

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

**DESARROLLO DE UN SISTEMA WEB PARA LA
ADMINISTRACIÓN DE UN CONDOMINIO**

FELIPE ALEJANDRO VALENZUELA LARA

INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO DE EJECUCIÓN EN INFORMÁTICA

Enero, 2014

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

DESARROLLO DE UN SISTEMA WEB PARA LA ADMINISTRACIÓN DE UN CONDOMINIO

FELIPE ALEJANDRO VALENZUELA LARA

Profesor Guía: **Pamela Hermosilla Monckton**
Profesor Co-referente: **Iván Mercado Bermúdez**

Enero, 2014

AGRADECIMIENTOS

A mis padres por su apoyo, por su ejemplo de perseverancia y constancia que ha sido el impulso para lograr mis metas, a mi abuela Juana por su infinita bondad y amor y a todos los que participaron de este proceso. Gracias a todos.

Índice

GLOSARIO DE TÉRMINOS.....	V
LISTA DE FIGURAS.....	VI
LISTA DE TABLAS.....	VIII
RESUMEN	IX
ABSTRACT.....	X
1 INTRODUCCIÓN.....	1
2 DESCRIPCIÓN DEL SISTEMA ACTUAL.....	2
2.1 DIAGNÓSTICO DE LA SITUACIÓN ACTUAL.....	2
3 ESTADO DEL ARTE	3
3.1 ANALIZANDO SOFTWARE KASTOR.....	3
4 DEFINICIÓN DE LOS OBJETIVOS	4
4.1 OBJETIVO GENERAL	4
4.2 OBJETIVOS ESPECÍFICOS	4
5 DESCRIPCIÓN DE LA SOLUCIÓN	5
5.1 PROPUESTA DE SOLUCIÓN	5
5.2 VENTAJAS Y BENEFICIOS DE DESARROLLAR EL SISTEMA.....	5
5.3 PARADIGMA DE DESARROLLO.....	6
5.4 METODOLOGÍA.....	6
5.4.1 Estructurada	6
5.4.2 Orientada a objetos	6
5.4.3 Elección de la metodología.....	6
5.5 ARQUITECTURA LÓGICA.....	7
5.6 ARQUITECTURA FÍSICA	8
6 ESTUDIO DE FACTIBILIDAD.....	9
6.1 FACTIBILIDAD MERCADO.....	9
6.2 FACTIBILIDAD TÉCNICA:	9
6.2.1 Hardware.....	9
6.2.2 Software	10
6.2.3 Recursos Humanos	10
6.3 FACTIBILIDAD ECONÓMICA:	10
6.4 FACTIBILIDAD LEGAL:	10
6.5 FACTIBILIDAD OPERATIVA:.....	10
7 ANÁLISIS DE RIESGOS.....	11
7.1 IDENTIFICACIÓN	11
7.2 MITIGACIÓN Y CONTINGENCIA	12
8 ESPECIFICACIÓN DE REQUERIMIENTOS	13
8.1 REQUERIMIENTOS FUNCIONALES	13
8.1.1 Administrador	13
8.1.2 Copropietario.....	14
8.2 REQUERIMIENTOS NO FUNCIONALES	15
8.3 DEFINICIÓN DE ACTORES	15

9	ANÁLISIS Y DISEÑO DEL SISTEMA.....	16
9.1	DIAGRAMA CASO USO GENERAL	16
9.2	CASO DE USO ADMINISTRAR GASTOS COMUNES.....	17
9.3	DIAGRAMA DE SECUENCIA AUTENTICAR USUARIO.....	17
9.4	DIAGRAMA DE ACTIVIDAD PRORRATEAR GASTO COMÚN.....	18
9.5	CASO DE USO GESTIONAR EXISTENCIAS.....	19
9.6	DIAGRAMA DE ACTIVIDAD INGRESAR EXISTENCIA	19
9.7	CASO DE USO GESTIONAR UNIDADES	20
9.8	DIAGRAMA DE ACTIVIDAD INGRESAR MONTO MASIVO UNIDADES.....	21
9.9	CASO DE USO ADMINISTRAR USUARIOS	22
9.10	CASO DE USO GESTIONAR PROVEEDORES	22
9.11	CASO USO GESTIONAR COMUNICATIVOS	23
9.12	CASO DE USO ADMINISTRAR CUENTAS.....	23
9.13	CASO DE USO GESTIONAR CONDOMINIO	24
9.14	CASO DE USO GESTIONAR PERFIL.....	24
9.15	CASO DE USO CONSULTAR GASTOS COMUNES	25
9.16	CASO DE USO ENVIAR MENSAJES	25
9.17	DIAGRAMA DE CLASES	26
9.18	MODELO DE DATOS LÓGICO.....	27
10	IMPLEMENTACIÓN	28
11	DISEÑO DE INTERFAZ	29
12	PLAN DE PRUEBA	30
12.1	OBJETIVOS.....	30
12.2	ENFOQUE DE LAS PRUEBAS.....	30
12.3	PLANIFICACIÓN DE LAS PRUEBAS:	30
13	CONCLUSIONES.....	31
14	REFERENCIAS	32
15	ANEXOS	33
ANEXO A:	ESTADO DEL ARTE.....	34
ANEXO B:	DESCRIPCIÓN DE LA SOLUCIÓN.....	41
ANEXO C:	FACTIBILIDAD.....	45
ANEXO D:	CASOS DE USO NARRATIVOS	49
ANEXO E:	ANÁLISIS Y DISEÑO DEL SISTEMA	57
ANEXO F:	MODELO LÓGICO DE BASE DE DATOS	61
ANEXO G:	DISEÑO DE LA INTERFAZ	67
ANEXO H:	CASOS DE PRUEBA	76

Glosario de Términos

Administrador: Es la persona natural o jurídica, designada por la asamblea de copropietarios, al cual le corresponden el cuidado de los bienes de dominio común y la ejecución de los actos de administración y conservación de tales bienes, conforme a las normas legales, reglamentos de copropiedad, acuerdos de la asamblea de copropietarios y comité de administración.

Alícuota: Es el porcentaje de contribución que posee una unidad en la comunidad como un todo. Dicho porcentaje de prorrato lo designa el arquitecto de la obra y en general sirve para distribuir los gastos comunes para cada unidad. Las alícuotas estarán estipuladas como un anexo del reglamento de copropietarios.

Comité de Administración: Es el órgano que representara a la asamblea en todas sus facultades (excepto sesiones extraordinarias). El comité de administración podrá también dictar normas que faciliten el buen orden y administración del condominio, como asimismo imponer las multas que estuvieren contempladas en el reglamento de copropiedad.

Condominios: Son las construcciones o los terrenos acogidos al régimen de copropiedad inmobiliaria regulado por la ley de Copropiedad Inmobiliaria N° 19.537. Se distinguen dos tipos de condominios en Chile, los cuales no podrán estar emplazados en un mismo predio: Tipo A, las construcciones, divididas en unidades, emplazadas en un terreno de dominio común, y Tipo B, los predios, con construcciones o con proyectos de construcción aprobados, en el interior de cuyos deslindes existan simultáneamente sitios que pertenezcan en dominio exclusivo a cada copropietario y terrenos de dominio común de todos ellos.

Fondo de Reserva: Todo condominio deberá tener un fondo común de reserva. Su objetivo será: reparaciones de los bienes de dominio común, certificación periódica de las instalaciones de gas o a gastos comunes urgentes o imprevistos. Se irá incrementando este fondo a través de: porcentajes de recargo sobre los gastos comunes, multas y uso de bienes comunes.

Gasto común: La ley distingue gastos ordinarios y extraordinarios. Los ordinarios son aquellos que se generan por la mantención, administración, reparación y uso de la copropiedad. Otros gastos, así como las sumas destinadas a nuevas obras, son calificadas por ley de: “gastos comunes extraordinarios”.

Todos los copropietarios están obligados al pago de los gastos comunes ordinarios y extraordinarios, en proporción al derecho que les corresponda en la copropiedad.

Predio: Superficie delimitada de terreno que pertenece a un propietario o a varios pro indiviso.

Prorratear: Repartir una cantidad entre varios según la parte que proporcionalmente le toca pagar a cada uno.

Lista de Figuras

Figura 3.1 Interfaz software “kastor” modulo principal [1]	3
Figura 5.1 Arquitectura de 3 capas	7
Figura 5.2 Arquitectura física del sistema [5]	8
Figura 9.1 Diagrama caso uso general.....	16
Figura 9.2 Caso de uso Administrar Gastos Comunes (Ver narrativo en anexo D).....	17
Figura 9.3 Diagrama de secuencia Autenticar Usuario	17
Figura 9.4 Diagrama de actividad Prorratear Gasto Común	18
Figura 9.5 Caso de uso Gestionar Existencias.....	19
Figura 9.6 Diagrama de actividad Ingresar Existencia.....	19
Figura 9.7 Caso de uso Gestionar Unidades (Ver narrativo en Anexo D)	20
Figura 9.8 Diagrama de Actividad Ingresar Monto Masivo Unidades.....	21
Figura 9.9 Caso de uso Administrar Usuarios (Ver narrativo en Anexo D)	22
Figura 9.10 Caso de uso Gestionar Proveedores	22
Figura 9.11 Caso de uso Gestionar Comunicativos.....	23
Figura 9.12 Caso de uso Administrar cuentas	23
Figura 9.13 Caso de uso Gestionar Condominio.....	24
Figura 9.14 Caso de uso Gestionar Perfil	24
Figura 9.15 Caso de uso Consultar Gastos Comunes.....	25
Figura 9.16 Caso de uso Enviar Mensajes.....	25
Figura 9.17 Diagrama de clases del sistema.....	26
Figura 9.18 Modelo de Datos Lógico.....	27
Figura 11.1 Interfaz del administrador: Ingresar gastos comunes.....	29
Figura 11.2 Interfaz del copropietario: Gastos de la comunidad.....	29
Figura A. 1 Interfaz del software Kastor: Módulo Biblioteca [1]	34
Figura A. 2 Interfaz del software Kastor: Módulo Mis Solicitudes [1].....	34
Figura A. 3 Interfaz del software Kastor: Módulo Generación Gastos Comunes [1]	35
Figura A. 4 Interfaz del software Tu Condominio al Día: Estado de Cuenta [2].....	36
Figura A. 5 Interfaz del software Tu Condominio al Día: Cuentas por Pagar [2].....	36
Figura A. 6 Interfaz del software Tu Condominio al Día: Mis Pagos [2]	37
Figura A. 7 Interfaz del software Tu Condominio al Día: Recibos del Condominio [2]	37
Figura A. 8 Interfaz del software Habitek: Inicio de Sesión [3].....	38
Figura A. 9 Interfaz del software Habitek: Secciones Copropietarios [3]	38
Figura A. 10 Interfaz del software Habitek: Mis Gastos Comunes [3]	39
Figura B. 1 Fases del paradigma UP [4].....	41
Figura E. 1 Diagrama de secuencia Ingresar Usuario	57
Figura E. 2 Diagrama de secuencia Agregar Proveedor.....	58
Figura E. 3 Diagrama de secuencia Ingresar Condominio	59
Figura E. 4 Diagrama de secuencia Modificar Contraseña	60
Figura F. 1 Modelo Relacional	61
Figura F. 2 Tablas sección 1	63
Figura F. 3 Tablas sección 2.....	63
Figura F. 4 Tablas sección 3.....	64

Figura F. 5 Tablas sección 4.....	64
Figura F. 6 Tablas sección 5.....	65
Figura F. 7 Tablas sección 6.....	66
Figura G. 1 Interfaz del administrador: Listar factores de interés del condominio.....	67
Figura G. 2 Interfaz del administrador: Creación de factor de interés del condominio	67
Figura G. 3 Interfaz del administrador: Creación de categorías en gastos comunes.....	67
Figura G. 4 Interfaz del administrador: Crear de subcategorías en gastos comunes.....	68
Figura G. 5 Interfaz del administrador: Listar parámetros gasto común.....	68
Figura G. 6 Interfaz del administrador: Listar viviendas.....	69
Figura G. 7 Interfaz del administrador: Listar gastos comunes.....	69
Figura G. 8 Interfaz del administrador: Ver y editar gastos comunes de un período	70
Figura G. 9 Interfaz del administrador: Formulario Crear Usuario.....	70
Figura G. 10 Interfaz del administrador: Formulario Crear Proveedor	71
Figura G. 11 Interfaz del administrador: Formulario Crear Existencia.....	71
Figura G. 12 Interfaz del administrador: Crear Información.....	72
Figura G. 13 Interfaz del administrador: Consultar y crear calendario	72
Figura G. 14 Interfaz del administrador: Cancelación de Gastos Vivienda	73
Figura G. 15 Interfaz del copropietario: Modificar contraseña	73
Figura G. 16 Interfaz del copropietario: Gastos comunidad	74
Figura G. 17 Interfaz del copropietario: Mis gastos comunes.....	75
Figura H. 1 Proceso gráfico de pruebas de caja negra.....	76

Lista de Tablas

Tabla 2.1 Problemas en una gestión deficiente	2
Tabla 3.1 Sistemas comerciales.....	3
Tabla 5.1 Beneficios de implementar el sistema a alto nivel	5
Tabla 5.2 Secciones en el detalle de gastos comunes.....	6
Tabla 6.1 Factibilidad de Hardware	9
Tabla 7.1 Riesgos del proyecto.....	11
Tabla 7.2 Mitigación y contingencia del proyecto	12
Tabla A. 1 Características de Kastor, Tu condominio al Día y Habitek	40
Tabla C. 1 Requisitos mínimos de Sublime Text	45
Tabla C. 2 Hardware desarrollador.....	45
Tabla C. 3 Requisitos de instalación Navegador Chrome	46
Tabla C. 4 Características del servicio de hosting.....	46
Tabla C. 5 Datos costo desarrollo.....	47
Tabla D. 1 Caso de uso Prorratear gasto común.....	49
Tabla D. 2 Caso de uso Autenticar usuario	50
Tabla D. 3 Caso de uso Crear calendario anual.....	51
Tabla D. 4 Caso de uso Adjuntar documento	52
Tabla D. 5 Caso de uso Ingresar monto masivo unidades.....	53
Tabla D. 6 Caso de uso Crear unidad	54
Tabla D. 7 Caso de uso Registrar usuario	55
Tabla D. 8 Caso de uso Gestionar vinculación a unidad	56
Tabla H. 1 Clase de equivalencia para el largo de la contraseña.....	76

Resumen

El condominio Nitella ubicado en la quinta región de Chile tiene una administración que no cuenta con un sistema que les permita sistematizar y organizar la información de cuentas, gastos comunes e informativos de la comunidad. Es por ello que surge la necesidad de crear una herramienta que apoye su gestión, automatice los procesos y ofrezca un mayor grado de organización de los datos.

Para diseñar el sistema se realizó un estudio de los procesos de la administración y los productos comerciales ofrecidos en el área identificando los requerimientos funcionales y no funcionales con que debe cumplir el software para satisfacer las necesidades del cliente. Para el desarrollo se utilizaron diversas tecnologías como el lenguaje de programación PHP, el sistema manejador de base de datos MYSQL y el servidor Web Apache 2.2.

En la culminación del proyecto se adquirió la experiencia de aplicar los conocimientos aprendidos durante la formación académica con usuarios finales y que la implantación de este nuevo software generará un impacto positivo en las labores de administración del condominio.

Palabras-clave: condominio, gastos comunes, requerimientos, patrones de diseño, modelamiento, arquitectura.

Abstract

The Nitella condominium located in the fifth region of Chile has an administration that does not have a system that allows them to systematize and organize the information of accounts, common expenses and informative of the community. That is why the need arises to create a tool that supports its management, automates the processes and offers a greater degree of organization of the data.

In order to design the system, a study of the management processes and the commercial products offered in the area was carried out, identifying the functional and non-functional requirements that the software must fulfill in order to satisfy the client's needs. Various technologies such as the PHP programming language, the MYSQL database management system and the Apache 2.2 Web server were used for the development.

At the culmination of the project the experience of applying the knowledge learned during the academic training with end users was acquired and that the implementation of this new software will generate a positive impact on the management of the condominium.

Key words: condominium, common expenses, requirements, design patterns, modeling, architecture.

1 Introducción

Un condominio se podría definir como un conjunto de viviendas poseído en régimen de comunidad de propietarios, los cuales son dueños de una alicuota del conjunto y comparten bienes de dominio común como son patios, plazas, ascensores, piscinas etc que requieren una mantención que debe ser solventada por la comunidad de propietarios, es aquí donde nace el concepto de gastos comunes. Los gastos comunes son emitidos en forma mensual y deben ser calculados por la administración.

Los procesos de la administración del condominio Nitella se llevan a cabo en su mayor parte manualmente lo cual tiene como consecuencia un mayor gasto de tiempo, eventuales pérdidas de información, una débil disposición de la información de interés a la comunidad y ralentización de procesos como es el control de pagos de las cuotas correspondientes a cada propietario. En nuestro país hay una importante cantidad de condominios que necesitan optimizar sus procesos de gestión administrativa y acceso a la información.

El propósito de este proyecto es realizar un estudio y desarrollo de un sistema que apoye la gestión administrativa del condominio Nitella, complementando la documentación tradicional con una plataforma informática que permita sistematizar en parte sus procesos logrando un acceso rápido y estructurado a la información tanto para el administrador como para los copropietarios.

Para el análisis y diseño del software se realizó un levantamiento de requerimientos con el usuario final además de un estudio del estado del arte obteniendo de esta forma los requerimientos funcionales y no funcionales para luego seleccionar las metodologías y tecnologías que permitan un desarrollo eficiente y de calidad.

Se proyecta que el sistema permita el cálculo de los gastos comunes, de intereses, ingresos y gastos en general, guardando un historial de las cuentas del condominio y cada unidad como también módulos para el manejo de información relacionada con los empleados, proveedores y sus productos, y las existencias.

2 Descripción del Sistema Actual

2.1 Diagnóstico de la Situación Actual

El condominio Nitella, ubicado en la ciudad de Viña del Mar, cuenta con 27 cabañas y tiene una administración la cual gestiona en forma manual la información de la comunidad debido a que no cuentan con un sistema de software que apoye sus procesos.

La forma tradicional de manejar la información en base a documentos físicos hace que el proceso de búsqueda sea lento y retrasa la atención en otras actividades necesarias para el bienestar de la comunidad.

Las operaciones como el cálculo de los gastos comunes que debe cancelar cada unidad se realiza con plantillas Excel lo cual consume un mayor tiempo y es susceptible a errores y/o pérdidas de información. El acceso a la información de los gastos de cada periodo se debe informar a los copropietarios en reuniones y/o por correo electrónico no siendo accesible oportunamente para su análisis. Al no contar un sistema se pueden visualizar varios problemas como:

- ✘ La planificación de las operaciones que se deben llevar a cabo posiblemente no se cumplan en los plazos estipulados.
- ✘ El canal actual de entrega de la información de ingresos, gastos y detalles de los cobros de gastos comunes no es eficiente.
- ✘ Hay un bajo nivel de control de los recursos del condominio, como son el fondo de reserva del mismo.
- ✘ La información no está debidamente respaldada y estructurada en una base de datos, pudiendo haber pérdidas o mal manejo.
- ✘ En un proceso no automatizado es más difícil llevar un control de los gastos comunes, pagados y por pagar de los copropietarios, así como consultar la información histórica en forma rápida.

En la siguiente tabla podemos ver a alto nivel algunos problemas que pueden ocurrir

Tabla 2.1 Problemas en una gestión deficiente

CONCEPTO	PROBLEMA
Error en el cálculo de los gastos comunes.	Capacidad de acceso a las fuentes de información para corroborar los cobros emitidos no es la más óptima.
Demora en la entrega de la información a la comunidad.	Copropietarios desconocen sus obligaciones a tiempo. Mayor dificultad en entregar información actualizada. Comunidad mal informada.
Integridad de la información	Al manejar diversas fuentes de información es posible que no haya una consistencia de los datos.

3 Estado Del Arte

Entre los sistemas comerciales para la administración de condominios se pueden encontrar varias opciones ofrecidas por diferentes empresas. Estos productos son en general sistemas web que en su publicidad ofrecen una gestión eficiente y automatizada de la información de la comunidad. Entre los sistemas ofrecidos se pueden nombrar los siguientes.

Tabla 3.1 Sistemas comerciales

Producto comercial	Página web
Kastor Software	https://www.kastor.cl/
Mi conserje	https://www.miconserje.cl/
Habitek	www.habitek.cl
Tu condominio al día	http://www.tucondominioaldia.com.ve/

Para el análisis de los diferentes productos se estudiara la publicidad y algunas demostraciones en línea que algunos tienen disponibles. Los sistemas a analizar son Kastor, Tu condominio al día y Habitek.

3.1 Analizando Software Kastor

El software “kastor” tiene una versión demo a la que se puede acceder en la dirección <http://demo.kol.cl/kol/index.php>. El sistema en su interfaz inicial pide una autenticación de usuario y contraseña que se encuentra en la misma pantalla. Al autenticar re-direcciona a la interfaz inicial que se muestra a continuación:



Figura 3.1 Interfaz software “kastor” modulo principal [1]

Al ingresar a biblioteca se pueden encontrar los reglamentos que rigen a la comunidad entre estos se pueden mencionar la ley de copropiedad y decretos establecidos por los mismos propietarios participes. Para más información acerca de los productos en análisis ver el Anexo A.

4 Definición de los Objetivos

4.1 Objetivo general

- ✓ Diseñar e implementar un sistema web para apoyar la gestión administrativa del condominio Nitella permitiendo tener un mejor control en sus procesos.

4.2 Objetivos específicos

- ✓ Estudiar los productos comerciales del área para la administración de condominios.
- ✓ Analizar la gestión y procesos que ocurren en la administración del condominio en estudio.
- ✓ Sistematizar los procesos de gestión de la comunidad.
- ✓ Identificar los requerimientos que satisfagan las necesidades del cliente, modelar una solución e identificar las herramientas tecnológicas para el desarrollo.
- ✓ Desarrollar el sistema web en base al análisis y diseño realizado.
- ✓ Lograr un software que sirva como herramienta para mantener los datos actualizados de los propietarios, personal, cuentas y gastos de la comunidad entre otros facilitando el manejo y acceso a la información.

5 Descripción de la Solución

5.1 Propuesta de solución

La solución para este cliente es crear un sistema web que trabaje con una base de datos en donde se estructure la información lo que permitirá al administrador y copropietarios tener un acceso sistematizado y actualizado desde cualquier computador con acceso a internet. Los procesos como son el prorrateo de gastos comunes serán realizados por el sistema lo que automatizará esta operación de cálculo. Con la aplicación web se busca lograr un mejor control de las cuentas del condominio y una agilización en los procesos de gestión de la administración.

5.2 Ventajas y beneficios de desarrollar el sistema

- ✓ Mantener un historial de la gestión del condominio mes a mes con fácil acceso a toda la información.
- ✓ Tener la información respaldada.
- ✓ Agilizar el control de gastos, pagos y asignación de estos al gasto común.
- ✓ Llevar un orden de sus proveedores
- ✓ Llevar un control de los Gastos Comunes, pagados y por pagar de sus copropietarios
- ✓ La entrega de los reportes de pagos y estados de cuentas a los copropietarios de los condominios será más accesible en todo momento y desde cualquier lugar.

Tabla 5.1 Beneficios de implementar el sistema a alto nivel

CONCEPTO	BENEFICIO
Mayor precisión en los cálculos.	Son calculados por la aplicación mediante algoritmos computacionales. Disminución de la tasa de errores en los cobros.
Mayor control de los ingresos y egresos de la comunidad.	Mayor confiabilidad. Incremento en la satisfacción de los usuarios.
Base de datos, almacenamiento y respaldo de la información.	La información se almacena en forma estructurada, hay integridad de los datos y se evitan las inconsistencias. Respaldo de la información.
Rapidez en el procesamiento de los datos.	Proporciona un medio para encontrar en instantes la información. Generación de informes más rápidos y confiables.
Aumentar la seguridad en el almacenamiento y consulta de registros.	Los usuarios tienen niveles de acceso a los archivos y al tipo de información que deseen manejar, gestionado mediante el uso de perfiles que establecen los privilegios de los mismos.
Entregar una Información actualizada, respaldada y oportuna a la comunidad.	Transparencia de ingresos y egresos. Comunidad bien informada. Rápido acceso a la información mediante procesos sistematizados.

A grandes rasgos el detalle de gastos comunes se organizará en 6 grandes grupos que se muestran a continuación.

Tabla 5.2 Secciones en el detalle de gastos comunes

1- GASTOS DE ADMINISTRACIÓN	3- GASTOS GENERALES	5- SEGUROS
2- GASTOS DE MANTENCIÓN Y REPARACIÓN	4- CONSUMOS BÁSICOS	6- OTROS

5.3 Paradigma de Desarrollo

En la ingeniería de software existen varios paradigmas para el desarrollo de proyectos informáticos entre ellos; el modelo cascada, evolutivo, Proceso Unificado etc.

Para el desarrollo de este proyecto se utilizará UP (Proceso unificado) que es un marco de desarrollo de software que se caracteriza por estar dirigido por casos de uso. El proceso Unificado es iterativo e incremental compuesto de cuatro fases denominadas inicio, elaboración, construcción y transición. Cada una de estas fases es a su vez dividida en una serie de iteraciones. Estas iteraciones ofrecen como resultado un incremento en el producto que añade o mejora la funcionalidad del sistema. Para ver más información acerca del paradigma utilizado y las herramientas y tecnologías revisar el Anexo B.

5.4 Metodología

Se puede definir como un conjunto de técnicas y herramientas que permiten facilitar la construcción del software. Como ejemplo se mencionarán dos metodologías:

5.4.1 Estructurada

- El sistema se modela con un enfoque orientado al flujo de datos.
- La estructura de los datos y comportamiento están relacionados de forma débil, ya que esta se enfoca principalmente en funciones.

5.4.2 Orientada a objetos

- En la metodología orientada a objetos el sistema se organiza como una colección de objetos que interactúan entre sí y contienen tanto estructura de datos como comportamiento.
- Proximidad de los conceptos de modelado respecto a objetos en el mundo real.

5.4.3 Elección de la metodología

Para el presente proyecto se utilizara la metodología orientada a objetos (OO) porque:

- La OO es una nueva forma de pensar y diseñar, y es la tendencia más influyente observada en los sistemas de software en los últimos años.
- La programación orientada a objetos es hoy una forma popular de programar y tiene gran acogida en el desarrollo de proyectos de software gracias a sus ventajas de

modelar la realidad, expresando un programa como un conjunto de objetos que colaboran entre ellos para realizar tareas, esto permite hacer programas más fáciles de escribir, mantener y reutilizar.

- Además esta metodología va de la mano con la utilización del Proceso Unificado como paradigma de desarrollo y UML como herramienta de modelado, lo que facilita aún más el desarrollo de un sistema informático

5.5 Arquitectura lógica

Para el desarrollo del software se utilizará la arquitectura lógica de tres capas. El principal motivo de escogerla es separar la aplicación en varios niveles simplificando la comprensión y organización del sistema. Las tres capas lógicas que se describen a continuación:

- Capa de Presentación: Esta capa se encarga de proveer una interfaz entre el sistema y el usuario. Se responsabiliza de que se le comunique la información al usuario por parte del sistema y viceversa. Este nivel se comunica únicamente con la capa de negocio.
- Capa de Negocio: Será la capa que contiene los procesos a realizar con la información recibida desde la capa de presentación, las peticiones que nuestro usuario ha realizado, y responsabilizándonos de que se le envíen las respuestas adecuadas a la capa de presentación.
- Capa de Datos: En último caso, manejaremos la capa en donde almacenaremos los datos de los usuarios, cuentas y fondos pertenecientes a la comunidad, como también información de pagos y transacciones registrando su gestor, información del personal, sueldos etc. La capa de negocio, se puede encargará de ofrecer, modificar, almacenar, borrar y recuperar datos, mediante el gestor de bases de datos.

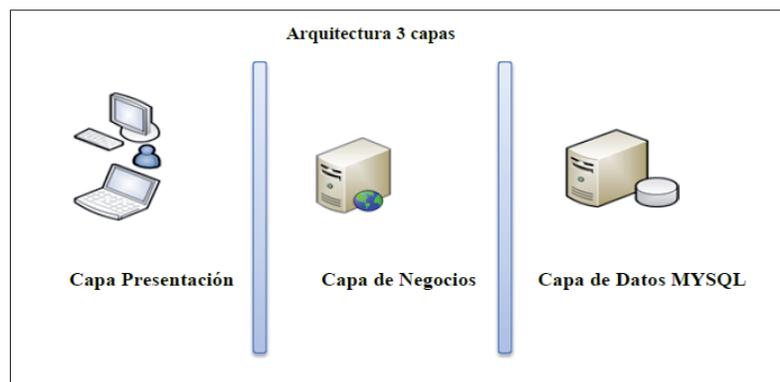


Figura 5.1 Arquitectura de 3 capas

Como se puede ver en la figura la arquitectura de tres capas consiste inicialmente en la capa de presentación (interfaz web) que se comunica con la capa de negocios y esta última con la base de datos en donde se almacenará la información del sistema.

5.6 Arquitectura Física

La arquitectura física de un sistema es la que representa la adecuación de todos los componentes físicos que intervienen en un sistema para su correcto funcionamiento y despliegue, entra dentro de la etapa de diseño de software y esta paralelo a los procesos de arquitectura de software. La siguiente imagen representa la infraestructura ofrecida del hosting donde se alojara la aplicación.

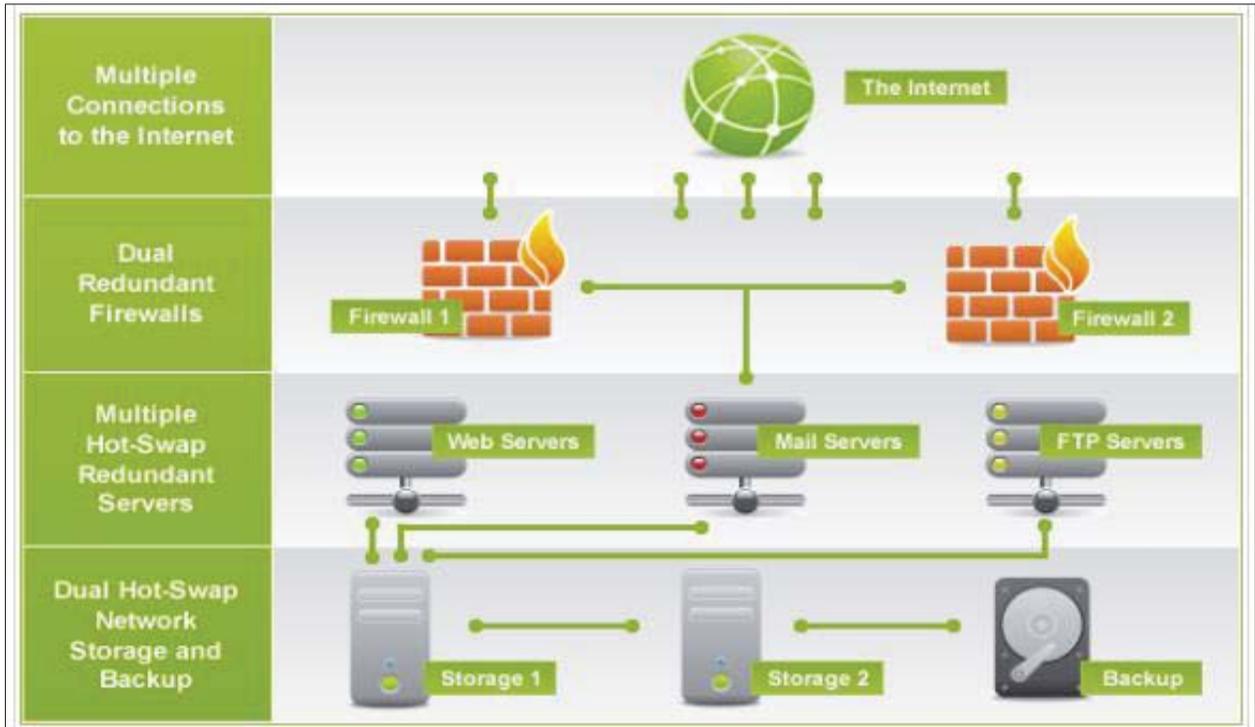


Figura 5.2 Arquitectura física del sistema [5]

La plataforma está diseñada de tal forma que si un servidor falla, automáticamente otro servidor entra en operación tomando el control y los procesos del servidor que falló. Además los datos son guardados en dos discos de red que están sincronizados así en el evento que si uno falla el otro continua entregando los datos a los servidores manteniendo el servicio en funcionamiento.

La arquitectura de red opera en RAID que es una forma de configurar el sistema y de esta forma si un disco falla es posible la recuperación de la información almacenada.

La arquitectura también cuenta con un firewall (llamado también "corta-fuego"), es un sistema que permite proteger a los servidores o la red de eventuales intrusiones que provenientes de otras redes (expresamente de Internet). El firewall es un sistema que permite filtrar los paquetes de datos que andan por la red. Se trata de un "puente angosto" que filtra, al menos, el tráfico entre la red interna y externa

Un firewall puede ser un programa (software) o un equipo (hardware) que actúa como intermediario entre la red local (o la computadora local) y una o varias redes externas.

El usuario podrá acceder al sistema desde cualquier computadora (cliente) que cuente con conexión a internet.

6 Estudio de Factibilidad

El estudio factibilidad se enfoca en determinar la disponibilidad de recursos necesarios para llevar a cabo un proyecto. El éxito de un proyecto está determinado por el grado de factibilidad que está presente en cada uno de sus aspectos. Este proceso es parte del de análisis total y diseño del sistema que busca determinar la viabilidad recopilando datos relevantes y en base a ello tomar la mejor decisión para el desarrollo del sistema.

Un estudio de factibilidad debe presentar las posibles ventajas para la entidad. Si el sistema cumple con el manejo de la información aumentando la eficacia de la gestión administrativa, disminuyendo costos administrativos entonces el sistema se justificaría; pero todo esto se debe demostrar detalladamente para estimar los recursos necesarios para el desarrollo y posterior puesta en marcha.

6.1 Factibilidad Mercado

Tiene como finalidad determinar si existe o no una demanda que justifique, bajo ciertas condiciones, la puesta en marcha de un programa de producción de ciertos bienes o servicios en un espacio de tiempo.

El sistema está pensado para un condominio pero puede ser un estándar que permita ser un producto comercializable a otras comunidades. En Chile Los condominios representan un porcentaje importante del parque habitacional destinado a vivienda social, estimándose que más de un millón de personas viven en ellos [6]. De hecho, sólo en la Región Metropolitana existirían más de 600 mil familias habitando en condominios sociales.

Analizando la cantidad de empresas que ofrecen servicios de Administración de condominios y basándose en clientes de estas mismas, se puede concluir que existe un amplio mercado y que esta progresivamente aumentando.

6.2 Factibilidad Técnica:

En la factibilidad técnica se evalúan dos enfoques que son muy importantes dentro de la informática como son el Hardware y el software.

6.2.1 Hardware

Dentro del Hardware se toma específicamente el servidor o servicio donde estará instalado el sistema. En este proyecto se contratara un proveedor del servicio que en su plan anual cuenta con las siguientes características entre otras.

Tabla 6.1 Factibilidad de Hardware

Espacio web - cuenta principal	30 GB
Servidores Dell	Plataforma Cloud con Servidores DELL PowerEdge de Alto Rendimiento
Bases de datos	MySQL / PostgreSQL
Transferencia Mensual	300 GB
Lenguajes	PHP, PYTHON

6.2.2 Software

Para operar con el sistema es necesario utilizar exploradores de internet como Google Chrome, Mozilla, Internet Explore o similar que están disponibles en cualquier computador y dispositivo móvil o se descargan de manera gratuita en sus sitios correspondientes.

El sistema final se podrá montar en una plataforma Windows o Linux sin inconvenientes.

6.2.3 Recursos Humanos

Para la realización de este proyecto se cuenta con un alumno que posee las siguientes competencias académicas:

- Lenguaje de Programación: C, PHP, Ajax, HTML, CSS.
- Modelamiento de Software: Lenguaje de modelado UML.
- Base de Datos: MYSQL versión 5.0.

6.3 Factibilidad Económica:

Esta factibilidad tiene por objetivo determinar si la implementación del sistema se justifica en su relación al costo / beneficio.

Para esto es necesario analizar los costos económicos necesarios para llevar a cabo los procesos y/o actividades para el desarrollo del proyecto contrastándolo con los beneficios que se producirán por la inversión y determinando así la rentabilidad del mismo. Para más detalles revisar la sección C.2 del Anexo C.

6.4 Factibilidad Legal:

Se refiere a la posibilidad de realizar el proyecto conforme a las leyes vigentes, es decir, que no esté prohibido expresamente por las leyes o que estén autorizadas las actividades a realizar de manera expresa.

La factibilidad legal nos permite determinar los derechos que tienen los autores sobre la documentación realizada por estos en este proyecto, la cual es exclusividad de los desarrolladores de este sistema, por tal motivo queda prohibida la distribución y reproducción de este documento, tales como la publicación impresa o su grabación.

Las licencias de los software a emplearse en la implantación del sistema informático son de libre acceso evitando así tener inconvenientes legales a futuro.

La creación del producto y documentación es creación personal así que no habrá riesgos de plagio como lo indica la ley de propiedad intelectual 17.336.

6.5 Factibilidad Operativa:

Esta factibilidad comprende una determinación de la probabilidad de si el nuevo sistema tendrá el uso que se espera. Los futuros usuarios del nuevo sistema, tanto el administrador como los copropietarios tiene un considerable interés por contar con una herramienta que les facilite la comunicación y administración en la comunidad. Para ver más acerca de esta factibilidad revisar la sección C.3 del Anexo C.

7 Análisis de Riesgos

El riesgo se podría definir como el área de incertidumbre que puede dar lugar a una desviación del plan proyecto, incluyendo la falla de este. Al analizar los riesgos de un proyecto se busca encontrar las vulnerabilidades del mismo y crear planes para evitar o minimizar sus efectos en el desarrollo del sistema.

La escala de evaluación según su probabilidad de ocurrencia será: **Muy bajo, Bajo, Moderado, Alto, Muy Alto** y según su rango de impacto: **Despreciable, Tolerable, Moderado, Crítico, Catastrófico**.

7.1 Identificación

Tabla 7.1 Riesgos del proyecto

N° del Riesgo	Descripción Riesgo	Probabilidad	Impacto
R1	Problemas de usabilidad del software	Moderado	Crítico
R2	Inexperiencia con las tecnologías	Medio	Alto
R3	El usuario cambia la idea del proyecto a desarrollar, cambio en los requerimientos	Moderado	Crítico
R4	Mala estimación del tiempo de desarrollo	Moderado	Catastrófico
R5	Aumenta el presupuesto de implementación	Bajo	Tolerable
R6	El sistema no cumple con la seguridad e integridad en los datos. Vulnerabilidades el en sistema web	Medio-Alto	Crítico
R7	Cambio en las tecnologías de desarrollo	Bajo	Crítico
R8	Falta de planeación del proyecto	Moderado	Catastrófico
R9	Resistencia al cambio por parte de los usuarios finales	Medio-Bajo	Tolerable

7.2 Mitigación y Contingencia

En esta sección se analizarán los riesgos anteriormente presentados ideando un plan de mitigación; prevenir el problema y contingencia; reducir o resolver la situación y su impacto al proyecto.

Tabla 7.2 Mitigación y contingencia del proyecto

N° del Riesgo	Mitigación	Contingencia
R1	Planificar un conjunto de evaluaciones de usabilidad con el usuario más un manual con el detalle del funcionamiento descriptivamente y gráficamente	Re-estructurar las pruebas de usabilidad. Brindar capacitación al usuario del software.
R2	Capacitación constante en las tecnologías a utilizar	Solicitar apoyo de un experto.
R3	Comunicación constante con el cliente	Minimizar la carga de un nuevo requerimiento.
R4	Estimar el tiempo de desarrollo con claridad	Recuperar el tiempo perdido y cumplir con los plazos
R5	Tener al cliente informado de actualizaciones en los valores del servicio como hosting y dominio	Buscar servicios de Hosting y dominios alternativos
R6	Aplicar diferentes técnicas para la seguridad como encriptación al menos a los datos confidenciales como contraseñas	Capacitación técnica en seguridad web
R7	Analizar bien en que plataforma se desarrollará la aplicación	Buscar herramientas y tecnologías de desarrollo compatibles
R8	Coordinar los esfuerzos y los recursos para lograr el logro de los objetivos, como propósito fundamental	Reestructurar las actividades asignando un nivel de urgencia
R9	Realizar pruebas de usabilidad	Impartir capacitación para el uso del sistema

8 Especificación de Requerimientos

Los requerimientos de un sistema son la declaración de los servicios que este proporcionará, estos reflejan las necesidades de los usuarios y buscan apoyar los procesos actuales como por ejemplo la búsqueda de información en forma automatizada.

Para la obtención de los requerimientos se debe trabajar en conjunto con el cliente y/o usuarios finales llevando a cabo un análisis de las necesidades y servicios que deberá proporcionar el software y el rendimiento esperado entre otros.

Los requerimientos de sistemas de software se pueden clasificar en funcionales y no funcionales. Los requerimientos para el proyecto se describen a continuación.

8.1 Requerimientos funcionales

8.1.1 Administrador

8.1.1.1 Intereses

- Ingresar y eliminar el factor de interés del condominio, manteniendo el historial y dejando como operativo el más reciente.

8.1.1.2 Personas:

- Ingresar, modificar y eliminar personas.
- Asociar un perfil único a cada persona para su autenticación (administrador, copropietario y personal).
- El sistema tendrá diferentes privilegios de acceso para perfiles como administrador, copropietarios, arrendatarios u otros.
- Buscar usuarios asociadas a un condominio, por su run o nombre.
- Ingresar, modificar y eliminar la información de contacto de cada usuario.
- Ver la información de contacto más reciente de cada usuario.

8.1.1.3 Proveedores y productos:

- Ingresar, modificar y eliminar los proveedores.
- Ingresar, modificar y eliminar productos, y asociarlos a un proveedor.
- Listar los proveedores y luego el detalle de sus productos.

8.1.1.4 Existencias

- Ingresar, modificar y eliminar existencias teniendo un historial de la cantidad y el precio al que fueron adquiridas.

8.1.1.5 Viviendas

- Ingresar, modificar y eliminar viviendas.

- Listar las viviendas y luego el detalle de sus moradores.
- Modificar el factor de alícuota.
- Vincular y desvincular moradores pudiendo consultar la información de los mismos.
- Cada vivienda tendrá una única cuenta en donde realizarán los cargos y abonos. Se debe almacenar información gastos comunes por periodo (mensual), multas, intereses y otros cobros.
- Tendrá una opción para dividir el factor de alícuota equitativamente para las viviendas registradas en un condominio.

8.1.1.6 Mensajes

- Los usuarios con un perfil activo podrán enviar mensajes a los usuarios registrados y activos.

8.1.1.7 Informativos

- El sistema permitirá generar informativos a la comunidad de distinta índole, gestionados por el administrador.
- Habrá un muro con las publicaciones que podrán ser consultadas por los usuarios habilitados.

8.1.1.8 Cuentas condominio

- Este permitirá ingresar gastos e ingresos, abonándolos o cargándolos a una o diferentes cuentas quedando un registro de las transacciones efectuadas.
- Permitirá adjuntar algún documento que valide y sea un respaldo para las transacciones realizadas.

8.1.1.9 Gastos comunes

- Ingresar, modificar y eliminar los gastos comunes para un periodo.
- Poder adjuntar un documento para cada gasto de la comunidad como medio de respaldo el que podrá ser consultado por los copropietarios.
- El sistema realizara el prorrateo de los gastos comunes.
- Permitirá cerrar el periodo cuando ya se hayan ingresado todos los gastos del mes y desde ahí podrá ser consultada la información por los copropietarios.

8.1.2 Copropietario

- El sistema permitirá gestionar los datos de contacto desde la sesión del usuario.
- Mostrará los gastos comunes de la vivienda para cada periodo (mes y año).
- Mostrará los gastos comunes de la comunidad para cada periodo (mes y año).
- Permitirá emitir un mensaje simple a otros usuarios del condominio.

8.2 Requerimientos no funcionales

Estos se pueden definir como aquellos requerimientos que no se refieren directamente a las funciones específicas que entrega el sistema sino a las propiedades de este como la fiabilidad, y tiempo de respuesta entre otros. En la siguiente lista se ven algunos requerimientos no funcionales que debe cumplir la plataforma a desarrollar.

- **Accesibilidad:** El software estará accesible a todos aquellos usuarios que dispongan de un computador con conexión a internet.
- **Estabilidad y fiabilidad:** El sistema debe ser estable con el fin de que no presente fallas y fiable en el sentido de que a entradas particulares muestren resultados e información correcta.
- **Interfaz del software:** El sistema será que ser lo más simple y fácil de utilizar con interfaces alusivas a las operaciones realizadas por el usuario.
- **Mostrar la información en forma clara y ordenada.**

8.3 Definición de Actores

Actor Administrador: Representa al encargado de la administración del sistema y la información que se registrará. Su trabajo consiste en dar altas, bajas y modificaciones de los usuarios (personal, copropietarios, proveedores) y existencias, administrar las cuentas del condominio y generar las cuentas de gastos comunes. Todo su trabajo podrá ser realizado a través de la aplicación, previa autenticación.

Actor Copropietario: Representa a todos los moradores de las unidades del condominio pudiendo ser estos copropietarios o arrendatarios, quienes podrán consultar mensualmente la información de los gastos comunes, generar graficas comparativas con respecto al historial, gestionar su respectivo perfil (datos de contacto y contraseña) y enviar mensajes a los usuarios registrados en el sistema. Todas las acciones descritas podrán ser realizadas previa autenticación.

9 Análisis y Diseño del Sistema

9.1 Diagrama caso uso General

En el siguiente diagrama se muestran las funcionalidades a alto nivel que tendrán el administrador y los moradores del condominio, en este se pueden apreciar las diferentes opciones para cada actor.



Figura 9.1 Diagrama caso uso general

9.2 Caso de Uso Administrar Gastos Comunes

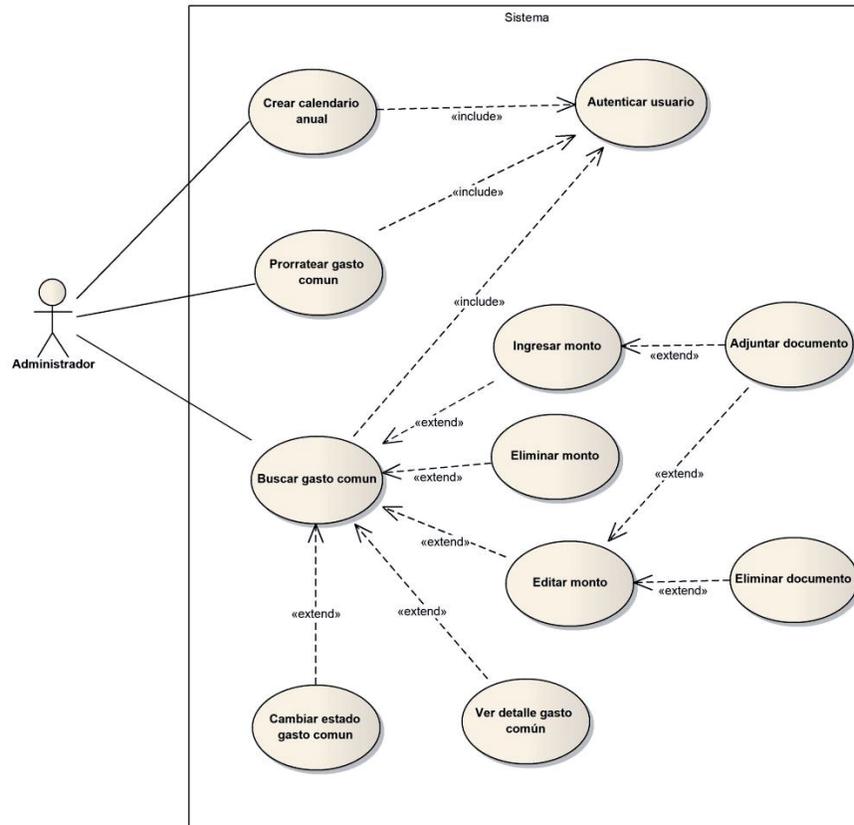


Figura 9.2 Caso de uso Administrar Gastos Comunes (Ver narrativo en anexo D)

9.3 Diagrama de secuencia Autenticar Usuario

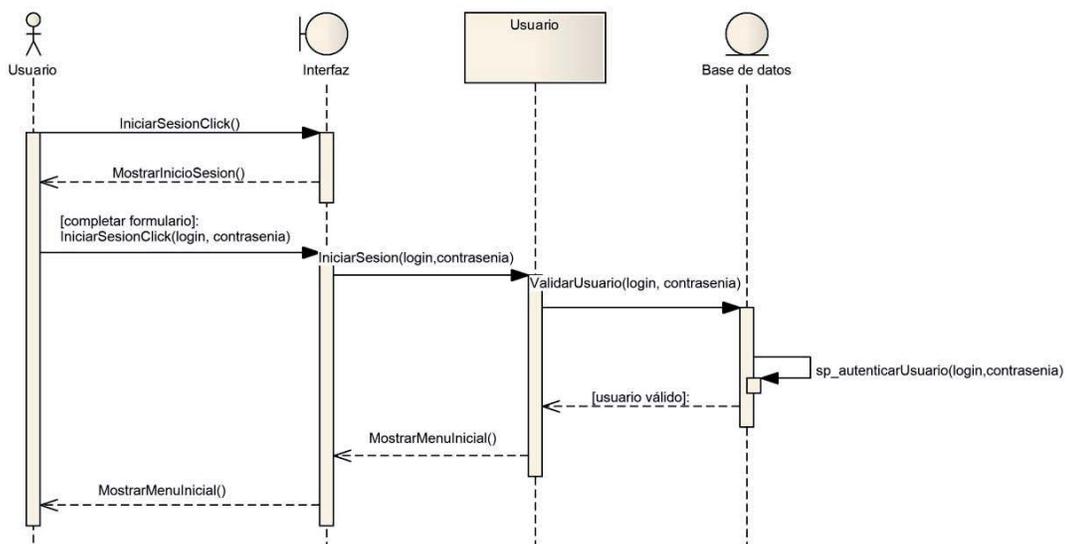


Figura 9.3 Diagrama de secuencia Autenticar Usuario

9.4 Diagrama de actividad Prorratear Gasto Común

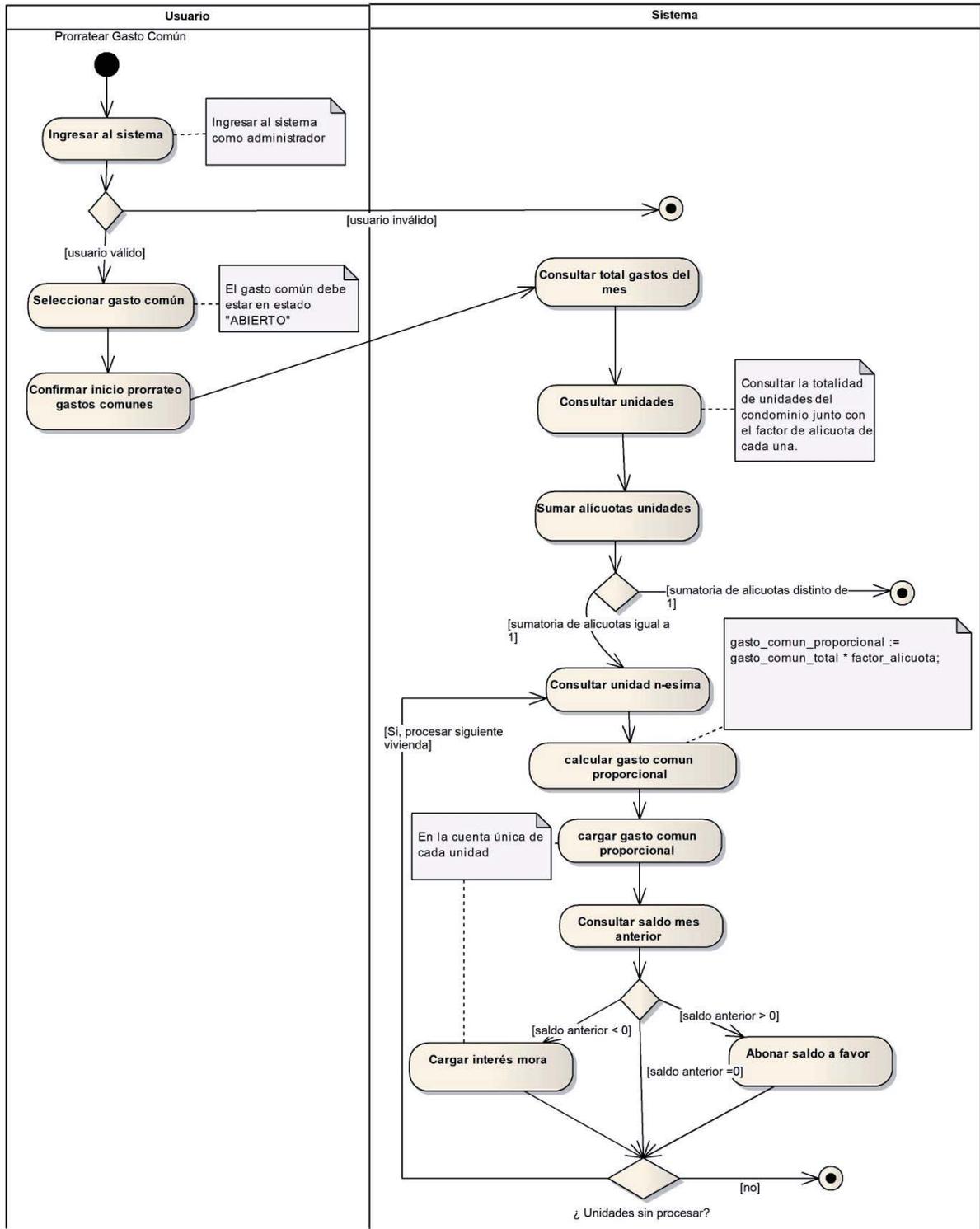


Figura 9.4 Diagrama de actividad Prorratear Gasto Común

9.5 Caso de Uso Gestionar Existencias

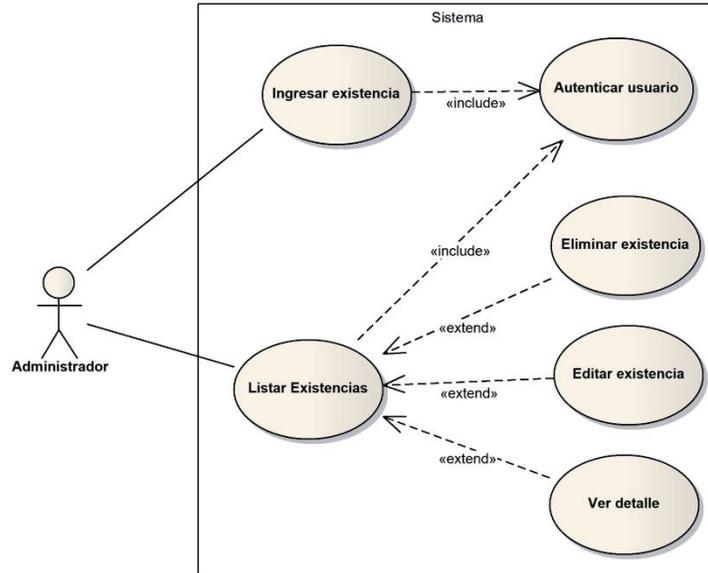


Figura 9.5 Caso de uso Gestionar Existencias

9.6 Diagrama de Actividad Ingresar Existencia

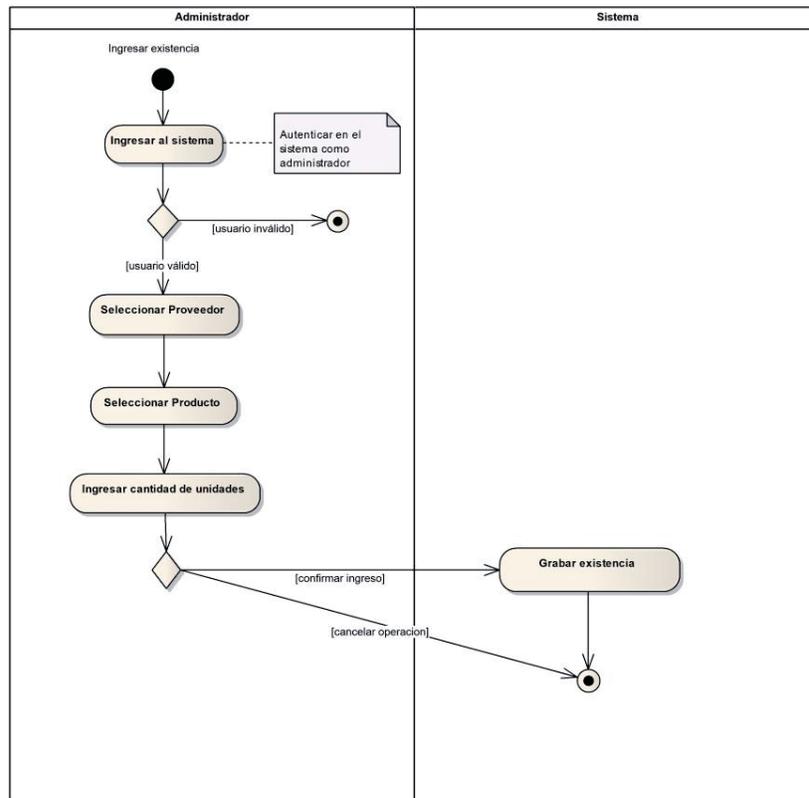


Figura 9.6 Diagrama de actividad Ingresar Existencia

9.7 Caso de Uso Gestionar Unidades

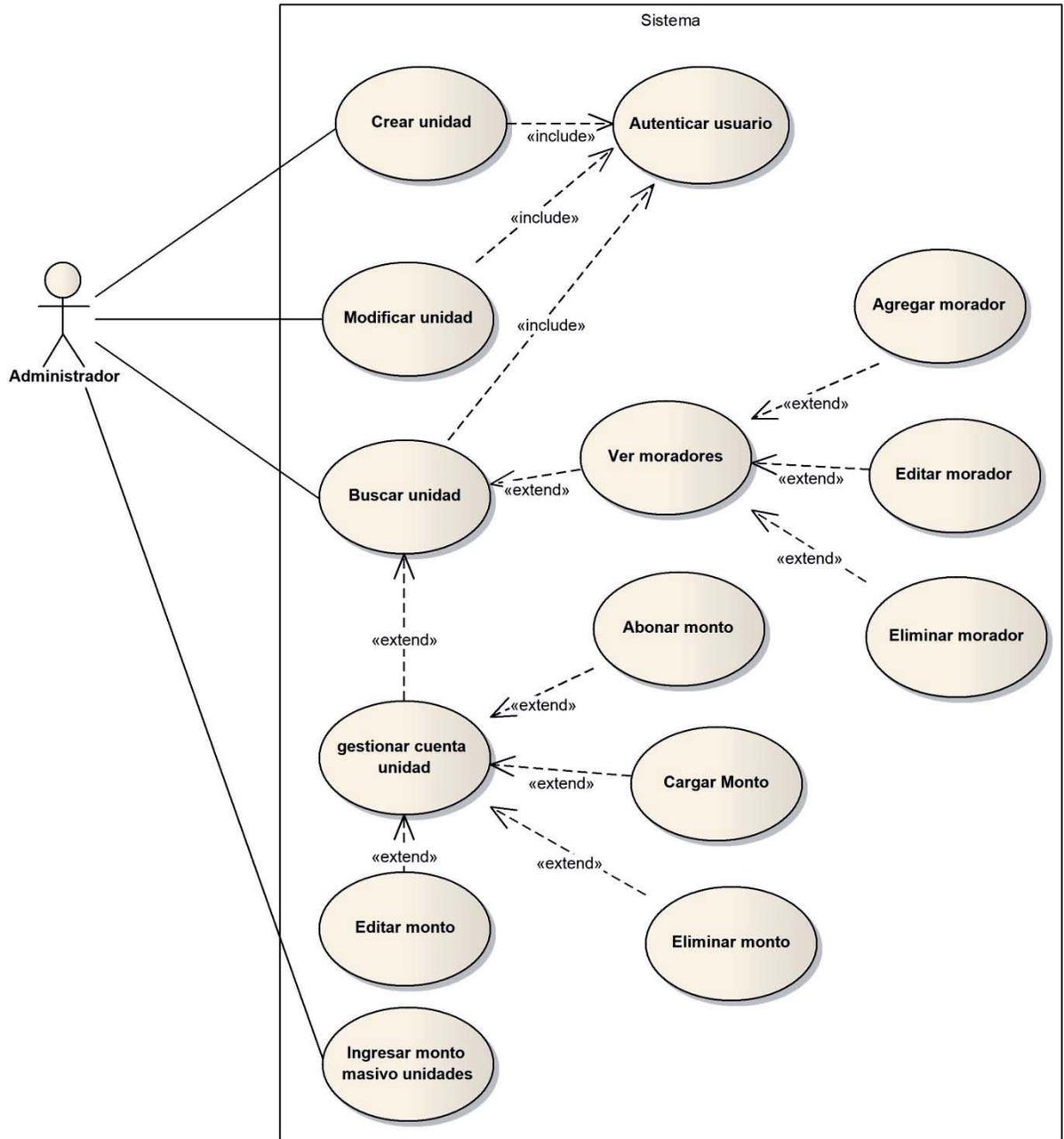


Figura 9.7 Caso de uso Gestionar Unidades (Ver narrativo en Anexo D)

9.8 Diagrama de Actividad Ingresar Monto Masivo Unidades

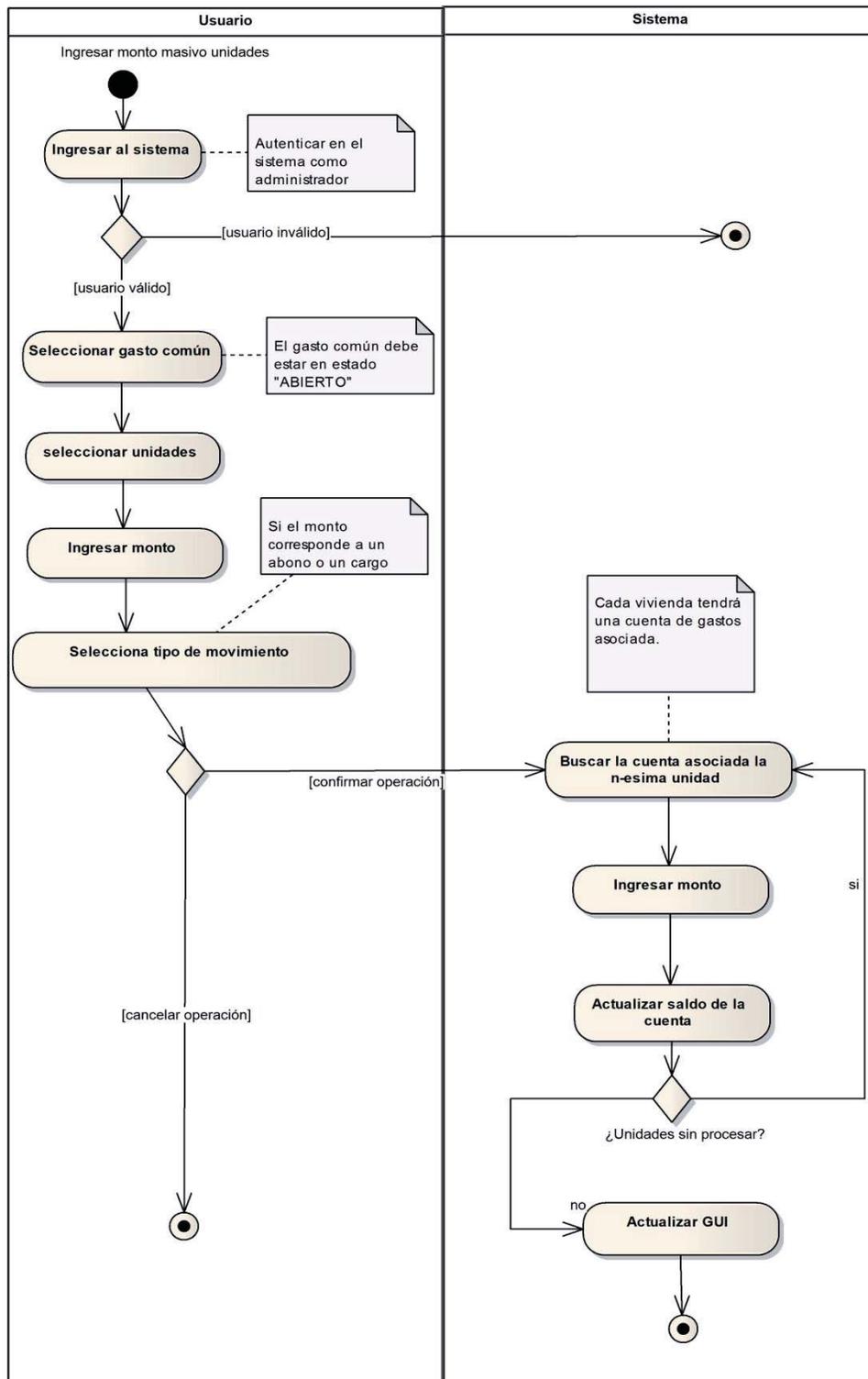


Figura 9.8 Diagrama de Actividad Ingresar Monto Masivo Unidades

9.9 Caso de uso Administrar Usuarios

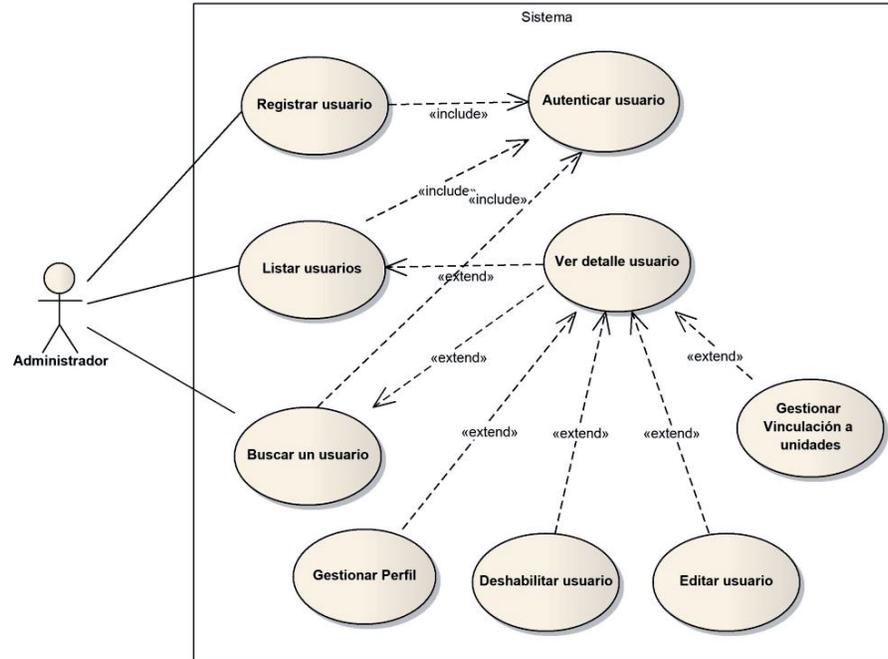


Figura 9.9 Caso de uso Administrar Usuarios (Ver narrativo en Anexo D)

9.10 Caso de uso Gestionar Proveedores

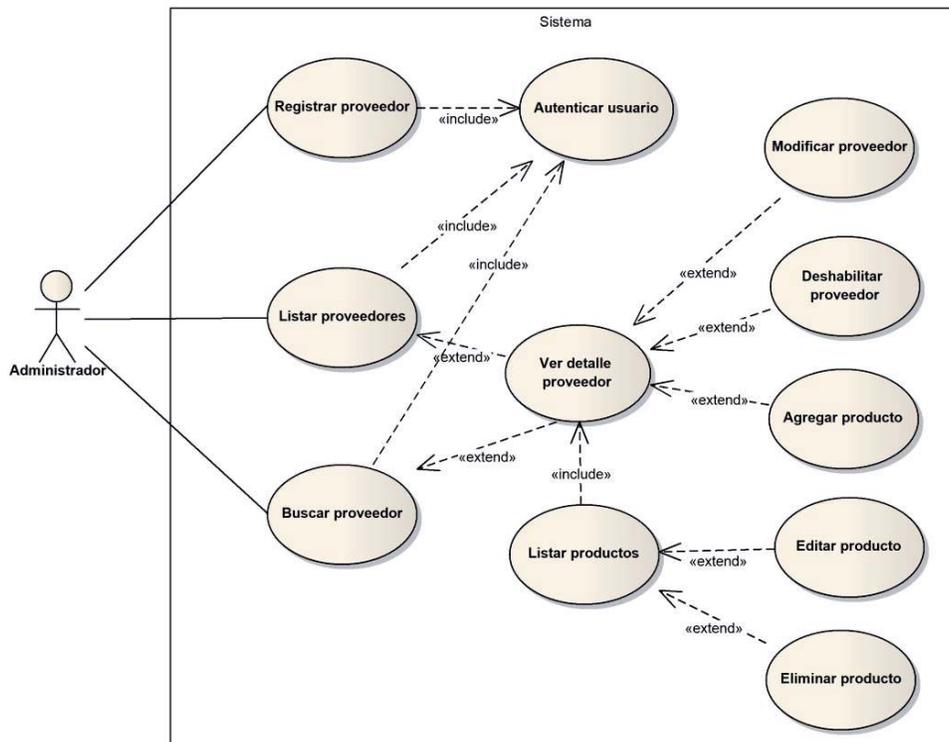


Figura 9.10 Caso de uso Gestionar Proveedores

9.11 Caso uso Gestionar Comunicativos

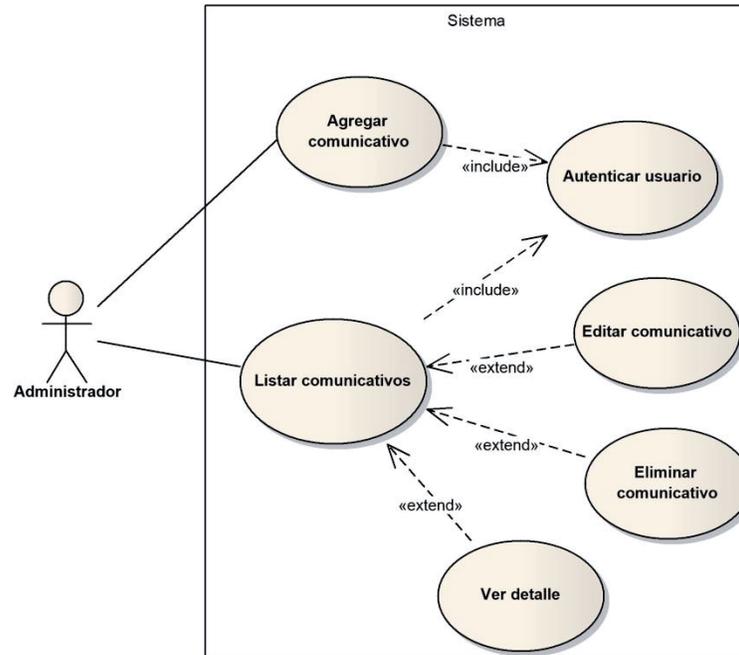


Figura 9.11 Caso de uso Gestionar Comunicativos

9.12 Caso de uso Administrar cuentas.

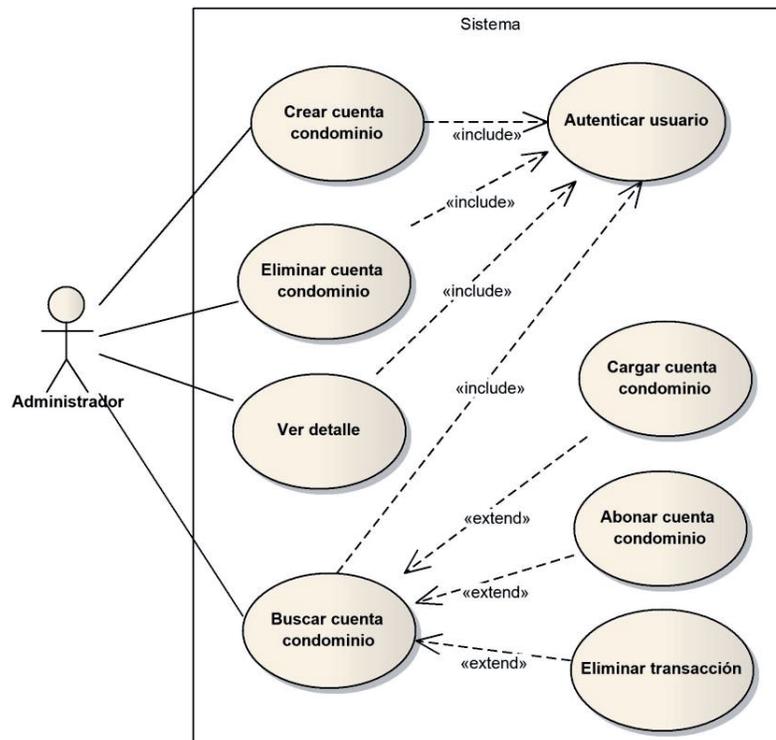


Figura 9.12 Caso de uso Administrar cuentas

9.13 Caso de uso Gestionar Condominio

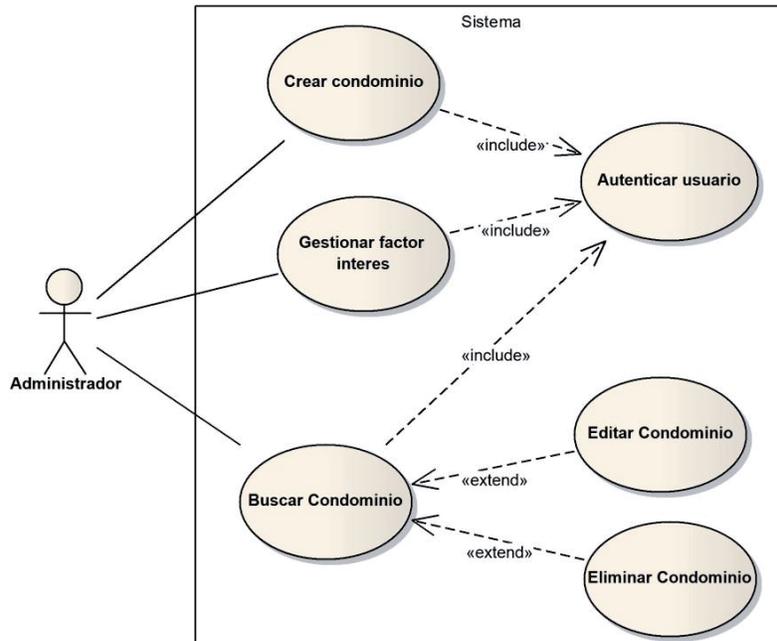


Figura 9.13 Caso de uso Gestionar Condominio

9.14 Caso de Uso Gestionar Perfil

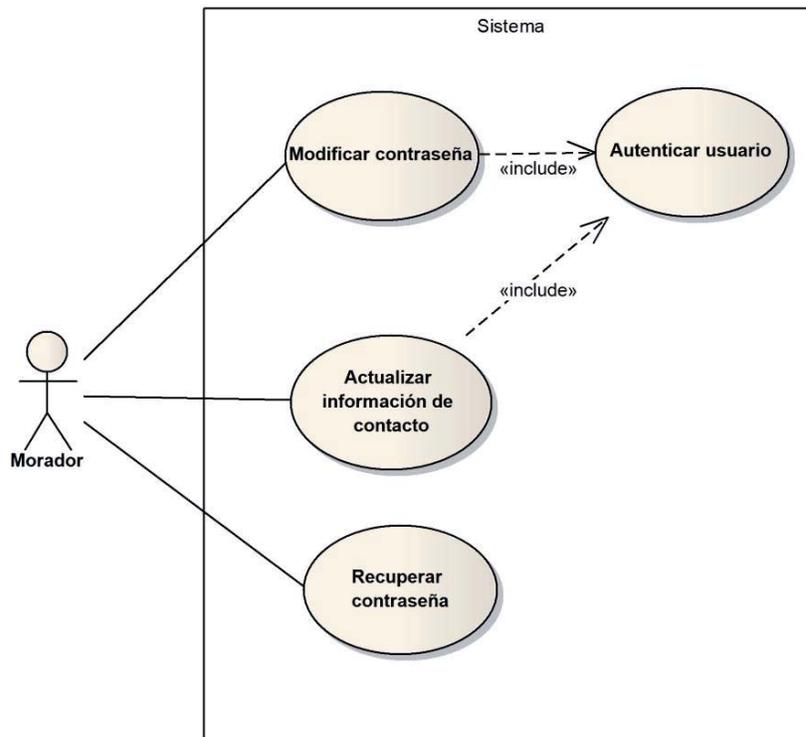


Figura 9.14 Caso de uso Gestionar Perfil

9.15 Caso de Uso Consultar Gastos Comunes

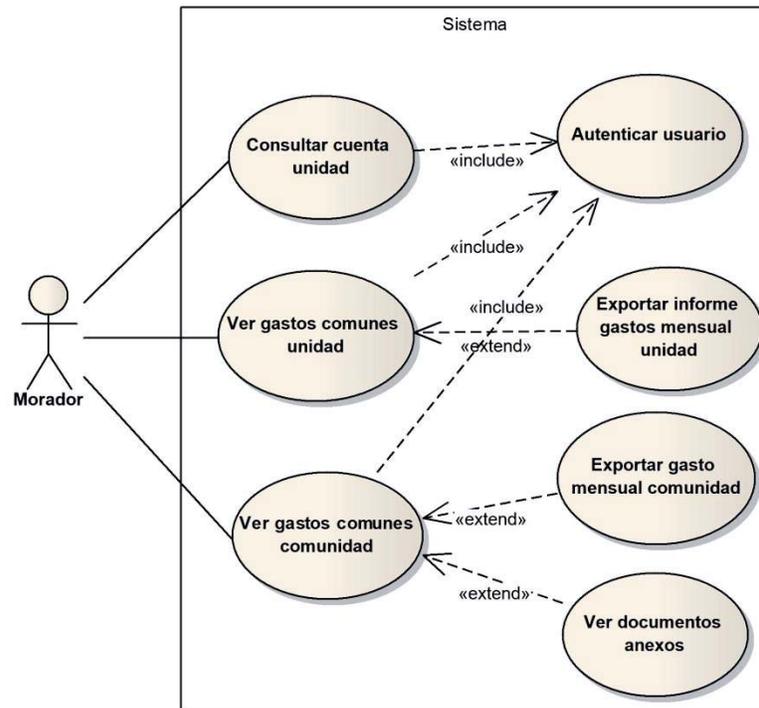


Figura 9.15 Caso de uso Consultar Gastos Comunes

9.16 Caso de Uso Enviar mensajes

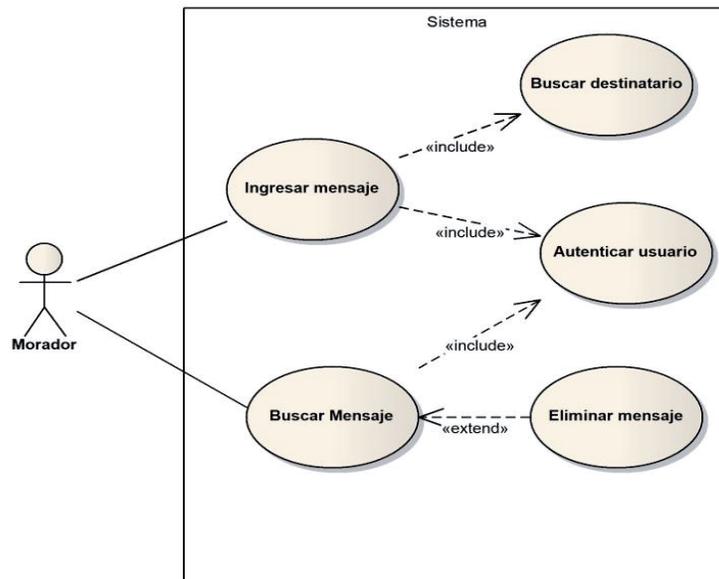


Figura 9.16 Caso de uso Enviar Mensajes

Para ver más diagramas revisar el Anexo E.

9.18 Modelo De Datos Lógico

Un modelo de datos es un lenguaje orientado a describir una base de datos, típicamente puede describir la estructura de los datos, en qué forma se relacionan y la restricción de integridad de estos datos. El siguiente modelo lógico se obtuvo a partir del diagrama clases y su mapeo al modelo relacional, aplicando ciertos niveles de normalización y tablas intermedias. Por lo amplio del modelo se optó por mostrar la relación entre las tablas y se agruparon en secciones. Para visualizar el detalle de las mismas revisar el Anexo F.

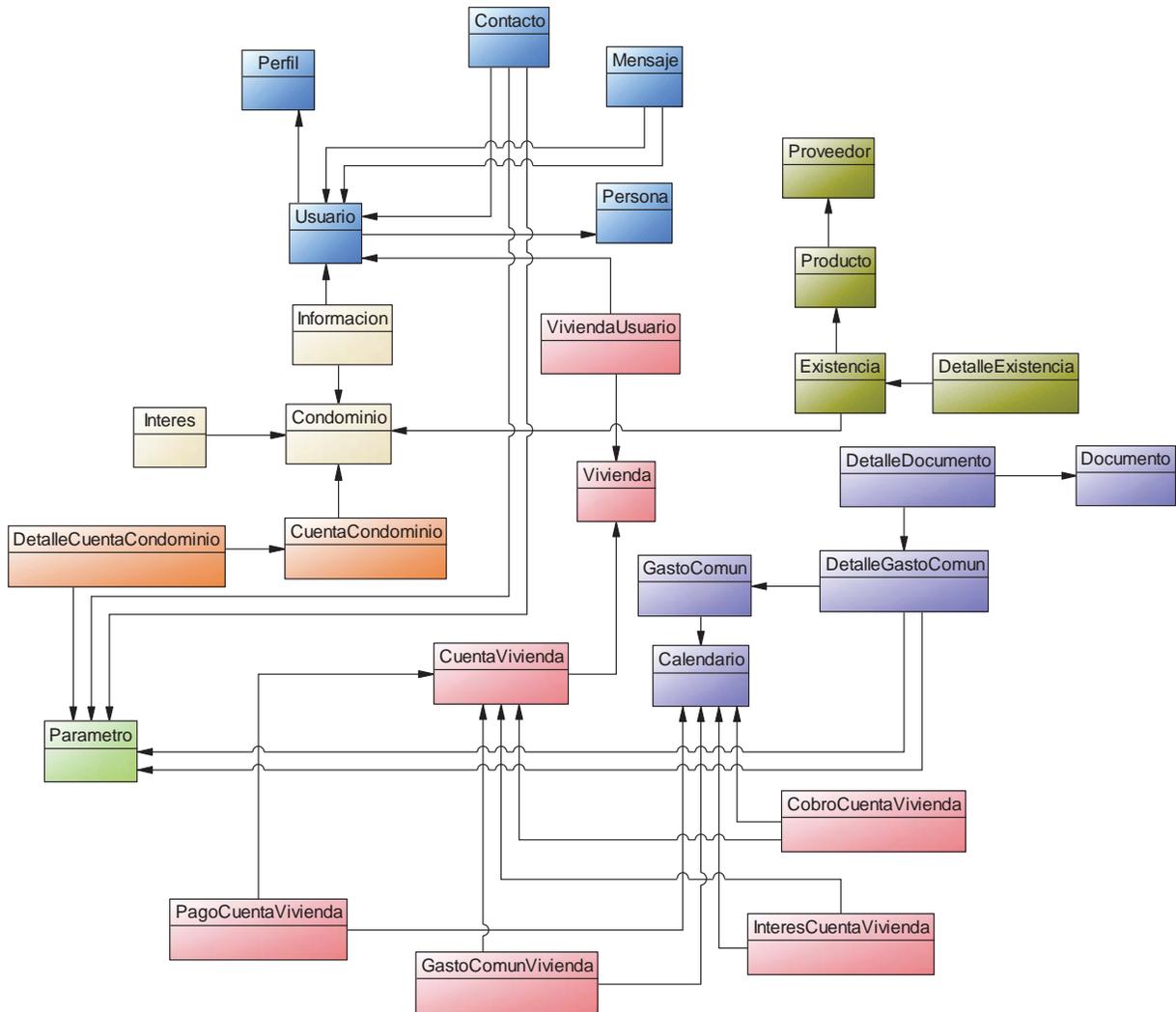


Figura 9.18 Modelo de Datos Lógico

10 Implementación

Tras analizar y diseñar la solución web para el condominio corresponde implementar la aplicación. Primeramente se llevó el modelo relacional al programa PowerDesigner adaptándolo a una base de datos MYSQL Versión 5.0 para conseguir el modelo físico y luego ejecutar el script generado por la aplicación en el motor.

Posteriormente se procedió a la configuración del entorno de programación que consistió en la instalación del lenguaje de servidor PHP, luego el servidor web APACHE 2.2 y por último el bloc de notas Sublime Text incorporándole algunos plugins recomendados para facilitar el desarrollo.

Se procedió a crear una clase para la conexión a la base de datos para empezar a desarrollar los mantenedores requeridos. En el siguiente código se muestra un extracto de la codificación de la clase Persona que es la clase base de Usuario.

```
class Persona {

    private $idPersona;
    private $run;
    private $sexo;
    private $nombres;
    private $apellidoPaterno;
    private $apellidoMaterno;
    private $fechaNacimiento;
    private $nacionalidad;

    function __construct($idPersona,$run,$sexo,$nombres,
    $apellidoPaterno,$apellidoMaterno, $fechaNacimiento,$nacionalidad) {
        $this->run = $run;
        $this->sexo = $sexo;
        $this->nombres = $nombres;
        $this->apellidoPaterno = $apellidoPaterno;
        $this->apellidoMaterno = $apellidoMaterno;
        $this->fechaNacimiento = $fechaNacimiento;
        $this->nacionalidad = $nacionalidad;
    }

    public final function getRun() {
        return $this->run;
    }
    public final function getNombres() {
        return $this->nombres;
    }
    public function getApellidoPaterno() {
        return $this->apellidoPaterno;
    }
    public function getApellidoMaterno() {
        return $this->apellidoMaterno;
    }
    // Muestra de implementación.....
}
```

11 Diseño de Interfaz

La interfaz web es el conjunto gráfico que permite la presentación y la navegación del sistema. Esto se consigue con la inclusión de elementos gráficos comunes a toda la web que son estándares, haciendo que los usuarios tengan completo control sobre las funcionalidades del sitio desde el momento mismo de entrar a él sin que para ello deba tener amplios conocimientos ni preparación anterior alguna. En las siguientes imágenes se muestra el diseño de las interfaces en donde el administrador podrá ingresar los gastos comunes para un periodo seleccionado y donde el copropietario visualizará los gastos. Para ver más diseños de interfaces ver Anexo G.

ENERO - 2017 Valor

* Categoría:
 * Subcategoría:
 Mostrar registros eliminados Ningún archivo seleccionado

GASTO COMÚN		Enero - 2017		ESTADO	
GASTOS DE ADMINISTRACIÓN					
PERSONAL DE CONSERJERÍA Y SEGURIDAD					
	2 conserjes Full Time		785,000		Detalle Editar Eliminar
HONORARIOS DE ADMINISTRACIÓN					
	Gastos de administración (factura 449)		1,051,226		Detalle Editar Eliminar
	TOTAL GASTOS DE ADMINISTRACIÓN		1,836,226		
CONSUMOS BÁSICOS					
ELECTRICIDAD Y ESPACIOS COMUNES					
	Factura 16335254		1,200,000		Detalle Editar Eliminar
	TOTAL CONSUMOS BÁSICOS		1,200,000		
GASTOS DE MANTENCIÓN Y REPARACIÓN					
MANTENCIÓN REPARACIÓN DE ASCENSORES					
	Factura 3543246		500,000		Detalle Editar Eliminar
	TOTAL GASTOS DE MANTENCIÓN Y REPARACIÓN		500,000		
	TOTAL DEL MES		3,536,226		

Figura 11.1 Interfaz del administrador: Ingresar gastos comunes

Condominio Nitella | Informaciones | Mis Gastos Comunes | Gastos Comunidad | Mensajes | Perfil

Mes: Año:

GASTO COMÚN		Enero - 2017		ESTADO	
GASTOS DE ADMINISTRACIÓN					
PERSONAL DE CONSERJERÍA Y SEGURIDAD					
	2 conserjes Full Time		785,000		
HONORARIOS DE ADMINISTRACIÓN					
	Gastos de administración (factura 449)		1,051,226		
	TOTAL GASTOS DE ADMINISTRACIÓN		1,836,226		
CONSUMOS BÁSICOS					
ELECTRICIDAD Y ESPACIOS COMUNES					
	Factura 16335254		1,200,000		
	TOTAL CONSUMOS BÁSICOS		1,200,000		
GASTOS DE MANTENCIÓN Y REPARACIÓN					
MANTENCIÓN REPARACIÓN DE ASCENSORES					
	Factura 3543246		500,000		
	TOTAL GASTOS DE MANTENCIÓN Y REPARACIÓN		500,000		
	TOTAL DEL MES		3,536,226		
OBSERVACIÓN					
Sin observaciones					

Figura 11.2 Interfaz del copropietario: Gastos de la comunidad

12 Plan de Prueba

Un sistema de pruebas está diseñado para encontrar posibles errores y determinar situaciones en donde algo pasa cuando no debería pasar.

Un sistema de pruebas implica la aplicación de una serie de casos de prueba los cuales intentan buscar errores en el software desarrollado. En una palabra, un sistema de pruebas está orientado a detectar. Las pruebas no pueden asegurar la ausencia de errores; sólo puede demostrar que existen defectos en el software.

12.1 Objetivos

Las pruebas en un desarrollo de software representa una etapa más del proceso y su objetivo es asegurar que el software cumpla con las especificaciones requeridas y eliminar los posibles defectos que éste pudiera tener.

Objetivos de las Pruebas:

- Encontrar defectos en el software.
- Una prueba es considerada exitosa si se descubre un defecto.
- Una prueba fracasa si hay un defecto y no las descubre.
- Asegurar la calidad del software.

12.2 Enfoque de las pruebas

Existen variados tipos de pruebas con diferentes enfoques, algunos de ellas son:

- **Pruebas de Caja Negra:** El sistema de pruebas de caja negra no considera la codificación dentro de sus parámetros a evaluar, es decir, que no están basadas en el conocimiento del diseño interno del programa. Estas pruebas se enfocan en los requerimientos establecidos y en la funcionalidad del sistema.
- **Pruebas de Caja Blanca:** Al contrario de las pruebas de caja negra, éstas se basan en el conocimiento de la lógica interna del código del sistema. Las pruebas contemplan los distintos caminos que se pueden generar gracias a las estructuras condicionales, a los distintos estados del mismo, etc.

12.3 Planificación de las pruebas:

Dentro de este proyecto se realizaron tipos de prueba que demanden un tiempo acotado y a la vez sea confiable a la hora de detectar errores. Como es habitual sobre el proyecto se realizan pruebas unitarias sobre todas las funcionalidades de manera informal para luego generar un documento que detalle las que se aplicarán.

Para realizar la planificación de pruebas en este sistema se utilizó el enfoque de caja negra, para ver más detalle de las pruebas aplicadas revisar el Anexo H.

13 Conclusiones

El desarrollo de un proyecto informático puede llevarse a cabo de distintas formas pero la aplicación de metodologías formales, el buen entendimiento del problemática, los requerimientos enfocados en satisfacer las necesidades del cliente junto con una planificación correcta conducen a la obtención de un buen sistema de software.

En el mercado actual se encontraron varias empresas que se dedican a ofrecer sistemas de software para la gestión condominios las cuales cobran generalmente licencias mensuales. A pesar de que aún hay condominios que operan con documentación tradicional estos tendrían varios beneficios importantes al migrar su información hacia un sistema automatizado que les ayude a manejar la información.

Para la fase de identificación de requerimientos es necesario tener una fluida comunicación con el cliente para identificar sus necesidades, documentarlas y estudiar que tecnologías que podrían adaptarse de mejor forma a la problemática y así proponer una solución.

La implementación del software para la administración en el condominio “Nitella” incidirá positivamente en la automatización del proceso porque generará un mejor desempeño en las labores de administración y lograrán tener mejor organizada la información y será más transparente para las partes interesadas.

El desarrollo del sistema influyó bastante en la aplicación, reforzamiento y aprendizaje de nuevos conocimientos, este proyecto representa un primer acercamiento a lo que podría ser un desarrollo en el mundo laboral. Al diseñar y modelar el sistema utilizando el lenguaje de unificado de modelado UML se obtuvo una visión detallada y explicativa de los requisitos definidos especificando tanto el modelo como el funcionamiento de la aplicación.

Durante la codificación del proyecto pueden ir surgiendo cambios menores en las necesidades o en el diseño inicial, es necesario documentar estos cambios para tener un mejor control del proyecto. También se debe tener una buena organización en los tiempos y buscar formas o herramientas que permitan agilizar las tareas de desarrollo del producto y tener más bien un enfoque en la lógica más que en el desarrollo de tareas repetitivas.

Por el tiempo dispuesto para la creación de este software este último apunta a resolver las principales gestiones de administración del condominio, sin embargo es posible seguir escalando la solución y lograr un producto más completo en el tiempo.

14 Referencias

- [1] <http://demo.kol.cl/kol/ingresos/login> Revisado por última vez el 4 Diciembre de 2016.
- [2] <https://tucondominioaldia.net/> Revisado por última vez el 4 Diciembre de 2016.
- [3] <https://s3.amazonaws.com/habitek-comunidades/Archivos/guia-rapida-uso-v2.11.pdf>
Revisado por última vez el 4 Diciembre de 2016.
- [4] <http://ingsoftware072301.obolog.es/up-proceso-unificado-2010775> Revisado por última vez el 7 Junio de 2017.
- [5] <http://www.easyhosting.cl/> Revisado por última vez el 4 Diciembre de 2016.
- [6] http://www.minvu.cl/opensite_20160405114506.aspx Revisado por última vez el 10 de Junio de 2016.
- [7] Schmuller, Aprendiendo UML en 24 Horas, Año 2000.
- [8] Pressman R, Ingeniería de Software: Un Enfoque Práctico, Editorial McGraw-Hill, Quinta Edición, Año 2002.
- [9] Kimmel Paul, Manual de UML: Guía de aprendizaje, Editorial McGraw-hill profesional, Primera Edición, Año 2008.
- [10] Martin Fowler, UML gota a gota, Editorial Pearson Educacion, Año 1999.

15 Anexos

Anexo A: Estado del Arte

A.1 Analizando Software Kastor

El producto comercial Kastor ofrece características como reporte de morosos, editar la información del usuario, análisis de los gastos, contacto de la comunidad y un módulo de pagos en línea, cabe destacar que la licencia tiene un valor aproximado de 3 UF para administrar un condominio y 5 UF en su versión ilimitada, información que fue proporcionada al tomar contacto con una de sus ejecutivas de venta, los valores son mensuales e incluyen el soporte.

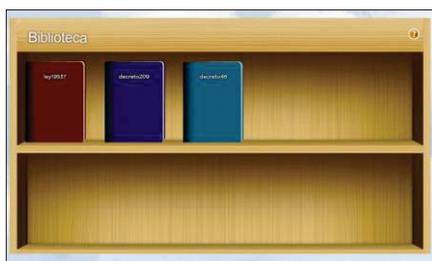


Figura A. 1 Interfaz del software Kastor: Módulo Biblioteca [1]

Al ingresar al módulo Mis solicitudes se visualiza una interfaz en la que se ven las informaciones publicadas por los usuarios en la que aparece la identificación de quien publicó.

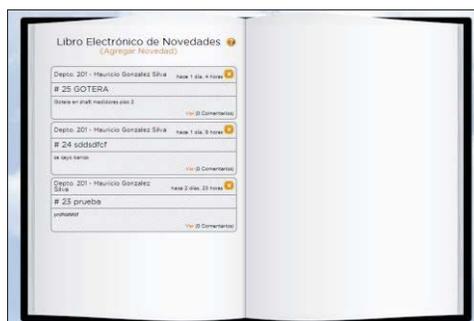


Figura A. 2 Interfaz del software Kastor: Módulo Mis Solicitudes [1]

Y por último en el módulo de gastos comunes se puede encontrar información detallada de los gastos mensuales clasificados en grupos como remuneraciones, administrativos, de servicios, de mantenimiento, generales y seguros.

Detalle Gasto Común	
Mes: Abril	Año: 2010
Gastos Comunes: Abril 2010	
Remuneraciones	
Buenos Ilíquidos	\$1.416.771
Anticipos	-
Lejes Sociales	\$372.901
	\$1.789.672
Gastos de Administracion	
Honorarios de Administracion (Documento Nº: 2140)	\$300.000
Descuento de Honorarios por Entregas según Contrato (Documento Nº: 2140)	-\$36.000
	\$264.000
Gastos de Servicios	
Agua (Documento Nº: 14102815)	\$358.309
Telefono (Documento Nº: 1970188)	\$48.944
Electricidad (Documento Nº: 6573569)	\$1.104.470
Gas (Documento Nº: 45777748)	\$1.261.680
	\$2.773.403
Gastos de Mantenion	
Areas Verdes (Documento Nº: 32)	\$285.600
Central Termica (Documento Nº: 1103)	\$117.504
Bombas (Documento Nº: 2672)	\$49.923
Ascensores (Documento Nº: 138054)	\$500.850
Sistema de Seguridad (Documento Nº: 145)	\$149.940
	\$1.103.817
Gastos de Reparacion	
Central Termica (Documento Nº: 1130)	\$341.280
	\$341.280
Gastos Generales	
Articulos de Aseo (Documento Nº: 113012)	\$115.902
Turnos Extras (Documento Nº: 042010, 10004)	\$140.000
Uniformes Personal (Documento Nº: 113458, 113459)	\$115.526
Turnos Extras Aseo (Documento Nº: 13042010, 410, 1004)	\$288.000
Articulos Electricos (Documento Nº: 113014)	\$114.787
Articulos de Escritorio (Documento Nº: 114542, 113013)	\$3.954
Insumos Personal (Documento Nº: 113011)	\$6.682
Contribuciones (Documento Nº: 042010, 201004)	\$1.689.940
Entrega de Correspondencia (Documento Nº: 042010)	\$20.000
Directiva de Funcionamiento (Documento Nº: 1975)	\$40.000
Impresion Gastos Comunes (Documento Nº: 168)	\$3.243
	\$2.538.044
Gastos Seguros	
Seguro Incendio (Documento Nº: 5810638)	\$412.163
	\$412.163
TOTAL GASTOS COMUNES	
	\$9.222.679
FONDO DE RESERVA	
	\$276.662
TOTAL	
	\$9.499.341
Observaciones: FAVORGIRAR CHEQUE CRUZADO Y NOMINATIVO A NOMBRE DE EDIFICIO LEONARDO.	
DATOS PARA TRANSFERENCIA: EDIFICIO LEONARDO, RUT: 53.312.283-3, BANCO SECURITY, CATA NÁ* E-1212117-91, MAIL DE AVISO INDICANDO NÁ* DE DEPTO: agm@harrison.cl	

Figura A. 3 Interfaz del software Kastor: Módulo Generación Gastos Comunes [1]

A.2 Analizando Software Tu condominio al día

Al buscar en internet se ha encontrado la página web <http://www.tucondominioaldia.com.ve/> que publicita un software con muchas características interesantes entre las cuales se pueden nombrar las siguientes: Fácil, Rápido, Amigable, Potente.

Es una plataforma web que integra mensajería de texto con una sincronización con una empresa que provea el servicio. Ofrece ayuda personalizada, un respaldo completo de la información y asesoría permanente.

También se pueden mencionar las funcionalidades de envíos por correo electrónico de los recibos, estados de cuenta, pagos realizados.

Su interfaz la promueven como un ambiente de Windows; gráfico, amigable e intuitivo, que le permite al usuario integrarse a las herramientas, métodos disponibles en el sistema y obtener todo el potencial desde un primer contacto. Estos ofrecen un presupuesto online a sus potenciales usuarios y una prueba demo del mismo producto. Algunas de las interfaces que se pueden obtener en la versión demo son las siguientes:

Fecha	Concepto	Debe	Haber	Saldo	Facturas
2012-02-29	febrero/12	135.41	135.41	0.00	,20
2012-02-29	VIGILANCIA	68.00	68.00	0.00	,20
2012-02-29	CUOTA EXTRA MANTENIMIENTO DE EDIFICIO	100.00	100.00	0.00	,20
2012-03-31	marzo/2012	85.12	85.12	0.00	,85
2012-03-31	VIGILANCIA	68.00	68.00	0.00	,85
2012-03-31	CUOTA EXTRA MANTENIMIENTO DE EDIFICIO	100.00	100.00	0.00	,85
2012-04-30	abril/2012	71.13	0.00	71.13	
2012-04-30	VIGILANCIA	68.00	0.00	68.00	
2012-04-30	CUOTA EXTRA DE MANTENIMIENTO	100.00	0.00	100.00	
Totales		795.66	556.53	239.13	

Figura A. 4 Interfaz del software Tu Condominio al Día: Estado de Cuenta [2]

Fecha	Concepto	Debe	Haber	Saldo	Facturas
2012-04-30	abril/2012	71.13	0.00	71.13	
2012-04-30	VIGILANCIA	68.00	0.00	68.00	
2012-04-30	CUOTA EXTRA DE MANTENIMIENTO	100.00	0.00	100.00	
Total		71.13	0.00	71.13	

Figura A. 5 Interfaz del software Tu Condominio al Día: Cuentas por Pagar [2]

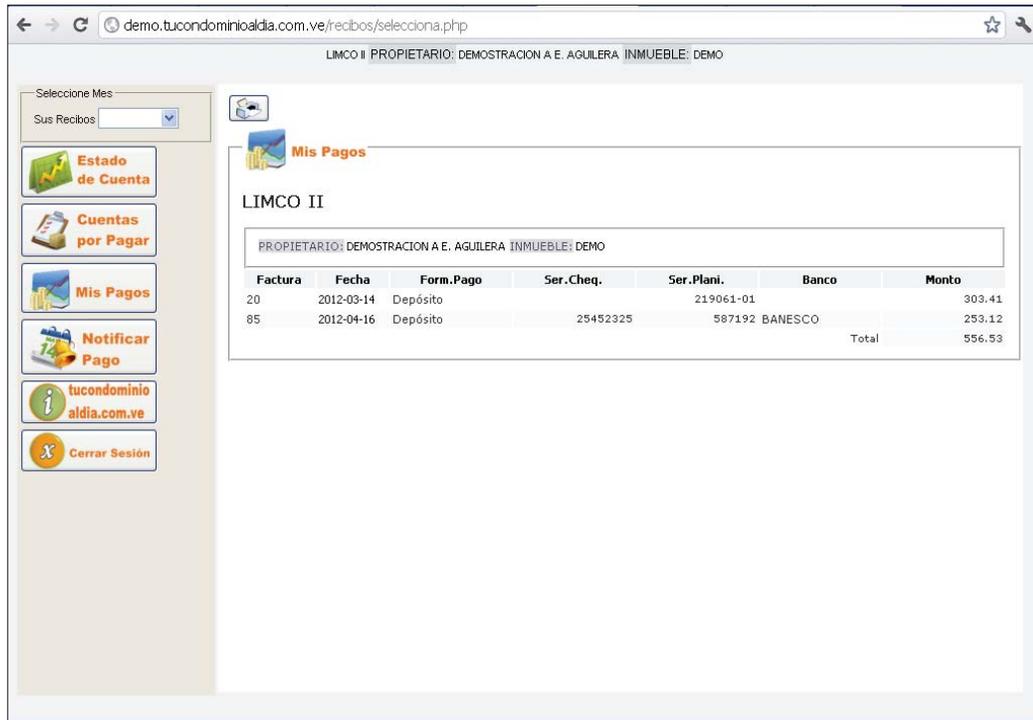


Figura A. 6 Interfaz del software Tu Condominio al Día: Mis Pagos [2]

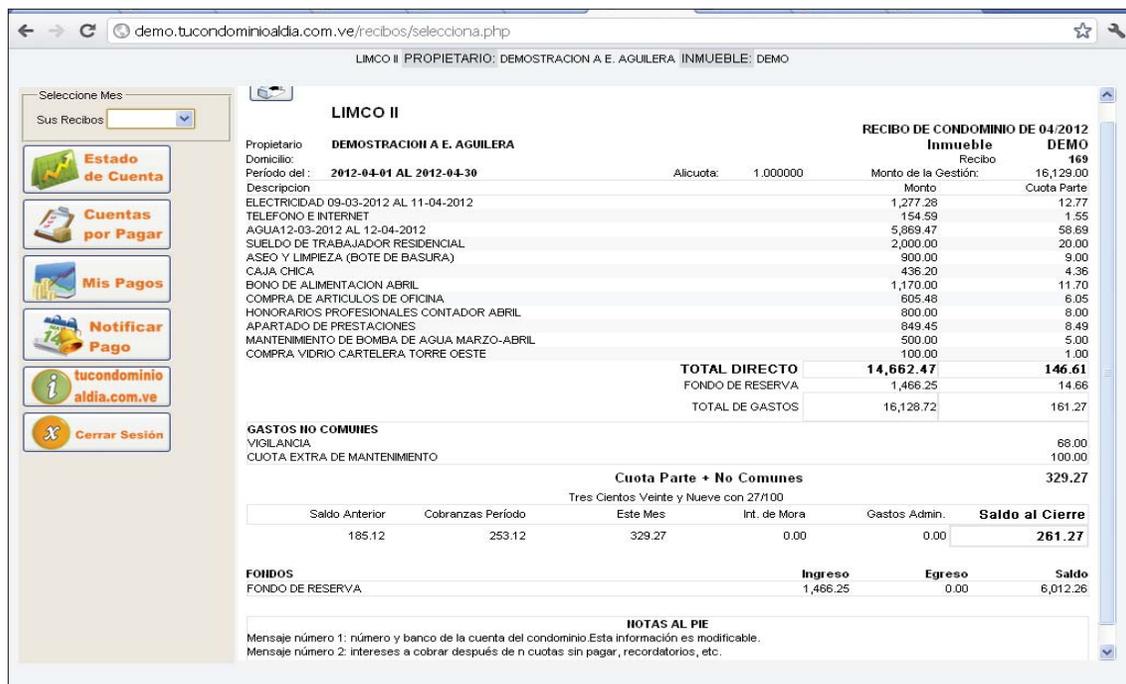


Figura A. 7 Interfaz del software Tu Condominio al Día: Recibos del Condominio [2]

A.3 Analizando Software Habitek

El software Habitek es una plataforma web que tiene un manual público del que se pueden observar algunas de sus características. En la siguiente imagen se visualiza la interfaz de inicio de sesión.

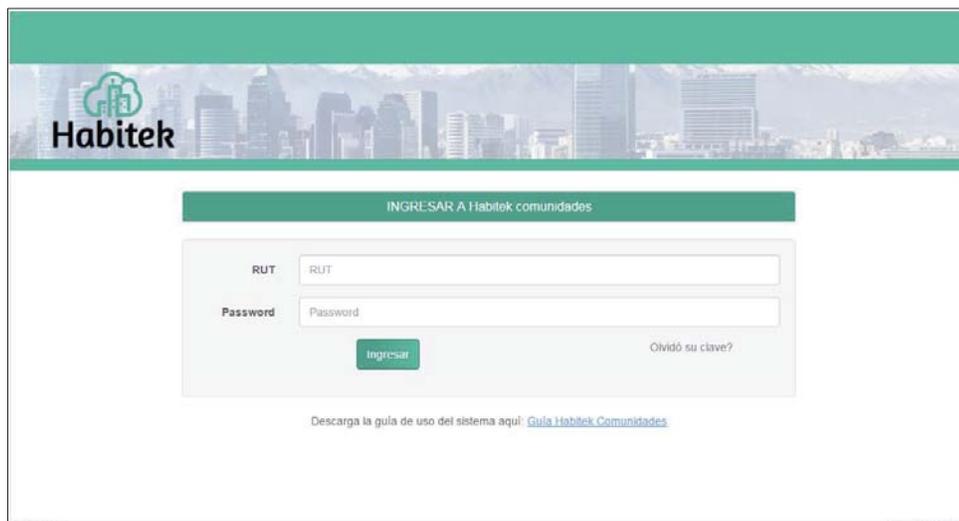


Figura A. 8 Interfaz del software Habitek: Inicio de Sesión [3]

En el módulo de los copropietarios se puede observar un menú con las siguientes secciones:

- Información
- Reglamentos
- Mis gastos comunes
- Gastos de la Comunidad
- Informes
- Perfil de Usuario



Figura A. 9 Interfaz del software Habitek: Secciones Copropietarios [3]

La sección Mis gastos comunes muestra el detalle de los gastos comunes a pagar por un departamento, estacionamiento y/o bodega. Tiene un selector que permite ver el gasto común de meses anteriores y el detalle es organizado en varias secciones. Algunas secciones tienen el ícono de una lupa la que al presionarla muestra más información asociada al cobro etc.



Figura A. 10 Interfaz del software Habitek: Mis Gastos Comunes [3]

En general los sistemas ofrecen funcionalidades similares que satisfacen las necesidades de los usuarios tanto administradores como copropietarios en la comunidad. En la siguiente tabla se realiza la comparación de algunas características.

Tabla A. 1 Características de Kastor, Tu condominio al Día y Habitek

	Kastor	Tu condominio al día	Habitek
Aplicación de escritorio	✓	✓	✗
Plataforma web	✓	✓	✓
Mensajería texto	✗	✓	✓
Pago en línea	✓	✗	✓

El software Kastor y Habitek ofrecen la modalidad de pago en línea, permitiéndole a los usuarios optimizar los tiempos en este concepto, sin embargo Tu condominio al día se centra en los recordatorios vía mensajería de texto de las cobranzas considerándolo como el medio más efectivo para realizar gestiones de cobranza, notificaciones e informaciones de índole en general.

Anexo B: Descripción de la solución

B.1 Paradigma de Desarrollo del Proceso Unificado

En la siguiente figura se muestra en forma gráfica el proceso UP.

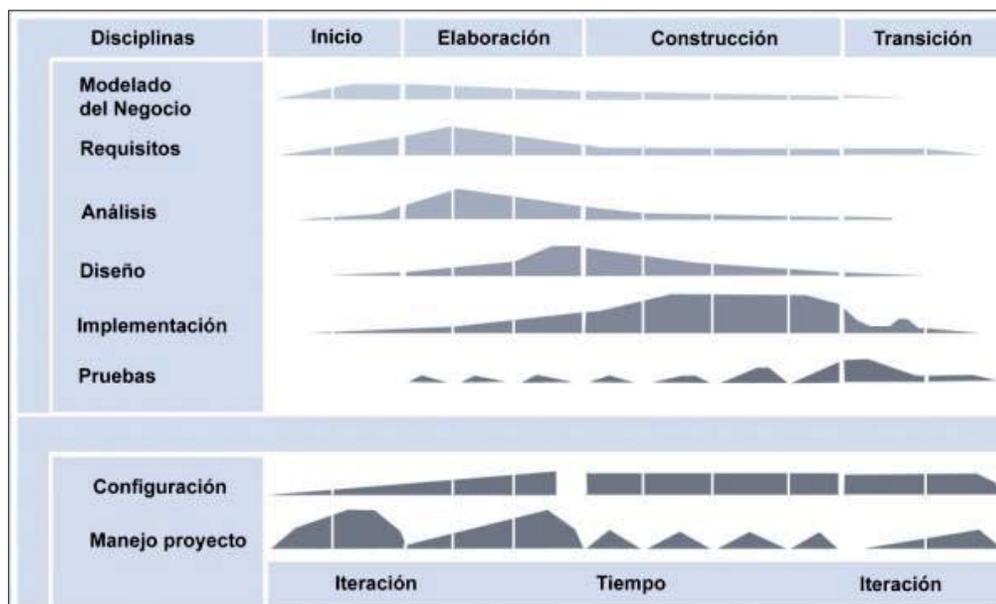


Figura B. 1 Fases del paradigma UP [4]

El UP (Proceso Unificado) es un modelo dividido en cuatro fases que se detallan a continuación:

Fase de Inicio: Agrupa actividades tanto de comunicación con el cliente como de planeación. Al colaborar con los participantes, se identifican los requerimientos del negocio, se propone una arquitectura aproximada para el sistema y se desarrolla un plan para la naturaleza iterativa e incremental del proyecto en cuestión. Los requerimientos fundamentales del negocio se describen por medio de un conjunto de casos de uso preliminares que detallan las características y funciones que desea cada clase de usuarios. La planeación identifica los recursos, evalúa los riesgos principales, define un programa de actividades y establece una base para las fases que se van a aplicar a medida que avanza el incremento del software.

Fase de Elaboración: Incluye las actividades de comunicación y modelado del modelo general del proceso. La elaboración mejora y amplía los casos de uso preliminares desarrollados como parte de la fase de inicio y aumenta la representación de la arquitectura.

Los objetivos de la Fase de elaboración son desarrollar una comprensión del dominio del problema, establecer un marco de trabajo arquitectónico para el proyecto e identificar los riesgos claves del proyecto. Durante la fase de elaboración se especifican en detalle la mayoría de los casos de uso del producto y se diseña la arquitectura.

Fase de Construcción: Esta fase fundamentalmente comprende avance en el diseño del sistema, la programación y las pruebas. Durante esta fase se desarrollan e integran las partes del sistema. Al finalizar esta etapa se deben tener un software operativo y su documentación lista para una entrega al usuario.

Con el uso del modelo de arquitectura como entrada, la fase de construcción desarrolla o adquiere los componentes del software que harán cada caso de uso sea operativo para los usuarios finales. Para lograrlo, se completan los modelos de requerimientos y diseño que se comenzaron durante la fase de elaboración, a fin de que reflejen la versión final del incremento del software. Luego se implementan en código fuente todas las características y funciones necesarias para el incremento del software. A medida de que se implementan los componentes, se diseñan y efectúan pruebas unitarias para cada uno. Se emplean casos de uso para obtener un grupo de pruebas de aceptación que se ejecutan antes de comenzar la siguiente fase del UP.

Fase de Transición: Incluye las últimas etapas de la actividad general de construcción y la primera parte de la actividad de despliegue general. Durante esta fase, se vigila el uso que se da al software, se brinda apoyo para el ambiente de operación (infraestructura) y se reportan defectos y solicitudes de cambio para su evaluación.

La perspectiva práctica en el UP describe buenas prácticas de la ingeniería del software que son aconsejables para el desarrollo del sistema entre las cuales se pueden nombrar:

- Desarrollo iterativo e incremental del software.
- Gestión de los requerimientos.
- Uso de arquitectura basada en componentes
- Modelamiento visual del sistema.
- Verificación de la calidad del software.
- Control de cambios

B.1.1 Elección del Paradigma

Debido a los modelos antes mencionados, se ha elegido el modelo Proceso Unificado debido a los siguientes motivos:

- Por ser un paradigma iterativo que además integra las ventajas del modelo en cascada dentro de cada una de sus iteraciones o fases.
- Al ser una de sus características ser iterativo e incremental, permite la detección de errores y riesgos importantes dentro del proyecto en forma temprana.
- Este va de la mano con el lenguaje de modelado UML el cual se utilizará en la documentación de este proyecto y la orientación a objetos que será la metodología escogida.

B.2 Herramientas y Tecnologías

B.2.1 Análisis y diseño

En esta sección se dar a una breve descripción de las tecnologías a utilizar para el desarrollo de este proyecto.

Microsoft Word 2007: Es un software procesador de texto con el cual se escribirá la documentación.

Microsoft Power Point 2007: Programa diseñado para hacer presentaciones con texto y animación e imágenes de fácil uso y con muchas características ventajosas.

Diagrammer UML: Este software sirve para crear los diagramas en lenguaje UML el cual es un lenguaje de especificación, visualización, construcción y documentación especializado en el área de sistemas de software.

Dia: Es una aplicación informática de propósito general para la creación de diagramas, desarrollada como parte del proyecto GNOME. Está concebido de forma modular, con diferentes paquetes de formas para diferentes necesidades.

B.2.2 Motor Bases de datos

MySQL es un sistema de gestión de bases de datos relacional, licenciado bajo la GPL de la GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente.

Las principales características de este gestor de bases de datos son las siguientes:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, etc).
- Gran portabilidad entre sistemas.
- Gestión de usuarios y password, manteniendo un muy buen nivel de seguridad en los datos.

B.2.3 Entornos de desarrollo

Geany para Windows: Es un editor de Texto ligero con características básicas de entorno de desarrollo integrado (IDE). Está disponible para distintos sistemas operativos y es distribuido como software libre bajo la Licencia Pública General de GNU.

B.2.4 Lenguajes de programación

PHP: Es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante.

JAVA SCRIPT: Es un lenguaje de programación interpretado. Se define como orientado a objetos, basado en prototipos, imperativo y dinámico. Se utiliza principalmente en su forma del lado del cliente (Client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas.

B.2.5 Servidor web local

APACHE: El servidor HTTP Apache es un servidor web HTTP de código abierto, para plataformas Unix (BSD, GNU/Linux, etc.), Microsoft Windows, Macintosh y otras, que implementa el protocolo HTTP/1.1. Este servidor será de gran utilidad para trabajar a nivel local.

Anexo C: Factibilidad

C.1 Factibilidad técnica

C.1.1 Base de datos

El gestor de base de datos a utilizar será MYSQL Community edition. Esta versión es de uso gratuito y es soportada por muchos hosting en donde se alojara el sistema.

Los requerimientos básicos para utilizar este gestor de bases de datos son:

- Un sistema operativo Windows de 32 bits, tal como 9x, Me, NT, 2000, XP, o Windows Server 2003.
- Se recomienda fuertemente el uso de un sistema operativo Windows basado en NT (NT, 2000, XP, 2003) puesto que éstos permiten ejecutar el servidor MySQL como un servicio.
- Soporte para protocolo TCP/IP.
- Suficiente espacio en disco rígido para descomprimir, instalar, y crear las bases de datos de acuerdo a sus requisitos. Generalmente se recomienda un mínimo de 200 megabytes.

C.1.2 Editor de texto

El editor de texto a utilizar es Sublime Text 2 que es compatible con la mayoría de sistemas operativos. En la siguiente tabla se muestran los requerimientos.

Tabla C. 1 Requisitos mínimos de Sublime Text

Sistema operativo	Windows XP Service Pack 2 o posterior Windows Vista Windows 7 Windows 8
Procesador	Intel Pentium 4 o posterior
Espacio libre en disco	30 MB
RAM	128 MB

C.1.3 Hardware Desarrollador

Tabla C. 2 Hardware desarrollador

Modelo	NOTEBOOK HP 430
CPU	Intel Pentium p6200 2,13 GHz
Memoria RAM	4 GB
Disco duro	500 GB
Sistema operativo	Windows 7 32 BITS

C.1.4 Navegador Chrome

Google Chrome está disponible para Windows, Mac y Linux. Para conseguir un rendimiento óptimo, es recomendable que el sistema cumpla los siguientes requisitos:

Tabla C. 3 Requisitos de instalación Navegador Chrome

	Requisitos de Windows	Requisitos de Mac	Requisitos de Linux
Sistema operativo	Windows XP Service Pack 2 o posterior Windows Vista Windows 7 Windows 8	Mac OS X 10.6 o posterior	Ubuntu 10.04 o posterior Debian 6+ OpenSuSE 11.3+ Fedora Linux 14
Procesador	Intel Pentium 4 o posterior	Intel	Intel Pentium 3 / Athlon 64 o posterior
Espacio libre en disco	100 MB		
RAM	128 MB		

C.2 Factibilidad económica

La factibilidad económica es donde se ven los costos y ganancias dentro de la empresa o institución en este caso la administración del condominio Nitella. Se hará un estudio de costo-beneficio para ver si el sistema a realizar será un beneficio en el ámbito económico en cuestión de ahorro y ganancias.

C.2.1 Costos

Para dejar el sistema operativo se arrendara un servicio de hosting y un domino que represente el sistema y sea fácil de recordar para los usuarios.

C.2.1.1 Hosting

Al cotizar en la página web www.bluehosting.cl se ofrece el siguiente plan y características con un costo anual de aproximado de \$10990 pesos. Las características del servicio cumplen técnicamente para subir el sistema a desarrollar.

Tabla C. 4 Características del servicio de hosting

Servidores Dell	Servidores Dell PowerEdge de alto rendimiento
Data center	Datacenter propio en Chile- WorldClass TIER III
Espacio Web	Ilimitado
Dominios permitidos	1
Transferencia manual	120 GB
Bases de datos MySQL / PostgreSQL	12
Cuentas de E-mail	Ilimitadas
Subdominios	Ilimitados

C.2.1.2 Dominio

En la página www.nic.cl ofrecen el registro de dominios que tiene un costo de 9990 pesos anuales.

C.2.1.3 Costos de Software

Para el desarrollo de este proyecto se utilizará el gestor de base de datos MYSQL en su versión community que no tiene costo comercial, editor de texto Sublime Text que es gratuito, el lenguaje de servidor PHP que es libre en general todas las tecnologías a utilizar son gratuitas y no representan costos para la implementación.

C.2.1.4 Analista / Programador

El proyecto se desarrollara por un integrante que trabajara durante 3 días a la semana por 4 horas diarias durante 10 meses, comenzando desde marzo con un valor de \$2500 la hora hombre.

Tabla C. 5 Datos costo desarrollo

Horas Hombre (HH)	\$2500
Horas trabajadas por día (H-D)	4
Días trabajados a la semana (D-S)	3
Meses trabajados en el proyecto(M)	10
Semas al mes (S-M)	4

Sueldo del Analista Desarrollador = $2500 (HH) * 4 (H-D) * 3 (D-S) * 10 (M) * 4 (S-M)$
= 1.200.000 pesos.

Pero este proyecto no es remunerado por ende el costo total real del proyecto para el cliente serán el valor de adquirir un servicio de hosting más el valor de comprar un dominio lo que se traduce en dinero a $\$10990 + \$9990 = \$20980$.

C.2.2 Beneficios

Con la implementación del sistema se pueden estimar los siguientes beneficios, a partir de información entregada por el cliente

1. Disminuir montos de mora y gestiones de cobranza por atraso de gastos comunes, en los cuales se gastan \$6000 mensuales en concepto de llamadas, ahorrando un 50% en las llamadas.
2. Se tendrá un ahorro en impresiones del 50% debido a que gran parte de la información será enviada a los correos de los copropietarios, ahorrándose en este concepto un 50% aproximadamente \$6000 que se incurren mensualmente.
3. Se esperan ahorros en materiales de consumo interno por el control de este stock con un ahorro del 10% de un total mensual de 28.000 pesos.

A modo de analizar la factibilidad económica se utilizara el método V.A.N (Valor actual neto) para la evaluación de proyectos con una TRMA del 10% (Tasa de retorno mínima atractiva) y una evaluación de 1 año. La evaluación será anual pues el servicio de Hosting y Dominio se renuevan cada año.

Datos:

- Costo anual de Hosting: \$10.990.
- Costo anual dominio: \$9.990.
- Ahorro anual punto1 = $6.000 * 0,5 * 12$ meses = \$36.000 anuales.
- Ahorro anual punto 2= $6.000 * 0,5 * 12$ meses = \$36.000 anuales.
- Ahorro anual punto 3= $\$28.000 * 0,1 * 12$ meses = \$33.600 anuales.

$VAN = -Inversión\ inicial + Ingresos\ anuales\ (P/F,\ 10\%,\ 1)$.

$VAN = -20980 + (36.000 + 36000 + 33.600) * 0.9091 = 75.020$ pesos anuales.

Como el VAN del proyecto es mayor que cero es totalmente factible realizarlo.

C.3 Operativa

Del estudio de factibilidad se pueden mencionar los siguientes puntos:

- En general la comunidad tiene interés en contar con una herramienta que les facilite la gestión a la administración y los mantenga al tanto de las informaciones.
- En cuanto a la usabilidad las interfaces deberán estar bien adecuadas a los requerimientos de usuario.
- Para operar con el sistema se generara un manual de capacitación, describiendo paso a paso las operaciones con ejemplos y detalles.
- Los usuarios copropietarios no deberán tener amplios conocimientos computacionales para manejar el sistema, solo deberán saber utilizar algún navegador como Internet Explorer o Chrome y autenticarse en el sistema.

Anexo D: Casos de Uso Narrativos

D.1 Caso de uso Administrar Gastos Comunes

Tabla D. 1 Caso de uso Prorratear gasto común

Caso de uso	Prorratear gasto común.
Actores	Administrador.
Propósito	Realizar la división proporcional de los gastos de un período entre las unidades del condominio.
Descripción	Cada comunidad tiene gastos mensuales para la mantención por ejemplo de espacios comunes, personal etc. Estos gastos deben repartirse cada mes proporcionalmente.
Precondición	<ol style="list-style-type: none">1) El usuario debe estar autenticado con el perfil de “Administrador” y tener los privilegios para la operación.2) Debe existir el periodo (gasto común creado) creado y además encontrarse en estado “ABIERTO”.
Post condición	Los gastos comunes de un periodo se encuentran prorrateados entre las cuentas asociadas a las unidades del condominio.
Escenario principal	<ol style="list-style-type: none">1) El Actor Administrador ingresa en la opción gastos comunes.2) El sistema muestra el calendario de meses que se pueden prorratear.3) El Administrador selecciona un período (mes-año) y da curso al prorrateo.4) El sistema inicia los cálculos de gastos que le corresponden a cada unidad.
Escenario alternativo	<ol style="list-style-type: none">1) Si la sumatoria total de los factores de alícuota no es el 100% la operación se termina y se muestra un mensaje al usuario.

Tabla D. 2 Caso de uso Autenticar usuario

Caso de uso	Autenticar usuario.
Actores	Usuarios: (administrador, copropietario).
Propósito	Garantizar la seguridad en la aplicación.
Descripción	El Sistema verifica el nombre de usuario y contraseña. La contraseña deberá ser alfanumérica.
Precondición	El usuario mediante un navegador web ingresa a la aplicación y se dirige a la sección iniciar sesión. La clave deberá estar compuesta por una secuencia de caracteres de números y letras (alfanumérica) con un largo mínimo de 6 caracteres y uno máximo de 10.
Post condición	El usuario puede ingresar a la aplicación y a las secciones donde tiene privilegios.
Escenario principal	<ol style="list-style-type: none"> 1) El sistema presenta interfaz de inicio de sesión. 2) El usuario ingresa su nombre de usuario y contraseña. 3) El usuario envía los datos para su validación. 4) El sistema válida los datos recibidos. 5) El sistema entrega una respuesta al usuario (autenticación exitosa). 6) El usuario ingresa a la página principal.
Escenario alternativo	<ol style="list-style-type: none"> 1) En caso de que la autenticación sea errónea (usuario y contraseña inválidos) la aplicación solicitará nuevamente los datos. 2) El usuario sale de la aplicación hace clic en el botón cancelar.

Tabla D. 3 Caso de uso Crear calendario anual.

Caso de uso	Crear calendario anual.
Actores	Administrador.
Propósito	Generar el calendario anual, que incluye los 12 meses para un año.
Descripción	Los gastos comunes se calculan en periodos mensuales por lo tanto se deberá crear un calendario de 12 periodos para cada anualidad.
Precondición	El usuario debe estar autenticado y tener los privilegios.
Post condición	Se crea los 12 periodos para el año ingresado.
Escenario principal	<ol style="list-style-type: none"> 1) El Administrador ingresar al módulo Calendario. 2) El Administrador selecciona la opción “Crear anualidad”. 3) El sistema muestra el formulario. 4) El Administrador ingresar un año y selecciona la opción “Crear Anualidad”. 5) El Sistema valida los datos. 6) El Sistema muestra un mensaje de respuesta de la operación.
Escenario alternativo	<ol style="list-style-type: none"> 1) La anualidad ya existe en el sistema

Tabla D. 4 Caso de uso Adjuntar documento

Caso de uso	Adjuntar documento
Actores	Administrador
Propósito	Adjuntar un documento a un ítem de un gasto común como respaldo.
Descripción	Al ingresar los gastos comunes de un período el usuario “Administrador” puede adjuntar un documento como anexo al gasto generado.
Precondición	<ol style="list-style-type: none"> 1) El usuario debe estar autenticado y tener los privilegios. 2) El estado del gasto común debe ser “ABIERTO”.
Post condición	
Escenario principal	<ol style="list-style-type: none"> 1) El Administrador ingresa al módulo de Gastos Comunes. 2) El sistema muestra la interfaz del módulo. 3) El Administrador selecciona un año y luego un periodo. 4) El Administrador ingresar gasto y selecciona adjuntar documento. 5) El Sistema valido el documento. 6) El Sistema muestra al usuario un mensaje de respuesta de la operación. 7) El Administrador confirma el ingreso del gasto.
Escenario alternativo	<ol style="list-style-type: none"> 1) El documento no cumple con las especificaciones requeridas, tanto en peso como formato. 2) La conexión falla.

D.2 Caso de uso Gestionar Unidades

Tabla D. 5 Caso de uso Ingresar monto masivo unidades

Caso de uso	Ingresar monto masivo unidades.
Actores	Administrador.
Propósito	Cargar un monto masivo a las unidades seleccionadas.
Descripción	El actor “Administrador” puede requerir ingresar el mismo monto a distintas unidades o viviendas.
Precondición	El usuario debe estar autenticado y tener los privilegios.
Post condición	Se ingresa el monto a las cuentas asociadas a las unidades seleccionadas.
Escenario principal	<ol style="list-style-type: none">1) El Administrador ingresa al módulo de Viviendas.2) El Sistema muestra la interfaz del módulo.3) El Administrador ingresa a la opción “Ingresar monto masivo unidades”.4) El Sistema muestra el formulario.5) El Administrador ingresa un monto y selecciona las viviendas.6) El Administrador confirma la operación.7) El sistema valido la información.8) El sistema procede a ingresar el monto a las cuentas asociada a cada unidad seleccionada.
Escenario alternativo	<ol style="list-style-type: none">1) El selecciona la opción cancelar y cancela la operación.

Tabla D. 6 Caso de uso Crear unidad

Caso de uso	Crear unidad.
Actores	Administrador.
Propósito	Ingresar a un condominio una nueva unidad o vivienda.
Descripción	El Administrador podrá crear nuevas unidades o viviendas asociadas a un condominio.
Precondición	El usuario debe estar autenticado y tener los privilegios.
Post condición	Se crea una vivienda o unidad asociada al condominio.
Escenario principal	<ol style="list-style-type: none"> 1) El Administrador ingresa al módulo Unidades y luego a la opción Crear. 2) El Sistema muestra el formulario. 3) El Administrador completa el formulario con la información requerida. 4) El Administrador confirma la operación. 5) El sistema valida los datos. 6) El Sistema muestra al Administrador un mensaje de respuesta de la operación.
Escenario alternativo	<ol style="list-style-type: none"> 1) La vivienda ya existe, en tal caso se despliega un mensaje al usuario. 2) El usuario cancela la operación de ingreso.

D.3 Caso de uso narrativo Administrar Usuarios

Tabla D. 7 Caso de uso Registrar usuario

Caso de uso	Registrar usuario.
Actores	Administrador.
Propósito	Registrar los copropietarios y moradores, tanto copropietarios como arrendatarios, de un condominio.
Descripción	Al registrar un usuario este tendrán un perfil asociado para autenticar en el sistema y podrá ser vinculado con una vivienda o unidad.
Precondición	El usuario debe estar autenticado y tener los privilegios.
Post condición	Se crea un nuevo usuario en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1) El Administrador ingresa al módulo Usuarios opción Registrar. 2) El Sistema muestra el formulario de registro. 3) El Administrador completa la información requerida y acepta la operación. 4) El Sistema valida la información. 5) Sistema retorna un mensaje de resultado.
Escenario alternativo	<ol style="list-style-type: none"> 1) El nuevo usuario ya existe en el sistema 2) El usuario cancela la operación

Tabla D. 8 Caso de uso Gestionar vinculación a unidad

Caso de uso	Gestionar vinculación a unidad.
Actores	Administrador.
Propósito	Vincular un usuario con una vivienda o unidad.
Descripción	En el sistema los usuarios estarán asociados a las unidades de forma de saber quiénes habitan una vivienda.
Precondición	El usuario debe estar autenticado y tener los privilegios.
Post condición	El usuario es vinculado o desvinculado de una unidad.
Escenario principal	<ol style="list-style-type: none"> 1) El administrador ingresa al módulo Usuarios opción vinculación. 2) El sistema muestra el formulario. 3) El Administrador selecciona un usuario. 4) El sistema muestra el estado actual del usuario. 5) El Administrador selecciona una vivienda del condominio con la opción de vincular o lo contrario y acepta. 6) El sistema valida la información y retorna un mensaje al Administrador.
Escenario alternativo	N/A

Anexo E: Análisis y Diseño del sistema

El diagrama de secuencia sirve para modelar la interacción de un conjunto de objetos en una aplicación a través del tiempo.

E.1 Caso de uso Administrar Usuarios

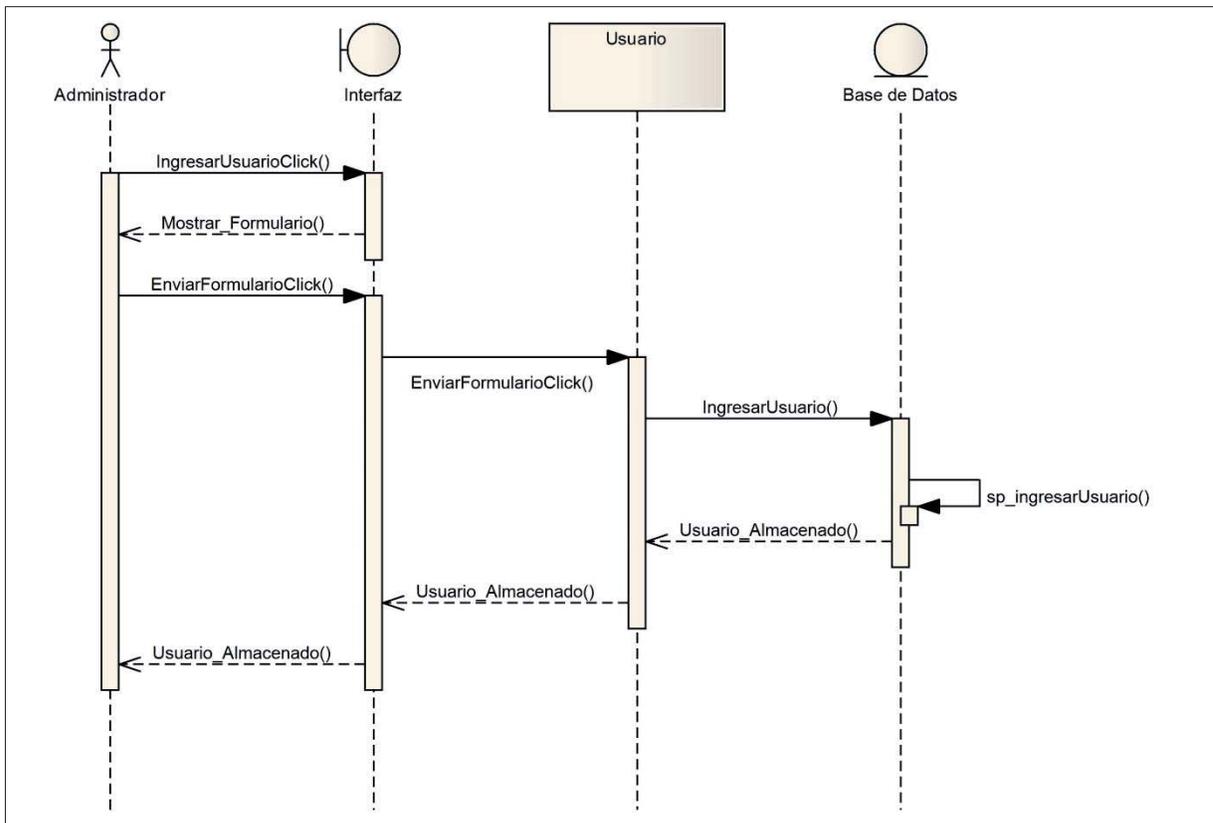


Figura E. 1 Diagrama de secuencia Ingresar Usuario

E.2 Caso de uso Gestionar Proveedores

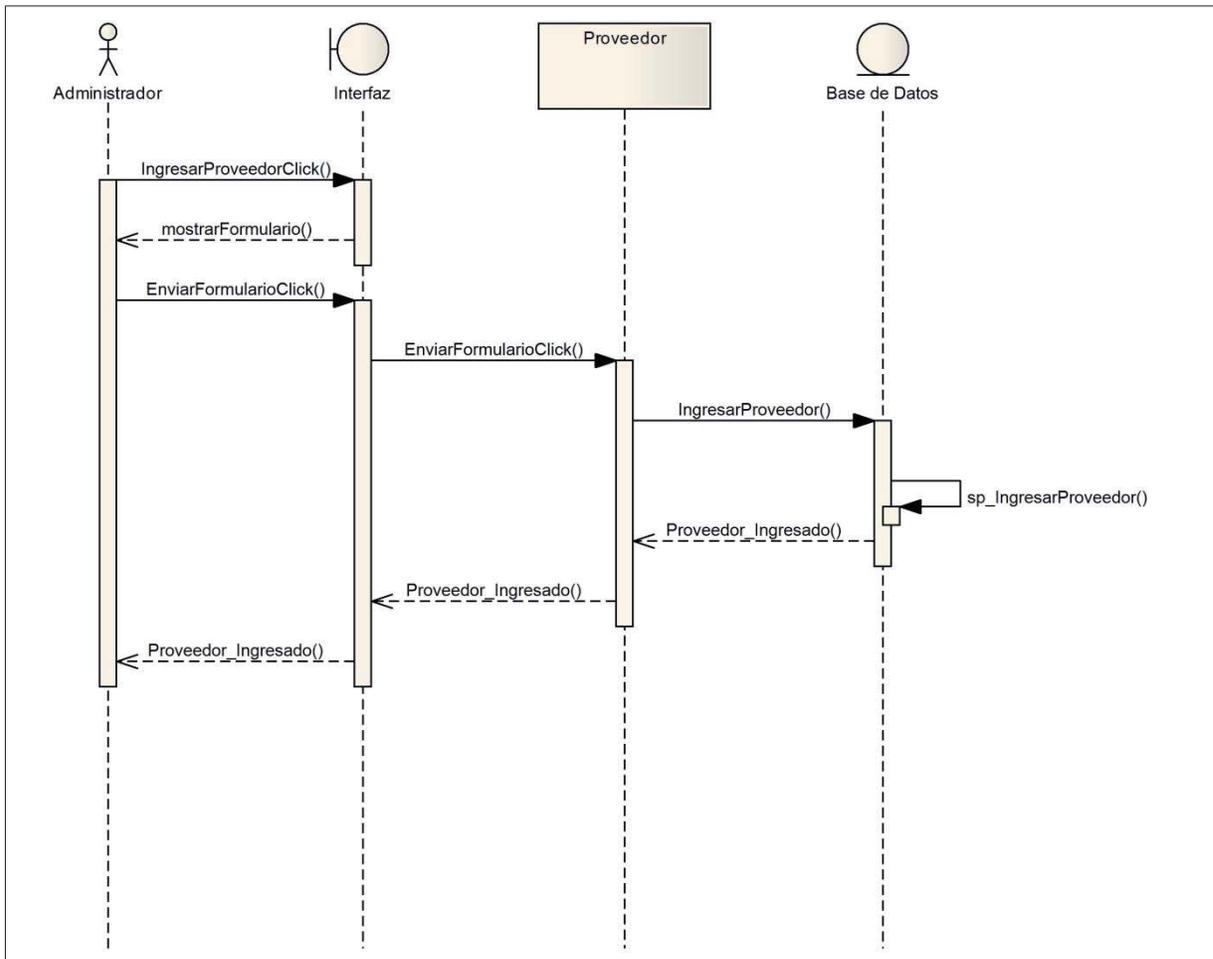


Figura E. 2 Diagrama de secuencia Agregar Proveedor

E.3 Caso de uso Gestionar Condominio

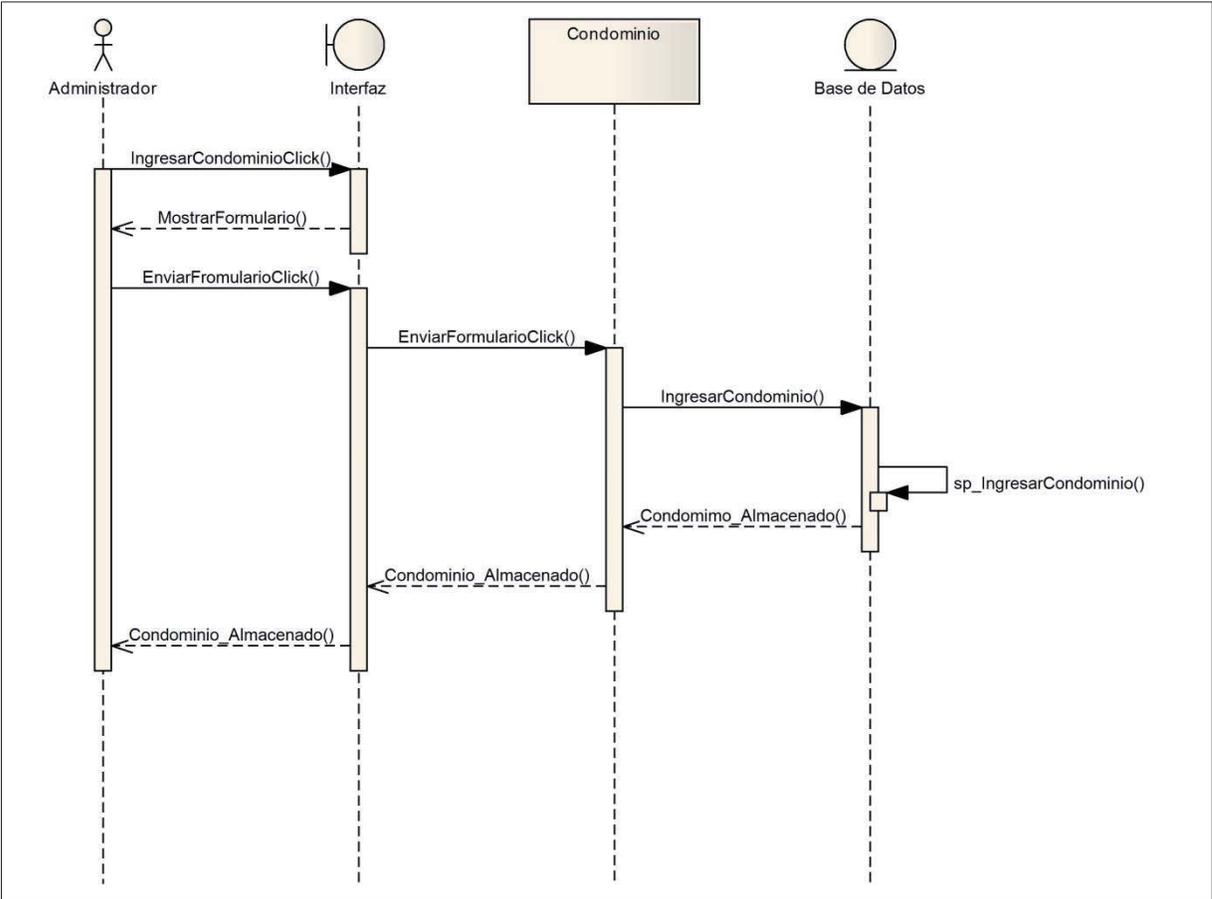


Figura E. 3 Diagrama de secuencia Ingresar Condominio

E.4 Caso de Uso Gestionar Perfil

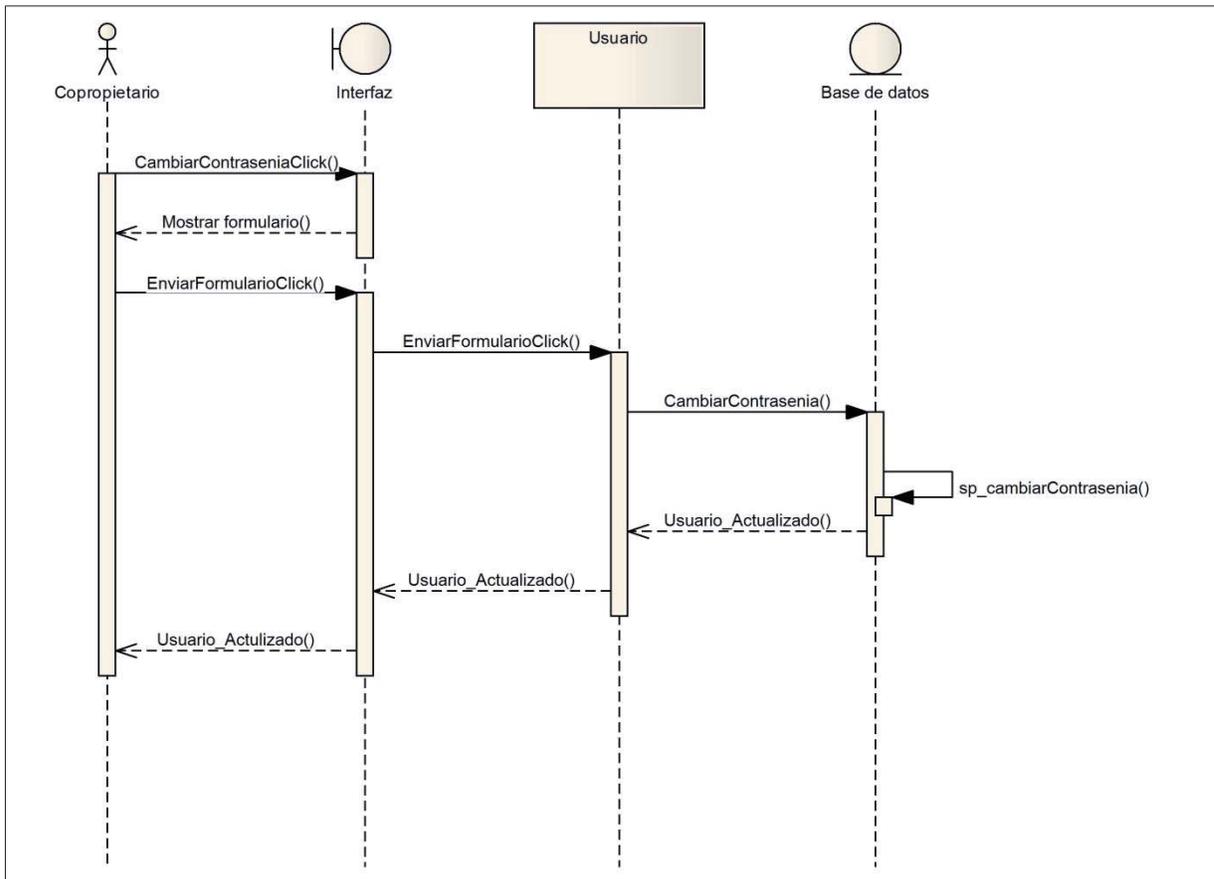


Figura E. 4 Diagrama de secuencia Modificar Contraseña

Anexo F: Modelo Lógico de base de datos

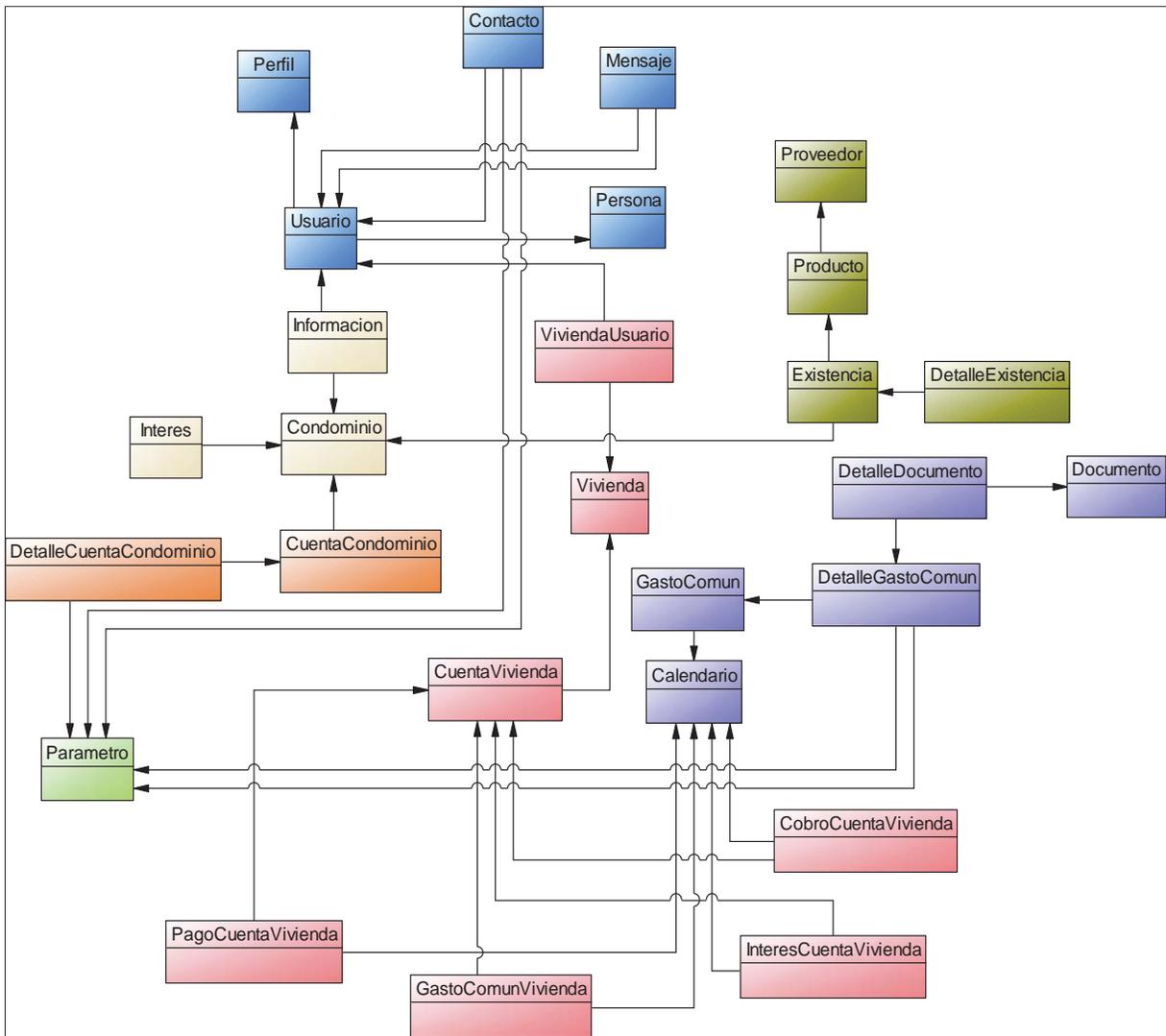


Figura F. 1 Modelo Relacional

Por lo amplio del diagrama se dividió en secciones las que se detallan y muestran en las siguientes figuras.

Tablas Sección 1:

- Usuario
- Persona
- Perfil
- Mensaje
- Contacto

Tablas Sección 2:

- Condominio
- Informacion
- Interes

Tablas Sección 3:

- CuentaCondominio
- DetalleCuentaCondominio

Tablas Sección 4:

- Calendario
- GastoComun
- DetalleGastoComun
- DetalleDocumento
- Documento

Tablas Sección 5:

- ViviendaUsuario
- Vivienda
- CuentaVivienda
- PagoCuentaVivienda
- GastoComunVivienda
- CobroCuentaVivienda
- InteresCuentaVivienda

Tablas Sección 6:

- Proveedor
- Producto
- Existencia
- DetalleExistencia

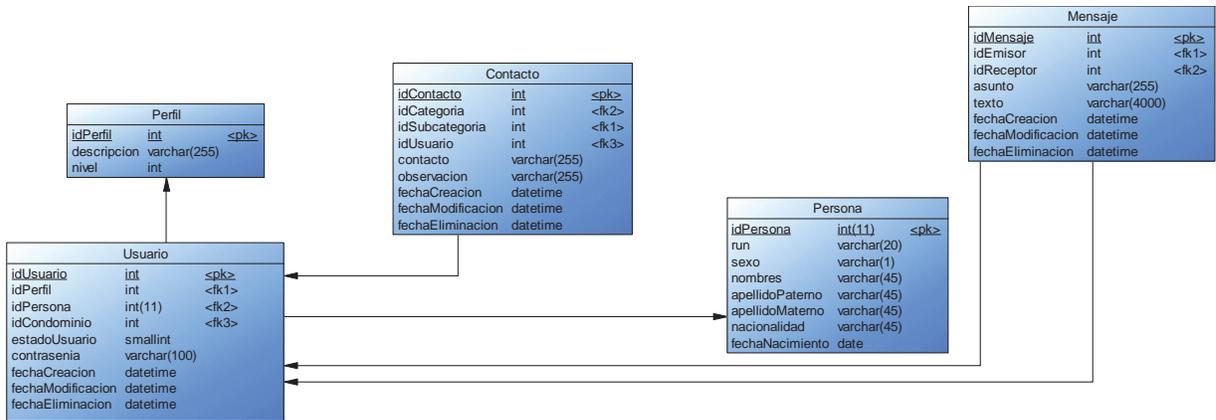


Figura F. 2 Tablas sección 1

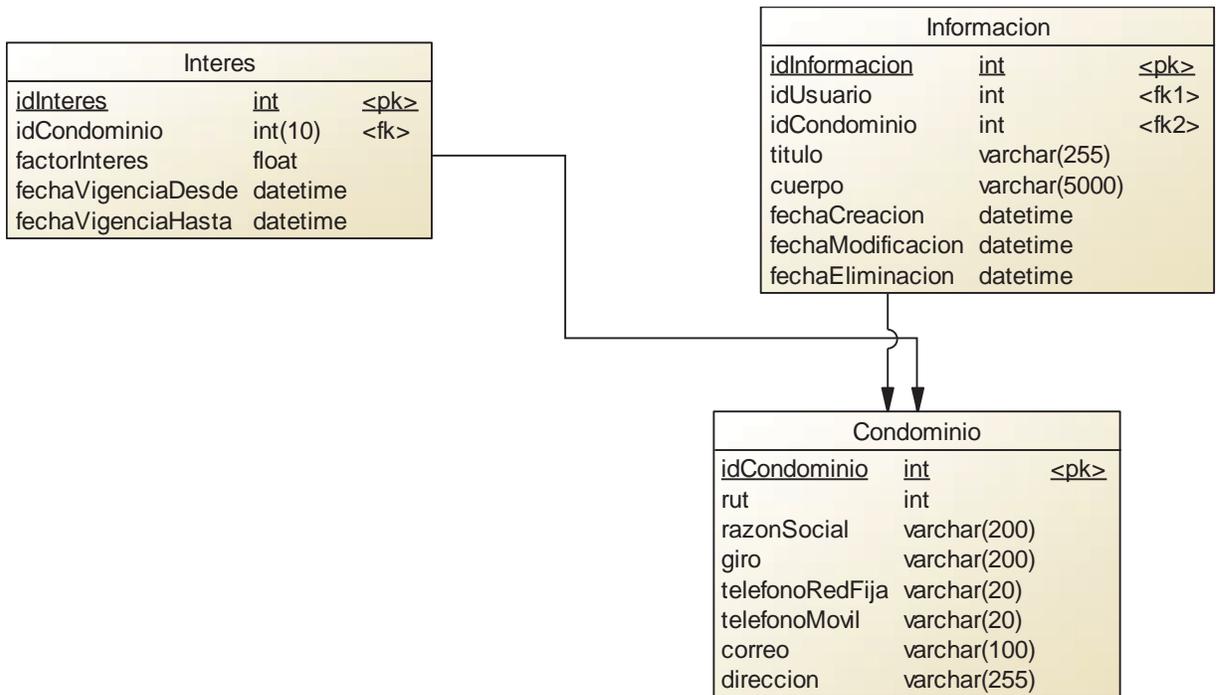


Figura F. 3 Tablas sección 2

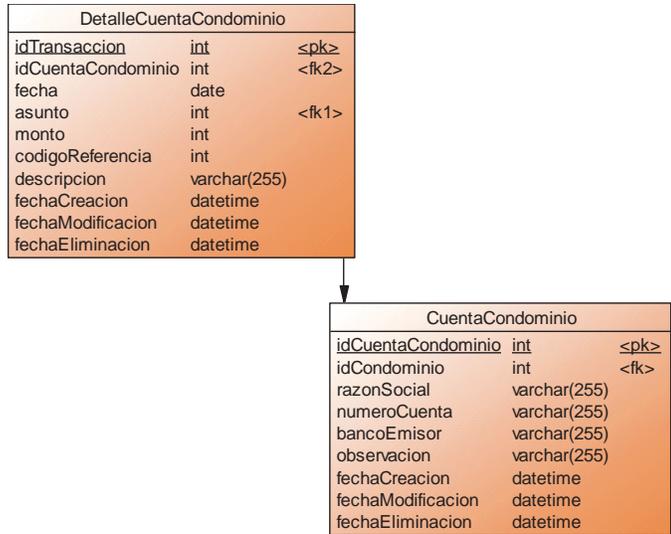


Figura F. 4 Tablas sección 3

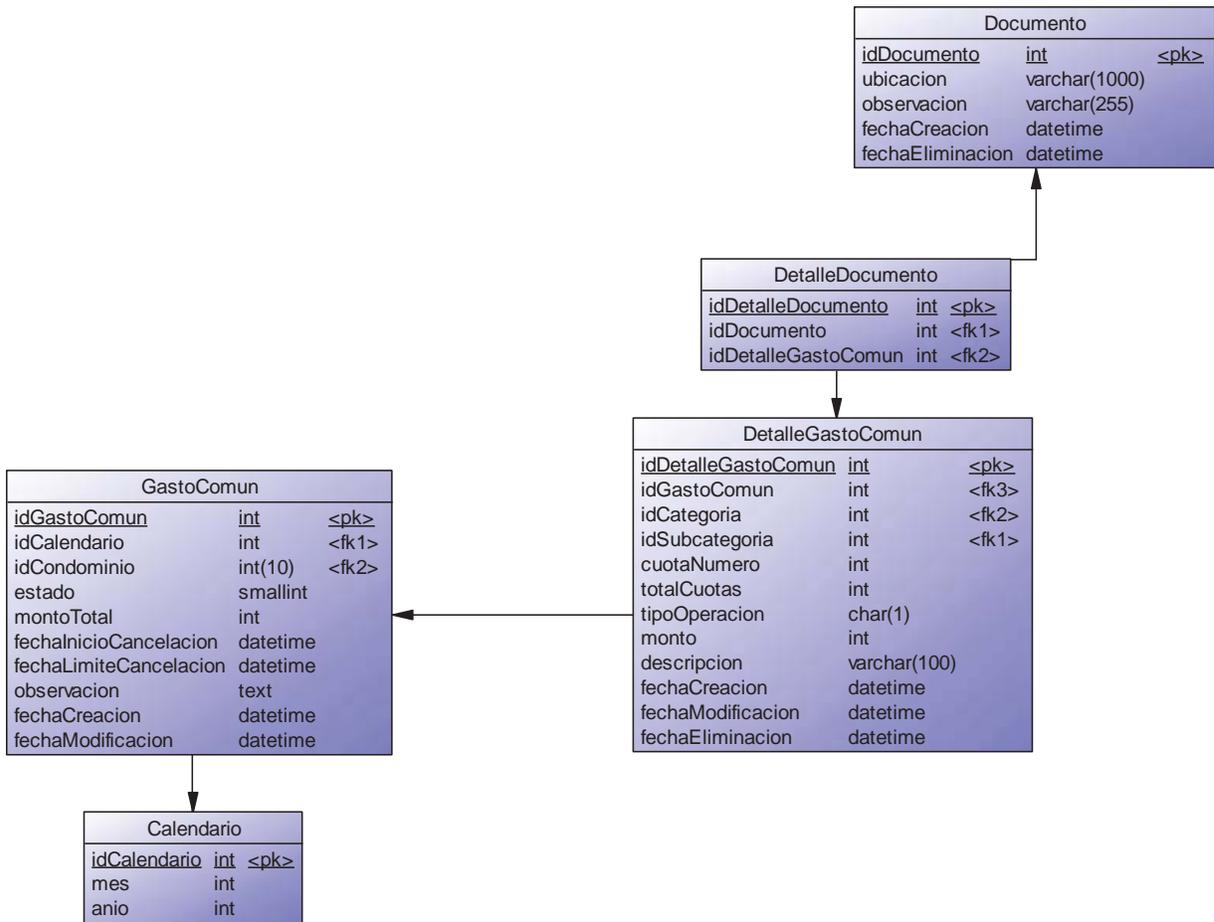


Figura F. 5 Tablas sección 4

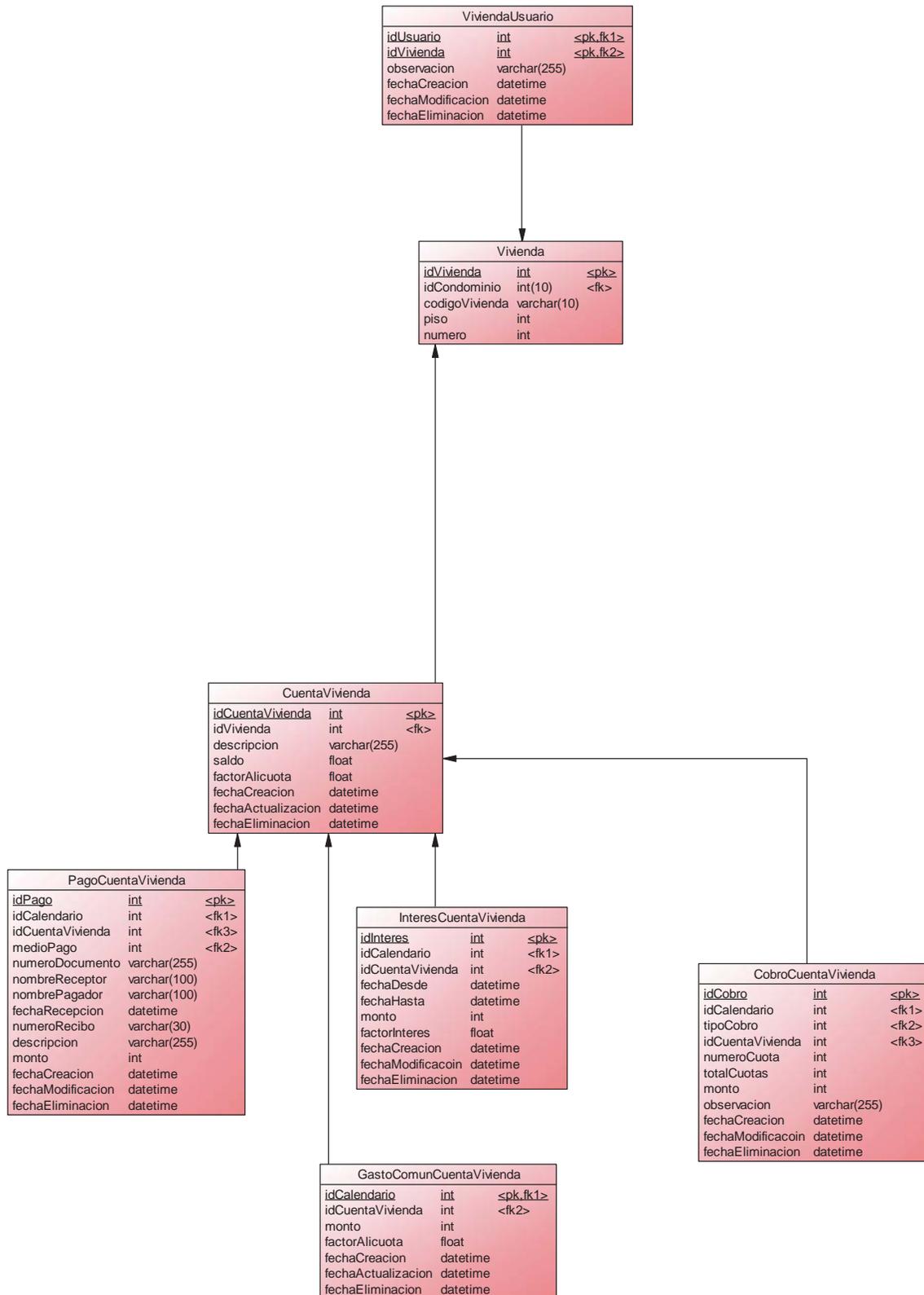


Figura F. 6 Tablas sección 5

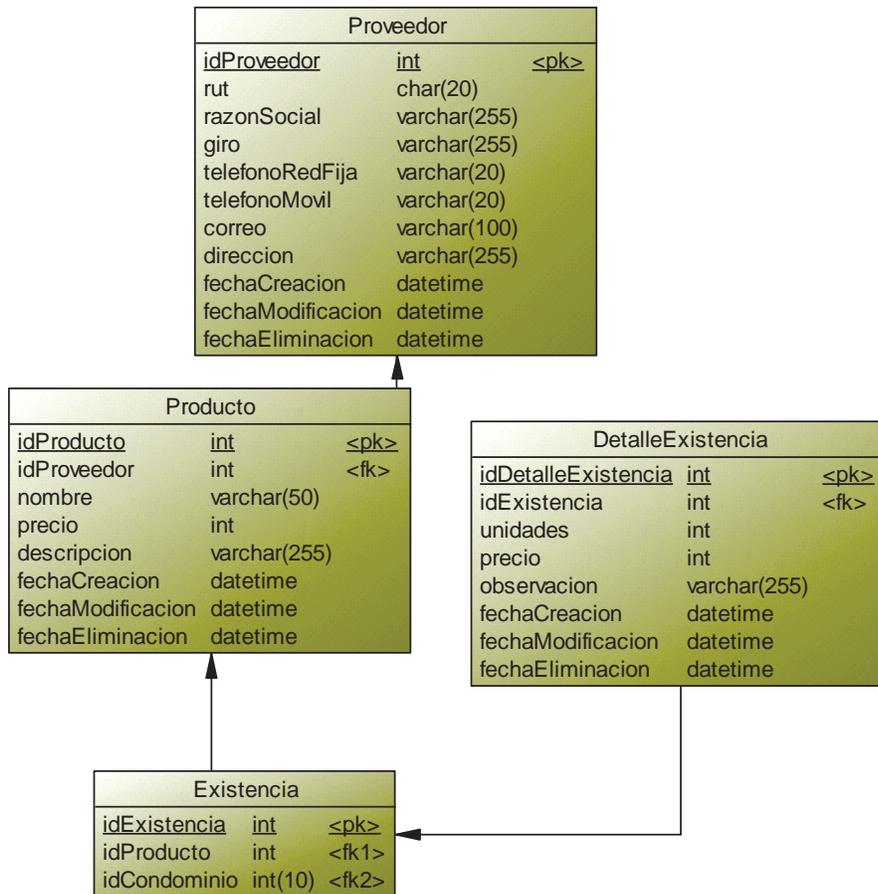


Figura F. 7 Tablas sección 6

Anexo G: Diseño de la interfaz

A continuación se muestran algunas interfaces del sistema que corresponden a la sesión del administrador.

G.1 Módulo de Interés

En la siguiente interfaz se visualiza el factor de interés que se aplica en el condominio. Cuando se crea un nuevo factor se elimina el anterior (modifica el estado) y queda como vigente el nuevo.



Condominio	Factor de interes	Fecha vigencia desde	Fecha vigencia hasta	Estado
Condominio Nitella	5.1 %	14-06-2017 23:02:33	14-06-2017 23:02:36	VIGENTE
Condominio Nitella	1.7 %	14-06-2017 23:02:16	14-06-2017 23:02:33	ELIMINADO

Figura G. 1 Interfaz del administrador: Listar factores de interés del condominio

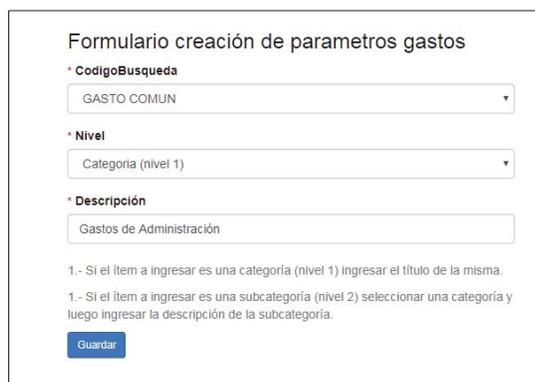
En la siguiente imagen de muestra el formulario para crear el factor de interés del condominio.



Figura G. 2 Interfaz del administrador: Creación de factor de interés del condominio

G.2 Módulo de Parámetros

En el siguiente formulario el administrador podrá crear las categorías y subcategorías de los gastos comunes.



Formulario creación de parametros gastos

* **CodigoBusqueda**

GASTO COMUN

* **Nivel**

Categoría (nivel 1)

* **Descripción**

Gastos de Administración

1.- Si el ítem a ingresar es una categoría (nivel 1) ingresar el título de la misma.
1.- Si el ítem a ingresar es una subcategoría (nivel 2) seleccionar una categoría y luego ingresar la descripción de la subcategoría.

Guardar

Figura G. 3 Interfaz del administrador: Creación de categorías en gastos comunes

Formulario creación de parametros gastos

* **CodigoBusqueda**

GASTO COMUN

* **Nivel**

Subcategoria (nivel 2)

* **Categoria**

Gastos de Administración

* **Descripción**

Personal de consejería y seguridad

1.- Si el ítem a ingresar es una categoría (nivel 1) ingresar el título de la misma.
 1.- Si el ítem a ingresar es una subcategoría (nivel 2) seleccionar una categoría y luego ingresar la descripción de la subcategoría.

Guardar

Figura G. 4 Interfaz del administrador: Crear de subcategorías en gastos comunes

GASTO COMUN

Buscar

Categoria	Subcategoria	Estado	Creación	Eliminación	
Gastos de Administración		ACTIVO	19-05-2017 18:12:37		Eliminar
Gastos de Administración	Personal de consejería y seguridad	ACTIVO	19-05-2017 18:13:31		Eliminar
Gastos de Administración	Honorarios de administración	ACTIVO	19-05-2017 19:13:50		Eliminar
Consumos básicos		ACTIVO	19-05-2017 18:14:44		Eliminar
Consumos básicos	Electricidad y espacios comunes	ACTIVO	19-05-2017 18:15:22		Eliminar
Gastos de mantención y reparación		ACTIVO	19-05-2017 18:15:46		Eliminar
Gastos de mantención y reparación	Mantención /Reparación de ascensores	ACTIVO	19-05-2017 19:13:27		Eliminar
Gastos Generales		ACTIVO	19-05-2017 19:11:45		Eliminar
Gastos Generales	Comisión Transbank	ACTIVO	19-05-2017 19:12:55		Eliminar

Figura G. 5 Interfaz del administrador: Listar parámetros gasto común

G.3 Módulo de Viviendas

A modo de ejemplo en la siguiente interfaz se crean 6 cabañas para un condominio, pero el número de unidades en el condominio Nitella corresponden a 27. Como se puede observar para este condominio las cabañas poseen el mismo factor de alícuota.

Dividir factor de alícuota proporcionalmente entre las viviendas		Crear vivienda			
Condominio	Vivienda	Piso	Número	Factor alícuota	
Condominio Nitella	Cabaña 1	1	1	16.6667 %	Moradores Gasto común Editar Eliminar
Condominio Nitella	Cabaña 2	1	2	16.6667 %	Moradores Gasto común Editar Eliminar
Condominio Nitella	Cabaña 3	1	3	16.6667 %	Moradores Gasto común Editar Eliminar
Condominio Nitella	Cabaña 4	1	4	16.6667 %	Moradores Gasto común Editar Eliminar
Condominio Nitella	Cabaña 5	1	5	16.6667 %	Moradores Gasto común Editar Eliminar
Condominio Nitella	Cabaña 6	1	6	16.6667 %	Moradores Gasto común Editar Eliminar
Sumatoria factores de alícuotas				100.0002 %	

Figura G. 6 Interfaz del administrador: Listar viviendas

G.4 Módulo de Gastos Comunes

En la siguiente imagen se muestra la interfaz que corresponde al módulo de gastos comunes en donde se ve un resumen de los gastos comunes de enero a marzo para un año seleccionado.

Año 2017		Consultar				
Mes	Año	Estado	Monto	Observación	Última actualización	
Enero	2017	ABIERTO	3,536,226		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Febrero	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Marzo	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Abril	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Mayo	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Junio	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Julio	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Agosto	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Septiembre	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Octubre	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Noviembre	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo
Diciembre	2017	ABIERTO	0		14-06-2017 23:40:38	Editar gastos Cerrar periodo

Figura G. 7 Interfaz del administrador: Listar gastos comunes

Al navegar por la opción “Editar gastos” el sistema lleva a la siguiente pantalla en donde se pueden agregar, eliminar o editar los gastos del período.

ENERO - 2017 [Volver](#)

* **Categoría**
 Seleccione un categoría

* **Subcategoría**
 Seleccione un subcategoría

Mostrar registros eliminados

* **Monte**
 Ingresar un monto

* **Descripción**
 Seleccione un archivo Ningún archivo seleccionado

* **Documento**
 Seleccione un archivo

* **Cuota número**
 1

* **Número total de cuotas**
 1

[Guardar](#) [Consultar](#)

GASTO COMÚN Enero - 2017 **ESTADO** ABIERTO

GASTOS DE ADMINISTRACIÓN

PERSONAL DE CONSERUERÍA Y SEGURIDAD

2 conserjes Full time	785,000	Detalle Editar Eliminar
-----------------------	---------	---

HONORARIOS DE ADMINISTRACIÓN

Gastos de administración (factura 449)	1,051,226	Detalle Editar Eliminar
TOTAL GASTOS DE ADMINISTRACIÓN	1,836,226	

CONSUMOS BÁSICOS

ELECTRICIDAD Y ESPACIOS COMUNES

Factura 16335254	1,200,000	Detalle Editar Eliminar
TOTAL CONSUMOS BÁSICOS	1,200,000	

GASTOS DE MANTENCIÓN Y REPARACIÓN

MANTENCIÓN /REPARACIÓN DE ASCENSORES

Factura 3543246	500,000	Detalle Editar Eliminar
TOTAL GASTOS DE MANTENCIÓN Y REPARACIÓN	500,000	
TOTAL DEL MES	3,536,226	

Figura G. 8 Interfaz del administrador: Ver y editar gastos comunes de un período

G.5 Módulo de Usuarios

Formulario Crear Usuario

* **run**
 ejemplo: 12345678-9

* **nombres**

* **Apellido Paterno**

* **Apellido Materno**

* **Sexo**
 Seleccionar

* **nacionalidad**

* **Fecha de nacimiento**
 dd/mm/yyyy

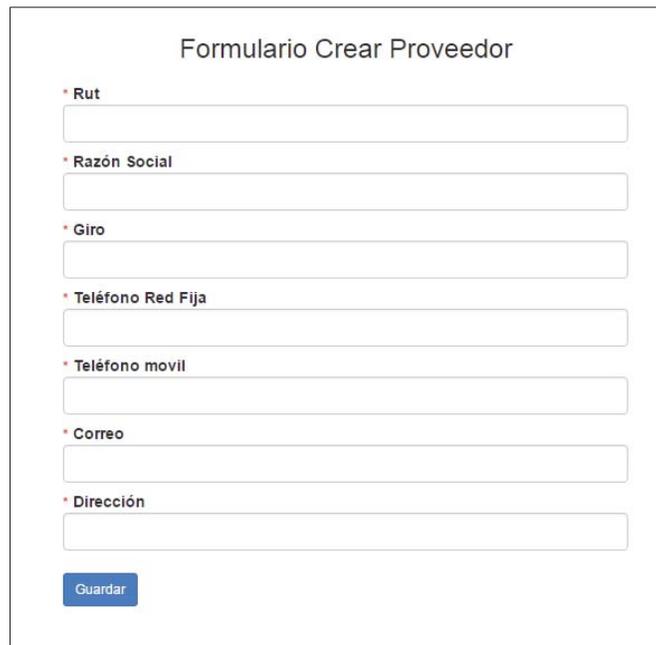
* **Contraseña**
 Ingresar una contraseña entre 4 y 8 caracteres

* **Perfil**
 Seleccionar

[Guardar](#)

Figura G. 9 Interfaz del administrador: Formulario Crear Usuario

G.6 Módulo de Proveedores



Formulario Crear Proveedor

* Rut

* Razón Social

* Giro

* Teléfono Red Fija

* Teléfono movil

* Correo

* Dirección

Guardar

Figura G. 10 Interfaz del administrador: Formulario Crear Proveedor

G.7 Módulo de Existencias



Formulario Crear Existencia

Proveedor

Seleccionar Proveedor ▼

* Producto

Seleccionar Producto ▼

* Precio

Ingresar precio del producto

* Observación

* Unidades

Cantidad de unidades ▼

Guardar

Figura G. 11 Interfaz del administrador: Formulario Crear Existencia

G.8 Módulo de Informaciones

Formulario Crear de información

* **Título**

Ingresar título

* **Cuerpo**

Ingrese la información

Guardar

Figura G. 12 Interfaz del administrador: Crear Información

G.9 Módulo Calendario

* **Calendario** 2017 Consultar Crear anualidad

Mes	Año
Enero	2017
Febrero	2017
Marzo	2017
Abril	2017
Mayo	2017
Junio	2017
Julio	2017
Agosto	2017
Septiembre	2017
Octubre	2017
Noviembre	2017
Diciembre	2017

Figura G. 13 Interfaz del administrador: Consultar y crear calendario

G.10 Módulo de Cuentas

The screenshot shows a web form titled "Cancelación de gastos Vivienda". It contains the following fields and controls:

- Unidad:** A dropdown menu.
- Periodo:** A dropdown menu.
- Nombre Receptor:** A text input field.
- Nombre Pagador:** A text input field.
- Medio de Pago:** A dropdown menu.
- Fecha Recepción:** A text input field.
- N° Documento:** A text input field.
- Monto:** A text input field.
- N° Recibo:** A text input field.
- Observacion:** A text input field.
- Guardar:** A blue button at the bottom left.

Figura G. 14 Interfaz del administrador: Cancelación de Gastos Vivienda

Las siguientes interfaces corresponden a la sesión de copropietarios.

G.11 Perfil Modificar Contraseña

The screenshot shows a web form titled "Modificar contraseña". It contains the following fields and controls:

- Actual Contraseña:** A text input field.
- Nueva Contraseña:** A text input field.
- Confirmar Contraseña:** A text input field.
- CAMBIAR CONTRASEÑA:** A blue button at the bottom center.

Figura G. 15 Interfaz del copropietario: Modificar contraseña

G.12 Gastos Comunidad

El copropietario en la sección Gastos Comunidad visualizara los gastos para un periodo, previa selección de un mes y año, podrá descargar la documentación adjunta.

Mes	Enero	Año	2017	Consultar
GASTO COMÚN		Enero - 2017	ESTADO	ABIERTO
GASTOS DE ADMINISTRACIÓN				
PERSONAL DE CONSERJERÍA Y SEGURIDAD				
2 conserjes Full time		785,000	 Ver Archivo	
HONORARIOS DE ADMINISTRACIÓN				
Gastos de administración (factura 449)		1,051,226	 Ver Archivo	
TOTAL GASTOS DE ADMINISTRACIÓN		1,836,226		
CONSUMOS BÁSICOS				
ELECTRICIDAD Y ESPACIOS COMUNES				
Factura 16335254		1,200,000	 Ver Archivo	
TOTAL CONSUMOS BÁSICOS		1,200,000		
GASTOS DE MANTENCIÓN Y REPARACIÓN				
MANTENCIÓN /REPARACIÓN DE ASCENSORES				
Factura 3543246		500,000	 Ver Archivo	
TOTAL GASTOS DE MANTENCIÓN Y REPARACIÓN		500,000		
TOTAL DEL MES		3,536,226		
OBSERVACIÓN				
Sin observaciones				

Figura G. 16 Interfaz del copropietario: Gastos comunidad

G.13 Mis Gastos Comunes

En esta interfaz el copropietario podrá ver los gastos comunes que corresponden a su vivienda para un período seleccionado, visualizando el detalle de cobros y el monto total a cancelar.

Vivienda Cabaña 1
Periodo Enero - 2017
Consultar

Vivienda	Factor Alicuota	Última actualización
Cabaña 1	3,7037 %	15-06-2017 13:03:49

Gasto Total Comunidad

Periodo	Estado	Monto	
Enero - 2017	ABIERTO	\$ 3.536.226	Q

Prorrateo Vivienda

Factor alicuota	Monto prorrateo
3,7037 %	\$ 130.971

Pendiente

Gasto Común anterior	Factor interés	Monto	
\$ 25.000	5,0000 %	\$ 1.250	Q

Otros cobros

Descripción	Fecha	N° Cuota	Monto	
Fondo de Reserva	05-01-17	1/1	\$2500	Q
Pintura Reja	05-01-17	1/10	\$7500	Q

Subtotal	\$ 142.221
Saldo Anterior	\$ 25.000
Total a pagar	\$ 167.221

Figura G. 17 Interfaz del copropietario: Mis gastos comunes

Anexo H: Casos de Prueba

Las pruebas de caja negra, es una técnica de pruebas de software en la cual la funcionalidad se verifica sin tomar en cuenta la estructura interna de código, detalles de implementación o escenarios de ejecución internos en el software. En la siguiente figura se grafica el proceso.

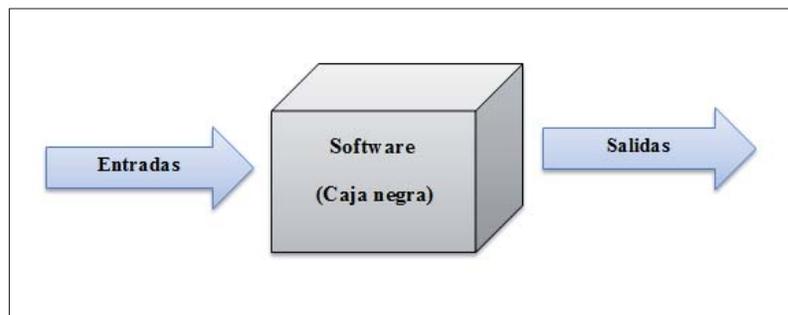


Figura H. 1 Proceso gráfico de pruebas de caja negra

Las pruebas de caja negra a aplicar en este proyecto son las siguientes:

Tabla H. 1 Clase de equivalencia para el largo de la contraseña

Condición de entrada	Clase Válida	Clave Inválida
Largo de cadena de caracteres en la contraseña entre 6 y 10.	$6 \leq \text{Largo} \leq 10$	Largo < 6 Largo > 10

Caso de prueba: 1

Funcionalidad: Autenticar Usuario

Descripción: El sistema funcionara solo con usuarios registrados y activos en el sistema.

Objetivo del caso de prueba: Comprobar que no deja entrar a un usuario inexistente y/o con clave con formato incorrecto. La clave deberá estar compuesta por una secuencia de caracteres con un largo mínimo de 6 caracteres y uno máximo de 10 como se muestra en la tabla H.1.

Caso 1.1:

- Datos de entrada:
Usuario válido: 11.111.111-1
Contraseña (cadena 13 caracteres): “Ingreso1Alsis”
- Resultado esperado: El sistema debe rechazar la contraseña y resetear el campo en el formulario. Además debe alertar al usuario con un mensaje.
- Resultado obtenido: El sistema responde acorde a lo esperado. Contraseña mayor a 10 caracteres.

Caso 1.2:

- Datos de entrada:
Usuario válido: 11.111.111-1
Contraseña largo válido pero incorrecta: “Ingreso1A”
- Resultado esperado: El sistema debe desplegar un mensaje indicando que la contraseña y/o usuario es incorrecto y resetear el campo que corresponde a la contraseña.
- Resultado obtenido: El sistema responde acorde a lo esperado.

Caso 1.3:

- Datos de entrada:
Usuario válido: 11.111.111-1
Contraseña correcta: “abc123Fa”
- Resultado esperado: El sistema debe autenticar al usuario y redirigir hacia el menú inicial según el perfil de usuario.
- Resultado obtenido: El sistema responde acorde a lo esperado.

Caso de prueba: 2

Funcionalidad: Exportar informe gasto mensual unidad

Descripción: Exportar el detalle del gasto común que corresponde a un periodo para una vivienda en la sesión de copropietario.

Objetivo del caso de prueba: Comprobar que se genere correctamente el informe para cada período y en formato PDF.

Condiciones de ejecución: El estado del gasto común a consultar (año-mes) deberá estar en “CERRADO”.

Caso 2.1:

- Procedimiento:
 - 1.- Ingresar al sistema como copropietario.
 - 2.- Seleccionar en el menú la opción mis gastos comunes.
 - 3.- Seleccionar un período y presionar buscar en el formulario.
 - 4.- Luego seleccionar la opción exportar.
- Resultado esperado: El sistema debe generar un informe en formato PDF el que se debe descargar desde el navegador.
- Resultado obtenido: El sistema responde acorde a lo esperado.

Caso de prueba: 3

Funcionalidad: Exportar informe gasto mensual comunidad

Descripción: Exportar el detalle del gasto común que corresponde a un periodo de la comunidad en la sesión de copropietario.

Objetivo del caso de prueba: Comprobar que se genere correctamente el informe para cada período y en formato PDF.

Condiciones de ejecución: El estado del gasto común a consultar (año-mes) deberá estar en “CERRADO”.

Caso 2.1:

- Procedimiento:
 - 1.- Ingresar al sistema como copropietario.
 - 2.- Seleccionar en el menú la opción gasto común comunidad.
 - 3.- Seleccionar un período y presionar buscar en el formulario.
 - 4.- Luego seleccionar la opción exportar.
- Resultado esperado: El sistema debe generar un informe en formato PDF el que se debe descargar desde el navegador.
- Resultado obtenido: El sistema responde acorde a lo esperado.