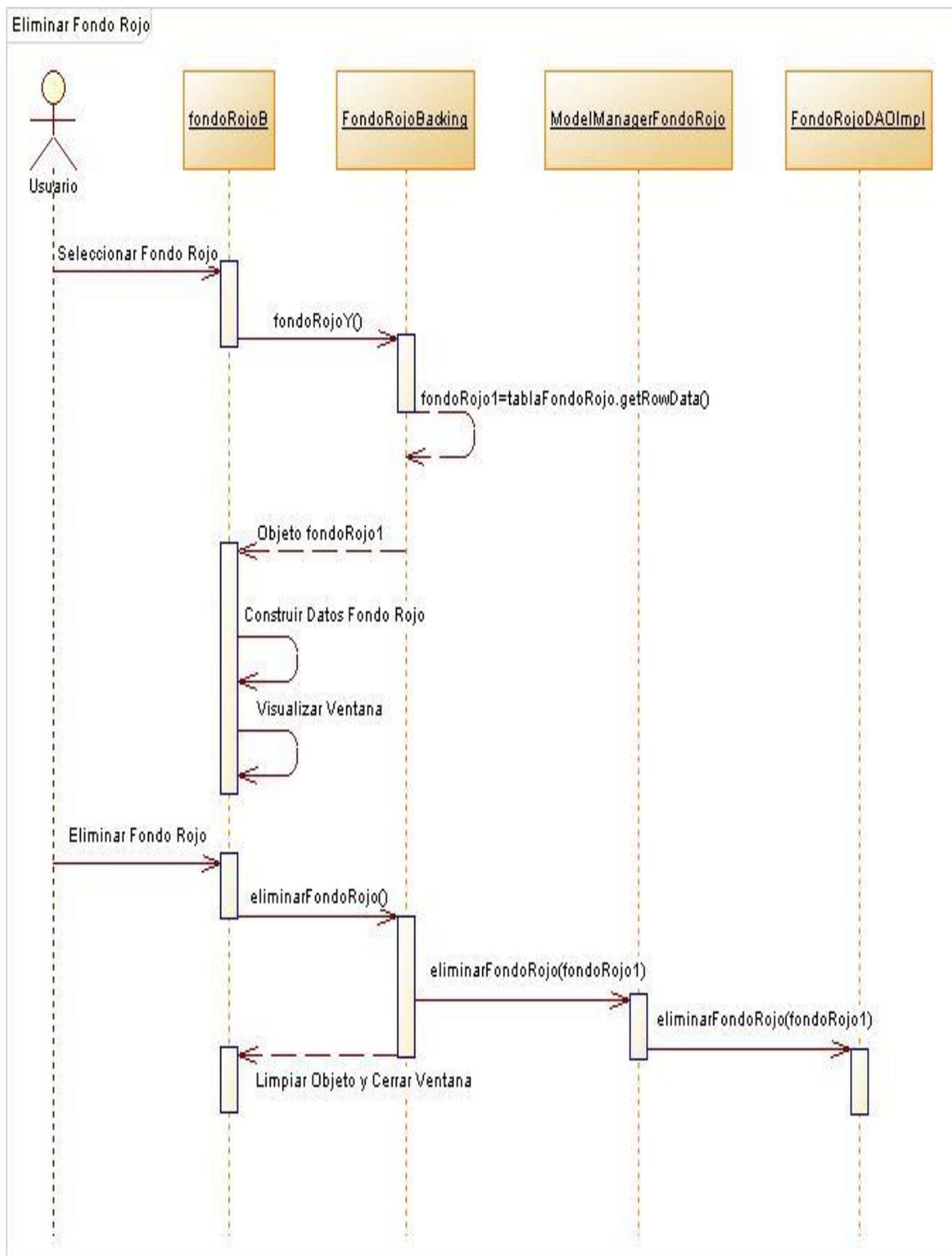


Ilustración 39.- Diagrama de Secuencia Caso de Uso: Eliminar Fondo Rojo.
Fuente: Andrés Cheza

El diagrama nos muestra cómo se realiza el proceso de eliminación de un fondo rojo del sistema.

5.12.4.19 Diagrama de Secuencia Caso de Uso: Generar Reporte Fondo Rojo

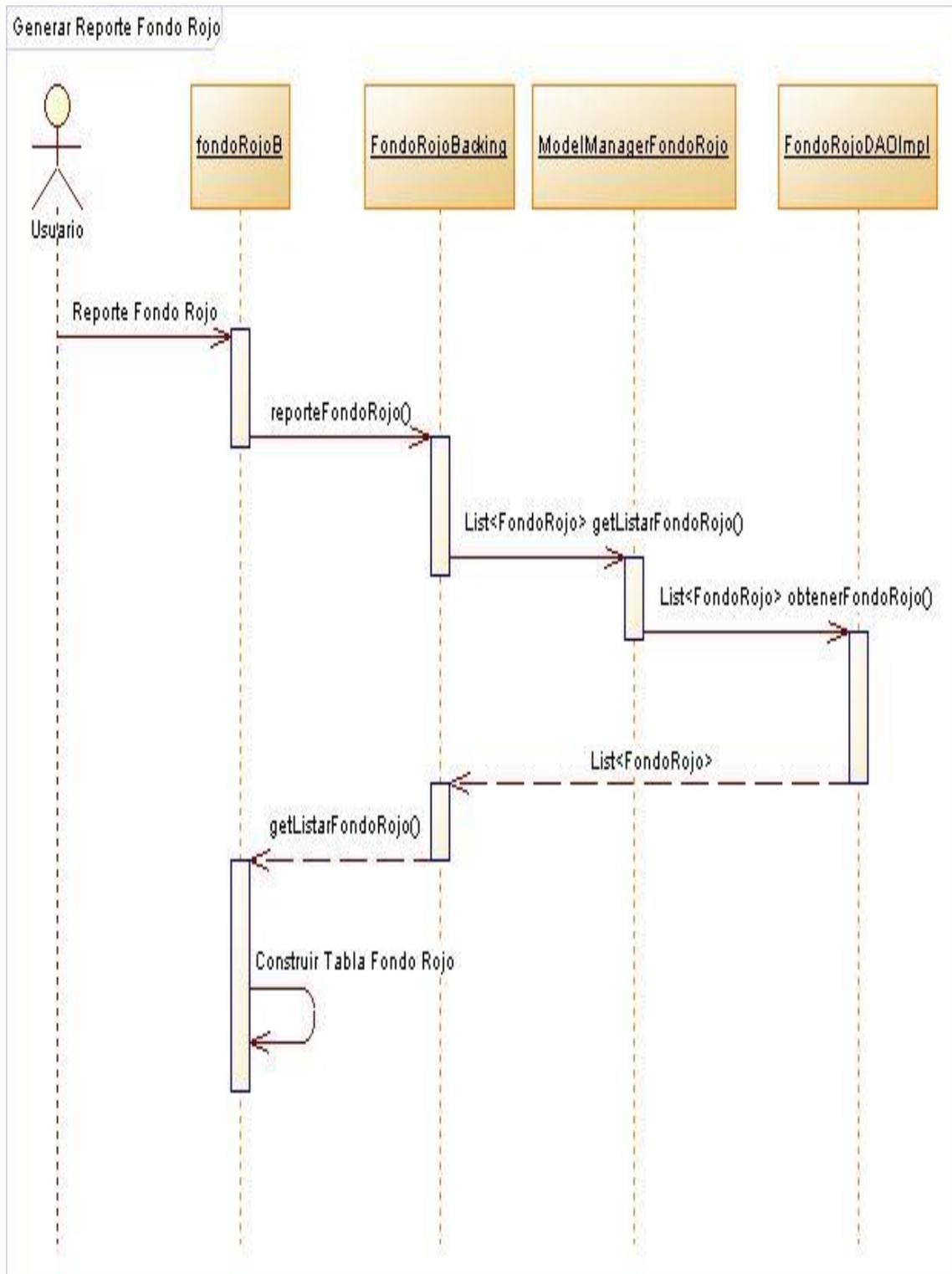


Ilustración 40.- Diagrama de Secuencia Caso de Uso: Generar Reporte Fondo Rojo.
Fuente: Andrés Cheza

Para realizar el proceso de reporte de todos los datos del fondo rojo registrados en el sistema el diagrama nos muestra este proceso en forma gráfica.

5.12.4.20 Diagrama de Secuencia Caso de Uso: Pagar Consumo Agua Potable

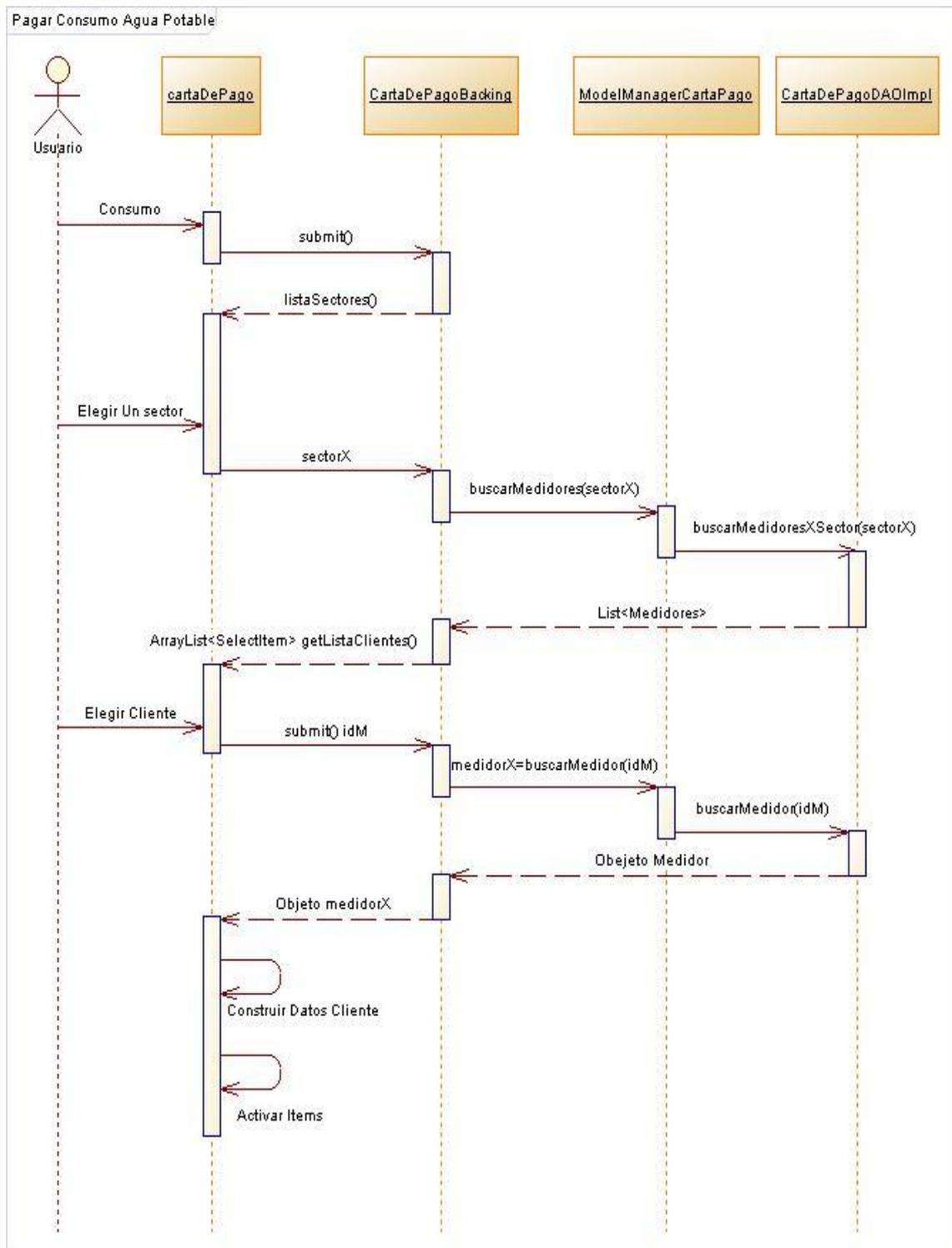


Ilustración 41.- Diagrama de Secuencia Caso de Uso: Pagar Consumo Agua Potable.
Fuente: Andrés Cheza

En el presente diagrama podemos ver cómo interactúan tanto el actor como el sistema para realizar el proceso del pago del consumo de agua potable para un cliente.

5.12.5 Vista Lógica

A continuación se mostrará los subsistemas, paquetes y componentes tomados en cuenta para la elaboración de esta vista:

5.12.5.1 Descomposición den subsistemas

En el presente apartado procedemos a desarrollar la descomposición del aplicativo en subsistemas, los mismos que son identificados en el paquete de análisis.

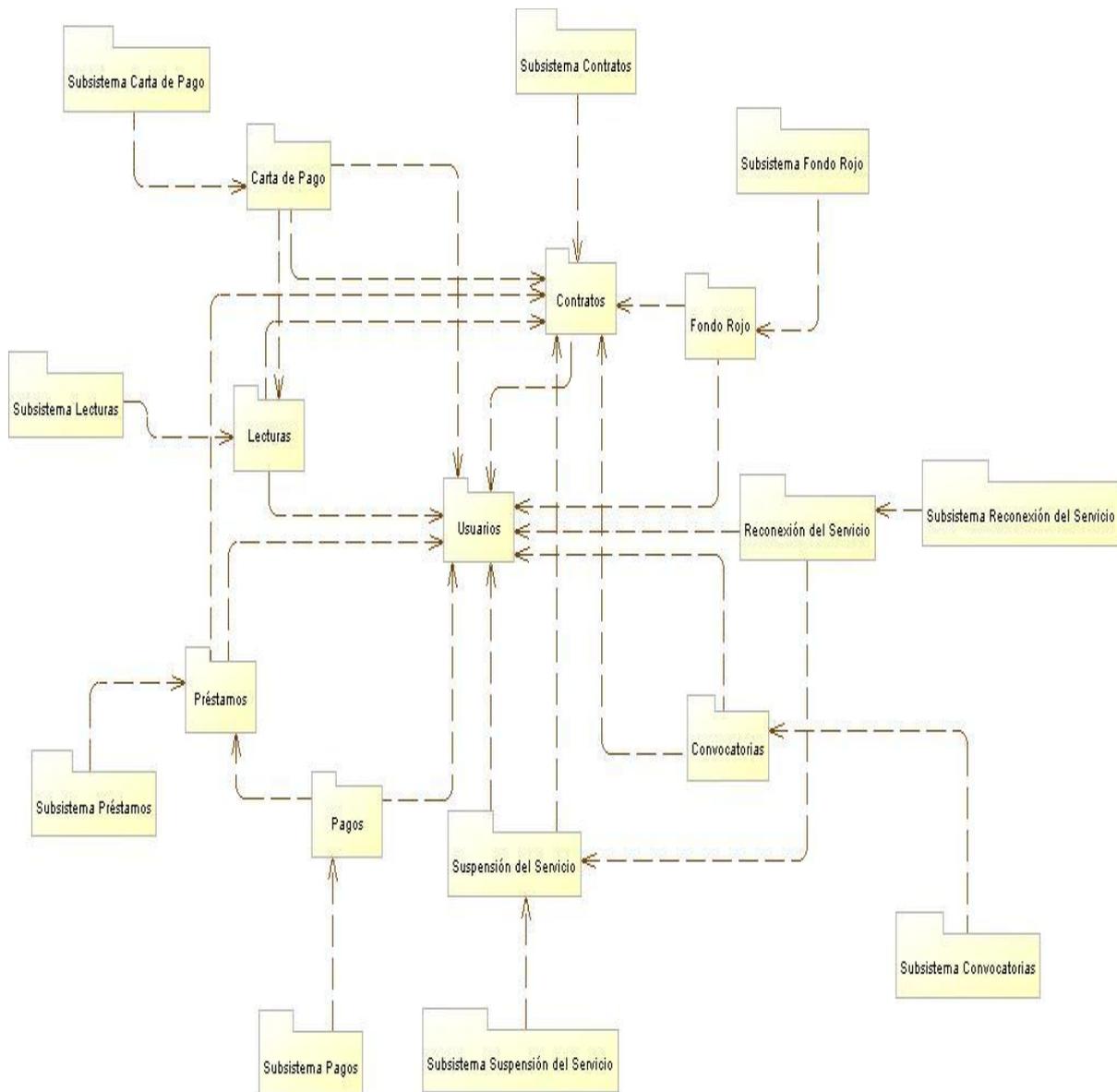


Ilustración 42.- Identificación de subsistemas de diseño a partir de paquetes de análisis.
Fuente: Andrés Cheza

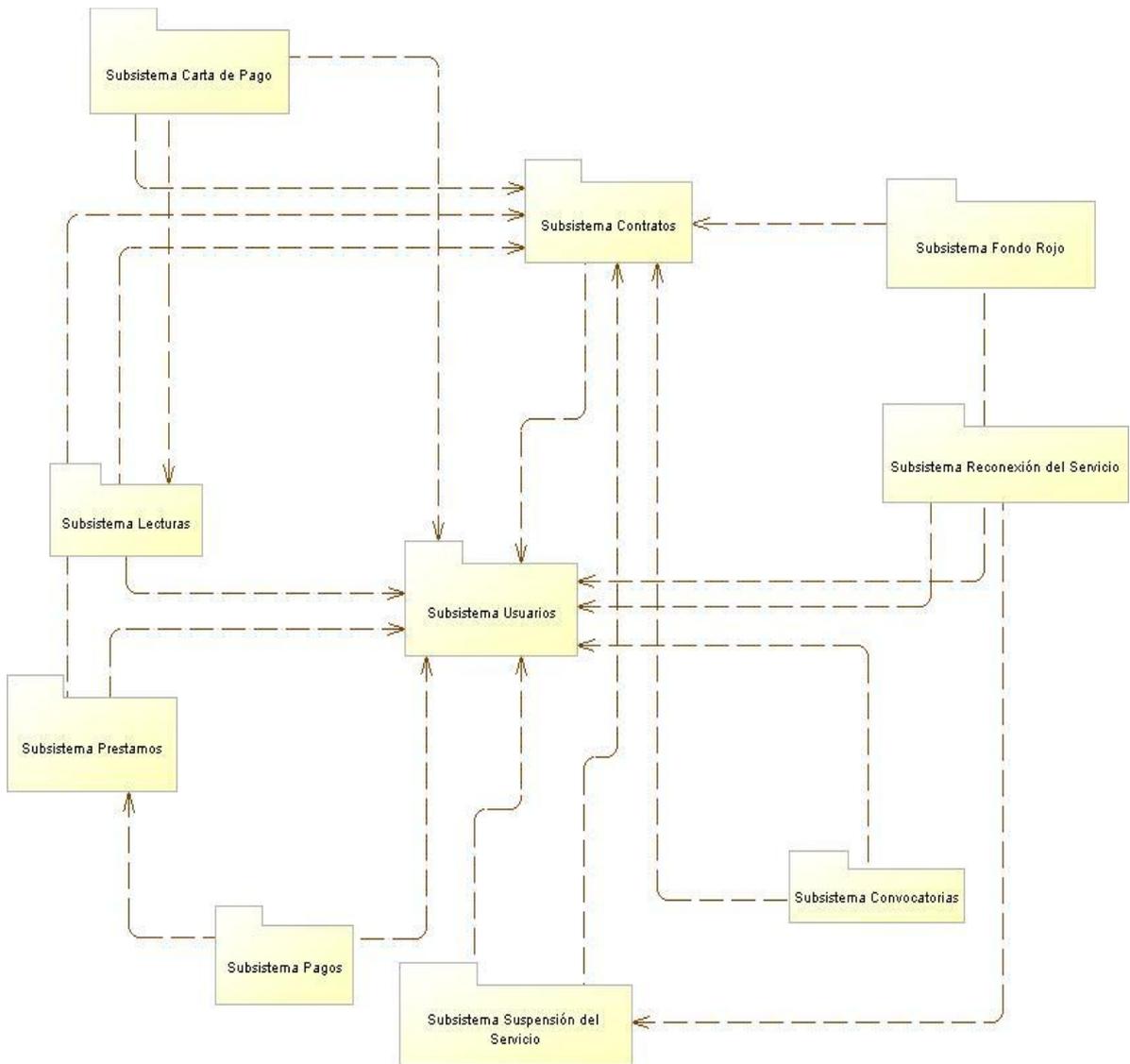


Ilustración 43.- Dependencia entre subsistema de diseño.

Fuente: Andrés Cheza

5.12.6 Vista de Procesos

En la presente sección mostraremos los diagramas de clases que utiliza cada caso de uso para desarrollar su proceso.

5.12.6.1 Diagrama de Clase: clase Subsistema Usuarios

Especifica las clases que conforman el paquete relacionado a los usuarios del sistema, los correspondientes atributos y su interacción entre ellos.

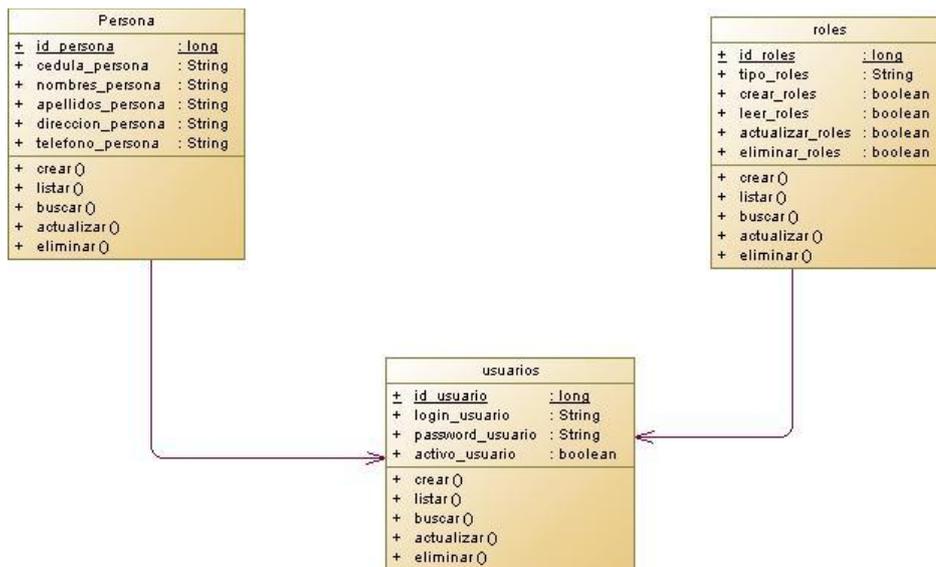


Ilustración 44.- Diagrama de Clase: clase Subsistema Usuarios.

Fuente: Andrés Cheza

5.12.6.2 Diagrama de Clase: clase Subsistema Contratos

En el presente diagrama se muestra cuáles son las clases que intervienen en los contratos, que función desarrollan y como están conformados.

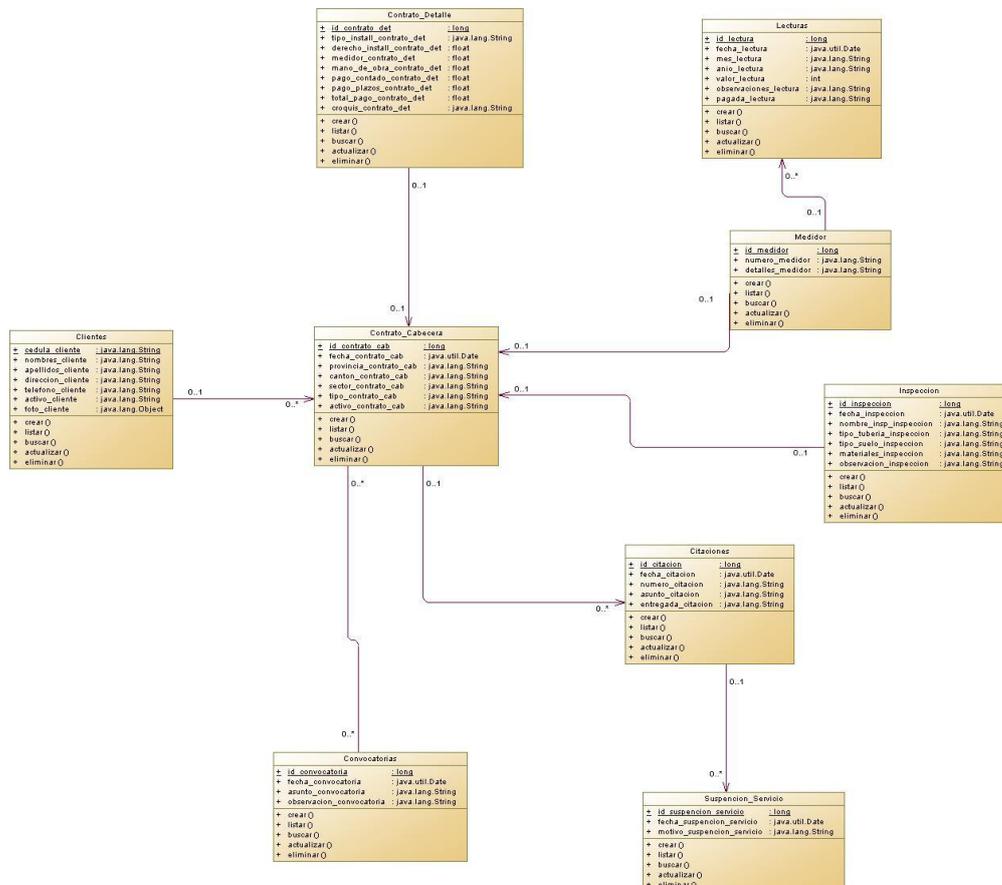


Ilustración 45.- Diagrama de Clase: clase Subsistema Contratos.

Fuente: Andrés Cheza

5.12.6.3 Diagrama de Clase: clase Subsistema Lecturas

En el subsistema de Lecturas podemos apreciar como las clases que los conforman así también sus atributos, la forma en que interactúan entre ellos.

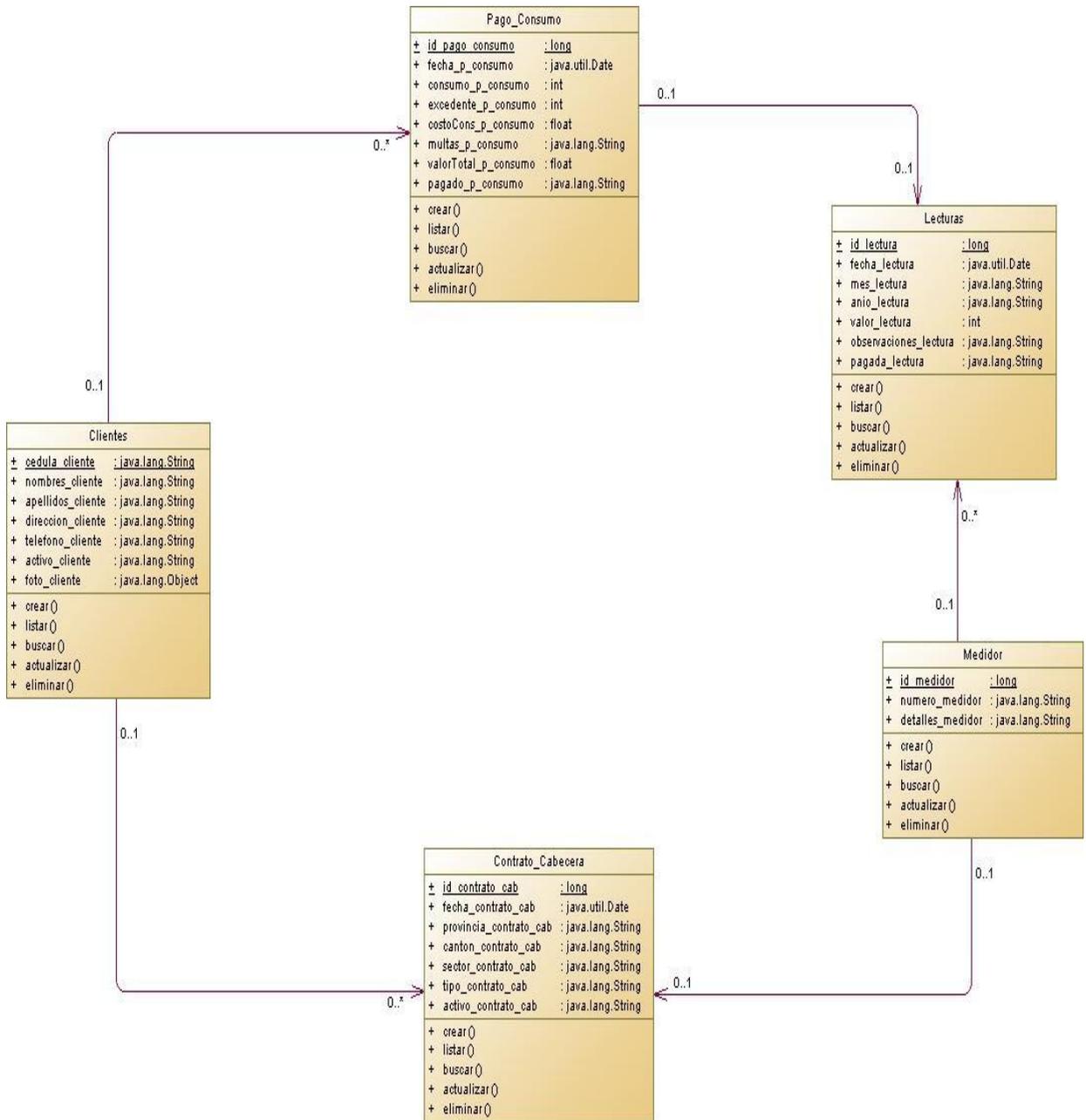


Ilustración 46.- Diagrama de Clase: clase Subsistema Lecturas.
Fuente: Andrés Cheza

5.12.6.4 Diagrama de Clase: clase Subsistema Fondo Rojo

El subsistema Fondo Rojo está constituido por las clases que se aprecia en el presente diagrama así como su relación entre las mismas.

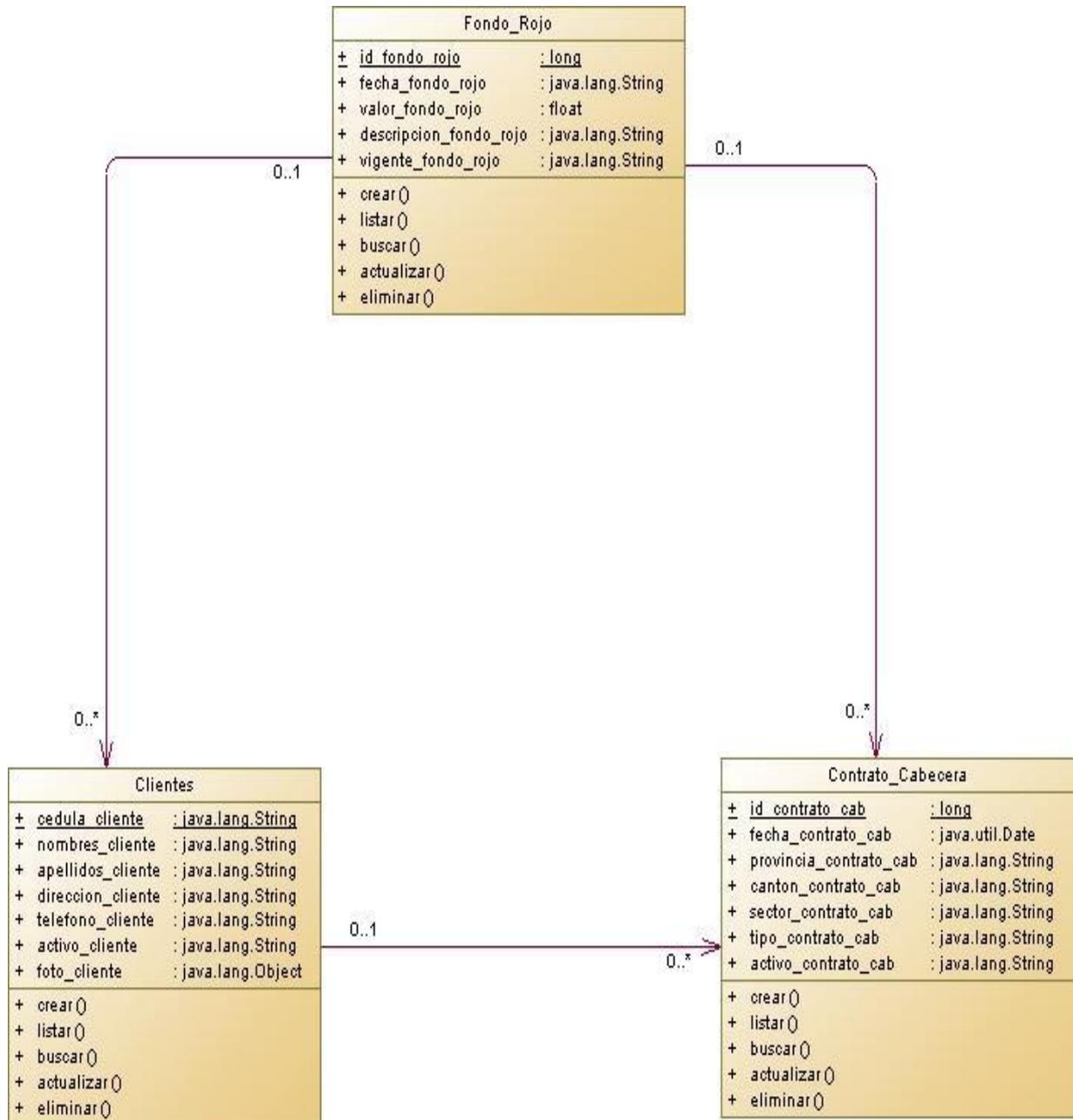


Ilustración 47.- Diagrama de Clase: clase Subsistema Fondo Rojo.

Fuente: Andrés Cheza

5.12.6.5 Diagrama de Clase: clase Subsistema Carta de Pago

Como podemos ver en el presente diagrama el subsistema Carta de Pago ejecuta varias interacciones con las diferentes clases que están asociadas entre ella.

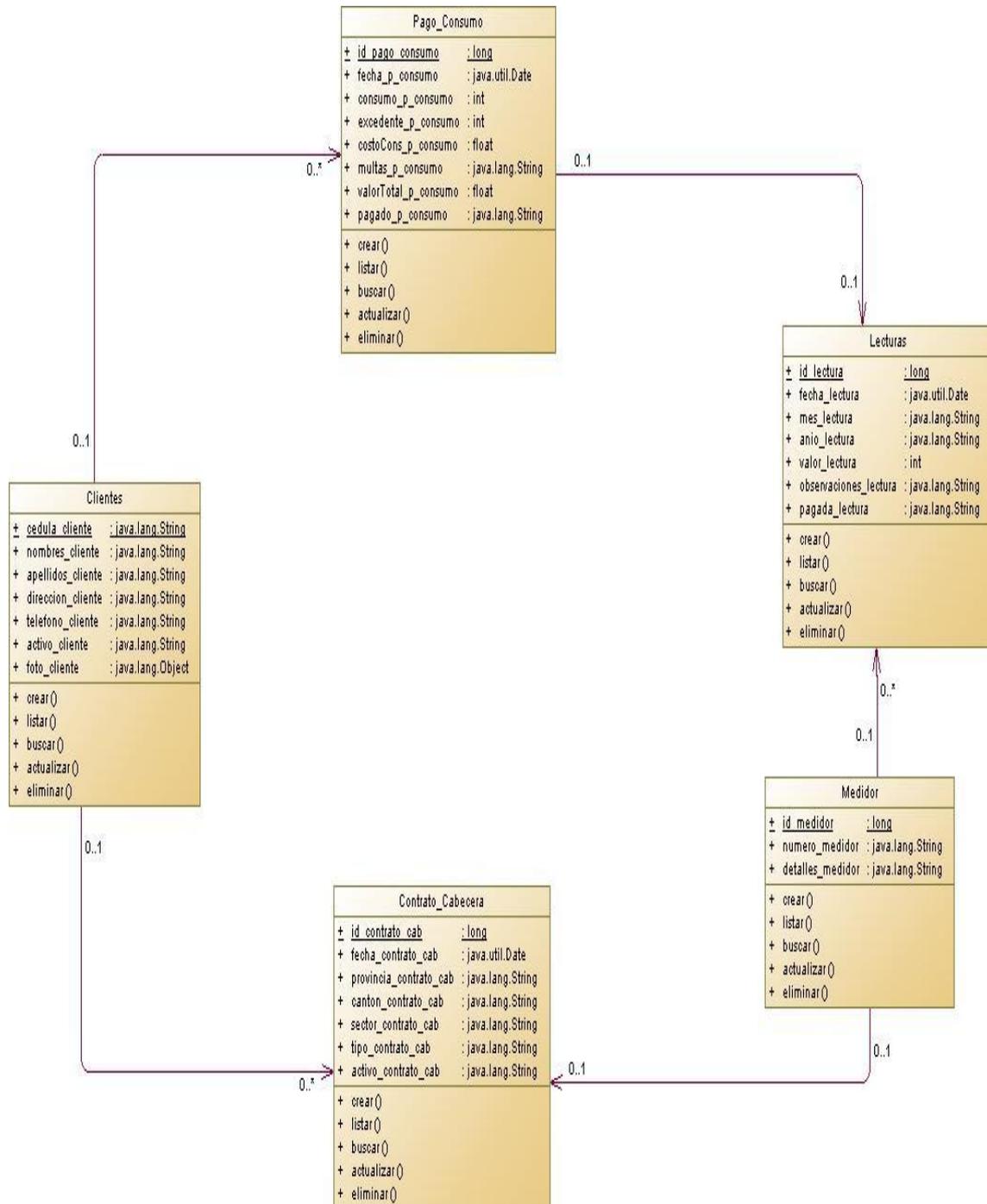


Ilustración 48.- Diagrama de Clase: clase Subsistema Carta de Pago.

Fuente: Andrés Cheza

5.12.6.6 Diagrama de Clase: clase Subsistema Suspensión del Servicio

En el diagrama que se muestra a continuación podemos apreciar los atributos con que cuenta el subsistema Suspensión del servicio, así también su relación con las clases que conforman este subsistema.

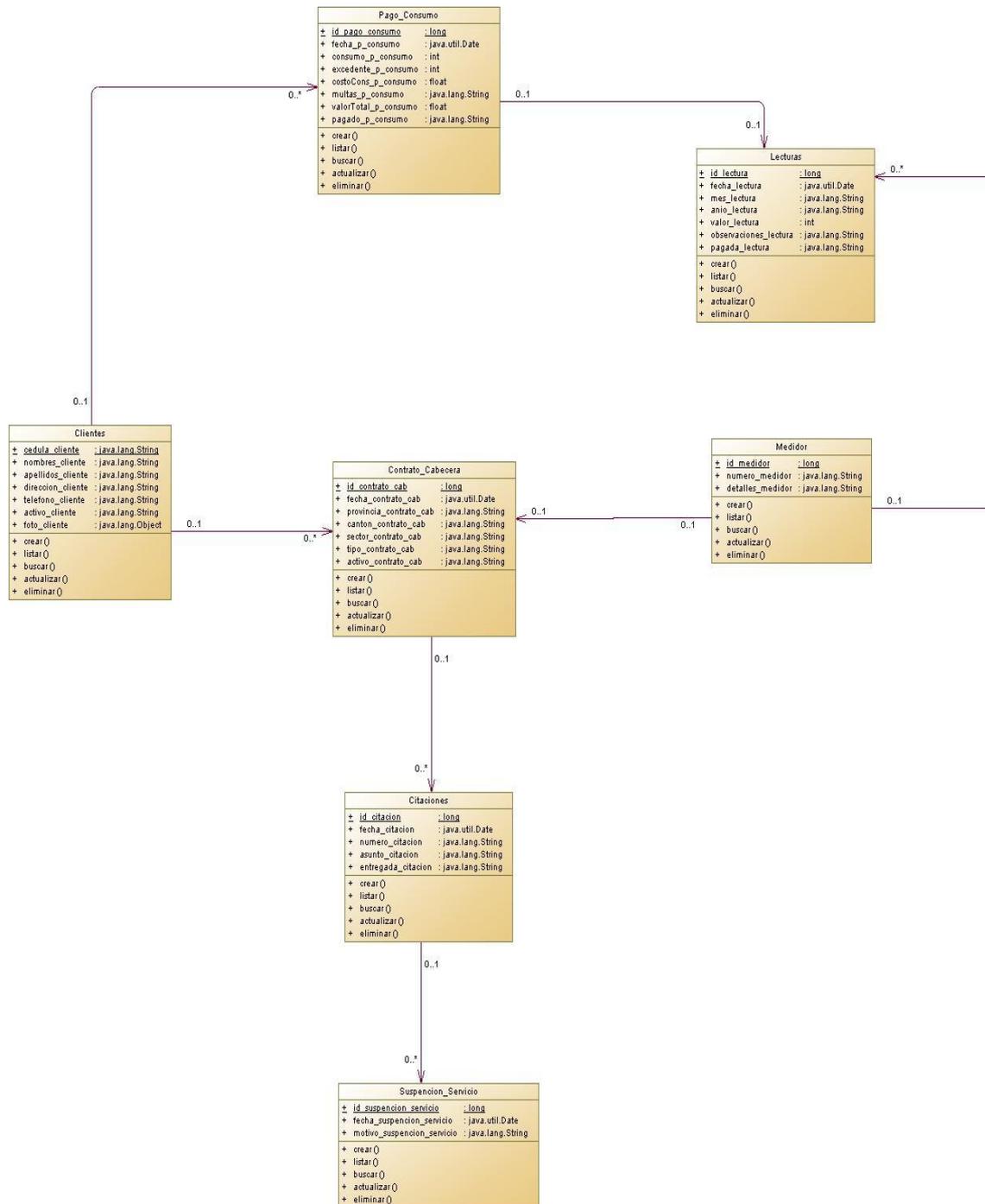


Ilustración 49.- Diagrama de Clase: clase Subsistema Suspensión del Servicio.
Fuente: Andrés Cheza

5.12.6.7 Diagrama de Clase: clase Subsistema Reconexión del Servicio

El diagrama correspondiente al subsistema Reconexión del Servicio cuenta con varias clases que se relacionan entre sí para poder interactuar.

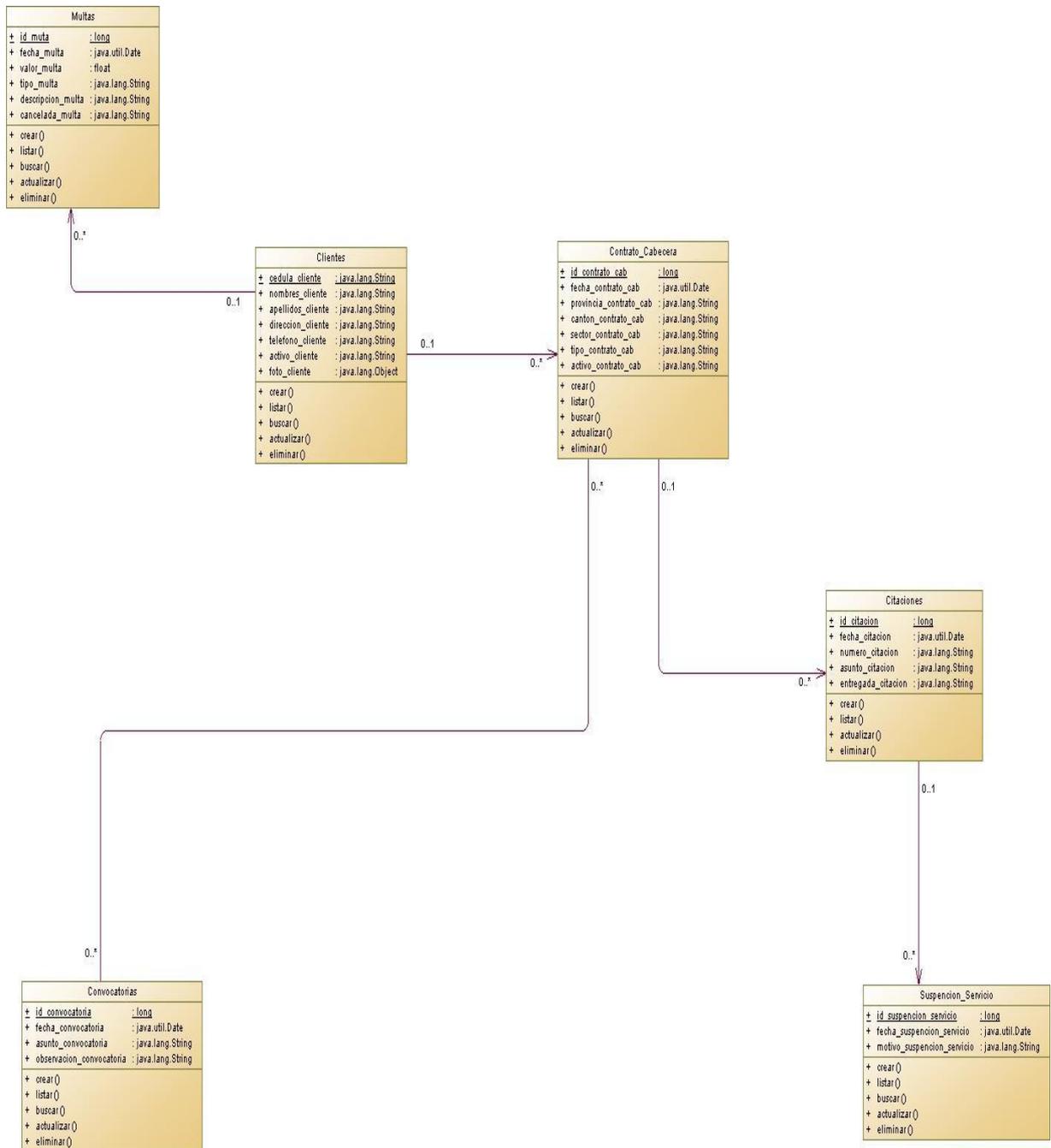


Ilustración 50.- Diagrama de Clase: clase Subsistema Reconexión del Servicio.
Fuente: Andrés Cheza

5.12.6.8 Diagrama de Clase: clase Subsistema Convocatorias

En el diagrama que se presenta a continuación podemos apreciar las clases que conforman el subsistema Convocatorias, también los atributos con los que cuenta dicho subsistema.

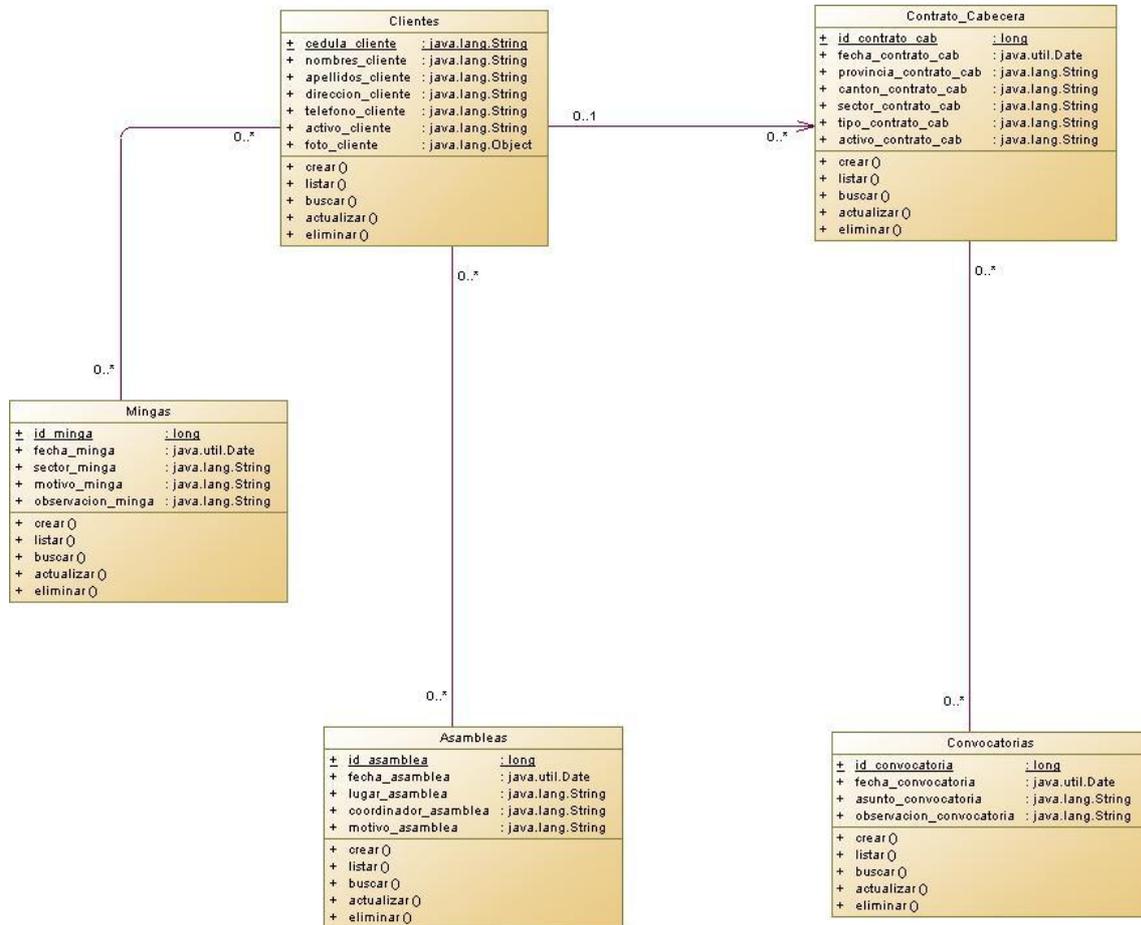


Ilustración 51.- Diagrama de Clase: clase Subsistema Convocatorias.

Fuente: Andrés Cheza

5.12.7 Vista de Datos

Se ejecutará la estructura de los correspondientes datos teniendo en cuenta el desarrollo de los siguientes modelos de datos los mismos que se iniciaron con el modelo entidad-relación, para posteriormente desarrollar el modelo físico.

5.12.8 Modelo Relacional

Ver anexo

5.12.9 Modelo Físico

Ver Anexo

5.13 Lista de Riesgos

5.13.1 Introducción

Una lista de riesgos nos permite realizar un análisis de cada uno de los posibles problemas que tendremos a lo largo del desarrollo del proyecto, la dimensión, su descripción, los impactos que estos puedan ocasionar, el indicador que nos permitirá verificar si el riesgo está por producirse o transformarse en un problema, una estrategia de mitigación para evitar que este riesgo se efectúe, y el plan de contingencia que se realizará si el riesgo se llegara a efectuar.

5.13.1.1 Propósito

El propósito del presente documento es detallar el proceso de administración de riesgos presentes a lo largo de la producción del proyecto, dicha lista nos permitirá revisar periódicamente para evitar que alguno de ellos se transforme en un problema. Este proceso de administración comprende la identificación, priorización, análisis, monitoreo y posterior mitigación de los riesgos.

5.13.1.2 Alcance

La lista de riesgos tiene como alcance precisar diferentes tipos de riesgos en nuestro caso analizaremos los siguientes tipos:

- Riesgo de recursos
- Riesgos tecnológicos

5.13.1.3 Definición, acrónimos y abreviaturas

Ver documento Glosario

5.13.1.4 Revisión General

En el siguiente plan de riesgos que vamos a procesar, analizaremos su magnitud, la misma que nos permitirá clasificar que tan perjudicial puede ser para el proyecto, el impacto que puede generar el proyecto, como podríamos monitorear determinado riesgo, que está haciendo para reducir el impacto creado por este riesgo y en caso que dicho riesgo afecte al proyecto cual sería el plan de contingencia para que el riesgo no se convierta en una dificultad y afecte de manera negativa al proyecto.

5.13.2 Riesgos

5.13.2.1 Deficiencia en la recolección de información

❖ **Magnitud de riesgo o “Ranking”**

Alta

❖ **Descripción**

Este riesgo se puede presentar en la obtención de la información referente a los procesos que lleva acabo la Junta de Agua.

❖ **Impactos**

Malos resultados en la información de los clientes que maneja la junta de agua.
Tener una deficiencia en el control de los clientes.

❖ **Indicadores**

❖ **Estrategia de mitigación**

Obtener la información lo más exacta posible de como maneja los proceso la junta de agua.

❖ **Plan de contingencia**

Ingresar la información manualmente realizando una actualización.

5.13.2.2 no llenar las expectativas de la Junta de Agua Potable

❖ **Magnitud de riesgo o “Ranking”**

Alta

❖ **Descripción**

Que los procesos que lleve la Junta de Agua no estén muy claros y tengan limitaciones, ya que el sistema se desarrolla de acuerdo a las especificaciones de los requerimientos.

❖ **Impactos**

Abarcar limitados proceso.

Proceso incompletos.

❖ **Indicadores**

Indicador:

$\% \text{Proceso} = \text{Procesos manejados por el sistema} / \text{Proceso manejados por la Junta de Agua} * 100$

Umbral: 90%

❖ **Estrategia de mitigación**

Rediseñar los procesos creados por el personal de la Junta de Agua integrando las funcionalidades que puede brindar el sistema.

❖ **Plan de contingencia**

Reformular los módulos y procesos que lleva a cabo el sistema.

5.13.2.3 El MVC no se acople a la metodología RUP

❖ **Magnitud de riesgo o “Ranking”**

Baja

❖ **Descripción**

Diferencia en el diseño de los objetos de acuerdo al patrón MVC.

❖ **Impactos**

Diagramación del modelo de clase abstracta.

Cambio de arquitectura de desarrollo.

❖ **Indicadores**

Indicador:

$\% \text{Porcentaje de acoplamiento de metodología RUP y patrón MVC.}$

Umbral: 100%

❖ **Estrategia de mitigación**

Utilizar una herramienta que cumpla las especificaciones del patrón de diseño para poder generar los objetos y posteriormente el código SQL para la creación de la base de datos.

❖ **Plan de contingencia**

Diagramar en su totalidad según genere el patrón MVC.

5.13.2.4 La base de datos desarrollada en PostgreSQL no se acople al modelo del patrón MVC.

- ❖ **Magnitud de riesgo o “Ranking”**

Baja

- ❖ **Descripción**

Lograr la integración de la base de datos en el componente del patrón MVC debido a la variación de datos.

- ❖ **Impactos**

Tener dificultad de utilizar software libre en el proyecto.

- ❖ **Indicadores**

Indicador:

%Acoplamiento de la base de datos en el componente del patrón MVC.

Umbral: 100%

- ❖ **Estrategia de mitigación**

Realizar un análisis minucioso del framework que permite la manipulación de la persistencia con la base de datos.

- ❖ **Plan de contingencia**

Cambiar de sistema gestor de base de datos por ejemplo a MySQL.

5.14 Prototipo de Interfaz de Usuario

5.14.1 Introducción

El artefacto que a continuación detallamos es una parte esencial del proyecto ya que es el que tiene íntima relación con el usuario debido a que se realiza la interacción entre usuario sistema.

Por tal motivo debemos realizar el desarrollo de este componente con los elementos adecuados ya que de esta manera permitirán el fácil manejo por parte del usuario.

Para tal desarrollo debemos realizar una investigación sobre la parte de diseño gráfico para tener un excelente medio de interacción.

5.14.1.1 Propósito

Mostrar la plantilla que se usará en el desarrollo del sistema, así también como los diferentes archivos de configuración.

5.14.1.2 Descripción

El presente documento muestra al interesado los siguientes aspectos importantes:

- Archivo y configuraciones necesarias para la personalización de interfaces graficas de usuario.
- Diseño de la plantilla estándar.

5.14.2 Archivos de Configuración

En nuestro sistema las configuraciones necesarias van de acuerdo al patrón MVC, por lo tanto hemos visto necesario el trabajo en estos componentes:

- La presentación XHTML del resultado de la acción.
- Lo otro es los siguientes elementos:
 - ✓ El título de la página: además de ser útil para los usuarios que tienen varias páginas abiertas del navegador también es de fundamental importancia para que los buscadores indexen la página de una manera eficiente.
 - ✓ Inclusión de hojas de estilo.
 - ✓ Layout: algunas acciones necesitan un layout personalizado.

5.14.13 Mapa de navegación

Tomando muy en cuenta los diferentes componentes con los que cuenta el sistema se indica a continuación un mapa de navegación.

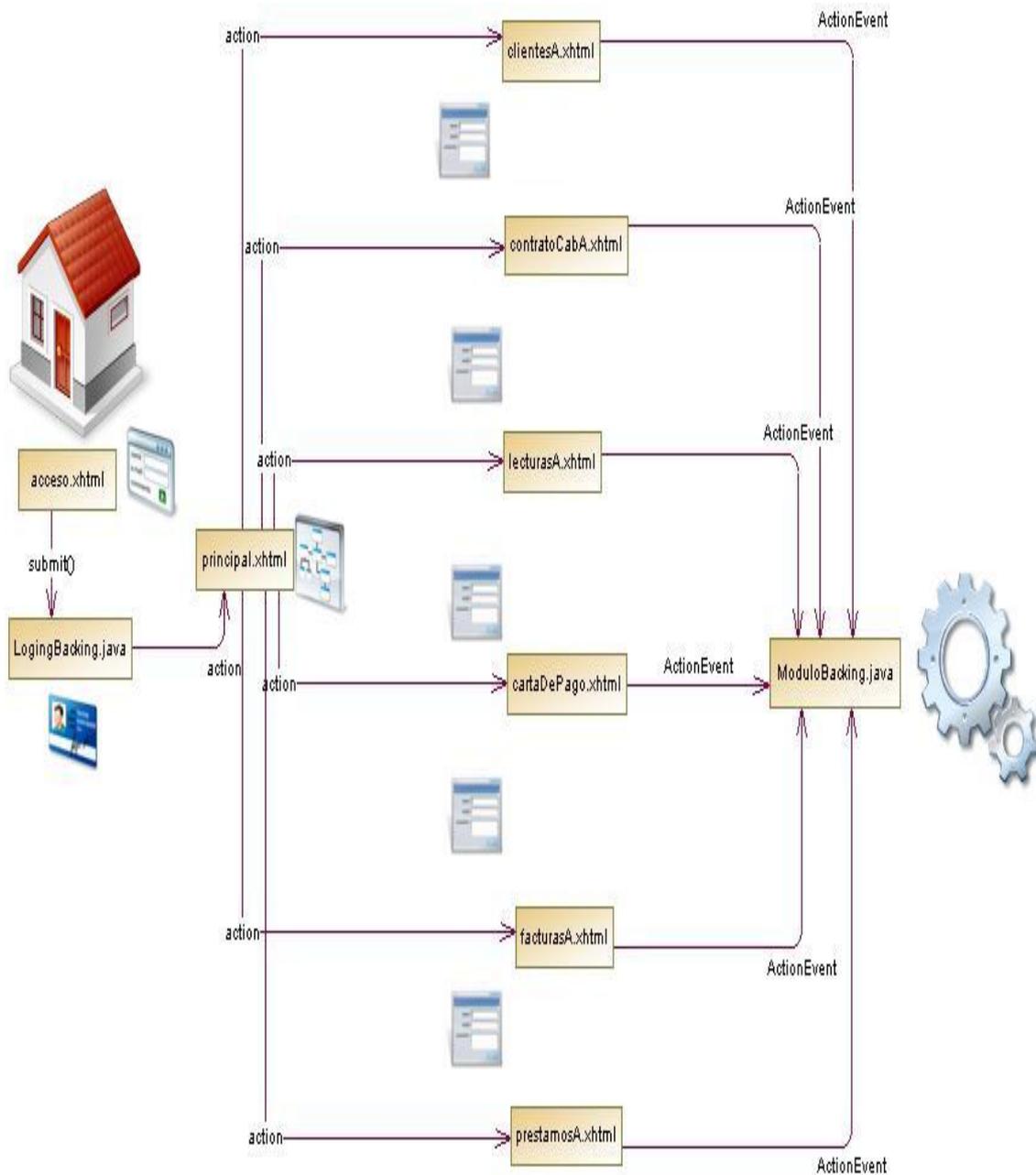


Ilustración 52.- Mapa de Navegación.

Fuente: Andrés Cheza

5.14.4 Estructura de las Páginas

layout.xhtml

La página del menú principal tendrá la siguiente estructura:

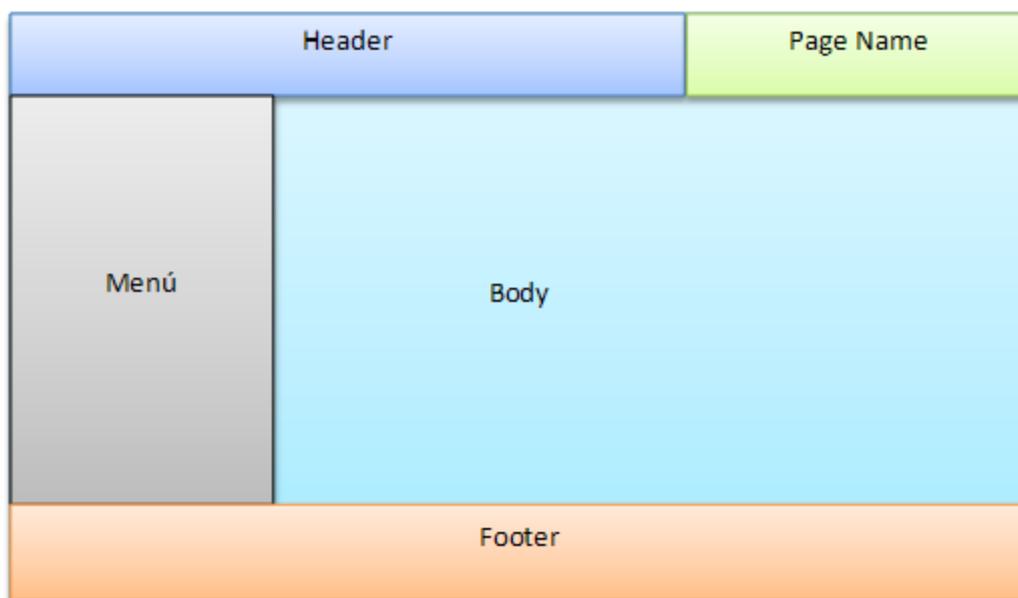


Ilustración 53.- Estructura de la página layout.xhtml
Fuente: Andrés Cheza

Campos	Descripción
Header	Contiene logotipo
Page Name	Muestra el nombre de la página que está siendo visualizada.
Menú	Contiene los distintos menús que serán mostrados por el sistema.
Body	El contenido de esta parte puede cambia respecto a lo que el usuario seleccione.
Footer	Contiene una breve descripción del sistema y muestra los derechos reservados.

Tabla 38.- Estructura de la página layout.xhtml.
Fuente: Andrés Cheza.

En la siguiente sección se muestra un diseño de la estructura de las páginas en formato xhtml con las que cuenta el sistema.

acceso.xhtml

Esta página será la que se muestre al usuario para que ingrese los datos necesarios para acceder al menú principal del sistema, dependiendo los privilegios que tenga el usuario.

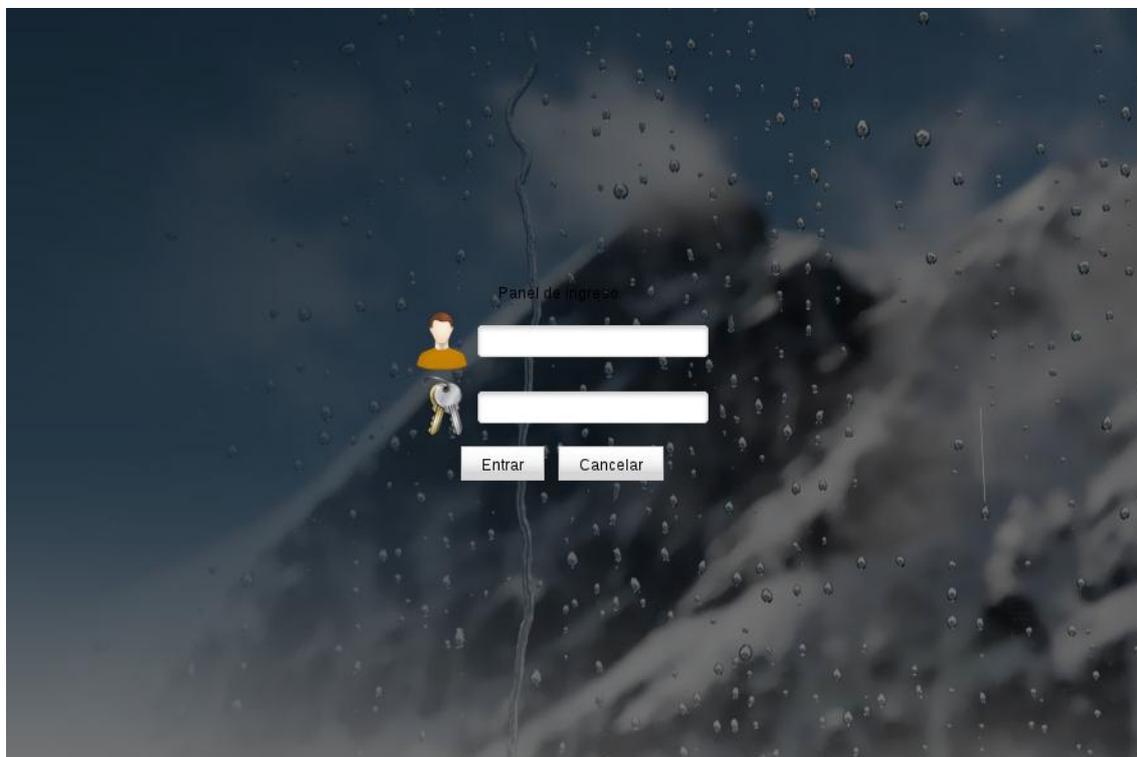


Ilustración 54.- Interfaz de Usuario. Validar Usuario
Fuente: Andrés Cheza

Tipo de Pagina	acceso.xhtml
Descripción	<p>Usuario.- Indica al usuario que debe digitar el nombre de acceso.</p> <p>Password.- Muestra al usuario el campo requerido para su clave secreta.</p> <p>Área de botones.- Indica la acciones que el usuario puede realizar.</p>
Acciones	Validar.- realiza una validación del nombre de usuario y la contraseña o clave secreta.

Tabla 39.- Estructura de Interfaz de Usuario. Validar Usuario.
Fuente: Andrés Cheza.

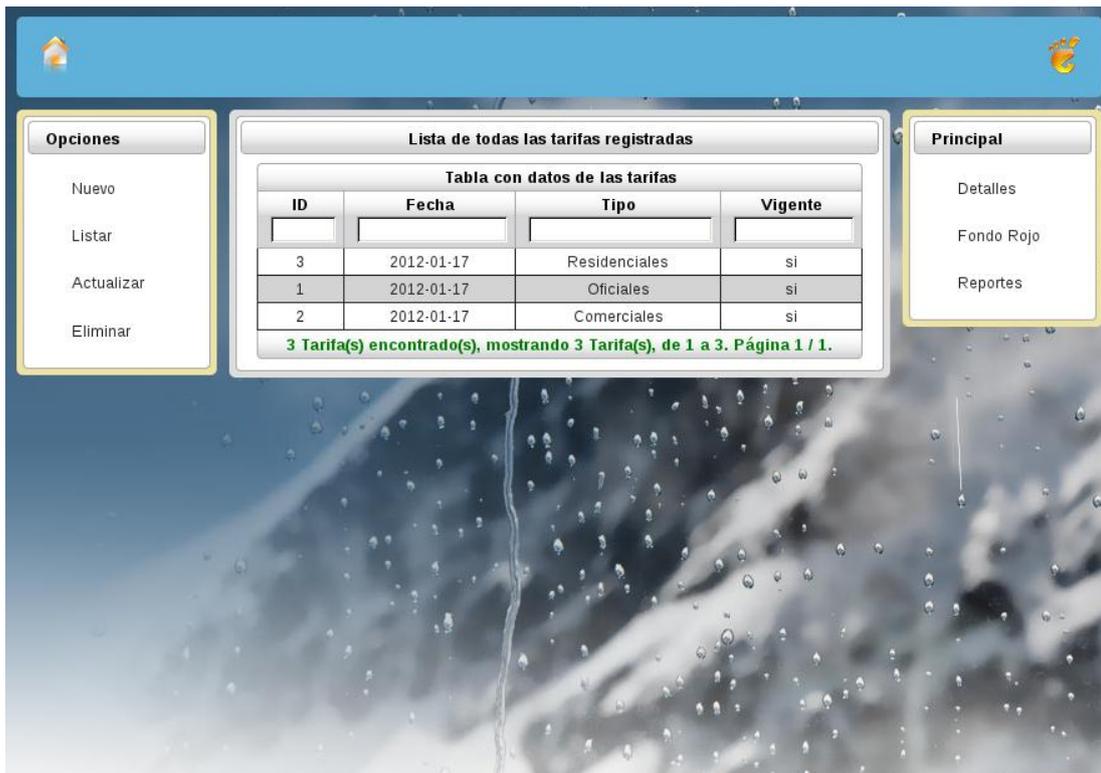


Ilustración 55.- Interfaz de usuario. Pagina moduloA.xhtml
Fuente: Andrés Cheza

Tipo de página	moduloA.xhtml
Descripción	<p>Panel.- Permite realizar una minimización del contenido que se encuentra dentro de dicho panel.</p> <p>Campos.- Son los elementos que permiten ingresar datos para ser almacenarlos por el sistema.</p> <p>Área de Botones.- Permiten realizar acciones de guardar datos y de limpiar el formulario.</p> <p>Campos de una Tabla.- Muestran la información pertinente de los atributos de cada tabla de la base de datos.</p> <p>Buscar.- Permite realizar las búsquedas por cada campo que se desee de una manera más eficiente.</p> <p>Seleccionar.- Al presionar en cualquier fila de la tabla nos permite seleccionarla cambiando así de diferente color individualmente.</p> <p>Paginado.- Nos permite desplazarnos interactivamente por los datos que se encuentran siendo visualizado en la tabla.</p> <p>Footer Table.- Nos muestra el número total de registros que contiene la tabla y cuantos están siendo visualizados en cada vista.</p>
Acciones	Crear.- A través del botón agregar podemos almacenar los datos del formulario.

Tabla 380.- Estructura del Prototipo de Interfaz de Usuario. Página moduloA.xhtml.
Fuente: Andrés Cheza.

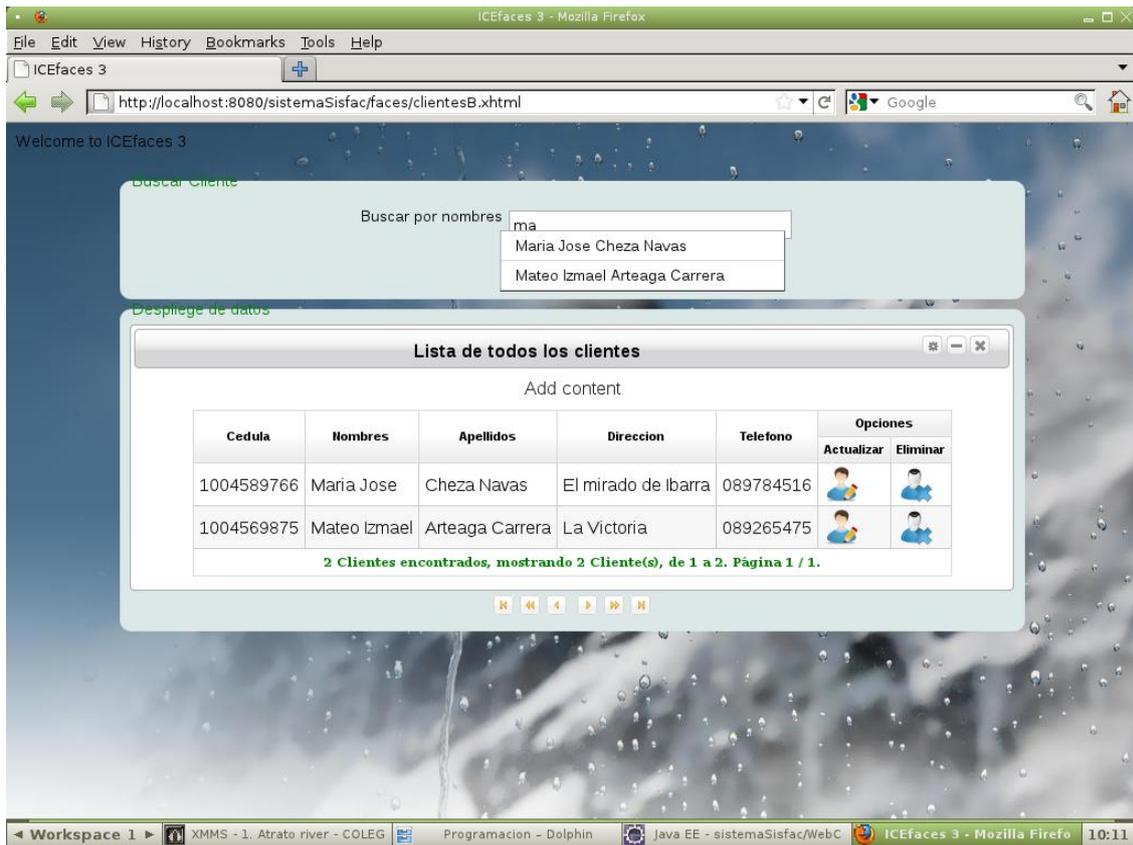


Ilustración 56.- Interfaz de Usuario. Página moduloB.xhtml
Fuente: Andrés Cheza

Tipo de página	moduloA.xhtml
Descripción	<p>Panel.- Permite realizar una minimización del contenido que se encuentra dentro de dicho panel.</p> <p>Área de Links.- Permiten realizar acciones de actualizar y eliminar datos.</p> <p>Campos de una Tabla.- Muestran la información pertinente de los atributos de cada tabla de la base de datos.</p> <p>Buscar.- Permite realizar las búsquedas.</p> <p>Seleccionar.- Al presionar en un campo determinado de la fila de la tabla nos permite seleccionarla capturando los datos de la misma.</p> <p>Paginado.- Nos permite desplazarnos interactivamente por los datos que se encuentran siendo visualizado en la tabla.</p> <p>Footer Table.- Nos muestra el número total de registros que contiene la tabla y cuantos están siendo visualizados en cada vista.</p>
Acciones	<p>Actualizar.- Al escoger el ítem por medio del link del campo de la fila, nos muestra una ventana modal con los datos para ser actualizados.</p> <p>Eliminar.- Al presionar en el campo de la tabla por medio del link nos muestra una ventana de aviso tipo modal para confirmar la eliminación de los datos.</p>

Tabla 41.- Estructura del Prototipo de Interfaz de Usuario. Página moduloB.xhtml.

Fuente: Andrés Cheza.

6. CAPÍTULO VI.- FASE DE TRANSICIÓN.

6.1 Implementación y Pruebas.

6.1.1 Introducción

Para poder conseguir el modelo de implementación se tomó como referencia los componentes principales que representan los componentes físicos, el modelo de componentes que se usarán para construir el sistema. Los cuales fueron analizados en el capítulo anterior en el desarrollo del documento referente a la arquitectura del software.

Para determinar cómo está conformado el sistema físicamente necesariamente utilizaremos el diagrama de componentes visto anteriormente.

6.2 Definición de Subsistemas e Implementación.

En el presente diagrama se presenta y detallan los subsistemas de implementación.

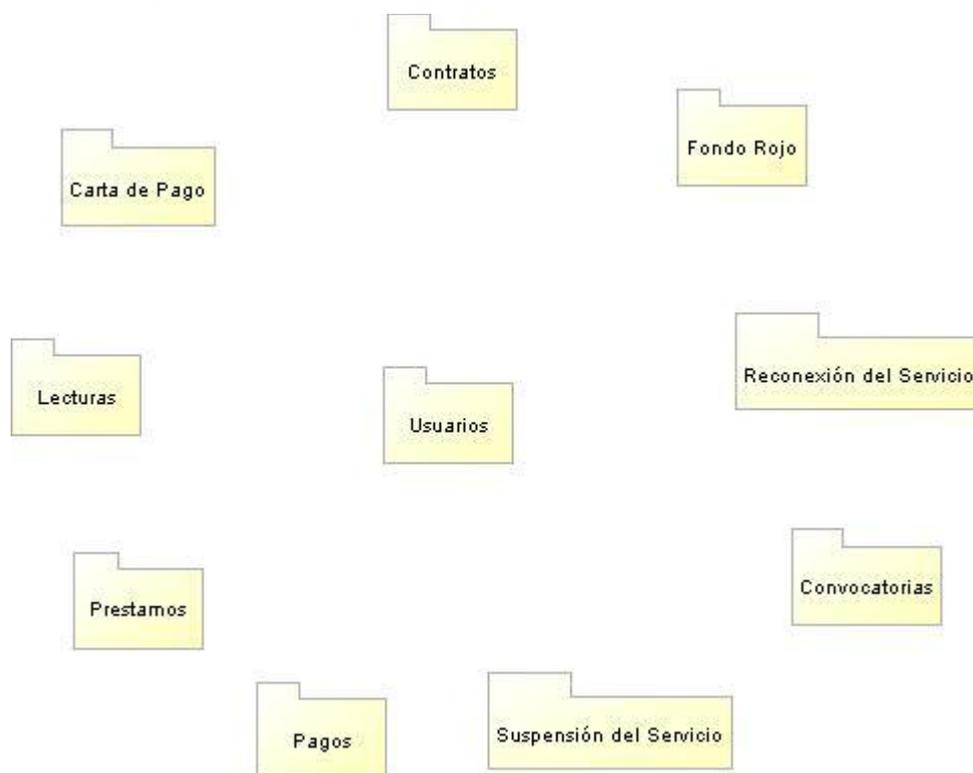


Ilustración 57.- Subsistemas. Módulo Diseño
Fuente: Andrés Cheza

6.3 Modulo Web.

6.3.1 Subsistema Usuarios.

Se conforma por los diversos componentes, los mismos que tienen como propósito la gestión de los usuarios que se encuentren registrados en el sistema, los cuales podrán

utilizar las diferentes funciones con las que cuenta el sistema. Este subsistema incluye las siguientes acciones:

Ingresar al Sistema

Validar Usuario

Mostrar Menú Principal del Sistema

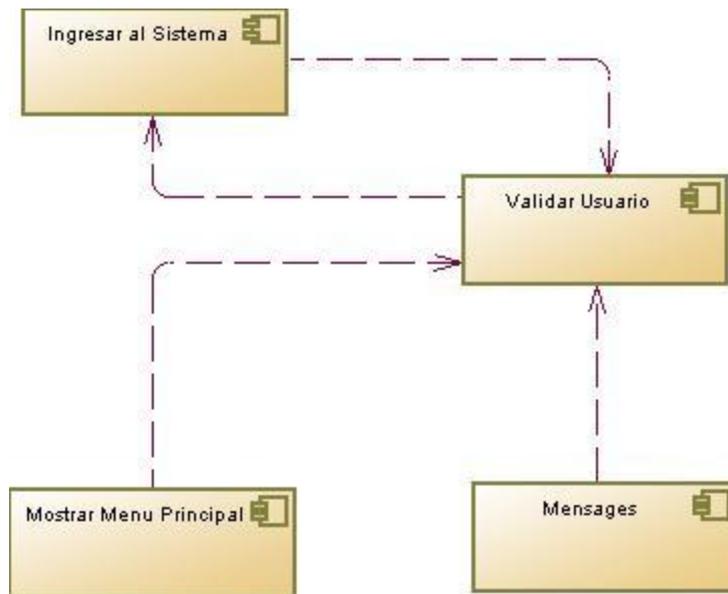


Ilustración 58.- Diagrama de componentes, Subsistema Usuarios.
Fuente: Andrés Cheza

6.3.2 Subsistema Contratos

Este subsistema está formado por varios componentes, los mismos que tienen como objetivo el registro de nuevos clientes y la creación de una cuenta para cada cliente, su ejecución se realizara por el personal de la Junta de Agua Potable, incluye las siguientes acciones:

Solicitar el Servicio de Agua Potable

Crear Contrato

Registrar Cobro por El Servicio de Agua Potable

Generar Factura por el Servicio de Agua Potable

Imprimir Factura por el Servicio de Agua Potable

Actualizar Contrato

Eliminar Contrato

Generar Reporte de Contratos

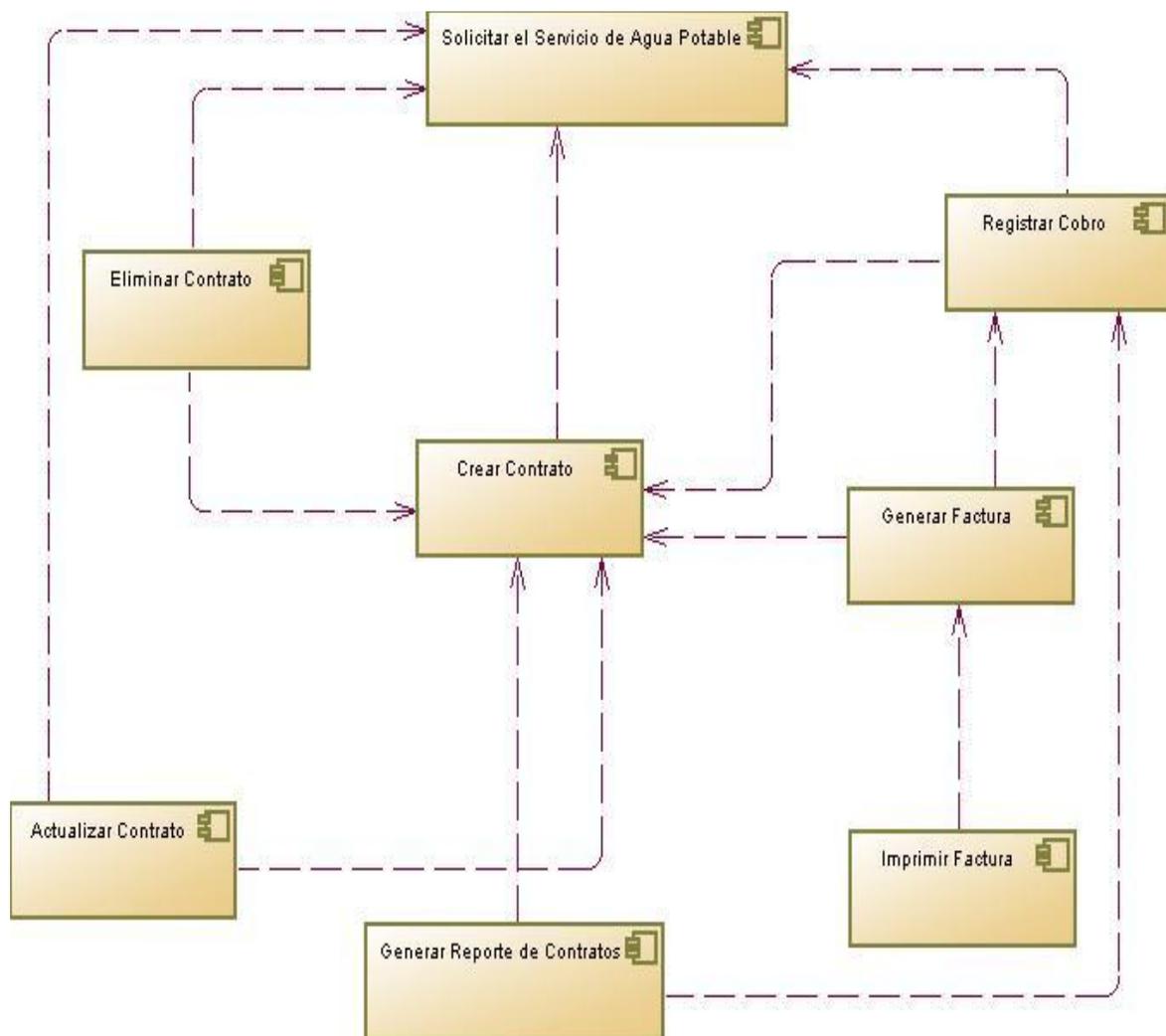


Ilustración 59.- Diagrama de componentes, Subsistema Contratos.

Fuente: Andrés Cheza

6.3.3 Subsistema Lecturas.

El subsistema lecturas está formado por diversos componentes, los cuales tienen como propósito el registro de las lecturas referentes al consumo de agua potable por los clientes, la ejecución se realizará por el encargado del registro de lecturas. Este subsistema incluye las presentes acciones:

Registrar Lectura

Actualizar Lectura

Eliminar Lectura

Generar Reporte Lecturas

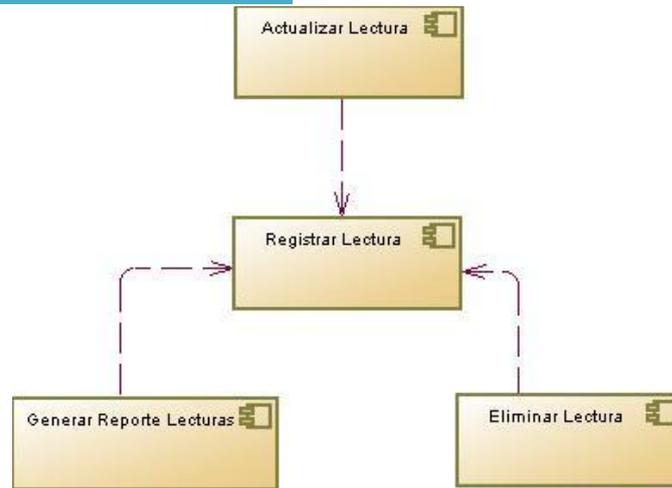


Ilustración 60.- Diagrama de componentes, Subsistema Lecturas
Fuente: Andrés Cheza

6.3.4 Subsistema Carta de Pago.

Se encuentra conformada por varios componentes, los cuales tienen como objetivo la gestión del cobro del consumo del agua potable, su ejecución estará a cargo del personal de la Junta de Agua Potable. Este subsistema contiene las siguientes acciones:

- Pagar Consumo Agua Potable
- Verificar Lectura
- Generar Carta de Pago
- Imprimir Carta de Pago
- Actualizar Carta de Pago
- Eliminar Carta de Pago
- Generar Reporte Carta de Pago

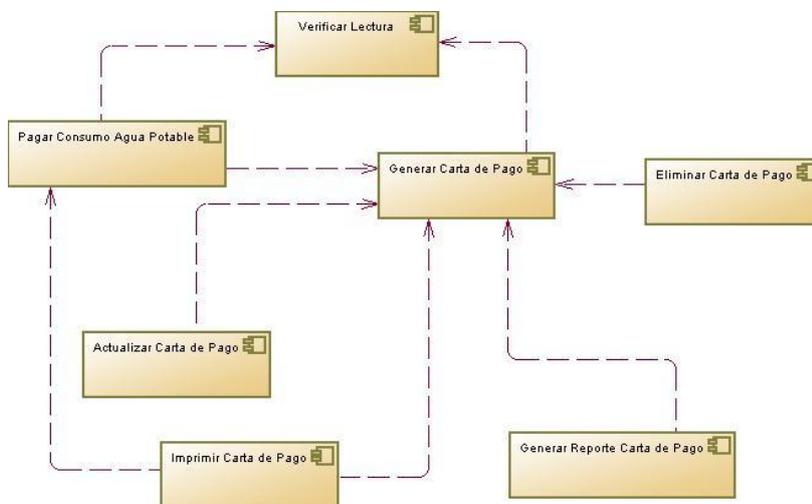


Ilustración 61.- Diagrama de componentes, Subsistema Carta de Pago
Fuente: Andrés Cheza

6.3.5 Subsistema Préstamos.

El subsistema prestamos está conformado por varios componentes, los mismos que tienen como función la gestión de préstamos que realiza la Junta de Agua a sus clientes, será ejecutado por el encargado de realizar los préstamos, además incluye las presentes acciones:

- Solicitar Préstamo**
- Verificar Contrato**
- Registrar Préstamo**
- Entregar Préstamo**
- Actualizar Préstamo**
- Eliminar Préstamo**
- Generar Reporte Préstamo**

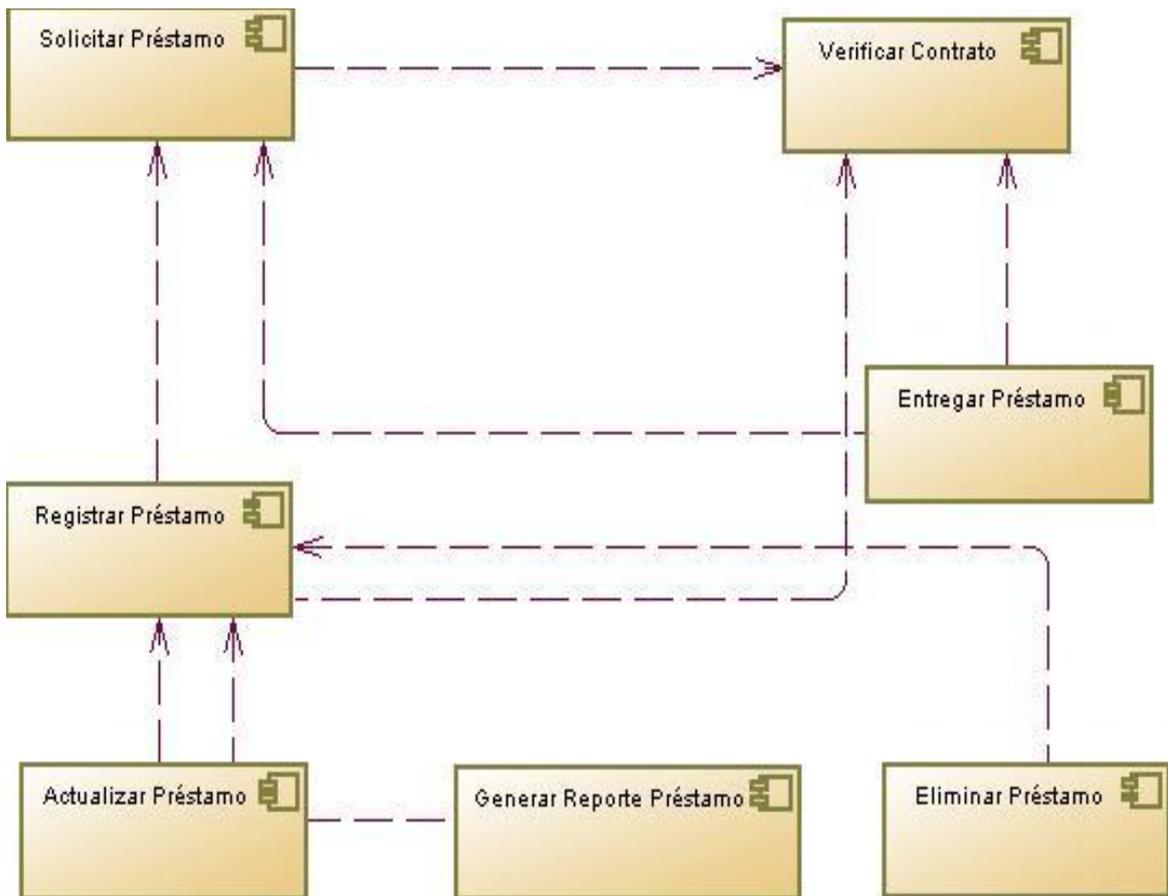


Ilustración 62.- Diagrama de componentes, Subsistema Préstamos.
Fuente: Andrés Cheza

6.3.6 Subsistema Pagos.

Este conformado por diversos componentes, los cuales tienen como objetivo la gestión de los correspondientes pagos realizados en la Junta de Agua Potable. La ejecución es realizada por el personal de la Junta de Agua Potable, incluye las subsiguientes operaciones:

Pagar Préstamo
Verificar Préstamo
Registrar Pago Préstamo
Generar Factura Pago Préstamo
Imprimir Factura Pago Préstamo
Entregar Factura Pago Préstamo
Actualizar Pago Préstamo
Eliminar Pago Préstamo
Generar Reporte Pago Préstamo
Pagar Intereses
Registrar Pago Intereses
Generar Factura Pago Intereses
Imprimir Factura Pago Intereses
Entregar Factura Pago Intereses
Actualizar Pago intereses
Eliminar Pago intereses
Generar Reporte Pago intereses
Pagar Multa
Verificar Multa
Registrar Pago Multa
Generar Factura Pago Multa
Imprimir Factura Pago Multa
Entregar Factura Pago Multa
Actualizar Pago Multa
Eliminar Pago Multa

Generar Reporte Pago Multa

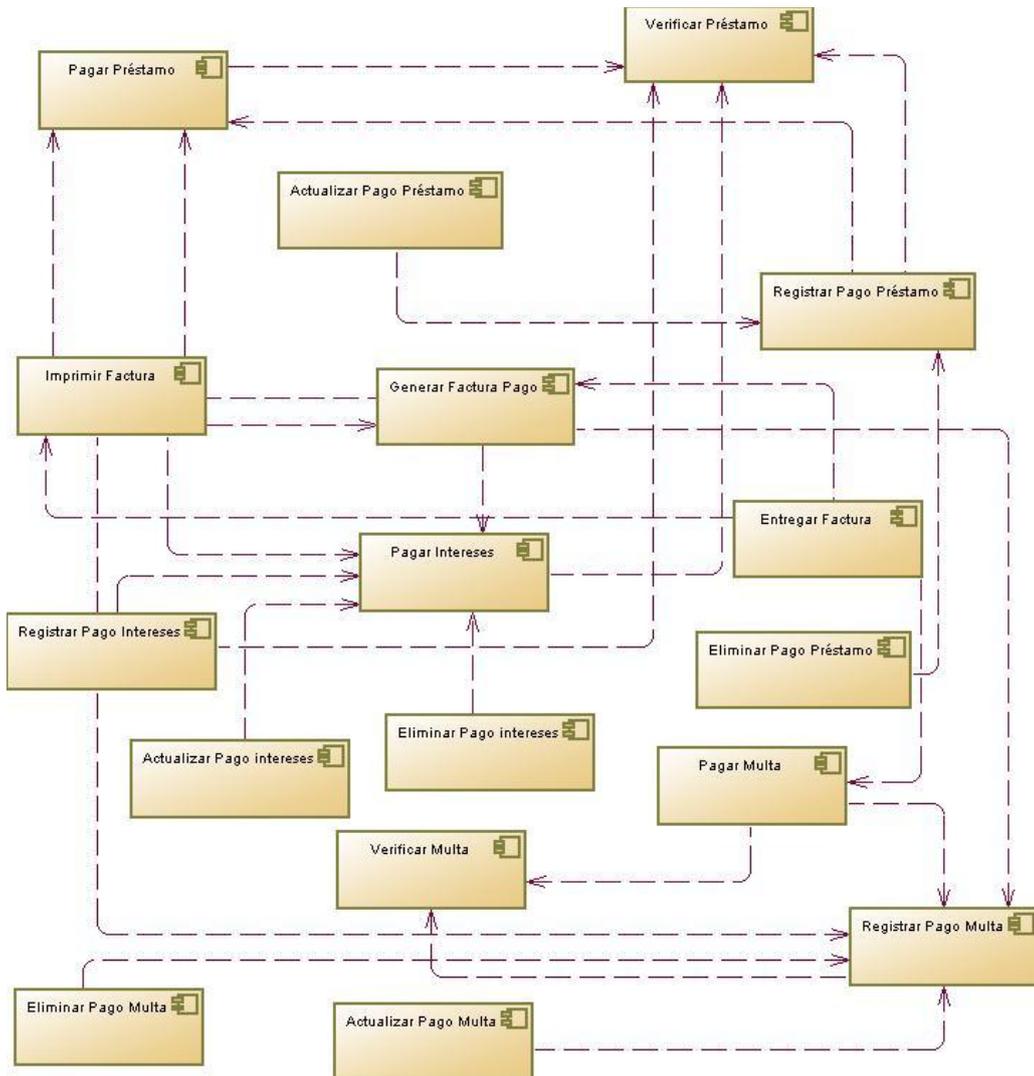


Ilustración 63.- Diagrama de componentes, Subsistema Pagos.
Fuente. Andrés Cheza

6.4 Caso de Prueba

6.4.1 Introducción

En un caso de prueba se especifican y comunican las condiciones específicas que se deben de tomar en cuenta para realizar la validación para luego habilitar una valoración de varios aspectos concretos del sistema que de acuerdo a esto permitirá ver claramente su comportamiento, en la interacción con el usuario.

6.4.1.1 Propósito

El propósito del presente documento es detallar las pruebas a las que se someterá al sistema para verificar su comportamiento.

6.4.1.2 Alcance

Especificar la evaluación que se debe realizar al caso de prueba, el mismo que se determina de los correspondientes casos de uso especificados anteriormente de los cuales en las siguientes secciones se realiza la explicación pertinente.

6.4.2 Casos de Prueba

6.4.2.1 Caso de Prueba para el Caso de Uso: Crear Contrato

Caso de Prueba	Crear Contrato
Entrada	Seleccionamos un cliente que se haya registrado anteriormente.
Resultado	El sistema debe mostrar la información del cliente seleccionado y además registrar los datos del nuevo contrato.
Condiciones	Datos del cliente registrados previamente en el sistema.

Tabla 39.- Caso de Prueba para el Caso de Uso: Crear Contrato.

Fuente: Andrés Cheza.

6.4.2.2 Caso de Prueba para el Caso de Uso: Generar Factura por el Servicio de Agua Potable

Caso de Prueba	Generar Factura por el Servicio de Agua Potable
Entrada	El sistema registró los datos del contrato creados anteriormente.
Resultado	El sistema debe mostrar la información del nuevo contrato registrado en formato pdf con sus correspondientes costos a ser cancelados.
Condiciones	Debe existir el contrato registrado previamente.

Tabla 40.- Caso de Prueba para el Caso de Uso: Generar Factura por el Servicio de Agua Potable.

Fuente: Andrés Cheza.

6.4.2.3 Caso de Prueba para el Caso de Uso: Registrar Lectura

Caso de Prueba	Registrar Lectura
Entrada	Se debe seleccionar el medidor del cliente que se va a registrar la nueva lectura.
Resultado	El sistema debe mostrar los datos del medidor y del cliente, para luego almacenar correctamente la información de la lectura del abonado.
Condiciones	Debe existir el contrato registrado previamente así también los datos del medidor.

Tabla 41.- Caso de Prueba para el Caso de Uso: Registrar Lectura.
Fuente: Andrés Cheza.

6.4.2.4 Caso de Prueba para el Caso de Uso: Pagar Consumo Agua Potable

Caso de Prueba	Pagar Consumo Agua Potable
Entrada	Seleccionamos el medidor del cliente del cual vamos a realizar el pago del consumo del agua potable.
Resultado	El sistema debe mostrar los datos del medidor y del cliente, para inmediatamente generar los costos y la información del consumo mensual del cliente.
Condiciones	Verificación de la lectura del cliente.

Tabla 42.- Caso de Prueba para el Caso de Uso: Pagar Consumo Agua Potable.
Fuente: Andrés Cheza.

6.4.2.5 Caso de Prueba para el Caso de Uso: Verificar Lectura

Caso de Prueba	Verificar Lectura
Entrada	Seleccionamos el medidor del cliente del cual vamos a realizar la validación de la lectura.
Resultado	El sistema debe mostrar los datos del medidor y del cliente, además debe mostrar la lectura especificada por el usuario.
Condiciones	Los datos de la lectura deben estar almacenados en el sistema.

Tabla 43.- Caso de Prueba para el Caso de Uso: Verificar Lectura.
Fuente: Andrés Cheza.

6.4.2.6 Caso de Prueba para el Caso de Uso: Generar Carta de Pago

Caso de Prueba	Generar Carta de Pago
Entrada	Seleccionamos el cliente, el año y el mes del que se va a realizar el cobro.
Resultado	El sistema debe mostrar los datos del medidor y del cliente, además debe mostrar los valores del consumo del mes y los costos a pagar.
Condiciones	Verificación de la lectura del mes.

Tabla 44.- Caso de Prueba para el Caso de Uso: Generar Carta de Pago.
Fuente: Andrés Cheza.

6.4.2.7 Caso de Prueba para el Caso de Uso: Generar Citaciones

Caso de Prueba	Generar Citaciones
Entrada	Seleccionamos algún cliente registrado previamente.
Resultado	El sistema debe mostrar los datos del cliente, además debe activar los casilleros para el ingreso de datos y posteriormente el sistema almacenará los mismos.
Condiciones	Se debe realizar la validación de que el cliente no ha realizado el pago del consumo de los meses.

Tabla 45.- Caso de Prueba para el Caso de Uso: Generar Citaciones.
Fuente: Andrés Cheza.

6.4.2.8 Caso de Prueba para el Caso de Uso: Registrar Suspensión

Caso de Prueba	Registrar Suspensión
Entrada	Seleccionamos algún cliente registrado previamente y que tenga las correspondientes citaciones emitidas.
Resultado	El sistema debe mostrar los datos del cliente, además debe activar los casilleros para el ingreso de datos y posteriormente el sistema almacenará los mismos.
Condiciones	Se debe validar que se encuentren registradas las correspondientes citaciones al cliente seleccionado.

Tabla 46.- Caso de Prueba para el Caso de Uso: Registrar Suspensión.
Fuente: Andrés Cheza.

6.4.2.9 Caso de Prueba para el Caso de Uso: Registrar Convocatoria

Caso de Prueba	Registrar Convocatoria
Entrada	Seleccionamos la fecha.
Resultado	El sistema debe activar los casilleros para el ingreso de datos y posteriormente el sistema almacenará los mismos.
Condiciones	Registrar convocatoria

Tabla 47.- Caso de Prueba para el Caso de Uso: Registrar Convocatoria.
Fuente: Andrés Cheza.

6.4.2.10 Caso de Prueba para el Caso de Uso: Solicitar Préstamo

Caso de Prueba	Solicitar Préstamo
Entrada	Seleccionamos el sector luego al cliente.
Resultado	El sistema nos mostrará los datos del cliente, además debe activar los casilleros para el ingreso de datos y posteriormente el sistema almacenará los mismos.
Condiciones	El cliente debe contar con un contrato registrado previamente.

Tabla 48.- Caso de Prueba para el Caso de Uso: Solicitar Préstamo.
Fuente: Andrés Cheza.

6.4.2.11 Caso de Prueba para el Caso de Uso: Verificar Contrato

Caso de Prueba	Verificar Contrato
Entrada	Seleccionamos el sector luego y después elegimos seleccionamos el contrato.
Resultado	El sistema nos mostrará los datos del cliente.
Condiciones	El cliente debe tener registrado en el sistema un contrato.

Tabla 49.- Caso de Prueba para el Caso de Uso: Verificar Contrato.
Fuente: Andrés Cheza.

6.4.2.12 Caso de Prueba para el Caso de Uso: Registrar Préstamo

Caso de Prueba	Registrar Préstamo
Entrada	Seleccionamos un cliente registrado previamente.
Resultado	El sistema nos mostrará los datos del cliente y además activara los casilleros necesarios para registrar los datos del préstamo.
Condiciones	Validación del contrato para el cliente.

Tabla 50.- Caso de Prueba para el Caso de Uso: Registrar Préstamo.
Fuente: Andrés Cheza.

6.4.2.13 Caso de Prueba para el Caso de Uso: Pagar Préstamo

Caso de Prueba	Pagar Préstamo
Entrada	Seleccionamos un sector.
Resultado	El sistema nos mostrará los clientes que han realizado algún préstamo.
Condiciones	Validación del préstamo por parte de un cliente.

Tabla 51.- Caso de Prueba para el Caso de Uso: Pagar Préstamo.
Fuente: Andrés Cheza.

6.4.2.14 Caso de Prueba para el Caso de Uso: Verificar Préstamo

Caso de Prueba	Verificar Préstamo
Entrada	Seleccionamos un sector y elegimos algún cliente registrado previamente.
Resultado	El sistema nos mostrará los datos del cliente y el préstamo.
Condiciones	La información referente al préstamo del cliente debe estar registrada en el sistema.

Tabla 52.- Caso de Prueba para el Caso de Uso: Verificar Préstamo.
Fuente: Andrés Cheza.

6.4.2.15 Caso de Prueba para el Caso de Uso: Registrar Pago Préstamo

Caso de Prueba	Registrar Pago Préstamo
Entrada	Seleccionamos un algún cliente registrado previamente.
Resultado	El sistema nos mostrará los datos del cliente y activará los casilleros necesarios para el ingreso de datos, luego el sistema procederá a almacenarlos correctamente.
Condiciones	Se debe haber realizado la validación de los datos del préstamo de algún cliente registrado en el sistema.

Tabla 53.- Caso de Prueba para el Caso de Uso: Registrar Pago Préstamo.
Fuente: Andrés Cheza.

6.4.2.16 Caso de Prueba para el Caso de Uso: Pagar Intereses

Caso de Prueba	Pagar Intereses
Entrada	Seleccionamos un sector.
Resultado	El sistema nos mostrará los clientes que han realizado algún préstamo.
Condiciones	Validación del préstamo por parte de un cliente.

Tabla 54.- Caso de Prueba para el Caso de Uso: Pagar Intereses.
Fuente: Andrés Cheza.

6.4.2.17 Caso de Prueba para el Caso de Uso: Registrar Pago Intereses

Caso de Prueba	Registrar Pago Intereses
Entrada	Seleccionamos un algún cliente registrado previamente.
Resultado	El sistema nos mostrará los datos del cliente y activará los casilleros necesarios para el ingreso de datos, luego el sistema procederá a almacenarlos correctamente.
Condiciones	Se debe haber realizado la validación de los datos del préstamo de algún cliente registrado en el sistema.

Tabla 55.- Caso de Prueba para el Caso de Uso: Registrar Pago Intereses.
Fuente: Andrés Cheza.

7. CAPÍTULO VII.- CONCLUSIONES Y RECOMENDACIONES

7.1 CONCLUSIONES.

Al terminar el proceso de desarrollo del presente aplicativo he logrado cumplir con el objetivo planteado al inicio del proyecto de desarrollar un sistema informático de facturación y registro de la información que será de suma utilidad para la “JUNTA ADMINISTRADORA DE AGUA POTABLE Y ALCANTARILLADO MIRADOR DEL OLIVO”.

Además se cumplieron los objetivos específicos de:

- Precisar, delimitar y analizar el proceso de facturación y registro de la información en la junta de agua.
- Plantear y estudiar las posibles soluciones informáticas para la solución del problema.
- Seleccionar e investigar las herramientas informáticas a utilizarse en el desarrollo de la solución informática.
- Realizar el diseño minucioso de la arquitectura y funcionamiento de la solución planteada.
- Desarrollar y probar el sistema hasta lograr los resultados esperados.
- Capacitar a los usuarios finales del sistema de facturación y registro de la información.
- Elaboración de la documentación pertinente relacionada con el uso y buen rendimiento del sistema.

Se realiza la entrega a la junta de agua una herramienta que permita optimizar el manejo de la información de los clientes y evitar la pérdida o duplicidad de dicha información.

Vale la pena mencionar que el uso de herramientas de software libre nos permitió conocer más afondo dichas herramientas y tener un alto desenvolvimiento y conocimientos en el tema de desarrollo web. Al utilizar la tecnología JSF con los frameworks ICEfaces y PrimeFaces para la parte de interfaz de usuario nos pudimos dar cuenta de la disminución de tiempo que toma en desarrollar las interfaces de usuario que si lo haríamos desde código css, javascript o ajax.

7.2 RECOMENDACIONES.

El presente trabajo da solución al problema planteado inicialmente y satisface los objetivos trazados al inicio, sin embargo existe un gran sin número de cosas que pueden ser incorporados en un futuro para que su eficiencia sea aún mayor, como por ejemplo

con las futuras versiones de las librerías tanto de ICEfaces como de PrimeFaces se tendrá muchas más funcionalidades que las que tenemos al momento.

Se recomienda tener un apropiado manejo de la documentación debido a que si en un futuro se realizaría un posible cambio en el sistema se tenga en claro donde realizar las modificaciones pertinentes.

A medida que se fue desarrollando el proyecto se pudo observar que la metodología de desarrollo de software RUP nos permite llevar una correcta organización del proyecto, además brinda una clara visión de las necesidades que tienen los usuarios.

A los lectores recomendamos el uso de herramientas de software libre como lo son PostgreSQL, JSF, Hibernate, ICEfaces y PrimeFaces por su alto desempeño y optimización de tiempo en el desarrollo de un proyecto.

En el uso de la tecnología JSF se recomienda utilizar los frameworks ICEfaces o PrimeFaces o los dos conjuntamente ya que para realizar el desarrollo de la interfaz gráfica de usuario nos brindan un considerable ahorro de tiempo ya que cuentan con formularios, etiquetas, cajas de mensajes, botones prediseñados con un alto nivel de desempeño y rendimiento evitándonos escribir código css, javascript o ajax. Por otro lado invitamos a investigar a fondo estos frameworks ya que cuentan con un sin número de posibilidades que podemos hacer uso y llegar a tener una excelente implementación en nuestra aplicación.

Para desarrollar un proyecto de software recomendamos hacer uso del patrón de diseño MVC, ya que de esta manera contaremos con un importante apoyo en la implementación de la aplicación para poder tener en un futuro un mantenimiento fácil del sistema.

GLOSARIO

8. GLOSARIO

Actor (UML)

Especifica un rol jugado por un usuario o cualquier otro sistema que interactúa con el sujeto.

UML

Lenguaje Unificado de Modelado

Aplicación web

Es una aplicación software que se codifica en un lenguaje soportado por los navegadores web en la que se confía la ejecución al navegador.

Base de datos

Una **base de datos** o **banco de datos** es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta.

Caso de uso

Un **caso de uso** es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso. Los personajes o entidades que participarán en un caso de uso se denominan actores. En el contexto de ingeniería del software, un caso de uso es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema. Los diagramas de casos de uso sirven para especificar la comunicación y el comportamiento de un sistema mediante su interacción con los usuarios y/u otros sistemas. O lo que es igual, un diagrama que muestra la relación entre los actores y los casos de uso en un sistema.

Clase (informática)

Una clase es una construcción que se utiliza como un modelo (o plantilla) para crear objetos de ese tipo. El modelo describe el estado y el comportamiento que todos los objetos de la clase comparten.

Complemento (plug-in)

Un complemento es una aplicación que se relaciona con otra para aportarle una función nueva y generalmente muy específica. Esta aplicación adicional es ejecutada por la aplicación principal e interactúan por medio de la API.

Diagrama de clases

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de

clases son utilizados durante el proceso de análisis y diseño de los sistemas, donde se crea el diseño conceptual de la información que se manejará en el sistema, y los componentes que se encargaran del funcionamiento y la relación entre uno y otro.

Diagrama de secuencia

El diagrama de secuencia es un tipo de diagrama usado para modelar interacción entre objetos en un sistema según UML.

Flujo de trabajo

Es el estudio de los aspectos operacionales de una actividad de trabajo: cómo se estructuran las tareas, cómo se realizan, cuál es su orden correlativo, cómo se sincronizan, cómo fluye la información que soporta las tareas y cómo se le hace seguimiento al cumplimiento de las tareas.

Framework

La palabra inglesa "**framework**" (marco de trabajo) define, en términos generales, un conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

Front-end y back-end

El front-end es la parte del software que interactúa con el o los usuarios y el back-end es la parte que procesa la entrada desde el front-end. La separación del sistema en "front ends" y "back ends" es un tipo de abstracción que ayuda a mantener las diferentes partes del sistema separadas. La idea general es que el front-end sea el responsable de recolectar los datos de entrada del usuario, que pueden ser de muchas y variadas formas, y procesarlas de una manera conforme a la especificación que el back-end pueda usar.

Herramienta CASE

Las **herramientas CASE** (*Computer Aided Software Engineering*, Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero. Estas herramientas pueden ayudar en todos los aspectos del ciclo de vida de desarrollo del software en tareas como el proceso de realizar un diseño del proyecto, cálculo de costos, implementación de parte del código automáticamente con el diseño dado, compilación automática, documentación o detección de errores entre otras.

ICEFaces

Es un framework de código abierto para construir aplicaciones web con AJAX tipo RIA. Permite al programador incluir una serie de Ajax-tags en sus JSP o xhtml de tal manera que el código Ajax es generado por el propio framework automáticamente.

Interfaz de usuario

Es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.

JSF

Java Server Faces.

Metodología

Hace referencia al conjunto de procedimientos racionales utilizados para alcanzar una gama de objetivos que rigen en una investigación científica. Alternativamente puede definirse la *metodología* como el estudio o elección de un método pertinente para un determinado objetivo.

Mitigación

El propósito de la mitigación es la reducción de la vulnerabilidad, es decir la atenuación de los daños potenciales sobre la vida y los bienes causados por un evento.

MVC

Modelo Vista Controlador.

Modelo Vista Controlador

Es un patrón o modelo de abstracción de desarrollo de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de negocio en tres componentes distintos. El patrón de llamada y retorno MVC (según CMU), se ve frecuentemente en aplicaciones web, donde la vista es la página HTML y el código que provee de datos dinámicos a la página. El modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio, y el controlador es el responsable de recibir los eventos de entrada desde la vista.

PrimeFaces

Es un componente para Java Server Faces (JSF) de código abierto que cuenta con un conjunto de componentes ricos que facilitan la creación de las aplicaciones web. Primefaces está bajo la licencia de Apache License V2. Una de las ventajas de utilizar Primefaces, es que permite la integración con otros componentes como por ejemplo RichFaces.

Proceso Unificado de Rational

El Proceso Unificado de Racional (RUP) es un proceso de desarrollo de software desarrollado por la empresa Rational Software, actualmente propiedad de IBM. Junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, diseño, implementación y documentación de sistemas orientados a objetos.

Recurso

Un **recurso** es una fuente o suministro del cual se produce un beneficio. Normalmente, los recursos son material u otros activos que son transformados para producir beneficio y en el proceso pueden ser consumidos o no estar más disponibles.

Riesgo

Es la vulnerabilidad ante un posible o potencial perjuicio o daño que puede surgir por un proceso presente o suceso futuro.

RIA

Rich, Internet, Application.

RUP

Rational Unified Process

Stakeholder

Es aquella persona o entidad que está en el funcionamiento del sistema de software.

Ejemplo usuario, cliente.

SISFAC

Sistema de Facturación y Registro de la Información.

Usuario

Es aquella persona que utiliza o trabaja con algún objeto o que es destinataria de algún servicio público o privado, empresarial o profesional. En este caso será la persona involucrada en el manejo de un software.

REFERENCIAS

9. REFERENCIAS

9.1 SITIOS WEB

Base de datos. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Base_de_datos

Actor (UML). (2012, 12 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 12 de abril de 2012 a las 10:20 de http://es.wikipedia.org/wiki/Actor_%28UML%29

Proceso Unificado de Rational. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 10:25 de http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational

Modelo Vista Controlador. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 10:25 de http://es.wikipedia.org/wiki/Modelo_Vista_Controlador

Front-end y back-end. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Front-end_y_back-end

Caso de uso. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Caso_de_uso

Clase (informática). (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Clase_%28inform%C3%A1tica%29

Complemento (informática). (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Complemento_%28inform%C3%A1tica%29

Diagrama de secuencia. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Diagrama_de_secuencia

Diagrama de flujo. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Diagrama_de_flujo

Diagrama de clases. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Diagrama_de_clases

Flujo de trabajo. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Flujo_de_trabajo

Framework. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de <http://es.wikipedia.org/wiki/Framework>

Herramienta CASE. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Herramienta_CASE

Interfaz de usuario. (2012, 21 de Abril) En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de abril de 2012 a las 09:17 de http://es.wikipedia.org/wiki/Interfaz_de_usuario

9.2 TESIS

Bucheli Gabriel (2011). Sistema de Información y Administración de Bienes Informáticos. Tesis de ingeniería en sistemas. UTN. Ibarra.

Suárez María (2010). Diseño e implementación de un sistema de facturación de Agua Potable para las parroquias Rurales del cantón Otavalo. Tesis de ingeniería en sistemas. UTN. Ibarra.

9.3 LIBROS

Kito M. (2005). JAVA SERVER FACES IN ACTION: Manning Publications Co.

John W. (2004). Mastering JavaServer Faces. Wiley Publishing, In.

Stephen S. R. (2005). Análisis y Diseño Orientado a Objetos con UML y el proceso unificado: Mc Graw Hill.

Martin C. R (2004). UML para Programadores Java. Person Educación.

Castaño M., y Paloma M. (2001). Diseño de Bases de Datos Problemas Resueltos: Alfaomega.

Herbert S. (2005). La biblia de Java 2 v5.0: The Mc Graw-Hill Companies.

Roger P. S. (2005). Ingeniería del Software: Mc Graw-Hill Interamericana.

Bruce E. (2007). Piensa En JAVA. Cuarta Edición: Pearson Education.

J. Paul, Harvey D. (2008). JAVA COMO PROGRAMAR: Pearson Educación.

Jim K. (2002). J2EE Manual de referencias: Mc Graw Hill Interamericana.

Howard P. (2010). PROGRAMACION UML: Business Analyst. Second Edition.

Martin F.,y Scott K. (2003). UML Distilled. A Brief Guide to the Standard Object Modeling Language, Second Edition: Addison-Wesley Professional.

Craig L. (2004). Applying UML and Patterns An Introduction to Object-Oriented Analysis and Design and Interactive Development: Prentice Hall PTR.

Grady B. (2005). The Unified Modeling Language User Guide: Addison-Wesley Professional.

ANEXOS

10. ANEXOS

10.1 Diccionario de datos.

10.1.1 Esquema de la base de datos.

Sisfac.

Esquema utilizado para el Sistema de Facturación y Registro de la Información.

10.1.2 Tablas.

Nombre	Descripción
pago_consumo	En la actual tabla se registrará el pago del consumo del agua potable.
lecturas	En la presente tabla se almacenará los datos correspondientes a las lecturas de cada medidor.
factura_cabecera	La tabla actual nos permitirá registrar los datos de la cabecera de una factura.
dontrato_detalle	En esta tabla se registran los detalles de los contratos creados previamente.
medidores	En la actual tabla nos permitirá registrar los datos correspondientes al medidor de agua potable.
factura_Detalle	En la siguiente tabla se registrará el detalle de la factura.
clientes	En esta tabla se registrará los datos personales de los clientes
contrato_cabecera	Contiene las cabeceras de los contratos que se crean para un cliente en el sistema.
multas	La presente tabla nos permitirá realizar el registro de las multas emitidas a un cliente.
inspeccion	Esta tabla nos permitirá registrar la inspección que se realiza previamente a la instalación del medidor de agua potable.
asambleas	En la presente tabla se realizará el registro de las asambleas que se crearan.
citaciones	En esta tabla se registrarán los datos de las citaciones que se emiten a un contrato.
convocatorias	En la presente tabla se almacena las convocatorias que se crean.

mingas	En la presente tabla se registrará los datos de las mingas, que se realicen.
fondo_rojo	
prestamos	En esta tabla se registrará los datos de los préstamos emitidos a los clientes.
suspension_servicio	La presente tabla nos permite registrar las correspondientes suspensiones del servicio de agua potable.
asamblea_detalle	En la presente tabla se registrará los detalles de las asambleas desarrolladas.
persona	En esta tabla se almacenara los datos personales de las personas que serán registradas en el sistema.
roles	En la presente tabla se registrara los privilegios que tendrá un usuario en el sistema.
usuarios	En esta tabla nos permitirá registrar los datos de los usuarios que tendrán acceso al sistema dependiendo los privilegios.
pagosPrestamo	La presente tabla nos permitirá registrar los datos de los pagos de los préstamos realizados por los clientes.
intereses_prestamo	En la presente tabla se realizará el registro de los datos de los pagos del interés de los préstamos solicitados por los clientes.
empleados	En la presente tabla se registrara los datos de los empleados que laboran en la institución
detalle_tarifa	En esta tabla se registrara los detalles de cada tarifa registradas en la tabla tarifa.
horas	En la presente tabla se realizara el registro de las horas que un empleado labora en la institución.
tarifa	Esta tabla nos permitirá registrar la cabecera de las tarifas que se manejan en la junta de agua potable.
contratos	En la actual tabla se almacenaran los datos de los contratos que se crea cuando un nuevo empleado ingresa a laborar en la institución.

Tabla A. 1.- Resumen de las tablas del sistema sisfac.

Fuente: Andrés Cheza.

10.2 Modelo Entidad – Relación.

