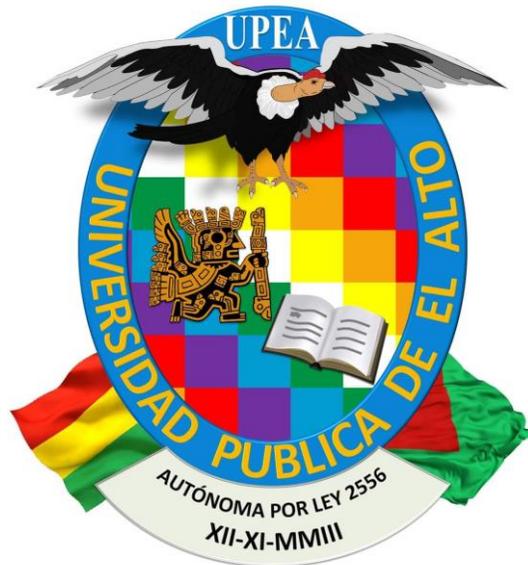


UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA DE GESTIÓN INMOBILIARIA” CASO: JALHISA LTDA.

Para Optar al Título de Licenciatura en Ingeniería de Sistemas
MENCIÓN: INFORMÁTICA Y COMUNICACIONES

Postulante: Omar Callejas Gutierrez
Tutor Metodológico: Ing. Marisol Arguedas Balladares
Tutor Especialista: MSc. Wendy Yomar Sarmiento Martínez
Tutor Revisor: Ing. Yuly Ramirez Limachi

EL ALTO – BOLIVIA

2022

Dedicatoria:

*El presente proyecto va dedicado a mis amados padres por estar siempre a mi lado,
dándome su apoyo incondicional y a mis queridas hermanas y hermanos.*

Agradecimientos

Primeramente agradecer a Dios por darme la vida y oportunidad para estar en este momento y estar rodeado de personas maravillosas e importantes que se encuentran a mi lado.

A mi mamá Julia Gutierrez Duran, agradecer su infinito amor, paciencia, dedicación, esmero y apoyo en todos los momentos de vida. Entregado su sabiduría y consejos.

A la Ing. Marisol Arguedas Balladares, por su gran calidad profesional y experiencia me acompañó en el desarrollo del presente proyecto de grado.

A la Lic. Wendy Yomar Sarmiento Martínez mi tutora especialista, quien con su gran profesionalismo realizo el seguimiento a este trabajo de grado, por dedicar su valioso tiempo, orientación, paciencia y colaboración.

A la Ing. Yuly Ramirez Limachi, por su valioso tiempo, colaboración, orientación y dedicación en el desarrollo del presente proyecto de grado.

Un sincero agradecimiento a la inmobiliaria JALHISA LTDA. en especial al gerente general Juan Carlos Laura Caljaro, por brindar su colaboración a lo largo de estos meses.

ÍNDICE GENERAL

	Pág.
CAPITULO I.....	1
MARCO PRELIMINAR.....	1
1.1. INTRODUCCIÓN.....	1
1.2. ANTECEDENTES DE LA INVESTIGACIÓN.....	2
1.2.1. Antecedentes Institucionales.....	2
1.2.2. Antecedentes de trabajos afines.....	3
1.3. PLANTEAMIENTO DEL PROBLEMA.....	4
1.3.1. Problema Principal.....	4
1.3.2. Problemas Secundarios.....	5
1.4. OBJETIVOS.....	6
1.4.1. Objetivo General.....	6
1.4.2. Objetivos Específicos.....	6
1.5. JUSTIFICACIÓN.....	7
1.5.1. Justificación Técnica.....	7
1.5.2. Justificación Económica.....	7
1.5.3. Justificación Social.....	8
1.6. METODOLOGÍA.....	8
1.6.1. UWE.....	8
1.6.2. Métrica de calidad.....	9
1.6.3. Estimación de costos - COSMIC.....	9
1.6.4. Seguridad del software ISO - 27001.....	9
1.6.5. Pruebas de software.....	10
1.7. MÉTODO.....	10
1.7.1. Modelo iterativo e incremental.....	10
1.8. HERRAMIENTAS.....	11
1.9. LÍMITES Y ALCANCES.....	13
1.9.1. Limites.....	13
1.9.2. Alcances.....	13
1.10. APORTES.....	14
1.10.1. Teórico.....	14

1.10.2. Práctico	15
CAPITULO II.....	16
MARCO TEÓRICO	16
2.1. SISTEMA.....	16
2.1.1. Sistema de información	16
2.2. GESTIÓN	17
2.3. INMOBILIARIA	17
2.4. INGENIERÍA DE REQUERIMIENTOS.....	18
2.4.1. Requerimientos funcionales	18
2.4.2. Requerimientos no funcionales	18
2.5. MODELO ITERATIVO E INCREMENTAL.....	19
2.5.1. Modelo incremental	19
2.5.2. Modelo iterativo	20
2.6. METODOLOGÍA UWE.....	21
2.6.1. Captura, análisis y especificación de requisitos.....	21
2.6.2. Diseño.....	22
2.6.3. Codificación.....	24
2.6.4. Pruebas.....	24
2.6.5. Instalación o Fase de Implementación.....	25
2.6.6. El Mantenimiento.....	25
2.7. UML.....	25
2.8. MÉTRICAS DE CALIDAD ISO-9126	25
2.8.1. Calidad externa e interna.....	26
2.8.2. Calidad en uso	29
2.9. SEGURIDAD del software iso - 27001	30
2.10. PRUEBAS DE FUNCIONAMIENTO	31
2.10.1. Prueba de caja blanca.....	31
2.10.2. Prueba de caja negra	35
2.11. ESTIMACIÓN DE COSTO CON COSMIC	37
2.11.1. Medición del software.....	38
2.11.2. Fase de estrategia de medición.....	38
2.11.3. Fase de mapeo	39
2.11.4. Fase de medición	40

2.11.5. Formula.....	41
2.12. SERVICIOS WEB.....	42
2.13. ARQUITECTURA DE SOFTWARE.....	42
2.13.1. Arquitectura API REST.....	43
2.13.2. Recursos URI's.....	44
2.13.3. Los métodos HTTP.....	45
2.13.4. Códigos de estados.....	46
2.13.5. Componentes para webapps.....	46
2.13.6. Patrón de Gestión de Estado.....	47
2.13.7. Arquitectura SPA.....	48
2.14. JWT.....	49
2.14.1. Estructura de JSON Web Token.....	49
2.15. HERRAMIENTAS.....	51
2.15.1. Lenguaje de programación.....	51
2.15.2. Base de datos.....	52
2.15.3. Framework.....	52
2.15.4. HTML.....	54
2.15.5. Css.....	55
CAPITULO III.....	56
MARCO APLICATIVO.....	56
3.1. APLICACIÓN DE LA METODOLOGÍA UWE.....	57
3.1.1. Captura, análisis y especificación de requisitos.....	57
3.1.2. Diseño del sistema.....	65
3.1.3. CODIFICACIÓN.....	101
3.1.4. IMPLEMENTACIÓN DEL SISTEMA.....	103
3.2. CALIDAD DEL SOFTWARE ISO-9126.....	103
3.2.1. Funcionalidad.....	103
3.3. SEGURIDAD DEL SOFTWARE ISO – 27001.....	109
3.4. MEDICIÓN Y ESTIMACIÓN DEL SOFTWARE MÉTODO COSMIC.....	110
CAPITULO IV.....	112
CONCLUSIÓN Y RECOMENDACIÓN.....	112
4.1. CONCLUSIÓN.....	112
4.2. RECOMENDACIONES.....	113

BIBLIOGRAFÍA.....	114
ANEXOS.....	118
ANEXO A: ORGANIGRAMA.....	119
ANEXO B: ÁRBOL DE PROBLEMAS.....	120
ANEXO C. ÁRBOL DE OBJETIVOS.....	121
ANEXO D. ENTREVISTA	122
ANEXO E. ENCUESTAS.....	124
ANEXO F. DOCUMENTOS DE LA EMPRESA.....	125
MANUAL.....	1
MANUAL DE USUARIO.....	1

ÍNDICE DE TABLAS

	Pág.
Tabla 1 Software.....	11
Tabla 2 Hardware	12
Tabla 3 Técnicas de investigación	12
Tabla 4 Modelado del negocio	12
Tabla 5 Valores de ajuste de complejidad.....	26
Tabla 6 JSON Web Token	51
Tabla 7 Obtención de Requerimientos.....	57
Tabla 8 Descripción de Actores	59
Tabla 9 Requerimientos Funcionales.....	61
Tabla 10 Requerimientos no Funcionales	64
Tabla 11 Descripción de actores de casos de uso	65
Tabla 12 Caso de uso, acceder al sistema	70
Tabla 13 Caso de uso, Gestionar empleados	72
Tabla 14 Caso de uso, Gestionar clientes.....	73
Tabla 15 Caso de uso, Gestionar urbanizaciones.....	74
Tabla 16 Caso de uso, Gestionar lote, casa y edificio.....	75
Tabla 17 Caso de uso, Seguimiento de asesores comerciales	76
Tabla 18 Caso de uso, Gestionar venta de lote urbanizado	77
Tabla 19 Caso de uso, Seguimiento de venta de inmuebles.....	78
Tabla 20 Caso de uso, Gestionar venta de lotes, casas y edificios al contado.....	79
Tabla 21 Caso de uso, Gestionar agenda de citas.....	80
Tabla 22 Gestionar servicio de publicidad de la empresa	81
Tabla 23 Caso de uso, Gestionar servicio de publicidad de venta, alquiler y anticrético de inmuebles	82
Tabla 24 Calculo de la cuenta total con factores de ponderación	104
Tabla 25 Calculo de los valores de ajuste de complejidad	104
Tabla 26 Encuesta sobre la usabilidad	108

ÍNDICE DE FIGURAS

	Pág.
Figura 1 Modelo incremental.....	19
Figura 2 Modelo iterativo	20
Figura 3 Identificación de actor	21
Figura 4 Requerimientos funcionales.....	22
Figura 5 Requerimientos no funcionales.....	22
Figura 6 Descripción de actores de caso de uso.....	23
Figura 7 Diagrama de casos de uso	23
Figura 8 Diagrama de clases	23
Figura 9 Modelo de navegación.....	24
Figura 10 Prueba de caja blanca	32
Figura 11 Diagrama de flujo.....	33
Figura 12 Grafo de flujo	33
Figura 13 Matriz de grafo.....	35
Figura 14 Prueba de caja negra.....	35
Figura 15 Notación de gráfico	36
Figura 16 Visión general del COSMIC	37
Figura 17 El proceso de medición COSMIC.....	38
Figura 18 Los cuatro tipos de movimientos de datos	40
Figura 19 Interacción con la API REST.....	44
Figura 20 Estructura de los URI.....	45
Figura 21 Patrón de gestión de estado	47
Figura 22 Patrón de gestión de estado de Vuex	48
Figura 23 El ciclo de vida de una SPA.....	49
Figura 24 Situación actual de la empresa	56
Figura 25 Diagrama de caso de uso general del sistema - A	68
Figura 26 Diagrama de caso de uso general del sistema - B	69
Figura 27 Diagrama de caso de uso, Acceder al sistema	70
Figura 28 Diagrama de caso de uso, Gestionar empleados.....	71
Figura 29 Diagrama de caso de uso, Gestionar clientes	72

Figura 30 Diagrama de caso de uso, Gestionar urbanizaciones	73
Figura 31 Diagrama de caso de uso, Gestionar lote, casa y edificio	74
Figura 32 Diagrama de caso de uso, Seguimiento de asesores comerciales.....	75
Figura 33 Diagrama de caso de uso, Gestionar venta de lote urbanizado	76
Figura 34 Diagrama de caso de uso, Seguimiento de venta de inmuebles	77
Figura 35 Diagrama de caso de uso, Gestionar venta de lotes, casas y edificios al contado	78
Figura 36 Diagrama de uso, Gestionar agenda de citas.	79
Figura 37 Diagrama de caso de uso, Gestionar servicio de publicidad de la empresa	80
Figura 38 Diagrama de caso de uso, Gestionar servicio de publicidad de venta, alquiler y anticrético de inmuebles para los clientes.....	81
Figura 39 Diagrama de clases	83
Figura 40 Modelo de navegación - Gerente general, A	84
Figura 41 Modelo de navegación - Gerente general, B	85
Figura 42 Modelo de navegación - Administrador, A.....	86
Figura 43 Modelo de navegación - Administrador, B.....	87
Figura 44 Modelo de navegación - Jefe de agencias, A.....	87
Figura 45 Modelo de navegación - Jefe de agencias, B.....	88
Figura 46 Modelo de navegación - Contador general.....	89
Figura 47 Modelo de navegación - Asesor comercial.....	90
Figura 48 Modelo de navegación - Agente inmobiliario.....	91
Figura 49 Modelo de navegación - Administrador de aginas web	92
Figura 50 Modelo de presentación - Inicio de sesión	93
Figura 51 Modelo de presentación - Sucursal.....	93
Figura 52 Modelo de presentación - Gestionar empleados	94
Figura 53 Modelo de presentación - Gestionar empleado, perfil	94
Figura 54 Modelo de presentación - Roles y permisos.....	94
Figura 55 Modelo de presentación - Gestionar cliente	95
Figura 56 Modelo de presentación - Gestionar cliente, perfil.....	95
Figura 57 Modelo de presentación - Gestionar inmueble, Registro LCD	96
Figura 58 Modelo de presentación -Gestionar inmuebles, lotes urbanizados.....	96
Figura 59 Modelo de presentación - Gestionar inmuebles, lotes urbanizados, manzanos	97

Figura 60 Modelo de presentación - Gestionar inmuebles, lotes urbanizados, manzanos, lotes.....	97
Figura 61 Modelo de presentación - Gestionar venta, registros LCD	98
Figura 62 Modelo de presentación - Gestionar venta, lotes urbanizados	98
Figura 63 Modelo de presentación - Gestionar venta, lote urbanizado, plan de pago	99
Figura 64	99
Figura 65 Modelo de presentación - Seguimiento de ventas, lotes urbanizados	100
Figura 66	100
Figura 67 Modelo de presentación - Seguimiento de asesor comercial, Comisión lotes Urbanizados	101
Figura 68 Modelo de presentación - Agenda de citas	101
Figura 70 Codificación - Página principal del software	102
Figura 71 Codificaron - Controlador empleado.....	102
Figura 72 Organigrama de la empresa JALHISA LTDA.	119

RESUMEN

El presente proyecto de grado se enfoca en la temática de bienes inmuebles que están muy inmersos en nuestra sociedad. La inmobiliaria JALHISA LTDA. realiza sus procesos administrativos de forma manual y semiautomático “utilizando cuadernos de anotaciones y programas ofimáticos”. Al cual se busca dar una solución, mediante la sistematización a través del desarrollo de software Web.

Utilizando de esta manera tecnologías emergentes; como ser: MariaDB para el almacenamiento de datos, framework Laravel, framework Vuejs, API REST, SPA “aplicaciones de una solo página”, arquitectura de componentes, entre otras cosas más.

El libro contiene cuatro capítulos muy importantes; los cuales describe desde la problemática hasta la solución, al problema planteado.

CAPITULO I

MARCO PRELIMINAR

1.1. INTRODUCCIÓN

A medida que el tiempo transcurre, la tecnología va avanzando causando que las personas tengan que adaptarse a estos cambios y depender de ellos, ya que se encuentran inmersas en nuestras vidas. En la actualidad tener una computadora o un celular inteligente ya no es un lujo, es una necesidad para acceder a la información y comunicación.

Las empresas inmobiliarias, manejan una gran cantidad de datos e información, de las cuales dependen, para su propio beneficio. Los sistemas informáticos accesibles e integrados de fácil manejo son una necesidad, para mejorar los procesos y la manipulación de los datos, es necesario sistematizar los procesos que ayudaran al desenvolvimiento y el control de los datos e información, teniendo de esta manera una mejor administración; una solución a estas demandas es el desarrollo de sistemas de información como sistemas web, que se puede acceder desde un navegador web en diferentes dispositivos como computadoras de escritorio, computadoras portables, dispositivos móviles y Tablet, por parte de los clientes y empleados.

Las aplicaciones web, presentan una serie de beneficios con respecto al software de escritorio, con los cuales se logran aprovechar los recursos de una empresa de una forma mucho más práctica, ya que permite el trabajo a distancia en cualquier momento y en cualquier lugar del mundo siempre que se tenga conexión a internet. Tienen una compatibilidad multiplataforma ya que puede ser accedido por distintos sistemas operativos que cuenten con un navegador web, también pueden ser accedidos por múltiples usuarios al mismo tiempo.

El desarrollo de un sistema informático es una necesidad primordial para las empresas inmobiliarias; al incorporar tecnología para mejorar la gestión de la información y los procesos administrativos, siendo más ordenado y confiable.

De esta manera el presente proyecto tiene el objetivo de optimizar los diferentes procesos administrativos relacionados con la venta y el servicio de publicidad de inmuebles; mejorando de esta manera el servicio hacia los clientes de la empresa JALHISA LTDA., lo que conlleva a obtener mayor utilidad, esto mediante el diseño y desarrollo del sistema informático. Haciendo uso de la metodología UWE para el desarrollo del sistema. Utilizando herramientas como: PHP, MariaDB, Laravel¹, Vue², Element-ui³, CSS, HTML y JavaScript.

1.2. ANTECEDENTES DE LA INVESTIGACIÓN

1.2.1. Antecedentes Institucionales

La empresa JALHISA LTDA inicia sus actividades el 8 de marzo del 2018, constituyéndose una Empresa Legalmente Establecida en el país, con Matrícula de Comercio No. 00393513, Número de Identificación Tributaria (NIT) 353452028, registrado en El ministerio de Trabajo con el código de Empleador N° 353452028-1, siendo su representante legal el Sr. Juan Carlos Laura Caljaro, mismo que se encuentra registrado con las siguientes actividades:

- **Actividad principal:** división automotriz; Importación, distribución y comercialización de vehículos automotores nuevos.
- **Actividad Secundaria:** la división inmobiliaria hace referencia a dos actividades importantes:
 - Venta: lotes, casas y edificios.

¹ Laravel: “marco de trabajo” para el Back-end

² Vue: “marco de trabajo” para el Front-end

³ Element-ui: “marco de trabajo” para el Front-end

- Servicio de publicidad: venta, alquiler y anticrético de lotes, casas y edificios; para dar a conocer al público en general mediante folletos y la red social Facebook.

Siendo su casa matriz en la Av. 6 de Marzo N° 657 entre las calles 5 y 6. Edificio Rojo Galería Aroma, Of. 4, Zona 12 de Octubre, se constituyó su primera sucursal de venta en las instalaciones del Edificio nb2.

MISIÓN: La Empresa JALHISA LTDA en su división Inmobiliaria dedica sus actividades a la venta y servicio de publicidad de inmuebles, enfocando sus directrices hacia la satisfacción de sus clientes, ofreciendo inmuebles en las mejores ubicaciones en el Departamento de La Paz, servicio calificado, los mejores precios, eficiencia en los tiempos de entrega, asesoría, respaldo y acompañamiento en los créditos requeridos en su conjunto.

VISIÓN: Para el año 2022 INMOBILIARIA JALHISA LTDA propenderá por la consolidación de su posicionamiento a nivel departamento de La Paz y Oruro, como una de las mejores empresas, destacándose por sus estándares de calidad. Alto nivel de satisfacción de sus clientes y compromiso con el desarrollo innovador, social y económico de las diferentes regiones.

La Inmobiliaria se dedica a la venta de inmuebles, en específico la venta de lotes urbanizados.

1.2.2. Antecedentes de trabajos afines

- (Pulido, 2018) “Sistema de información para la gestión de los procesos de la inmobiliaria RE/MAX” Universidad Piloto de Colombia, para llevar un mejor seguimiento y organización de la información. Hizo uso de SCRUM, HTML5, javascript, CSS3, MySQL y Python.
- (Guanolema, 2019) “Desarrollo de un sistema web para automatizar el proceso de compra y venta en la microempresa RAZA”. Riobamba, Ecuador. Hizo uso de PHP, HTML, CSS3, javascript Y MySQL.

- (Latorre y Castañeda, 2016) “Desarrollo e implementación de sistema de administración de ventas para Humane Escuela de Negocios para optimizar los procesos de gestión de ventas de servicios académicos”. Guayaquil, Ecuador. Hizo uso de UML, arquitectura MVC, Java y SQLServe.
- (Cuellar, 2016) “Diseño de un sistema de gestión para la administración y control de proyectos de construcción basado en la metodología del PMBOK”. Sucre, Bolivia. Hizo uso de la metodología PMBOK.
- (Sarco, 2017) “Sistema de control de compra, venta e inventarios”. Universidad UMSA, para la empresa Protec. Utilizando la metodología de desarrollo de software Ágil XP, PHP, HTML, framework CodeIgniter, arquitectura MVC y MySQL.
- (Forra, 2020) “Sistema web para la venta de ladrillos y control de personal”. Universidad UPEA, para la Fábrica de cerámica Emanuel. Utilizando la metodología UWE, PHP, javascript, HTML, CSS, MariaDB y servidor web apache.

1.3. PLANTEAMIENTO DEL PROBLEMA

1.3.1. Problema Principal

Al analizar el flujo de la información del procesamiento de la información respecto a los clientes y bienes inmuebles en la inmobiliaria, se realiza en programas ofimáticos como Microsoft Excel y Word, existe una agenda de citas, que la maneja cada asesor comercial de manera personal en cuaderno de anotaciones es decir de forma manual. Los “asesores comerciales”⁴ por cada venta elaboran un contrato en Word realizan periódicamente su respectivo reporte. Esto ocasiona mayor inversión de tiempo, inadecuado manejo de la información, errores y la acumulación de tareas en determinados periodos. Por ende clientes insatisfechos, derivando en pérdidas económicas.

⁴ Persona dedicada a la venta de propiedades

1.3.2. Problemas Secundarios

- Se tiene datos e información fragmentada de la inmobiliaria, debido al inadecuado control en los procesos administrativos, por lo que es necesario realizar un diagnóstico.
- La utilización de diferentes plantillas de Excel no permite llevar una integridad de la información, lo que ocasiona pérdida de tiempo para los administradores y asesores comerciales en cruzar los datos.
- Para los administradores es complicado realizar el seguimiento de sus asesores comerciales, lo cual provoca que se pierdan oportunidades de vender algún inmueble.
- La información y datos del control y seguimiento de la actividad de venta y servicio de publicidad; se encuentran dispersados, tiene con efecto perdida de información y una inadecuada administración.
- El plan de pagos y contrato se lo realiza de manera manual, lo que provoca que el asesor comercial y el cliente se tarden más tiempo en reunirse, retrasando la venta de dicho inmueble.
- La agenda de citas la maneja cada asesor comercial en cuaderno de anotaciones de forma manual, lo que genera borrones y pérdida de datos e información.
- La información de los clientes e inmuebles se encuentra en diferentes plantillas de Excel, lo que conlleva mayor tiempo en integrar estos datos.
- La inmobiliaria carece de una página de presentación de publicidad, tiene como efecto la pérdida de clientes.

De esta manera se formula la siguiente interrogante:

¿De qué manera se pueden optimizar los procesos administrativos, tener información actualizada y brindar un servicio de calidad a los clientes de la inmobiliaria?

1.4. OBJETIVOS

1.4.1. Objetivo General

Desarrollar un “Sistema de Gestión Inmobiliaria” para optimizar los procesos administrativos y brindar servicio de calidad al cliente por parte de la empresa JALHISA LTDA., haciendo uso de las tecnologías, para incrementar la utilidad.

1.4.2. Objetivos Específicos

- Realizar el diagnóstico para examinar la estructura de la Inmobiliaria, de esta manera obtener los requerimientos funcionales y no funcionales.
- Integrar la información en una base de datos, para agilizar los procesos de los administradores y asesores comerciales.
- Sistematizar el proceso de seguimiento de asesores comerciales, para tener mayor oportunidad de vender algún inmueble.
- Automatizar el control y seguimiento de la venta y servicio de publicidad de inmuebles, para tener información confiable y una buena administración.
- Sistematizar el proceso de plan de pagos y el contrato, para que el asesor comercial y el cliente puedan reunirse lo más antes posible y así realizar la venta del inmueble.
- Automatizar y modelar la agenda de citas, para que el asesor comercial tenga información confiable y oportuna.
- Integrar los datos e información de los clientes e inmuebles, para tener una administración productiva.
- Automatizar y modelar la página de presentación de publicidad, para que sea conocido al público en general.

1.5. JUSTIFICACIÓN

1.5.1. Justificación Técnica

El proyecto es viable ya que la inmobiliaria JALHISA LTDA., cuenta con los recursos tecnológicos adecuados, como equipos de computación, hardware y software adecuado, además cuenta con internet y está dispuesto a adquirir un Hosting para el sistema.

No será necesario tener equipos de última generación para poder acceder a este sistema, ya que se podrá acceder a través de un navegador web, con conexión a internet.

1.5.2. Justificación Económica

Los principales procesos se optimizarán, mejorando la experiencia de usuario y la calidad del servicio.

➤ Costo

- ◆ La utilización de software libre para el desarrollador del sistema, es un beneficio económico para la institución.
- ◆ Minimizar el gasto de los recursos económicos en la compra de material de escritorio.

➤ Beneficio

- ◆ Facilitar el alcance a la información.
- ◆ Tener un mejor control de los datos e información.
- ◆ Mejorar la experiencia de las tareas administrativas por medio de la Automatizar.
- ◆ Proteger los datos e información con mecanismos de seguridad informática.

- ◆ Aumento de la utilidad

1.5.3. Justificación Social

El diseño y desarrollo de un sistema de gestión inmobiliaria, ayudara al flujo de datos e información a la empresa JALHISA LTDA.

Los beneficiara a los administrativos, asesores de ventas y clientes, teniendo de esta manera una mejor calidad de experiencia de usuario con el sistema, y principalmente a los propietarios quienes podrán acceder a la información en cualquier momento y saber la situación económica de la empresa en tiempo real.

1.6. METODOLOGÍA

1.6.1. UWE

Es una herramienta que nos permitirá modelar aplicaciones web, utilizada en la ingeniería web, prestando especial atención en sistematización y personalización (sistemas adaptativos). UWE es una propuesta basada en el proceso unificado y UML, pero adaptados a la web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito. (Vallejos, Larios y Sandoval, p. 33)

Etapas de la metodología UWE:

- Captura, análisis y especificación de requisitos
- Diseño del sistema
- Codificación del software
- Pruebas
- La Instalación o Fase de Implementación
- El Mantenimiento

1.6.2. Métrica de calidad

1.6.2.1. ISO 9126

Esta Norma permite especificar y evaluar la calidad del software desde distintas perspectivas, las cuales están asociadas a la adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad, y auditoría del software. (Scalone, p. 215)

La norma ISO/IEC 9126 presenta dos modelos de calidad, la primera referida a la calidad interna y externa, y el segundo modelo referido a la calidad en uso.

- Calidad externa e interna; tiene 6 características: funcionalidad, fiabilidad, usabilidad, eficiencia, facilidad de mantenimiento y portabilidad.
- Calidad en uso; el modelo de calidad en uso que muestra un conjunto de 4 características: efectividad, productividad, integridad y satisfacción.

Por tal razón se utilizará este método para la evaluación de la calidad del software.

1.6.3. Estimación de costos - COSMIC

COSMIC es un método de medición de tamaño funcional, para la estimación del software. A partir de los requerimientos funcionales del usuario en diferentes representaciones. Este método ofrece un nivel de confiabilidad compatible con todos los tipos de software. (Vazquez, p. 4)

Es el más apropiado para obtener el costo del software, tamaño del software y el tiempo estimado de duración del proyecto.

1.6.4. Seguridad del software ISO - 27001

En cuanto a la seguridad del software se aplicará el estándar ISO 27001 para resguardar los datos e información de la empresa.

El sistema de gestión para la seguridad de información, sirve para brindar soporte de los datos que se encuentran registrados en la organización. El estándar 27001 cuida la información brindando confidencialidad, disponibilidad e integridad de los datos para el buen uso de la información y no divulgación del mismo en organizaciones ya sean grandes o pequeñas. (Cadme y Duque, p. 17)

1.6.5. Pruebas de software

Una vez generado el código fuente, el software debe probarse para descubrir (y corregir) tantos errores como sea posible antes de entregarlo al cliente. La meta es diseñar una serie de casos de pruebas que tengan una alta probabilidad de encontrar errores. (Pressman, p. 411)

Las pruebas de software tienen como objetivo proporcionar información objetiva e independiente sobre la calidad del producto, es una actividad más en el proceso de control de calidad.

- Caja blanca: Se lo realiza sobre las funciones internas de un módulo
- Caja negra: Se lo realiza sobre los requisitos funcionales desde el exterior del módulo.

1.7. MÉTODO

1.7.1. Modelo iterativo e incremental

Es un modelo de desarrollo, que no es más que un conjunto de tareas agrupadas en pequeñas etapas repetitivas (iteraciones), es uno de los más utilizados en los últimos tiempos ya que; como se relaciona con novedosas estrategias de desarrollo de software y una programación extrema, es empleado en metodologías diversas.

1.8. HERRAMIENTAS

En el presente proyecto se utilizará las siguientes herramientas para el desarrollo del software:

Tabla 1

Software

	Descripción	Objetivo
MariaDB	Es un gestor de base de datos.	Almacenar los datos del sistema.
PHP	PHP “Pre Procesador de Hipertexto” Es un lenguaje de programación multiplataforma.	Back-end para la lógica del modelo de negocio con el sistema.
Laravel	Es un framework para desarrollar aplicaciones web.	
CSS	Hoja de estilo en cascada, es un lenguaje usado para definir y crear la interfaz grafica de un archivo HTML.	
Vue	Es un framework para construir interfaces de Usuario para aplicaciones web	Front-end para la interfaz gráfica de usuario amigable.
Element-ui		
HTML	Esta colección representa la manera en que se presenta la información en el navegador de internet	

Fuente: [Elaboración propia, 2021]

Tabla 2*Hardware*

	Descripción	Objetivo
Computadora	Para el desarrollo del sistema se utilizará una computadora de escritorio con un microprocesador i3	Construcción del sistema

Fuente: [Elaboración propia, 2021]

Tabla 3*Técnicas de investigación*

	Descripción	Objetivo
Observación	Permite analizar el actual proceso administrativo.	Obtener datos e información de los procesos administrativos.
Entrevista	Permite la recolección de datos e información mediante la entrevista al personal de la empresa.	
Cuestionario	Permite la recolección de datos e información a través de cuestionario al personal de la empresa.	

Fuente: [Elaboración propia, 2021]

Tabla 4*Modelado del negocio*

	Descripción	Objetivo
StartUML	Programa informático para el diseño de diagramas UML	Realizar el modelado del negocio a través de

MagicDraw	Programa informático para el diseño de diagramas UML y diagramas UWE	diagramas UML y UWE
-----------	--	---------------------

Fuente: [Elaboración propia, 2021]

1.9. LÍMITES Y ALCANCES

1.9.1. Límites

Por los requerimientos de la inmobiliaria JALHISA LTDA., se tiene las siguientes limitaciones:

- El sistema no contemplará el control de ingreso y salidas de los empleados para gestionar su pago al final del mes.
- El sistema no contemplará registro contable, libros diarios y balances financieros.
- El sistema no permitirá compras PayPal.
- Los pagos no estarán enlazados con impuestos.
- No estará enlazado al sistema de la ASFI⁵ para la verificación de deudores.

1.9.2. Alcances

El sistema contemplará todos los procesos que conlleva la venta de lotes, casas y edificios y el servicio de publicidad de venta, alquiler y anticrédito de lotes, casas y edificios. Contemplará los siguientes módulos:

- Módulo de Inicio de Sesión, para entrar al sistema.
- Módulo de gestión de usuarios, para el registro y asignación de; roles y permisos.
- Módulo de proyectos de urbanización, para el registro de inmuebles.

⁵ ASFI: Autoridad de Supervisión del Sistema Financiero

- Página de presentación de publicidad que brindará servicio de información al público en general.
- Módulo de clientes, donde se registrar a todos los clientes.
- Módulo de inmuebles, donde se encontrar todos los inmuebles de la inmobiliaria.
- Módulo de venta; al contado o con un plan de pagos.
- Módulo de seguimiento del inmueble vendido por plan de pagos.
- Módulos de agenda de citas.
- Reporte, documento privado y minuta de transferencia.

1.10. APORTES

1.10.1. Teórico

En el presente proyecto se aportará con el conocimiento adquirido durante el proceso de formación académico.

Se aplicará el decreto supremo 1793 del Estado Plurinacional de Bolivia, en la ley N° 164; específicamente se dará prioridad al uso de software libre y las tecnologías emergentes para el desarrollo de tecnologías de información y comunicación. Utilizando contenido con información digitalizada generada bajo cualquier modo o forma de expresión que puede ser distribuida.

Se utilizará arquitectura de componentes y el Patrón de Gestión de Estado “sirve como una tienda centralizada para todos los componentes de una aplicación”, en el FRONTEND⁶ del sistema.

⁶ FRONTEND: La interfaz gráfica del sistema.

1.10.2. Práctico

El sistema de gestión inmobiliaria para la empresa JALHISA LTDA., se desarrolló en atención a una necesidad, de controlar la información; para brindar un servicio de calidad y excelencia al usuario, gracias al uso de las nuevas tecnologías.

El sistema es escalable ya que se puede aumentar más tareas con el transcurso del tiempo para seguir satisfaciendo las necesidades de la empresa mientras crece.

El sistema es trazable, ya que el diseño se realiza a partir de los requerimientos funcionales y no funcionales, ayudando a guiar y controlar los procesos para futuras auditorias. Teniendo así un producto específicamente para la empresa JALHISA LTDA.

CAPITULO II

MARCO TEÓRICO

En el presente capítulo se tocarán los conceptos fundamentales teóricos para sustentar el proyecto, los cuales serán; conceptos relacionados con el título del proyecto, método iterativo e incremental, metodología de desarrollo UWE, métrica de calidad, seguridad del software, pruebas de funcionamiento, estimación de costos, arquitectura del software y las herramientas que se utilizarán para el desarrollo del software.

Para comprender el título del proyecto, tomemos como referencia los siguientes conceptos.

2.1. SISTEMA

“Conjunto de cosas que relacionadas entre sí ordenadamente contribuyen a determinado objeto”. (Real académica española, 2014)

Un sistema es un conjunto de elementos relacionados entre sí, que funciona como un todo. Existen diferentes tipos de sistemas dependiendo de las necesidades de los usuarios final como, por ejemplo: sistema de información (SI) sistemas de procesamiento de transacciones (TPS), sistemas de automatización de oficinas (OAS), sistemas de trabajo de conocimiento (KWS), sistema de gestión de relaciones (e-CRM) y sistema de información administrativo (MIS). (Kendall y Kendall, p. 2)

2.1.1. Sistema de información

Se puede plantear la definición técnica de un sistema de información como un conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar los procesos de toma de decisiones y de control en una organización. Además de apoyar la toma de decisiones, la coordinación y el control, los sistemas de información también pueden

ayudar a los gerentes y trabajadores del conocimiento a analizar problemas, visualizar temas complejos y crear nuevos productos. (Kenneth, p. 15)

Un Sistema de información (SI) es un sistema automático o manual, que comprende personas, máquinas y/o métodos organizados para agrupar, procesar, transmitir y diseminar datos que representan información para el usuario. (Gonzalez, p. 1)

Por lo tanto, se puede decir que un sistema de información de un conjunto de componentes o elementos relacionados entre sí, con un solo fin de brindar información coherente.

2.2. GESTIÓN

La gestión constituye en sí mismo el proceso administrativo o conjunto de actividades especializadas permanentes y previsibles que de acuerdo con los elementos básicos de las organizaciones podemos sistematizar así: “organización, planeación, control, dirección”. (Montano et al., p. 8)

La gestión es un conjunto de operaciones que se realizan para dirigir y administrar un negocio o una empresa, una buena gestión hace que la empresa funcione. La administración tiene cuatro puntos fundamentales que son: planear, organizar, dirigir y controlar todos los recursos pertenecientes a una organización.

2.3. INMOBILIARIA

Las empresas inmobiliarias se dedican a la venta de bienes inmuebles (casas, departamentos, terrenos). También brindan otros servicios como: alquileres. Son intermediarios entre el propietario y el cliente, obteniendo de esta manera un beneficio por esta transacción.

2.4. INGENIERÍA DE REQUERIMIENTOS

Se denomina ingeniería de requerimientos. Desde la perspectiva del proceso del software, la ingeniería de requerimientos es una de las acciones importantes de la ingeniería de software que comienza durante la actividad de comunicación y continúa en la de modelado. Debe adaptarse a las necesidades del proceso, del proyecto, del producto y de las personas que hacen el trabajo. (Presman, p. 102)

La ingeniería de requerimientos proporciona el mecanismo apropiado para entender lo que desea el cliente, analizando las necesidades y evaluando la factibilidad para negociar una solución razonable sin ambigüedades.

Los requerimientos se clasifican en funcionales y no funcionales.

2.4.1. Requerimientos funcionales

Expresan la naturaleza del funcionamiento del sistema, los requerimientos funciones definen que debe hacer el sistema, a partir de la información adquirida en la especificación de requisitos.

Se especifica en el lenguaje natural, organizado de forma jerárquico y se enumera para facilitar su gestión.

2.4.2. Requerimientos no funcionales

Los requerimientos no funciones se refiere a como debe ser el sistema; las cualidades del sistema dividido en dos categorías.

- Cualidades de ejecución: como la seguridad o la usabilidad, observables en tiempo de ejecución.
- Cualidades de evolución: como la testabilidad, mantenibilidad, extensibilidad o escalabilidad, determinado por la estructura estática del software.

En otras palabras, son aquellas características del sistema que el usuario final no ve, pero el sistema lo realiza.

2.5. MODELO ITERATIVO E INCREMENTAL

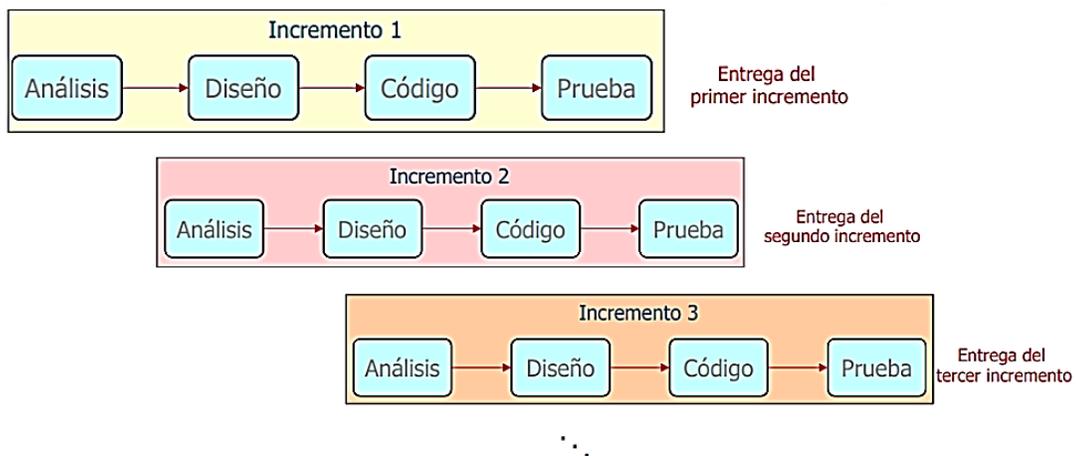
El ciclo de vida del desarrollo de software, consiste en una serie de fases que conducen al sistema final; cada fase está compuesto por un número de iteraciones que generan versiones del sistema.

2.5.1. Modelo incremental

El sistema, tal y como está especificado en la especificación de requisitos del software, se divide en subsistemas de acuerdo a su funcionalidad. Las versiones se definen comenzando con un subsistema funcional pequeño y agregando funcionalidad con cada nueva versión. Cada nueva parte entregada se denomina incremento, combina elementos del modelo en cascada con la filosofía interactiva de construcción de prototipos, aplica secuencias lineales de forma escalonada mientras progresa el calendario del proyecto. (García et al., p. 31)

Figura 1

Modelo incremental



Fuente: [García et al., p. 32]

2.5.2. Modelo iterativo

Entrega un sistema completo desde el principio, para posteriormente cambiar la funcionalidad de cada subsistema con cada versión.

Las características del ciclo iterativo son:

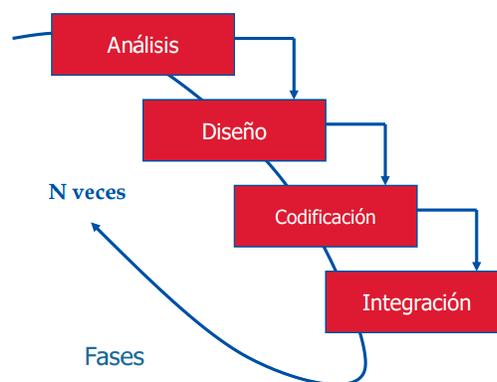
- Se basa en la evolución de prototipos ejecutables, mensurables y evaluables
- Se van incorporando cambios en cada iteración
- Exige más atención e implicación de todos los actores del proyecto

Mini cascada:

- Cada iteración reproduce el ciclo de vida en cascada, pero a una escala menor
- Los objetivos de cada iteración se establecen en función de la evaluación de las iteraciones precedentes
- Las actividades internas se solapan porque dentro de una iteración no necesitan terminarse de golpe, siendo la transición entre dos actividades progresiva.

Figura 2

Modelo iterativo



Fuente: [García et al., p. 34]

2.6. METODOLOGÍA UWE

Metodología UWE es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML. (Vallejos et al., p. 33)

A continuación, veremos las etapas de la metodología UWE.

2.6.1. Captura, análisis y especificación de requisitos

En esta etapa; se adquieren, reúnen y especifican. Las características que debe cumplir el sistema, a través de; requerimientos funcionales, requerimientos no funcionales, identificación de actores y diagrama de caso de uso. Esta etapa será la referencia para el diseño del sistema.

- **Identificación de actores:** se utiliza tablas, que contendrá el actor y la descripción, para la identificación de actores que interactúen con el sistema.

Figura 3

Identificación de actor

ACTOR	DESCRIPCIÓN

Fuente: [Vallejos et al., p. 33]

- **Requerimientos funcionales:** se utilizará para definir del funcionamiento del sistema “lo que debe hacer el sistema”. A través de tablas que contengan: rol “es el número del requerimiento funcional”, descripción “lo que el sistema debe hacer” y función “lo que tiene que hacer el sistema al ejecutar ese requerimiento”.

Figura 4

Requerimientos funcionales

ROL	DESCRIPCIÓN	FUNCIÓN

Fuente: [Vallejos et al., p. 33]

- **Requerimientos no funcionales:** se utilizará para capturar las propiedades o características del sistema como ser: el rendimiento, disponibilidad, seguridad entre otros. A través de tablas que contengan: rol “los requerimientos no funcionales” y función “restricciones del sistema”.

Figura 5

Requerimientos no funcionales

ROL	FUNCIÓN

Fuente: [Vallejos et al., p. 33]

2.6.2. Diseño

Se basa en la especificación de requisitos producido por el análisis de los requerimientos (fase de análisis), el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web.

- **Descripción de actores:** se utilizará para describir las tareas que puede realizar los actores dentro del sistema, según el rol al que pertenece.

Figura 6

Descripción de actores de caso de uso

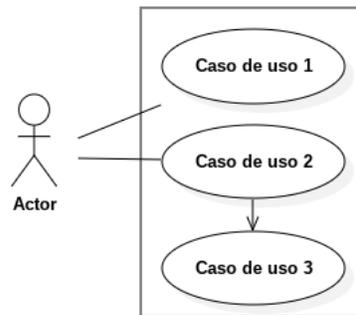
ROL	DESCRIPCIÓN

Fuente: [Vallejos et al., p. 34]

- **Diagrama de casos de uso:** se utilizará para la descripción de las acciones que debe contemplar el sistema desde el punto de vista del usuario. Modelando la funcionalidad del sistema; usando actores y casos de uso. Los casos de uso son servicios o funciones del sistema para el usuario “actor”.

Figura 7

Diagrama de casos de uso

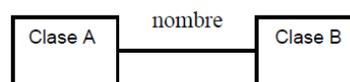


Fuente: [Vallejos et al., p. 34]

- **Modelo Conceptual:** especificará cómo se encuentran relacionados los contenidos del sistema, es decir define la estructura de los datos. Se modela a través del diagrama de clases.

Figura 8

Diagrama de clases

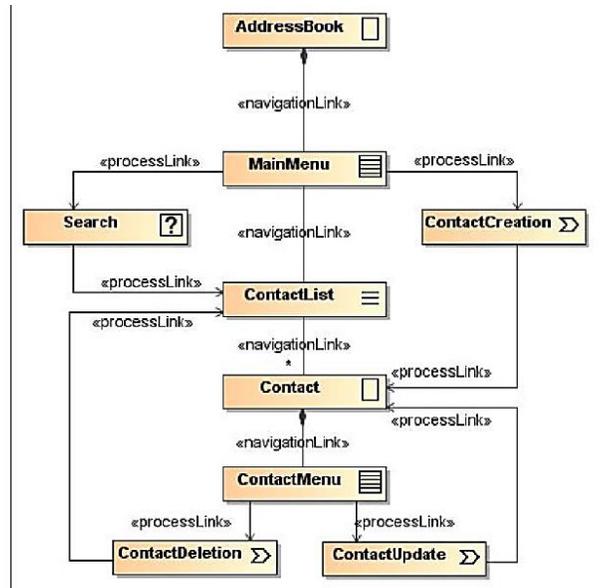


Fuente: [Vallejos et al., p. 35]

- **Modelo de navegación:** mostrará como el sistema web se encuentra relacionado internamente; los enlaces de los elementos de navegación. Dando así un panorama general del funcionamiento.

Figura 9

Modelo de navegación



Fuente: [Vallejos et al., p. 37]

2.6.3. Codificación

Durante esta etapa se realizan las tareas que comúnmente se conocen como programación; que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación, todo lo diseñado en la fase anterior. (Vallejos et al., p. 37)

2.6.4. Pruebas

Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código. (Vallejos et al., p. 37)

2.6.5. Instalación o Fase de Implementación

Es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados, y, eventualmente, configurados; todo ello con el propósito de ser ya utilizados por el usuario final. (Vallejos et al., p. 37)

2.6.6. El Mantenimiento

Es el proceso de control, mejora y optimización del producto ya desarrollado e instalado, que también incluye depuración de errores y defectos que puedan haberse filtrado de la fase de pruebas de control. (Vallejos et al., p. 37)

2.7. UML

El lenguaje unificado de modelado UML, está compuesto por diversos elementos gráficos que se combinan para conformar diagramas. Debido a que el UML es un lenguaje, cuenta con reglas para combinar tales elementos. (Vallejos et al., p. 32)

La finalidad de los diagramas es presentar diversas perspectivas del sistema, se les conoce como modelo que es una representación simplificada de la realidad; el modelo UML describe lo que hará el sistema.

A continuación, los diagramas más comunes del UML que se utilizarán:

- Diagrama de casos de uso

- Diagrama de clases

2.8. MÉTRICAS DE CALIDAD ISO-9126

La ISO 9126 describe el modelo de calidad del producto de software, aplicables a todo tipo de software. Permitiendo especificar y evaluar la calidad del software desde distintas perspectivas, las cuales están asociadas a la adquisición, requerimientos,

desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad, y auditoria del software. (Scalone, p. 215)

La norma ISO 9126 presenta dos modelos de calidad, la primera referida a la calidad interna y externa y el segundo modelo referido a la calidad en uso.

2.8.1. Calidad externa e interna

- **La calidad externa:** evalúa que el software satisfaga las necesidades del usuario teniendo en cuenta las condiciones especificadas. Esta calidad es medible en el comportamiento del producto.
- **La calidad interna:** evalúa el total de atributos que un software debe satisfacer teniendo en cuenta condiciones especificadas.³⁰ Esta calidad es medible a partir de las características intrínsecas.

La calidad externa e interna tiene 6 características: funcionalidad, fiabilidad, usabilidad, eficiencia, facilidad de mantenimiento y portabilidad.

- **Funcionalidad:** son aquellas que satisfacen las necesidades implícitas o explícitas, ya que son un conjunto de funciones y sus propiedades del sistema.

Para obtener la funcionalidad del sistema tomemos como referencia las siguientes tablas y formulas.

A continuación, para calcular el factor de ajuste, se toma en cuenta 14 características de la siguiente tabla:

Tabla 5

Valores de ajuste de complejidad

IMPORTANCIA	0%	20%	40%	60%	80%	100%
--------------------	-----------	------------	------------	------------	------------	-------------

Escala	No influencia	Incidencias	Moderado	Medio	Significativo	Esencial	Nivel de influencia
Factor	0	1	2	3	4	5	
Respaldo y recuperación							
Comunicaciones de datos							
Procesamiento distribuido							
Crítico el rendimiento							
Existencia de entorno operativo							
Entrada de datos en línea							
Transacción de entrada sobre múltiples pantallas							
Archivos maestros actualizados en línea							
Complejo de valores de dominio de información							
Complejo el procesamiento interno							
Código diseñado para reúso							
Conversión/instalación en diseño							
Instalaciones múltiples							
Aplicación diseñada para cambio							
Total $\sum(F_i)$							Nº

Fuente: [Presman, p. 603]

- ◆ Factor de ajuste

$$\text{Factor de ajuste} = \left(\sum (Fi) * 0,01 \right) + 0,65$$

- ◆ Punto de Función (PF)

Donde, **cuenta total**; se calcula a través de los factores de ponderación del sistema

$$PF = \text{cuenta total} * \left(\left(\sum (Fi) * 0,01 \right) + 0,65 \right)$$

- ◆ Punto de Función máximo (PF')

Donde el PF', alcanza $\sum(Fi) = 70$

$$PF' = \text{cuenta total} * \left(\left(\sum (Fi) * 0,01 \right) + 0,65 \right)$$

- ◆ Funcionalidad del software

$$\text{Funcionalidad} = \left(\frac{PF}{PF'} \right) * 100$$

- **Fiabilidad:** o confiabilidad del software es la probabilidad que un programa realice su objetivo satisfactoriamente (sin fallos) en un determinado periodo de tiempo.

- ◆ Para encontrar la fiabilidad del sistema se utilizará las siguientes formulas:

Probabilidad de fallos

$$\begin{aligned} P(T \leq t) &= F(t) \\ F(t) &= PF * e^{-\lambda * t} \end{aligned}$$

Probabilidad de éxito

$$\begin{aligned} P(T \leq t) &= 1 - F(t) \\ F(t) &= 1 - (PF * e^{-\lambda * t}) \end{aligned}$$

- **Usabilidad:** consiste en la simplicidad con que la gente puede aprender a usar el software y aplicarlo para resolver problemas. También incluye la facilidad de instalación, operación y monitoreo.

Para determinar la usabilidad del software se realiza un cuestionario relacionado con el uso del software.

- ◆ Usabilidad:

$$U = \left(\frac{\sum xi}{n} \right) * 100$$

- **Eficiencia:** es la habilidad del software para poner la cantidad mínima de demanda sobre los recursos de hardware como sea posible, tales como el tiempo de procesador, espacio ocupado en memorias internas o externas, ancho de banda usado en dispositivos de comunicación.
- **Facilidad de mantenibilidad:** es la facilidad con la cual se puede corregir el software si se encuentra un error, adaptarlo si su entorno cambia, o mejorarlo si el cliente desea un cambio en los requisitos. En otras palabras, es la "facilidad de comprender, corregir, adaptar y mejorar el software".

- ◆ Para hallar la mantenibilidad tomemos las siguientes variables y formula:

$$M = \frac{(Mt - (Fa + Fc + Fd))}{Mt}$$

- **Portabilidad:** consiste en la facilidad de transportar de software a varios ambientes de hardware y software.

2.8.2. Calidad en uso

Las métricas de calidad en uso miden los efectos de uso del software en un contexto específico de uso. Esta métrica mide si el producto se corresponde con las

necesidades específicas de los usuarios para así obtener los objetivos específicos con eficiencia, productividad, seguridad y satisfacción en un contexto de uso específico. (Scalone, p. 227)

El modelo de calidad en uso muestra un conjunto de 4 características:

- **Efectividad:** Permitir a los usuarios alcanzar objetivos especificados con exactitud, en un contexto de uso especificado.
- **Productividad:** Atributo relacionado con el rendimiento en las tareas cotidianas realizadas por el usuario.
- **Seguridad:** Alcanzar niveles aceptables del riesgo de hacer daño a personas, al negocio, al software, a las propiedades o al medio ambiente.
- **Satisfacción:** Satisfacer a los usuarios con el uso del software.

2.9. SEGURIDAD DEL SOFTWARE ISO - 27001

Es un sistema de gestión para la seguridad de información que sirve para brindar soporte de los datos que se encuentran registrados en la organización. Para ello se implementa un SGSI con estándar 27001 para el cuidado de la información brindando confidencialidad, disponibilidad e integridad de los datos para el buen uso de la información y no divulgación del mismo en organizaciones ya sean grandes o pequeñas. (Cadme y Duque, p. 17)

La norma ISO 27001 es un estándar de buenas prácticas de la seguridad informática, tiene las siguientes características:

- **Confidencialidad de datos:** tiene como propósito garantizar la seguridad al momento de ingresar a la información, no divulgación de la información a personas ajenas a la institución; con ello se busca que accedan a la información solo aquellos usuarios pertenecientes a la institución.

- **Disponibilidad de datos:** tiene como propósito que los datos e información del sistema informático estén disponibles y accesibles para los usuarios en lo posible 24hrs. del día, los 365 días de año.
- **Integridad de datos:** hace referencia a que los datos no pueden ser alterados por cualquier usuario. Teniendo como referencia el rol al cual pertenezca el usuario y la jerarquía en la institución; tendrá permisos para alterar los datos.

2.10. PRUEBAS DE FUNCIONAMIENTO

La meta de probar es encontrar errores, y una buena prueba es aquella que tiene una alta probabilidad de encontrar uno. Por tanto, un sistema basado en computadora o un producto debe diseñarse e implementarse teniendo en mente la “comprobabilidad”. Al mismo tiempo, las pruebas en sí mismas deben mostrar un conjunto de características que logren la meta de encontrar la mayor cantidad de errores con el mínimo esfuerzo. (Pressman, p. 412)

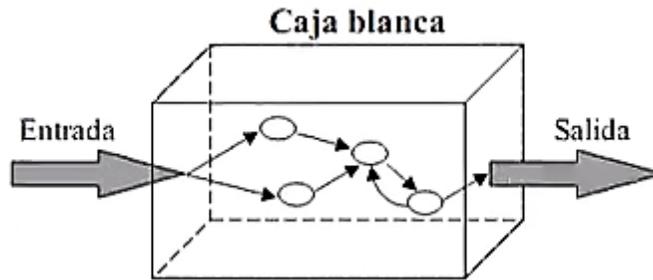
Existen diferentes métodos de pruebas del funcionamiento del software, a continuación se mencionara los más sobre salientes; prueba de caja blanca, prueba caja negra y prueba de estrés.

2.10.1. Prueba de caja blanca

También se le llama prueba de caja de vidrio, es una filosofía de diseño de casos de prueba que usa la estructura de control descrita como parte del diseño a nivel de componentes para derivar casos de prueba. (Presman, p. 414)

Figura 10

Prueba de caja blanca



Fuente: [Pressman, p. 414]

Al usar este método, se puede derivar casos de prueba que se:

- Garantizar que todas las rutas independientes dentro de un módulo se revisaran al menos una vez.
- Revisará todas las decisiones lógicas en sus lados verdadero y falso.
- Ejecutará todos los bucles en sus fronteras y dentro de sus fronteras operativas.
- Revisará estructuras de datos internas para garantizar su validez.

2.10.1.1. Prueba de ruta básica

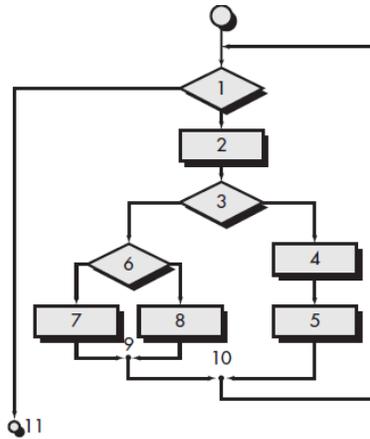
La prueba de ruta o trayectoria básica es una técnica de prueba de caja blanca. El método de ruta básica que permite obtener una medida de complejidad lógica de un diseño de procedimiento y usar esta medida como guía para definir un conjunto básico de rutas de ejecución.

Se tomará los siguientes pasos para realizar esta prueba:

- a) **Notación de grafo de flujo:** Se utilizará para la representación del flujo de control lógico. En el diseño de procedimiento se usa un diagrama de flujo para mostrar la estructura de control del programa, con el objetivo de mapear el grafo de flujo correspondiente.

Figura 11

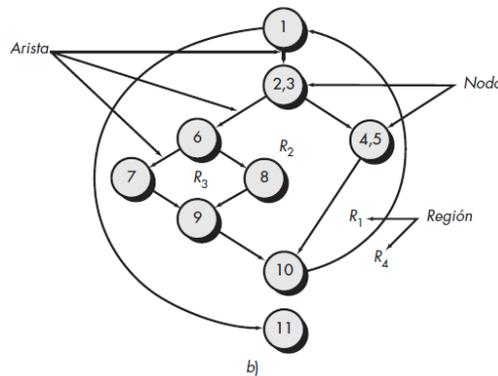
Diagrama de flujo



Fuente: [Pressman, p. 415]

Figura 12

Grafo de flujo



Fuente: [Pressman, p. 415]

- Rutas de programa independientes: cuando se establece un grafo de flujo, una ruta independiente debe moverse a lo largo de al menos una arista que no se haya recorrido antes de definir la ruta.

b) Complejidad Ciclomática: tiene fundamentos en la teoría de grafos y proporciona una medición de software extremadamente útil. La complejidad se calcula en una de tres formas:

- El número de regiones del gráfico de flujo corresponde a la complejidad ciclomática.
- La complejidad ciclomática $V(G)$ para un grafo de flujo G se define como:

$$V(G) = E - N + 2$$

Dónde: **E** es el número de aristas del grafo de flujo.

N el número de nodos del grafo de flujo.

- La complejidad ciclomática $V(G)$ para un gráfico de flujo G también se define como:

$$V(G) = P + 1$$

Dónde: **P** es el número de nodos predicado contenidos en el grafo de flujo G

c) Matrices de Grafos: es una matriz cuadrada cuyo tamaño es igual al número de nodos del grafo de flujo. Cada fila y columna corresponde a un nodo identificado y las entradas de la matriz corresponden a conexiones entre nodos. (Pressman, p. 420)

La matriz de grafo es la representación tabular de un gráfico de flujo, convirtiéndose en una poderosa herramienta para evaluar durante las pruebas. En su forma más simple, el enlace ponderado es 1 (existe una conexión) o 0 (no existe conexión).

Figura 13

Matriz de grafo

		Conectado a nodo				
		1	2	3	4	5
Nodo	1			a		
	2					
	3		d		b	
	4		c			f
	5		g	e		

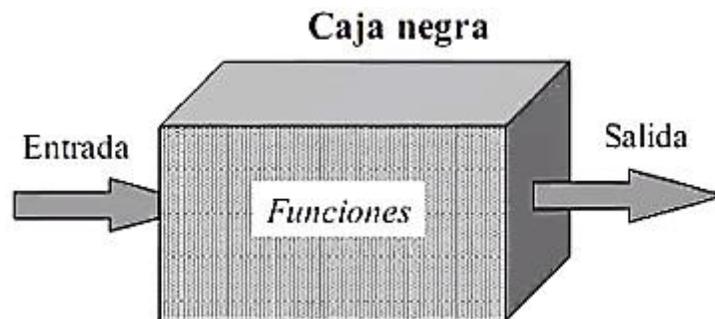
Fuente: [Pressman, p. 420]

2.10.2. Prueba de caja negra

Se enfoca en los requerimientos funcionales del software; es decir, la técnica de prueba de caja negra permite derivar un conjunto de condiciones de entrada que revisarán por completo todos los requerimientos funcionales del programa, para intentar encontrar errores que no se encontraron en la prueba de caja blanca. (Pressman, p. 423)

Figura 14

Prueba de caja negra



Fuente: [Pressman, p. 423]

Las pruebas de caja negra intentan encontrar errores en las categorías siguientes:

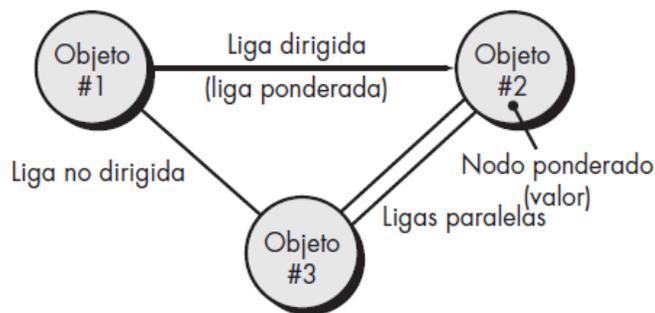
- Funciones incorrectas o faltantes.
- Errores de interfaz.
- Errores en las estructuras de datos o en el acceso a bases de datos externas.
- Errores de comportamiento o rendimiento.
- Errores de inicialización y terminación.

Se tomará los siguientes pasos para realizar esta prueba:

- a) **Método de Prueba basados en grafos:** la prueba de software comienza con la creación de un gráfico de objetos importantes y sus relaciones, y luego diseña una serie de pruebas que cubrirán el gráfico, de modo que cada objeto y relación se revise y se descubran errores.

Figura 15

Notación de gráfico



Fuente: [Pressman, p. 424]

- b) **Partición Equivalente:** es un método de prueba de caja negra que divide el dominio de entrada de un programa en clases de datos de los que pueden derivarse casos de prueba. Un caso de prueba ideal descubre de primera mano una clase de errores.

- c) **Análisis de Valor Frontera:** un mayor número de errores ocurre en las fronteras del dominio de entrada y no en el “centro”. Por esta razón es que el análisis de valor

de frontera, conduce a una selección de casos de prueba que revisan los valores de frontera.

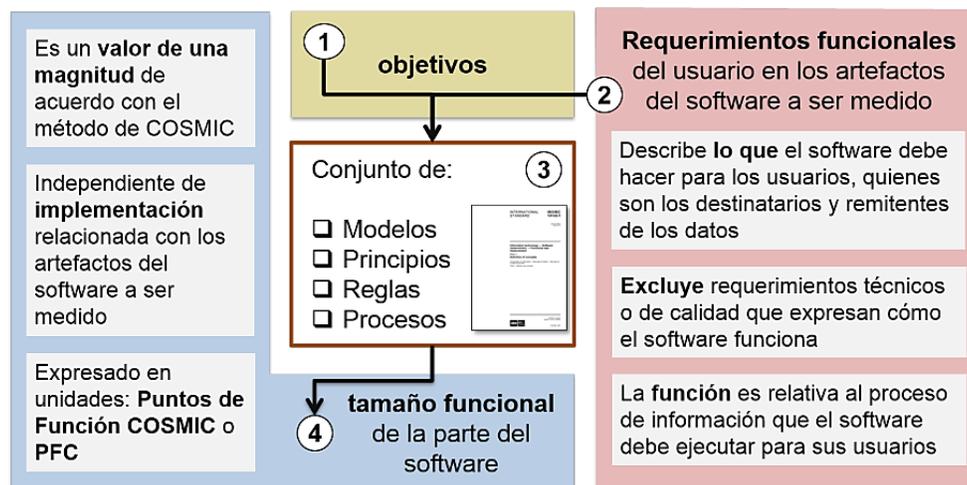
2.11. ESTIMACIÓN DE COSTO CON COSMIC

COSMIC es un método de medición de tamaño funcional, para la estimación del software. A partir de los requerimientos funcionales del usuario en diferentes representaciones. Este método ofrece un nivel de confiabilidad compatible con todos los tipos de software. (Vazquez, p. 4)

El método COSMIC es un estándar internacional bajo la ISO 19761, de dominio público y el acceso a su documentación no tiene costo y es compatible con la ingeniería de software moderna.

Figura 16

Visión general del COSMIC



Fuente: [Vazquez, p. 5]

2.11.1. Medición del software

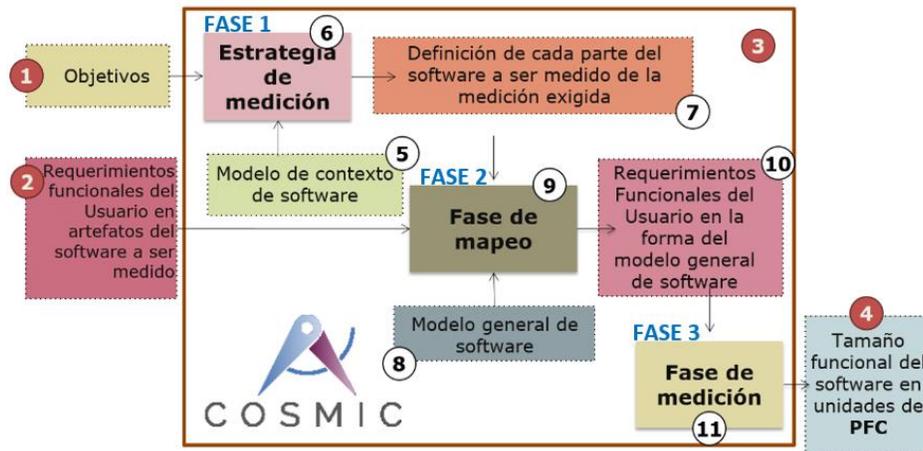
Tiene como principio la medición del software, los FUR⁷. Cuando se trata de una función, se debe considerar al usuario, ya que los requerimientos funcionales describen lo que el software debe hacer para el usuario. (Vazquez, p. 5)

El método de medición consiste en un conjunto de modelos, principios, reglas y procedimientos. Todo esto, con el propósito de generar el valor de la magnitud para la funcionalidad entregada por el software expresado en puntos de función. El resultado de la medición es el valor de una cantidad de funcionalidad entregada por el software en puntos de función.

El proceso de medición COSMIC tiene tres fases representadas como se muestra en la siguiente figura.

Figura 17

El proceso de medición COSMIC



Fuente: [Vazquez, p. 5]

2.11.2. Fase de estrategia de medición

Se describe el contexto en el cual el software es adicionado de acuerdo al objetivo de la medición. Además, delimita el software a ser medido y el usuario externo, que no

⁷ FUR: Requerimientos Funcionales del Usuario

es necesariamente una persona, definiendo cada parte del software a ser medido. (Vazquez, p. 5)

A continuación, los puntos que se deben considerar:

- Primero se debe definir lo qué se medirá.
- El tamaño de una parte del software depende de los usuarios funcionales, es decir, los humanos, dispositivos de hardware u otro software que interactúa con el software a medir.
- Para medir el tamaño de la parte de software se debe, por lo tanto, acordar primero el propósito de la medición, que lleva a definir el alcance de los FUR⁸ del software a medir y sus usuarios funcionales
- Es imprescindible documentar los parámetros de la estrategia de medición para que las medidas resultantes sean interpretadas de manera correcta por todos los futuros usuarios.

2.11.3. Fase de mapeo

La tarea en esta fase es crear el modelo COSMIC de los FUR, a partir de cualquier artefacto de software disponible. Para crear el modelo, aplicamos los principios del Modelo de Software Genérico COSMIC a los FUR a medir.

Este modelo de los FUR del software se basa en cuatro principios:

- La funcionalidad del software: consta de procesos funcionales. La tarea de cada proceso funcional es responder a un evento que ha sucedido en el mundo de los usuarios funcionales del software.
- Los procesos funcionales: constan de subprocesos. Éstos hacen solo dos cosas: se mueven y manipulan los datos. Los subprocesos de movimiento de datos que

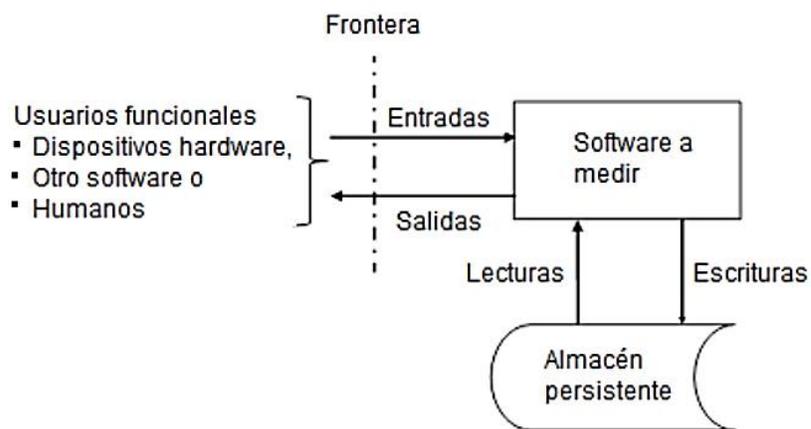
⁸ FUR: Requerimientos Funcionales del Usuario

mueven datos hacia y desde el almacenamiento persistente se les llama Escritura y Lectura.

- Cada uno de los movimientos de datos: (Entrada, Salida, Lectura o Escritura) mueve un solo grupo de datos cuyos atributos describen una sola “cosa” (un objeto de interés).
- Los subprocesos de manipulación de datos son contados por el movimiento de datos con el que están asociados. La manipulación de datos no se mide por separado. Un proceso funcional termina de ejecutarse cuando ha hecho todo lo que se requiere hacer para responder a los datos que recibió sobre el evento.

Figura 18

Los cuatro tipos de movimientos de datos



Fuente: [Vazquez, p. 5]

2.11.4. Fase de medición

La fase de medición tiene por objetivo consolidar los movimientos identificados considerando el equivalente a un Punto de Función COSMIC para cada movimiento de datos identificado.

La unidad de medida del método COSMIC es el CFP⁹. Cada uno de los movimientos de datos es medido como 1 CFP. En la fase de medición, medimos el tamaño de una nueva parte del software identificando todos los movimientos de datos (Entradas, Salidas, Lecturas y Escrituras) de cada uno de los procesos funcionales y se suman estos en todos sus procesos funcionales.

Un proceso funcional debe tener al menos dos movimientos de datos (una Entrada y una Salida o una Escritura) para proporcionar un servicio mínimo pero completo. Por lo tanto, el tamaño mínimo de un proceso funcional es 2 CFP. No hay un límite superior para el tamaño de un proceso funcional.

2.11.5. Formula

Para la estimación de costo del software con el método COSMIC, tomemos como referencia las siguientes formulas:

- Costo por Punto de Función COSMIC
 - ◆ Costo mes del equipo de trabajo: sueldo promedio del desarrollador de software
 - ◆ CFP: Punto de Función COSMIC
 - ◆ CFP mes: Punto de Función COSMIC por mes

$$\text{Costo por CFP} = \frac{\text{Costo mes del equipo de trabajo}}{\text{CFP mes}}$$

- Tamaño del software:
 - ◆ Tamaño del software: se determina por el número de CFP; de los requerimientos funciones.

⁹ CFP: Punto de Función COSMIC

$$\text{Consto del proyecto de software} = \text{tamaño del software} * \text{Costo por CFP}$$

➤ Tiempo de duración del proyecto en meses:

$$\text{Duración del proyecto} = \frac{\text{CFP}}{\text{CFP mes}}$$

2.12. SERVICIOS WEB

Una pieza del software que utiliza un conjunto de protocolos y estándares que sirven para el intercambio de datos entre aplicaciones que pueden estar desarrollados en diferentes lenguajes de programación y que pueden ser ejecutados sobre plataforma que utilicen el estándar HTTP. (García, Lozano, Mendoza, p. 1)

En la actualidad sea llegado a la conclusión que SOAP¹⁰ es demasiado complicado, y los servicios web basados en REST¹¹ es mucho más práctico; para el intercambio de datos masivos en muy poco tiempo.

XML y JSON son dos formatos para el intercambio de datos, que se utilizan en los servicios web. La ventaja de JSON sobre XML es que resulta mucho más fácil, sencillo y rápido, el flujo de datos entre cliente y servidor.

2.13. ARQUITECTURA DE SOFTWARE

La arquitectura del software es la estructura o estructuras del sistema, lo que comprende a los componentes del software, sus propiedades externas visibles y las relaciones entre ellos. (Pressman, p. 207)

¹⁰SOAP: Protocolo Simple de Acceso a Objetos

¹¹REST: Transferencia de Estado Representacional

2.13.1. Arquitectura API REST

- **API:** Interfaz de Programación de Aplicaciones, son un conjunto de definiciones y protocolos que se utilizan para diseñar e integrar el software de las aplicaciones, permitiendo la comunicación entre dos diferentes tipos de software. (Amodeo, 2013, p. 1)

Se considera como el contrato entre el proveedor de información y el usuario, donde se establece el contenido que se necesita por parte del consumidor (la llamada) y el que requiere el producto (la respuesta).

- **REST:** Transferencia de Estado Representacional, propuesta por Roy Fielding en el año 2000, como un enfoque arquitectónico para el diseño de servicios web que se apoya en el estándar HTTP, para obtener datos o generar operaciones sobre esos datos. (Amodeo, 2013, p. 2)

La REST se describe mediante un conjunto de limitaciones arquitectónicas que tratan de minimizar la latencia y las comunicaciones de la red, al mismo tiempo, maximizar la independencia y escalabilidad de las implementaciones de los componentes. Las seis limitaciones de la REST incluyen lo siguiente:

- ◆ **Cliente-servidor:** interfaz de comunicación, requiere que un servicio ofrezca una o más operaciones, y que los servicios esperen que los clientes soliciten dichas operaciones.
- ◆ **Sin estado:** requiere que las comunicaciones entre el consumidor del servicio (cliente) y el proveedor del servicio (servidor) no tengan estado. El servidor no almacena ningún tipo de información (lo que se denomina **estado**) del cliente entre las distintas peticiones.
- ◆ **Caché:** requiere que las respuestas (datos) se etiqueten claramente como con caché o sin caché.
- ◆ **Interfaz uniforme:** requiere que todos los proveedores del servicio y consumidores dentro de una arquitectura que cumple con la REST

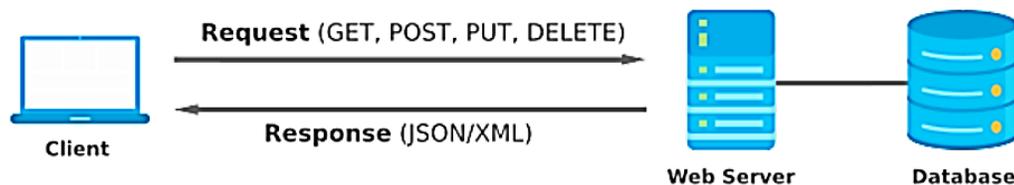
compartan una sola interfaz común para todas las operaciones (POST, GET, PUT, DELETE) sobre los recursos.

- ◆ **Sistema en capas:** requiere la capacidad de agregar o quitar intermediarios en el tiempo de ejecución sin alterar el sistema.
- ◆ **Código según la demanda (opcional):** permite que la lógica dentro de los clientes (como exploradores web) se actualice independientemente de la lógica del servidor utilizando un código ejecutable enviado por los proveedores del servicio a los consumidores.

Por lo tanto, la API REST, es una interfaz de programación de aplicaciones que se ajusta a los límites de la arquitectura REST y permite la interacción con los servicios web. Por esta razón, son más rápidas y ligeras, cuentan con mayor capacidad de ajuste, y por ende resultan ideales para el desarrollo de aplicaciones web y dispositivos móviles.

Figura 19

Interacción con la API REST



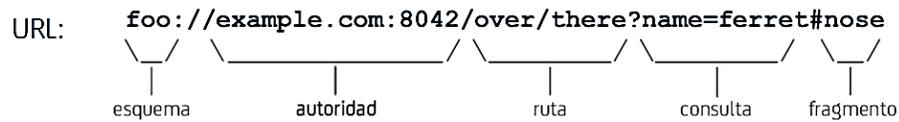
Fuente: [Bonilla, p. 7]

2.13.2. Recursos URI's

URI (identificador uniforme de recursos) cadena compacta de caracteres que permiten identificar un recurso (). Para la identificación de los recursos URI tiene la siguiente estructura:

Figura 20

Estructura de los URI



Fuente: [Amodeo, 2013, p. 8]

- **Esquema:** identifica el protocolo (FTP, HTTP, HTTPS o IRC).
- **Autoridad:** se refiere a la resolución del DNS (sistema de nombres de dominio) real del servidor (nombre de dominio o dirección IP).
- **Ruta:** se refiere a una secuencia de segmentos separados por una barra (“/”).
- **Consulta:** contiene información de identificación adicional que no es jerárquica por naturaleza y, a menudo, está separada por un signo de interrogación (“?”).
- **Fragmento:** proporciona la orientación hacia un recurso secundario dentro de uno principal identificado por la autoridad y la ruta, y está separado del resto mediante el signo de numeral (“#”).

2.13.3. Los métodos HTTP.

La arquitectura REST tiene como objetivo centrarse más en interactuar con los recursos “modelos de la base de datos”. Esto se realiza mediante operaciones de crear, leer, actualizar o eliminar (llamado de forma general operaciones CRUD), que asignan el estándar HTTP los métodos: POST, GET, PUT, DELETE.

- **POST (Crear):** solicita al servidor de origen la creación del recurso identificado por el URI de solicitud.
- **GET (Leer):** solicita al servidor de origen la obtención de la información que se haya identificado mediante el URI de solicitud.

- **PUT (Actualizar):** solicita al servidor de origen la actualización el recurso identificado por el URI de solicitud.
- **DELETE (eliminar):** solicita al servidor de origen elimine el recurso identificado por el URI de solicitud.

2.13.4. Códigos de estados

Los códigos de estado HTTP es un código numérico que informa al cliente sobre el resultado de la operación Ante cada transacción que se realiza con el servidor (Arias, p. 24).

Los códigos de estados que se utilizarán en la API REST son los siguientes:

- **200 OK:** mensajes asociados con operaciones realizadas correctamente.
- **201 Created:** La operación ha sido realizada correctamente, y como resultado se ha creado un nuevo objeto
- **401 Unauthorized:** La petición requiere una autorización especial
- **403 Forbidden:** Está prohibido el acceso a este recurso. No es posible utilizar una clave para modificar la protección.
- **404 Not Found:** La URL solicitada no existe.

2.13.5. Componentes para webapps

Es frecuente que cuando se trata de sistemas y aplicaciones basados en web, la frontera entre el contenido y la función sea borrosa. El diseño del contenido en el nivel de componentes se centra en objetos de contenido y en la forma en la que se empacan para su presentación a un usuario final de webapps. (Pressman, p. 251)

Un componente de webapp es una función cohesiva bien definida que manipula contenido, procesamiento de cómputo o de datos y funciones; que brindan al usuario final alguna capacidad solicitada.

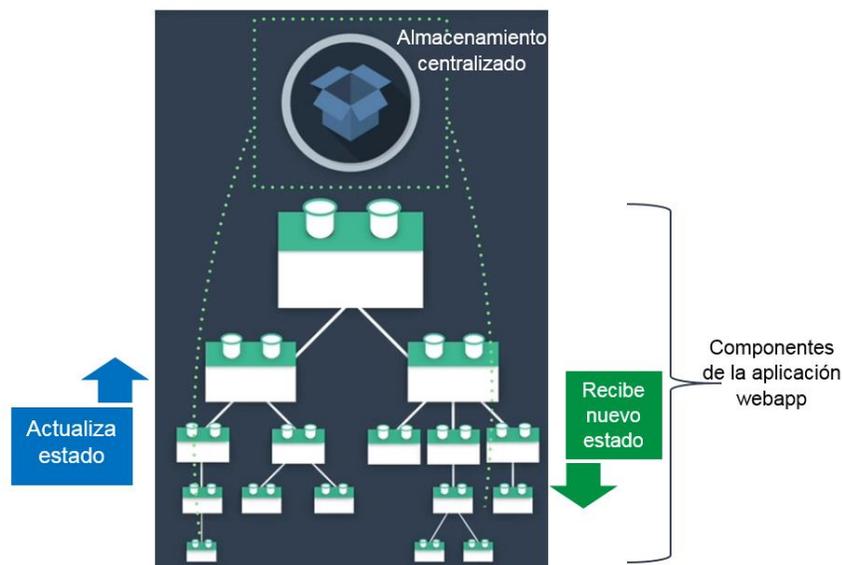
2.13.6. Patrón de Gestión de Estado

Al crear aplicaciones webapps de gran tamaño, como las aplicaciones de una sola página (SPA), que constan de muchos componentes reutilizables, se pueden volver difíciles de construir y mantener rápidamente. El intercambio de datos y el estado entre estos componentes también.

El patrón de gestión de estado sirve como un almacén centralizado “tienda” para todos los componentes de una aplicación webapps, con reglas que garantizan que el estado solo se puede mutar de manera predecible. (Stack Overflow Contributors, p. 88).

Figura 21

Patrón de gestión de estado



Fuente: [Stack Overflow Contributors, p. 88]

- **Vuex:** es un complemento oficial para Vue que ofrece un almacén de datos centralizado para usar dentro de su aplicación, ya que hace uso del patrón de gestión de estados.

Vuex tiene un objeto global llamado **store** que contiene los cuatro elementos principales para administrar el estado de la aplicación webapp:

- ◆ **State:** es un objeto JavaScript que contiene los objetos de la aplicación.
- ◆ **Getters:** son funciones que devuelven los datos contenidos en el estado.
- ◆ **Mutations:** son funciones síncronas que actualizan el estado, son el único mecanismo existente para hacerlo.
- ◆ **Actions:** son funciones asíncronas que actualizan el estado a través de una mutación existente.

Figura 22

Patrón de gestión de estado de Vuex



Fuente: [Stack Overflow Contributors, p. 89]

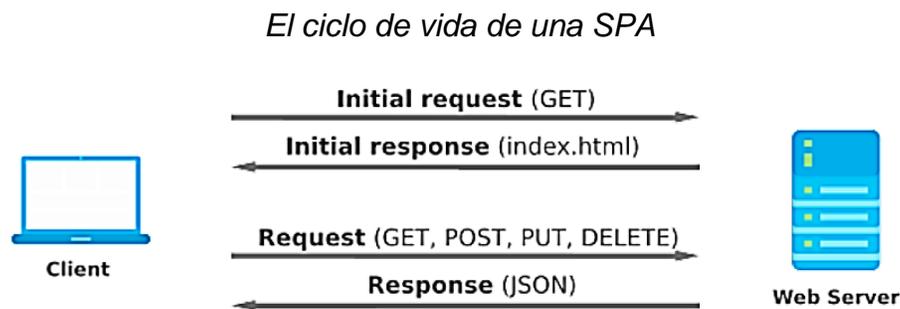
Al utilizar un almacén de datos de aplicaciones centralizado, el estado completo de la aplicación se puede representar en un solo lugar, lo que hace que la aplicación esté más organizada.

2.13.7. Arquitectura SPA

SPA (Single Page Application) es una aplicación de una sola página basada en la web que utiliza solo una página HTML como la columna vertebral de todas las sub páginas de la aplicación, no necesita actualizarse durante su utilización, una vez cargado el fichero HTML se descarga todo el código fuente necesario para la funcionalidad de la aplicación. (Bonilla, p. 9)

Las SPA realizan la petición del marcado (HTML) y los datos (JSON) de forma independiente, una vez cargado el fichero HTML y los datos, el navegador es responsable de renderizar y visualizar la página web. Una SPA interactúa con el servidor web únicamente con la transferencia de datos, lo que reduce la sobrecarga en la red, permitiendo a la aplicación ser mucho más rápida en comparación a las aplicaciones web tradicionales.

Figura 23



Fuente: [Bonilla, p. 9]

2.14. JWT

JSON Web Token (JWT) es un estándar abierto (RFC 7519) que define una forma compacta y autónoma de transmitir información de forma segura entre el cliente y servidor. Esta información puede ser verificada y confiable porque está firmada digitalmente. El servidor se encarga de firmar el token con una clave predefinida en donde el cliente y servidor podrán validar la autenticidad del token enviado. (Platero y Soñez, p 7)

Los tokens presentan un formato del tipo JSON para el intercambio de datos.

2.14.1. Estructura de JSON Web Token

En su forma compacta, los JSON web tokens constan de tres partes separadas por puntos (.), que son:

- **Encabezamiento (Header):** El encabezado generalmente consta de dos partes: el tipo de token, que es JWT, y el algoritmo de firma que se utiliza, como HMAC, SHA256 o RSA. Luego, este JSON está codificado en Base64Url para formar la primera parte del JWT.
- **Carga útil (Payload):** La segunda parte del token es la carga útil, que contiene las notificaciones. Las afirmaciones son declaraciones sobre una entidad (normalmente, el usuario) y datos adicionales. Luego, la carga útil se codifica en Base64Url para formar la segunda parte del JSON Web Token.
- **Firma (Signature):** Para crear la parte de la firma, debe tomar el encabezado codificado, la carga útil codificada, un secreto, el algoritmo especificado en el encabezado y firmarlo. La firma se utiliza para verificar que el mensaje no se haya modificado en el camino.

El resultado son tres cadenas de Base64-URL separadas por puntos que se pueden pasar fácilmente en entornos HTML y HTTP,

A continuación, se muestra un JWT que tiene codificados el encabezado y la carga útil anteriores, y está firmado con un secreto.

Tabla 6

JSON Web Token

CODIFICADO		
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MzkwMDF5Q.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c		
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9	eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MzkwMDF5Q	SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c
DESCIFRADO		
Encabezamiento (header)	Carga útil (payload)	Firma (Signature)
{ "alg": "HS256", "typ": "JWT" }	{ "sub": "1234567890", "name": "John Doe", "iat": 1516239022, }	HMACSHA256(base64UrlEncode(header) + "." + base64UrlEncode(payload),)

Fuente: [Platero y Soñez, p 8]

2.15. HERRAMIENTAS

2.15.1. Lenguaje de programación

2.15.1.1. PHP

PHP¹² es un lenguaje de programación orientado a objetos multiplataforma interpretado del lado del servidor que surge dentro de la corriente denominada

¹² PHP: Pre Procesador de Hipertexto

código abierto (Open Source). Se caracteriza por su potencia, versatilidad, robustez y modularidad. (Cobo, Gómez, Pérez y Rocha, p. 23)

2.15.1.2. Javascript

JavaScript es un lenguaje de programación orientado a objetos, que se utiliza principalmente para crear páginas web dinámicas; incorpora efectos como texto que aparece y desaparece, animaciones, acciones que se activan al pulsar botones y ventanas con mensajes de avisos al usuario. (Eguiluz, p. 5)

2.15.2. Base de datos

Una base de datos es un conjunto estructurado de datos que representan entidades y sus interrelaciones. La representación será única e integrada, debe poder ser utilizada de forma compartida por muchos usuarios de distintos tipos. (Camps, Casillas, Costal, Ginesta, Escofet y Pérez, p. 8)

2.15.2.1. MariaDB

MariaDB es un sistema de gestión de bases de datos relacional multiplataforma con una licencia GPL¹³ que permite que sea un software libre. Tiene dos motores de almacenamiento Aria y XtraDB. MariaDB permitirá el almacenamiento de datos del sistema a construir.

2.15.3. Framework

Un framework¹⁴ es un conjunto de archivos y directorios; personalizables e intercambiables, que facilitan la creación de aplicaciones ya que incorporan funcionalidades ya desarrolladas y probadas, implementadas en un determinado lenguaje de programación. (AcensTechnologies, p. 3)

¹³ GPL: Licencia Publica General

¹⁴ Marco de trabajo

El objetivo principal del framework es facilitar las cosas a la hora de desarrollar una aplicación, haciendo que nos centremos en el verdadero problema y nos olvidemos de implementar funcionalidades que son de uso común.

2.15.3.1. Backend

➤ Laravel

Laravel es un framework que utiliza como base el lenguaje de programación PHP, para la creación de aplicaciones web con sintaxis expresiva y elegante, para que el desarrollo sea una experiencia agradable y creativa. Es fácil de usar, fácil de desplegar y se le puede encontrar en muchos de los sitios web modernos que las personas utilizan día a día. (López, p. 3)

2.15.3.2. Frontend

➤ Vue

Es un marco de front-end para JavaScript, que ofrece un diseño simplificado de interfaz de usuario, manipulación y profunda reactividad. Se describe como un marco modelado MVVM, Model-View View-Model, que se basa en el concepto de datos de enlace bidireccional a componentes y vistas, es rápido y fácil de usar. (Stack Overflow Contributors, p. 2)

Vue basa su funcionamiento en el aislamiento de estados y comportamientos en pequeños componentes que se encarguen de llevar a cabo el ciclo de vida entero de una mínima parte de la UI¹⁵.

A continuación las características de Vue:

- ◆ Proporciona componentes visuales de forma reactiva, piezas de interfaz de usuario bien encapsulados.

¹⁵ UI Interfaz de Usuario

- ◆ Permite crear CSS para un componente específico.
- ◆ Cuenta con un sistema de efectos de transición y animación.
- ◆ Manejo de ruteo con Vue-router.
- ◆ Gestión de estado con Vuex.
- ◆ Tiene soporte para TypeScript.
- ◆ Permite mecanismos para que los componentes puedan ser renderizados en tiempo de servidor.
- ◆ Permite la creación de aplicaciones SPA¹⁶

➤ Element-ui

Element-ui es una librería de componentes basada en Vue para el diseño y desarrollo de interfaces gráficas de usuario para la web, específicamente dirigido al ámbito administrativo.

Element-ui se utilizará para la creación de la interfaz de usuario, como por ejemplo: los colores, estilo de letra, botón, barra de navegación, entre otros más; ya que hace uso de HTML, CSS y JavaScript, con el propósito de tener una interfaz amigable para el usuario final.

2.15.4. HTML

HTML (Lenguaje de Modelado de Híper Texto), es un lenguaje de etiquetas para construir páginas web y está a cargo de la estructura o el esqueleto de la página. Estas etiquetas HTML son palabras clave y atributos rodeados de los signos mayor y menor. La mayoría de las etiquetas HTML se utilizan en pares, una etiqueta de apertura y una de cierre. (Gauchat, p. 3)

¹⁶ SPA Aplicaciones de una Solo Pagina

2.15.5. Css

CSS es un lenguaje de hojas de estilos creado para controlar el aspecto o presentación de los documentos electrónicos definidos con HTML y XHTML. CSS es la mejor forma de separar los contenidos y su presentación y es imprescindible para crear páginas web complejas. (Eguíluz, p. 5)

Al crear una página web, se utiliza en primer lugar el lenguaje HTML/ XHTML para marcar los contenidos, es decir, para designar la función de cada elemento dentro de la página. Después se utiliza el lenguaje CSS para definir el aspecto de cada elemento: color, tamaño y tipo de letra del texto, separación horizontal y vertical entre elementos, posición de cada elemento dentro de la página, entre otros.

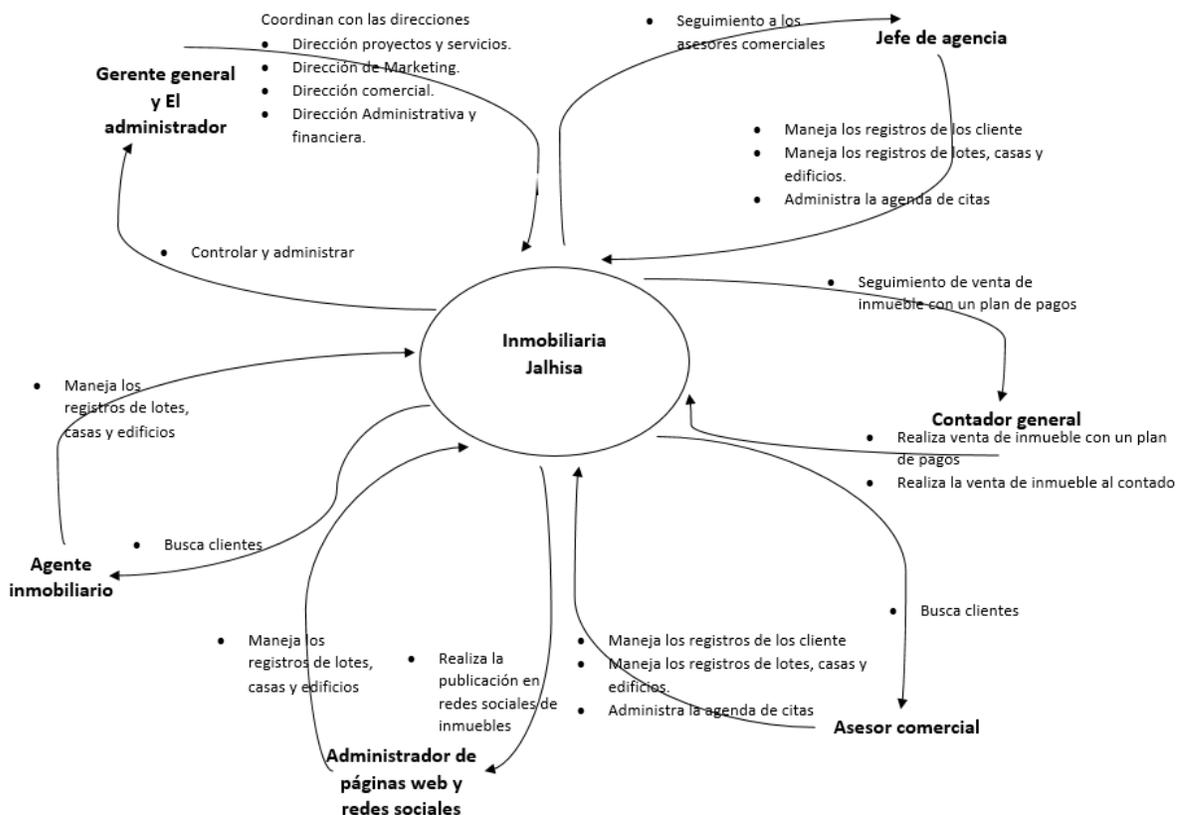
CAPITULO III

MARCO APLICATIVO

En este capítulo se dará paso al diseño y desarrollo del sistema. Haciendo uso de la metodología UWE y sus diferentes etapas para dar solución al problema que se planteó en el capítulo I y tomando como referencia los conceptos teóricos del capítulo II.

Figura 24

Situación actual de la empresa



Fuente: [Elaboración propia, 2022]

Dando como prioridad los módulos de: Acceso al sistema, gestión de lotes urbanizados, Pagina de presentación de publicidad, gestión de: usuarios, empleado, clientes, inmuebles, roles y permisos. Seguimiento de: inmueble, asesores

comerciales, venta de inmuebles y plan de pagos. Agenda de citas, realizar la copia de seguridad de los datos del sistema, entre otras tareas más.

3.1. APLICACIÓN DE LA METODOLOGÍA UWE

3.1.1. Captura, análisis y especificación de requisitos

Se utilizará como base la ingeniería de requerimientos; requerimientos funciones y no funciones. Para obtener la información clara y precisa a través de la captura, análisis y especificación de requisitos, con el propósito de tener en claro lo que se desea construir “que debe hacer el sistema”, teniendo como prioridad las necesidades del cliente dentro de la empresa relacionado con el sistema.

A continuación se describe la obtención de requerimientos a través de las entrevistas, encuestas, observación y la documentación proporcionada por la empresa JALHISA LTDA.

Tabla 7

Obtención de Requerimientos

TAREAS	DESCRIPCIÓN
Entrevistas	<p>Se realizó las entrevistas al personal de la empresa JALHISA LTDA.:</p> <ul style="list-style-type: none"> ➤ Gerente general (Jefe de la empresa JALHISA LTDA.) ➤ Administrador (sub gerente) ➤ Jefe de agencia. ➤ Contador general. ➤ Asesor comercial. ➤ Administrador de páginas web y redes sociales. ➤ Agente inmobiliario.

Encuestas	<p>Se realizó las encuestas al siguiente personal de empresa JALHISA LTDA.:</p> <ul style="list-style-type: none"> ➤ Gerente general (Jefe de la empresa JALHISA LTDA.) ➤ Administrador (sub gerente) ➤ Jefe de agencia. ➤ Contador general. ➤ Asesor comercial. ➤ Administrador de páginas web y redes sociales. ➤ Agente inmobiliario.
Observaciones	<p>En los diferentes procesos del manejo de la información, como ser: venta y alquiler de inmuebles, agenda de citas, plan de pagos, documentos privados, minuta de transferencia y seguimiento a los asesores comerciales; se lo realiza en cuadernos de anotaciones y programas ofimáticos Excel y Word. Esto ocasiona mayor inversión de tiempo, inadecuado manejo de la información y la acumulación de tareas en determinados periodos.</p>
Documentación	<p>Me proporcionaron el organigrama de la empresa JALHISA LTDA., y algunos documentos como ser: plan de pago, documento privado, documentación y plano de lote de una urbanización. La mayoría de la documentación está restringido por motivos de seguridad de la empresa.</p>

Fuente: [Elaboración propia, 2022]

3.1.1.1. Identificación de actores

Este proceso permitirá la identificación y descripción de los actores que interactuarán con el sistema.

Tabla 8

Descripción de Actores

ACTOR	DESCRIPCIÓN
Gerente general	<p>Pertenece a la Dirección de venta, proyectos y servicios. Es el representante legal de la empresa, encargado de administrar. Elabora: los planes, programas, proyectos, reglamentos, presupuestos, políticas generales y particulares de la empresa en general. Coordina con el administrador sobre los proyectos de inmuebles.</p> <p>Controla y coordina con las direcciones:</p> <ul style="list-style-type: none">➤ Dirección proyectos y servicios.➤ Dirección de Marketing.➤ Dirección comercial.➤ Dirección Administrativa y financiera.
Administrador	<p>Pertenece a la Dirección de proyectos y servicios. Controla y coordina con las direcciones:</p> <ul style="list-style-type: none">➤ Dirección de proyectos y servicios.➤ Dirección de Marketing➤ Dirección comercial➤ Dirección Administrativa

<p>Jefe de agencia</p>	<p>Pertenece a la dirección comercial. Es el encargado de las ventas de inmuebles al cliente.</p> <p>Controlar la ejecución de los planes y programas de la unidad central de recursos humanos, planificando y aplicando los lineamientos técnicos en el desarrollo de los planes y programas, para adaptar las normas y procedimientos según las necesidades de la Empresa.</p>
<p>Contador general</p>	<p>Pertenece a la dirección administrativa y financiera. Tiene como función principal el seguimiento de cobranzas de inmuebles vendidos; al contado o con un plan de pagos, reporte de ventas; mensuales y anuales, de manera general.</p>
<p>Asesor comercial</p>	<p>Pertenece a la dirección comercial. Su función principal es de vender inmuebles al contado o con un plan de pagos al público en general. Brindando una excelente atención, asesoramiento y un claro entendimiento con el cliente.</p>
<p>Administrador de páginas web y redes sociales</p>	<p>Pertenece a la dirección de Marketing. Su función principal es administrar: página web, redes sociales y realizar diseños gráficos para la empresa.</p>
<p>Agente inmobiliario</p>	<p>Persona natural externa a la empresa. Su función principal es atraer clientes a la empresa que quieran comprar un inmueble. Por cada venta que se realiza el agente recibe una comisión.</p>

Fuente: [Elaboración propia, 2022]

3.1.1.2. Requerimientos funcionales

Los requerimientos funcionales (RF), describen las características del funcionamiento del sistema “lo que debe hacer el sistema”, en la siguiente tabla.

Tabla 9

Requerimientos Funcionales

ROL	DESCRIPCIÓN	FUNCIÓN
RF-1	Acceso al sistema	Accede al sistema 7 tipo de usuarios (gerente general, administrador, jefe de agencia, contador general, asesor comercial, “administrador de páginas web y redes sociales” y agente inmobiliario). Mediante su usuario y contraseña.
RF-2	Gestionar empleados	Permite crear, listar, actualizar y eliminar el registro de los empleados.
RF-3	Gestionar clientes	Permite crear, listar, actualizar y eliminar el registro de los clientes.
RF-4	Gestionar usuarios	Permite crear, listar, actualizar, activar, inactivar y eliminar el registro de los usuarios.
RF-5	Gestionar roles	Permite la asignación de roles a los usuarios
RF-6	Gestionar permisos	Permite la asignación de permisos a los usuarios dependiendo del rol
RF-7	Administración del software	Permite al usuario con el perfil de Gerente general, tener el acceso total de los módulos y operaciones de gestión de información; creación, listado, actualización, inactivación y eliminación de los recursos del sistema.
RF-8	Perfil de usuario	Muestra la información del usuario y permite la

		actualización de su contraseña.
RF-9	Restablecer contraseña	Permite restablecer la contraseña del usuario.
RF-10	Gestionar lotes	Permite crear, listar, actualizar, activar, inactivar y eliminar el registro de los lotes.
RF-11	Gestionar casas	Permite crear, listar, actualizar, activar, inactivar y eliminar el registro de los casas.
RF-12	Gestionar edificios	Permite crear, listar, actualizar, activar, inactivar y eliminar el registro de los edificios.
RF-13	Servicio de publicidad	Permite la publicación de lotes, casa y edificios que estan disponibles a la página web de presentación para el público en general.
RF-14	Seguimiento de asesores comerciales	Permite el seguimiento de los asesores comerciales en relación con la venta de inmuebles.
RF-15	Seguimiento de venta de inmuebles	Permite listar las ventas de inmuebles (casas, lotes y edificios).
RF-16	Gestionar venta por plan de pagos	Permite generar un documento privado de la venta del inmueble, generar un plan de pagos del inmueble vendido y generar una minuta de transferencia al terminar de pagar todas las cuotas.
RF-17	Seguimiento del plan de pagos	Permite listar los planes de pagos y registrar las cuotas a pagar de cada uno de planes de pagos.
RF-18	Gestionar venta al contado	Permite generar una minuta de transferencia del inmueble vendido.

RF-19	Gestionar agenda de citas	Permite crear, listar, actualizar, activar, inactivar y eliminar el registro de la agenda de citas.
RF-20	Página de presentación de publicidad	Permite mostrar la información de empresa y todos los inmuebles disponibles al público en general.
RF-21	Búsqueda de información de un empleado.	Permite buscar información en la base de datos de un determinado empleado y mostrarlo en pantalla.
RF-22	Búsqueda de información de un cliente.	Permite buscar información en la base de datos de un determinado cliente y mostrarlo en pantalla.
RF-23	Búsqueda de información de un plan de pagos	Permite buscar información en la base de datos el plan de pagos de un determinado cliente.
RF-24	Generar reporte de venta de lotes	Permite mostrar información de las ventas de lotes.
RF-25	Generar reporte de venta de casas	Permite mostrar información de las ventas de casas.
RF-26	Generar reporte de venta de edificios	Permite mostrar información de las ventas de edificios.
RF-27	Copia de seguridad	Permite generar una copia de seguridad de los datos que se encuentra en la base de datos de manera automática.

Fuente: [Elaboración propia, 2022]

Son un total de 27 los requerimientos funcionales, características esenciales para el funcionamiento del software.

3.1.1.3. Requerimientos no funcionales

Requerimientos no funciones (RNF), en la siguiente tabla se describen las cualidades y características del sistema que el usuario no ve, pero el sistema lo realiza.

Tabla 10

Requerimientos no Funcionales

ROL	FUNCIÓN
RNF-1	El sistema debe funcionar correctamente en los navegadores web modernos como ser: Safari, Google Chrome, Mozilla Firefox IE/Edge y otros.
RNF-2	El sistema debe tener una interfaz gráfica de usuario amigable para que el usuario pueda manejarlo correctamente.
RNF-3	El sistema debe tener un menú lateral para que el usuario puede navegar a través de ella por diferentes páginas.
RNF-4	El sistema debe tener la ruta de navegación; donde se muestra al usuario la ubicación de la página actual, lo que facilita la navegación hacia atrás
RNF-5	El sistema debe tener pestañas de navegación para tener un mejor seguimiento de las páginas.
RNF-6	El sistema debe responder a todas las solicitudes del usuario.
RNF-7	El consumo y el ancho de banda de la red será mínimo ya que el sistema es un SPA (Aplicación de una Sola Página)
RNF-8	El sistema debe encriptar la contraseña de los usuarios.
RNF-9	El intercambio de datos entre el cliente y el servidor será seguro y confiable ya que el sistema utiliza JWT (JSON Web Token).
RNF-10	La sesión tendrá un determinado periodo de tiempo activo para que

	después el usuario vuelva a iniciar sesión
RNF-11	La sesión activa entre el cliente y el servidor estarán protegidos por un token encriptado (conjunto de caracteres alfanuméricos).
RNF-12	El software debe permitir agregar nuevos módulos y funciones en el futuro.

Fuente: [Elaboración propia, 2022]

3.1.2. Diseño del sistema

Para el diseño del sistema se utilizará; descripción de actores, diagrama de casos de uso, diagrama de clases, modelo conceptual y el modelo navegacional.

3.1.2.1. Descripción de actores

Las tareas que podrá realizar los actores dentro del sistema.

Tabla 11

Descripción de actores de casos de uso

ROL	DESCRIPCIÓN
Gerente general.	Realizara la gestión de usuarios (crear, listar, actualizar, activo, inactivo y eliminación)
	Realizara la gestión de empleados (crear, listar, actualizar, activo, inactivo y eliminación)
	Realizara la gestión de clientes (crear, listar, actualizar, activo, inactivo y eliminación)
	Realizara la gestión de roles (asignación de roles a los usuarios)
	Realizara la gestión de permisos (asignar permisos a los usuarios dependiendo del rol)

	<p>Crear, listar, actualizar y eliminar el registro de las urbanizaciones que contienen (manzanos, lotes).</p> <p>Crear, listar, actualizar, activo, inactivo y eliminar el registro de lotes, casas y edificios.</p> <p>Realizara el seguimiento a los asesores comerciales.</p> <p>Realizar el seguimiento del plan de pagos.</p> <p>Lista las ventas al contado.</p> <p>Realizara la gestión de servicio de publicidad.</p>
Administrador (sub gerente)	<p>Crear, listar, actualizar y eliminar el registro de las urbanizaciones que contienen (manzanos, lotes).</p> <p>Crear, listar, actualizar, activo, inactivo y eliminar el registro de lotes, casas y edificios.</p> <p>Realizara el seguimiento a los asesores comerciales.</p> <p>Realizar el seguimiento del plan de pagos</p>
Jefe de agencia	<p>Realizara la gestión de clientes (crear, listar, actualizar)</p> <p>Realizara el seguimiento a los asesores comerciales.</p> <p>Lista: lotes, casas y edificios, con un estado activos para la venta.</p> <p>Gestionará la agenda de citas (crea, lista, actualiza, activa e inactiva)</p> <p>Lista, crear, actualizar, activar e inactivar; lotes, casas y edificios para el servicio de publicidad, en las categorías de venta, alquiler y anticrético.</p>
Contador general	Gestionar la venta por plan de pagos

	<p>Realizar el seguimiento del plan de pagos</p> <p>Gestionar la venta al contado</p>
Asesor comercial	<p>Realizara la gestión de clientes (crear, listar, actualizar)</p> <p>Lista: lotes, casas y edificios, con un estado activos para la venta.</p> <p>Gestionará la agenda de citas (crea, lista, actualiza, activa e inactiva)</p>
Administrador de páginas web y redes sociales.	<p>Lista y realiza la publicación: lotes, casas y edificios; con un estado activos para la venta, alquiler y anticrético.</p>
Agente inmobiliario	<p>Lista: lotes, casas y edificios, con un estado activos para la venta.</p>

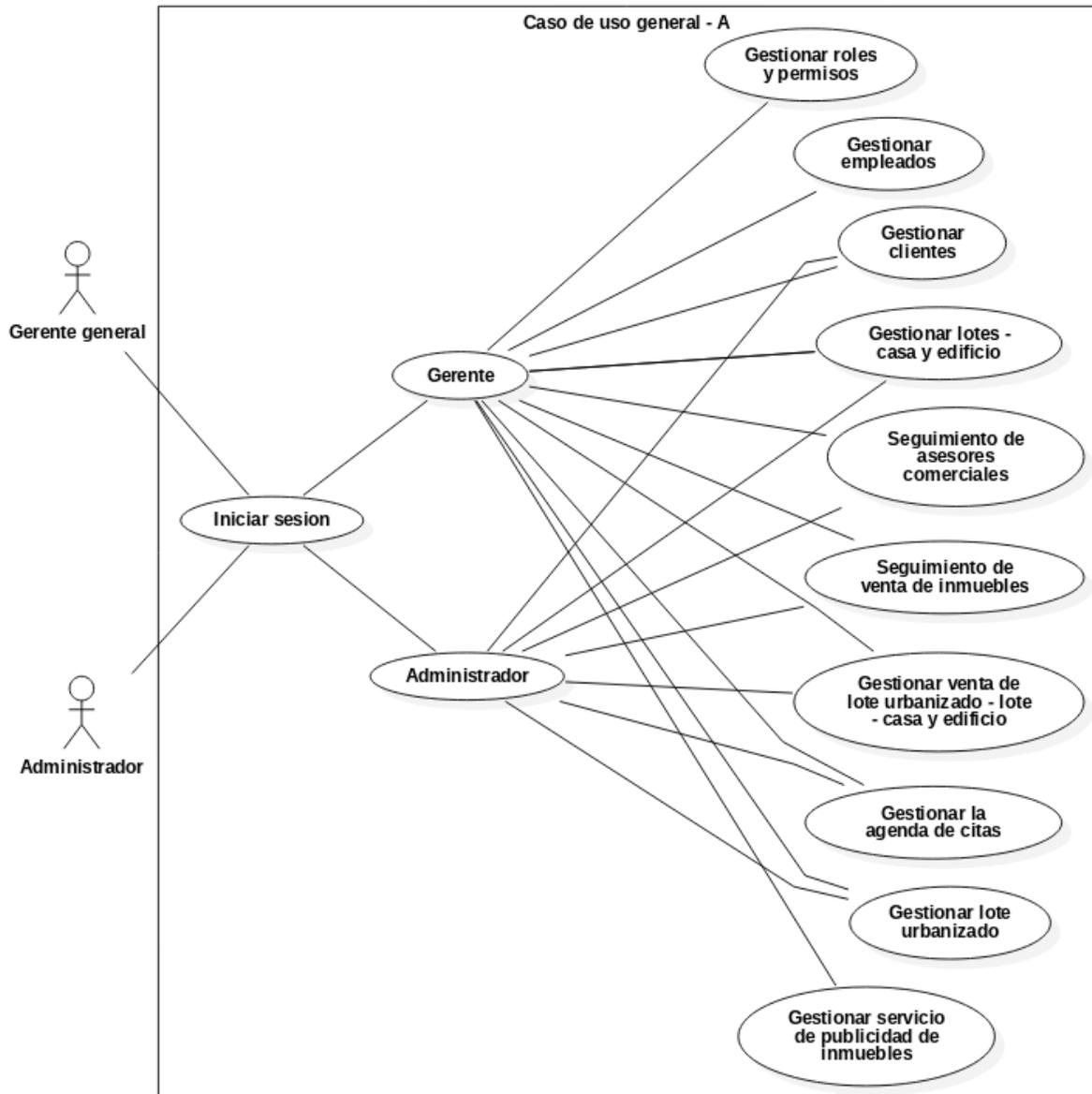
Fuente: [Elaboración propia, 2022]

3.1.2.2. Diagrama de casos de uso general del sistema

En la figura se muestra el diagrama de caso de uso general, el cual representa la interacción de los actores con el sistema.

Figura 25

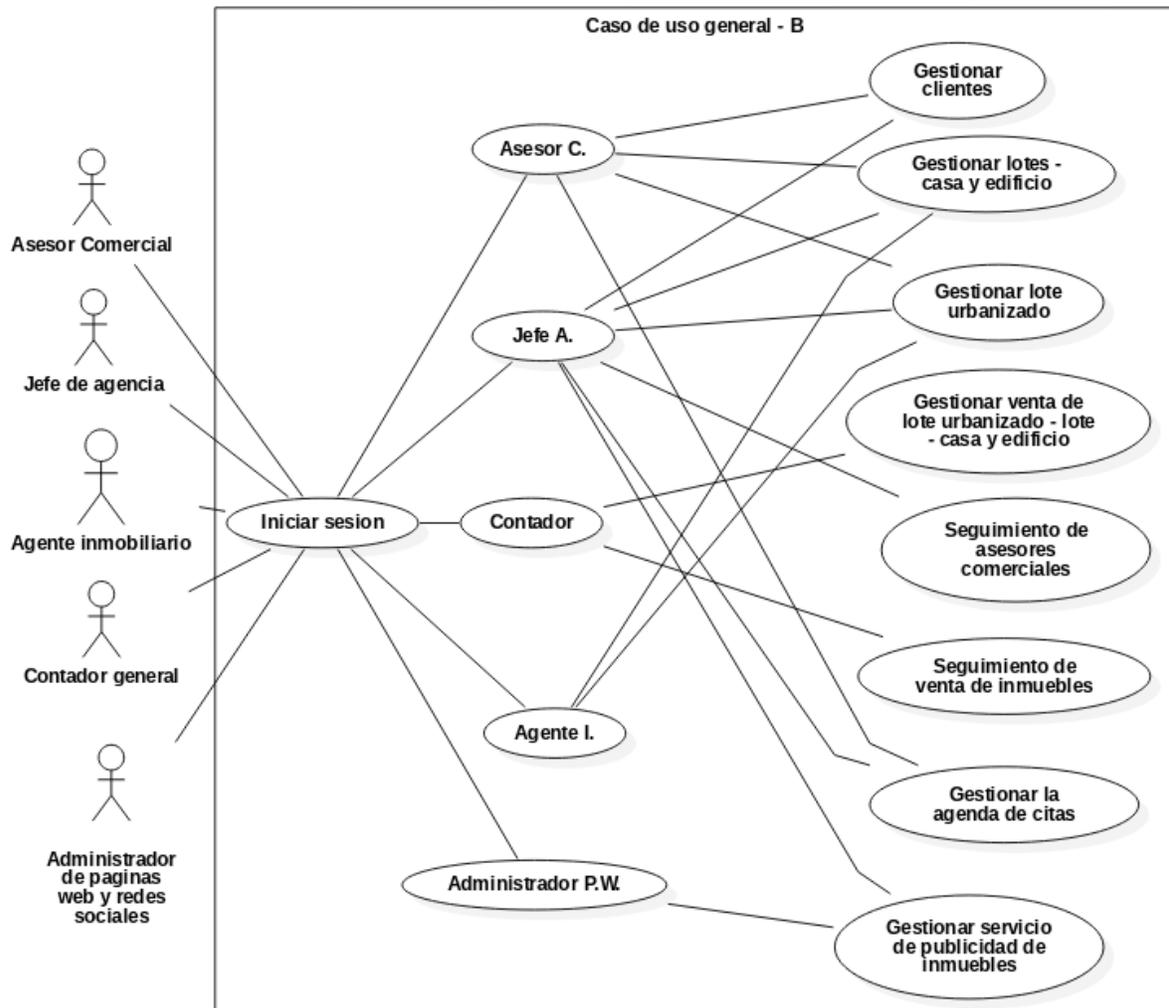
Diagrama de caso de uso general del sistema - A



Fuente: [Elaboración propia, 2022]

Figura 26

Diagrama de caso de uso general del sistema - B



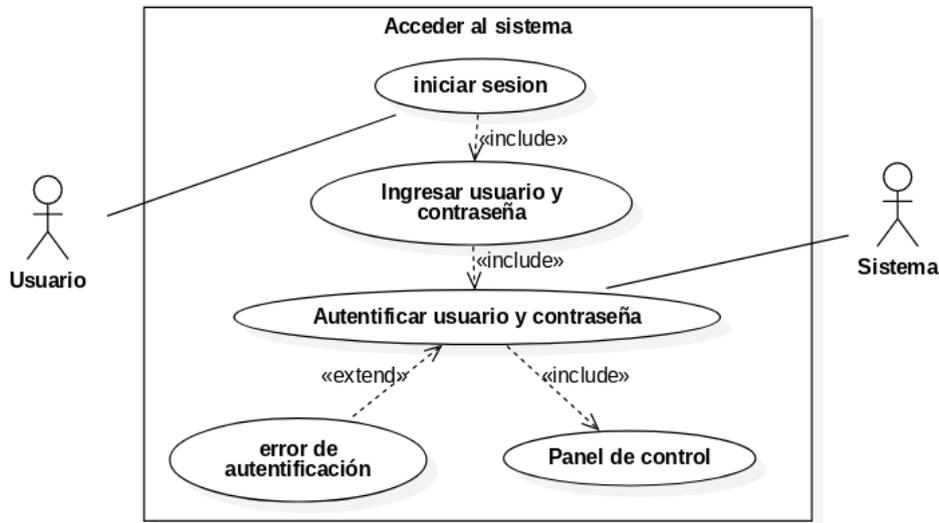
Fuente: [Elaboración propia, 2022]

3.1.2.3. Diagrama de casos de uso específicos del sistema

➤ Diagrama de caso de uso: Acceder al sistema

Figura 27

Diagrama de caso de uso, Acceder al sistema



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Acceso al sistema, en la siguiente tabla:

Tabla 12

Caso de uso, acceder al sistema

CASO DE USO: Acceder al Sistema	
Actores	Usuario: Gerente general, Administrador, Jefe de agencia, Contador general, Asesor comercial, Administrador de páginas web y redes sociales, y Agente inmobiliario.
Tipo	Primario esencial
Descripción	El usuario Inicia sesión, ingresando sus credenciales “usuario y contraseña”.

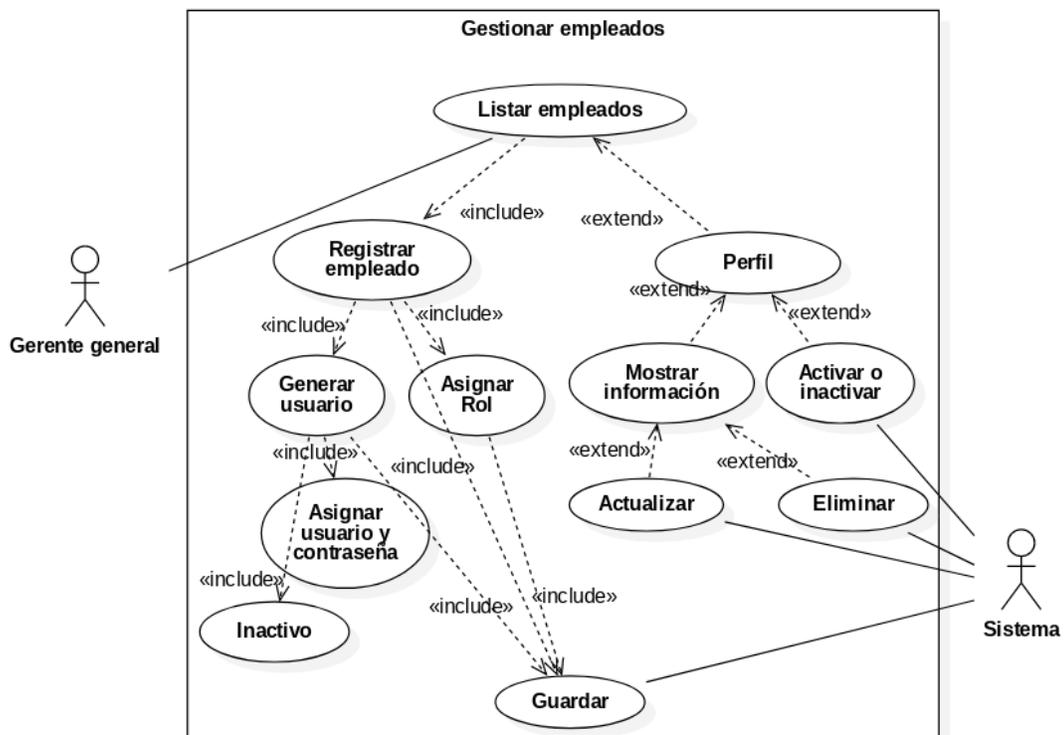
El sistema verifica sus credenciales; en el caso que sea correcto, el sistema carga el rol y los permisos asignados al usuario para luego Acceder al sistema

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso de uso:** Gestionar empleado

Figura 28

Diagrama de caso de uso, Gestionar empleados



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar empleados, en la siguiente tabla:

Tabla 13

Caso de uso, Gestionar empleados

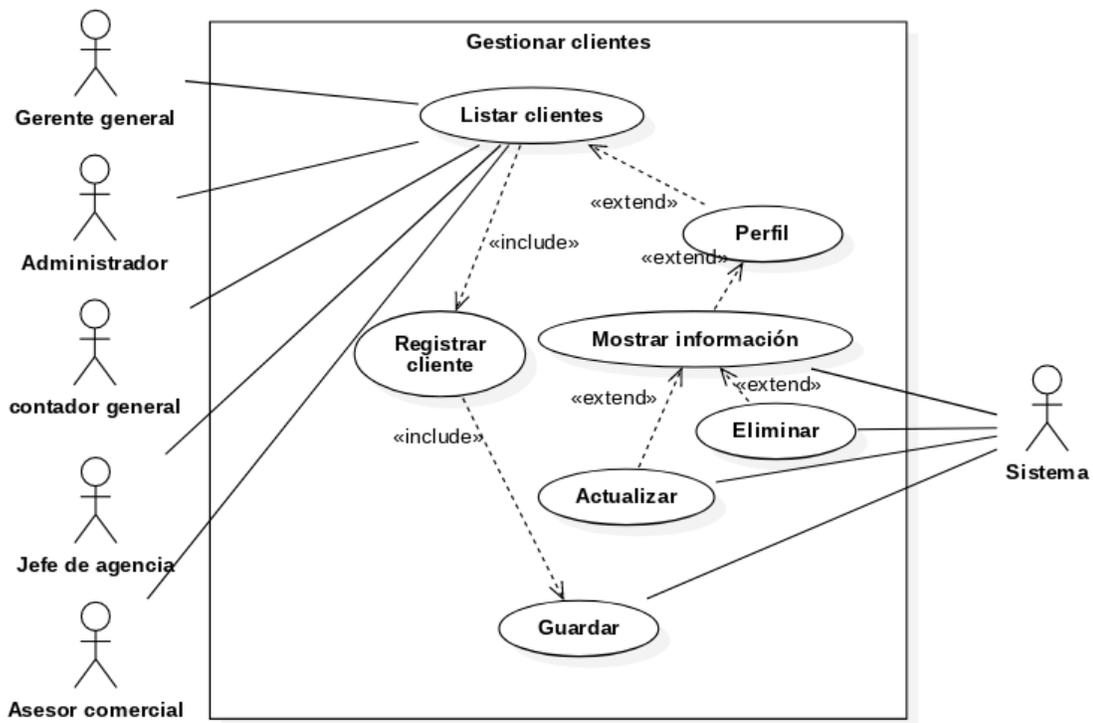
CASO DE USO: Gestionar empleados	
Actores	Gerente general.
Tipo	Primario esencial.
Descripción	Tiene acceso a los registros de los empleados. Puede realizar las operaciones de: crear, actualizar, activo, inactivo y eliminación.

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso de uso: Gestionar clientes**

Figura 29

Diagrama de caso de uso, Gestionar clientes



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar clientes, en la siguiente tabla:

Tabla 14

Caso de uso, Gestionar clientes

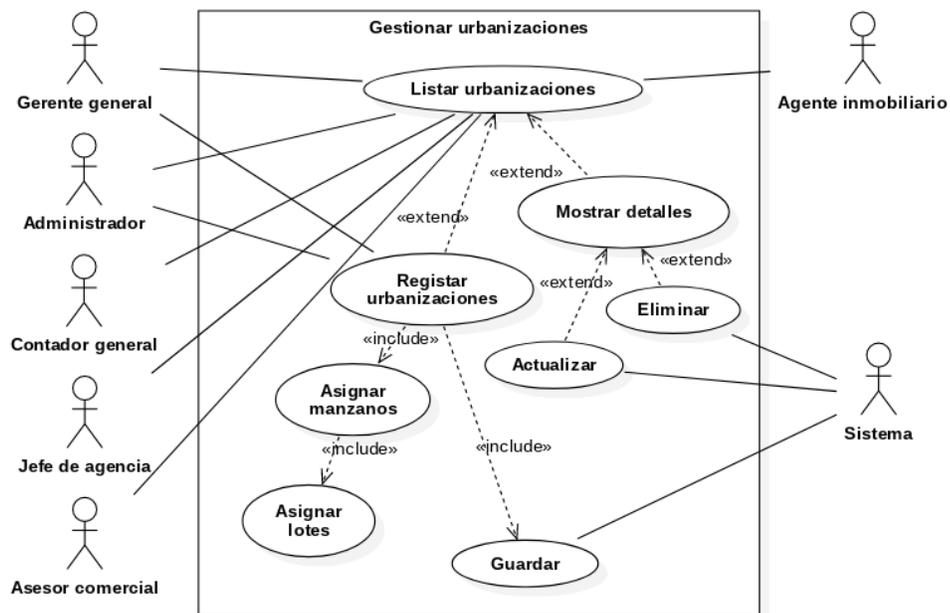
CASO DE USO: Gestionar clientes	
Actores	Gerente general, Administrador, Contador general, Jefe de agencia y Asesor comercial.
Tipo	Primario esencial.
Descripción	Tienen acceso a los registros de los clientes. Puede realizar las operaciones de: crear y actualizar. Solo el Gerente general puede y eliminar a los clientes.

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso de uso: Gestionar urbanizaciones**

Figura 30

Diagrama de caso de uso, Gestionar urbanizaciones



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar urbanizaciones, en la siguiente tabla:

Tabla 15

Caso de uso, Gestionar urbanizaciones

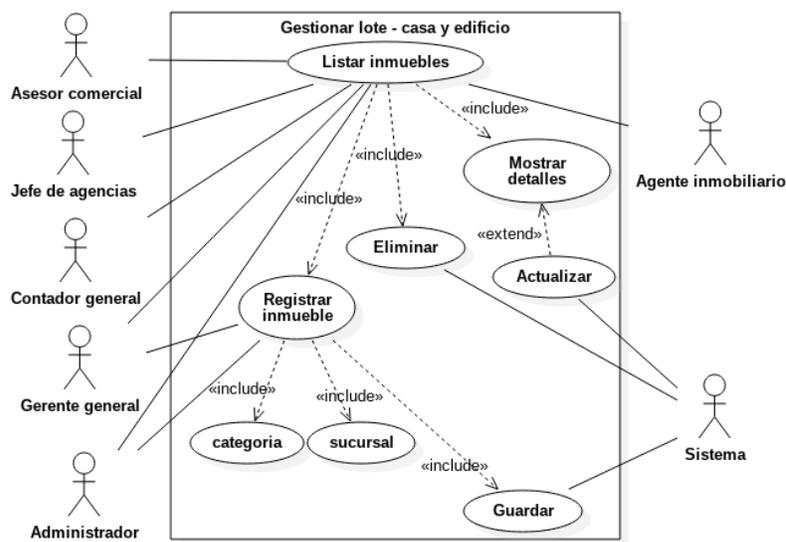
CASO DE USO: Gestionar urbanizaciones	
Actores	Gerente general, Administrador, Contador general, Jefe de agencia, Asesor comercial y Agente inmobiliario.
Tipo	Primario esencial.
Descripción	Tienen acceso a los registros de las urbanizaciones. Solo él Gerente general y él Administrador, pueden realizar el registro, eliminación y actualización las urbanizaciones.

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso de uso:** Gestionar lote, casa y edificio

Figura 31

Diagrama de caso de uso, Gestionar lote, casa y edificio



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar lote, casa y edificio, en la siguiente tabla:

Tabla 16

Caso de uso, Gestionar lote, casa y edificio

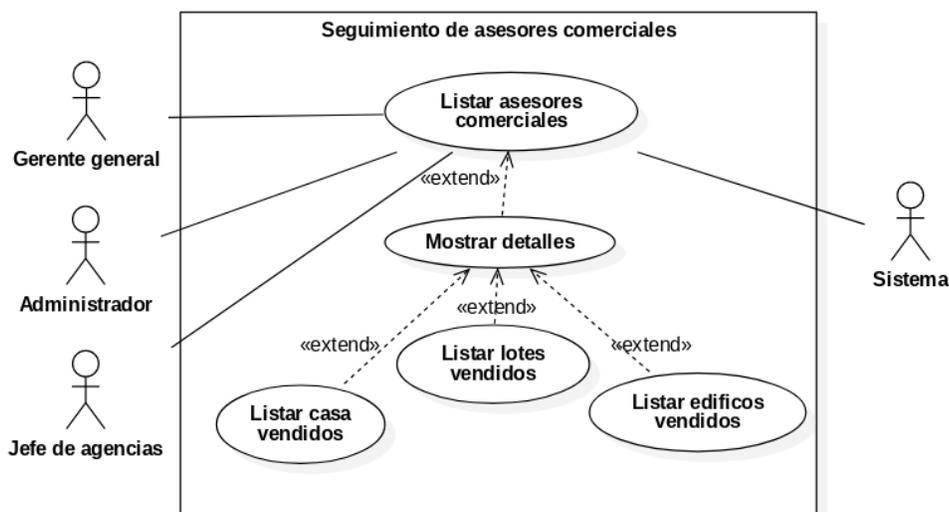
CASO DE USO: Gestionar lote, casa y edificio	
Actores	Gerente general, Administrador, Contador general, Jefe de agencia, Asesor comercial y Agente inmobiliario.
Tipo	Primario esencial.
Descripción	Tienen acceso a los registros de los lotes, casas y edificios. Solo él Gerente general y él Administrador, pueden realizar el registro, eliminación y actualización de los lotes, casas y edificios.

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso de uso:** Seguimiento de asesores comerciales

Figura 32

Diagrama de caso de uso, Seguimiento de asesores comerciales



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Seguimiento de asesores comerciales, en la siguiente tabla:

Tabla 17

Caso de uso, Seguimiento de asesores comerciales

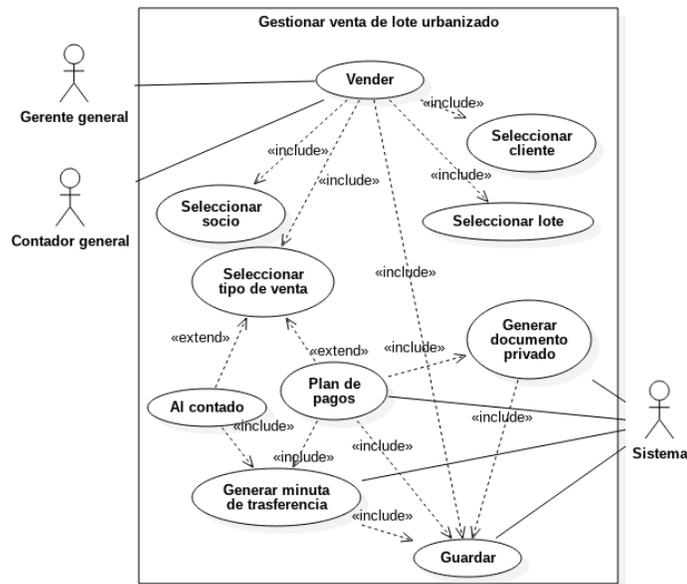
CASO DE USO: Seguimiento de asesores comerciales	
Actores	Gerente general, Administrador y Jefe de agencia.
Tipo	Primario esencial.
Descripción	Tienen acceso a los registros de los asesores comerciales, donde pueden acceder a los registros de casas, lotes y edificios vendidos por asesor comercial.

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso de uso:** Gestionar venta de lote urbanizado.

Figura 33

Diagrama de caso de uso, Gestionar venta de lote urbanizado



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar venta de lote urbanizado, en la siguiente tabla:

Tabla 18

Caso de uso, Gestionar venta de lote urbanizado

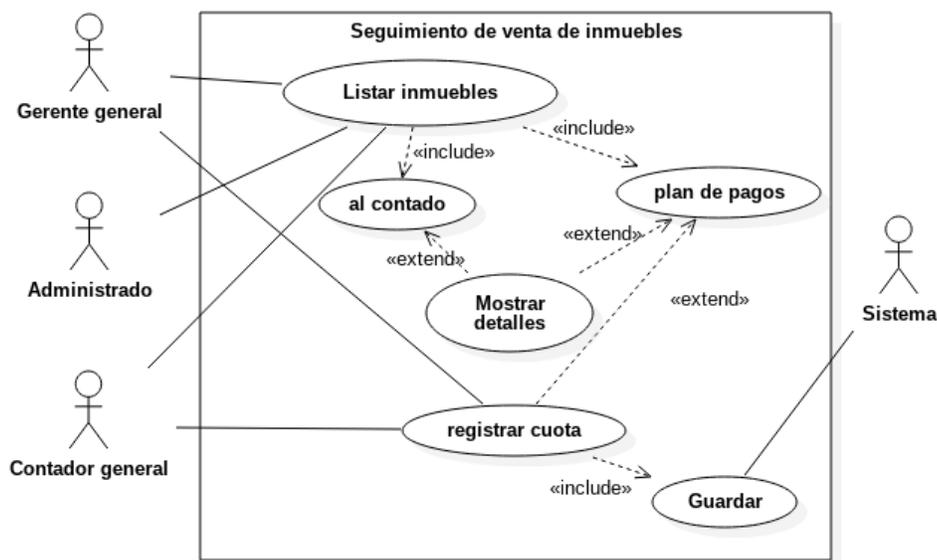
CASO DE USO: Gestionar venta de lote urbanizado	
Actores	Gerente general y Contador general.
Tipo	Primario esencial.
Descripción	Realizan la venta de un lote urbanizado; Si la venta es al contado el sistema genera una minuta de transferencia. Si la venta es con plan de pagos el sistema genera un documento privado y minuta de transferencia.

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso de uso:** Seguimiento de venta de inmuebles

Figura 34

Diagrama de caso de uso, Seguimiento de venta de inmuebles



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Seguimiento de venta inmuebles, en la siguiente tabla:

Tabla 19

Caso de uso, Seguimiento de venta de inmuebles

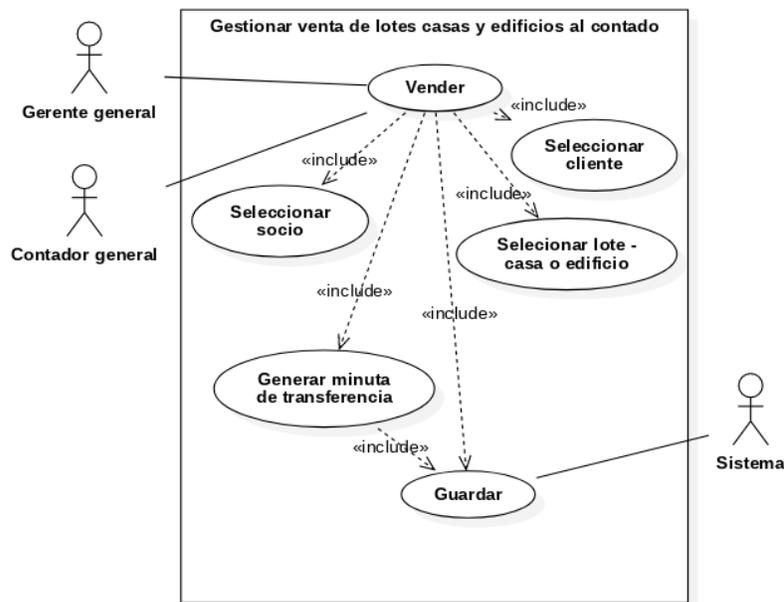
CASO DE USO: Seguimiento de venta de lote inmuebles	
Actores	Gerente general, Administrador y Contador general.
Tipo	Primario esencial.
Descripción	Tienen acceso a los registro de inmuebles vendidos al contado o con un plan de pagos. Solo él Gerente general y él Contador general pueden registrar las cuotas.

Fuente: [Elaboración propia, 2022]

- **Diagrama de caso uso:** Gestionar venta de lotes, casas y edificios al contado.

Figura 35

Diagrama de caso de uso, Gestionar venta de lotes, casas y edificios al contado



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar venta de lotes, casas y edificios al contado, en la siguiente tabla:

Tabla 20

Caso de uso, Gestionar venta de lotes, casas y edificios al contado

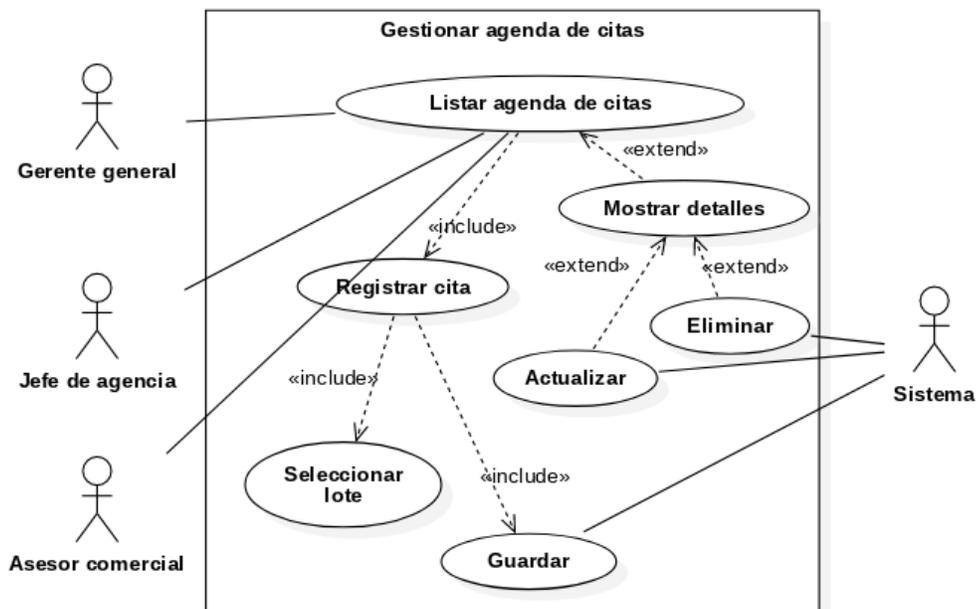
CASO DE USO: Gestionar venta de lotes, casas y edificios al contado	
Actores	Gerente general y Contador general.
Tipo	Primario esencial.
Descripción	Realizan la venta de un lote, casa o edificio al contado. Donde el sistema genera la minuta de transferencia.

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso uso:** Gestionar agenda de citas.

Figura 36

Diagrama de uso, Gestionar agenda de citas.



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar agenda de citas, en la siguiente tabla:

Tabla 21

Caso de uso, Gestionar agenda de citas

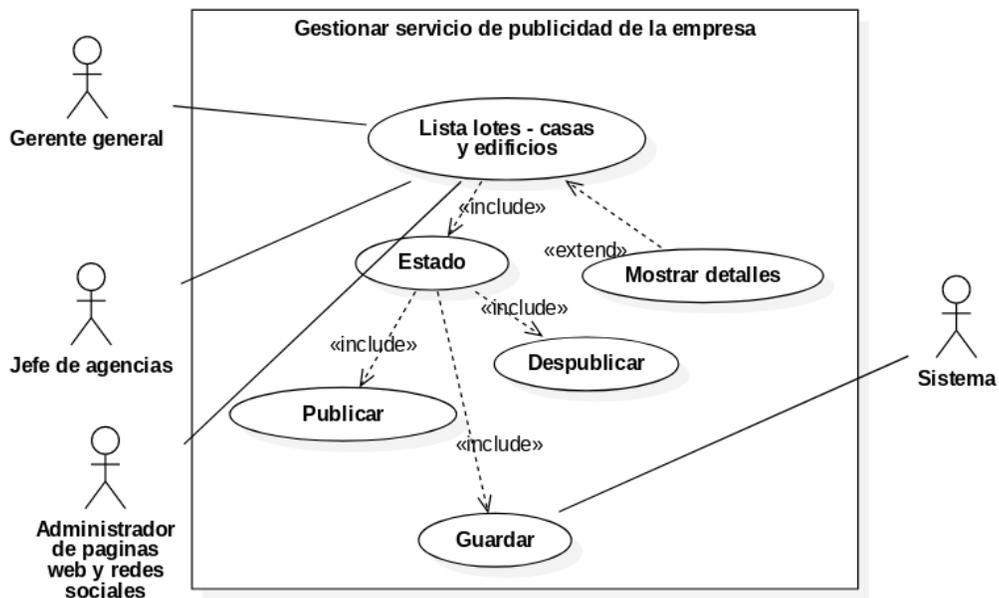
CASO DE USO: Gestionar agenda de citas	
Actores	Gerente general, Jefe de agencia y Asesor comercial.
Tipo	Primario esencial.
Descripción	Tiene acceso a los registros de la agenda. Pueden realizar el registro de citas, actualizar. Solo el Gerente general puede eliminar los registros.

Fuente: [Elaboración propia, 2022]

➤ **Diagrama de caso de uso:** Gestionar servicio de publicidad de la empresa.

Figura 37

Diagrama de caso de uso, Gestionar servicio de publicidad de la empresa



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar servicio de publicidad de la empresa, en la siguiente tabla:

Tabla 22

Gestionar servicio de publicidad de la empresa

CASO DE USO: Gestionar servicio de publicidad de la empresa	
Actores	Gerente general, Jefe de agencia y Administrador de páginas web y redes sociales.
Tipo	Primario esencial.
Descripción	Tiene acceso a los registros de lotes, casas y edificios. Pueden realizar la publicación, para que el público en general lo pueda ver desde la página web.

Fuente: [Elaboración propia, 2022]

- **Diagrama de caso uso:** Gestionar servicio de publicidad de venta, alquiler y anticrético de inmuebles para los clientes.

Figura 38

Diagrama de caso de uso, Gestionar servicio de publicidad de venta, alquiler y anticrético de inmuebles para los clientes



Fuente: [Elaboración propia, 2022]

A continuación, se describe el caso de uso Gestionar servicio de publicidad de venta, alquiler y anticrético de inmuebles para los clientes, en la siguiente tabla:

Tabla 23

Caso de uso, Gestionar servicio de publicidad de venta, alquiler y anticrético de inmuebles

CASO DE USO: Gestionar servicio de publicidad de venta, alquiler y anticrético de inmuebles para los clientes	
Actores	Gerente general, Jefe de agencia y Administrador de páginas web y redes sociales.
Tipo	Primario esencial.
Descripción	Tiene acceso a los registros inmuebles “lotes, casa y edificios”. Pueden realizar el registro y actualización de inmuebles con la categoría de servicio “alquiler, anticrético o venta”. Solo el Gerente general pobra eliminar los registros.

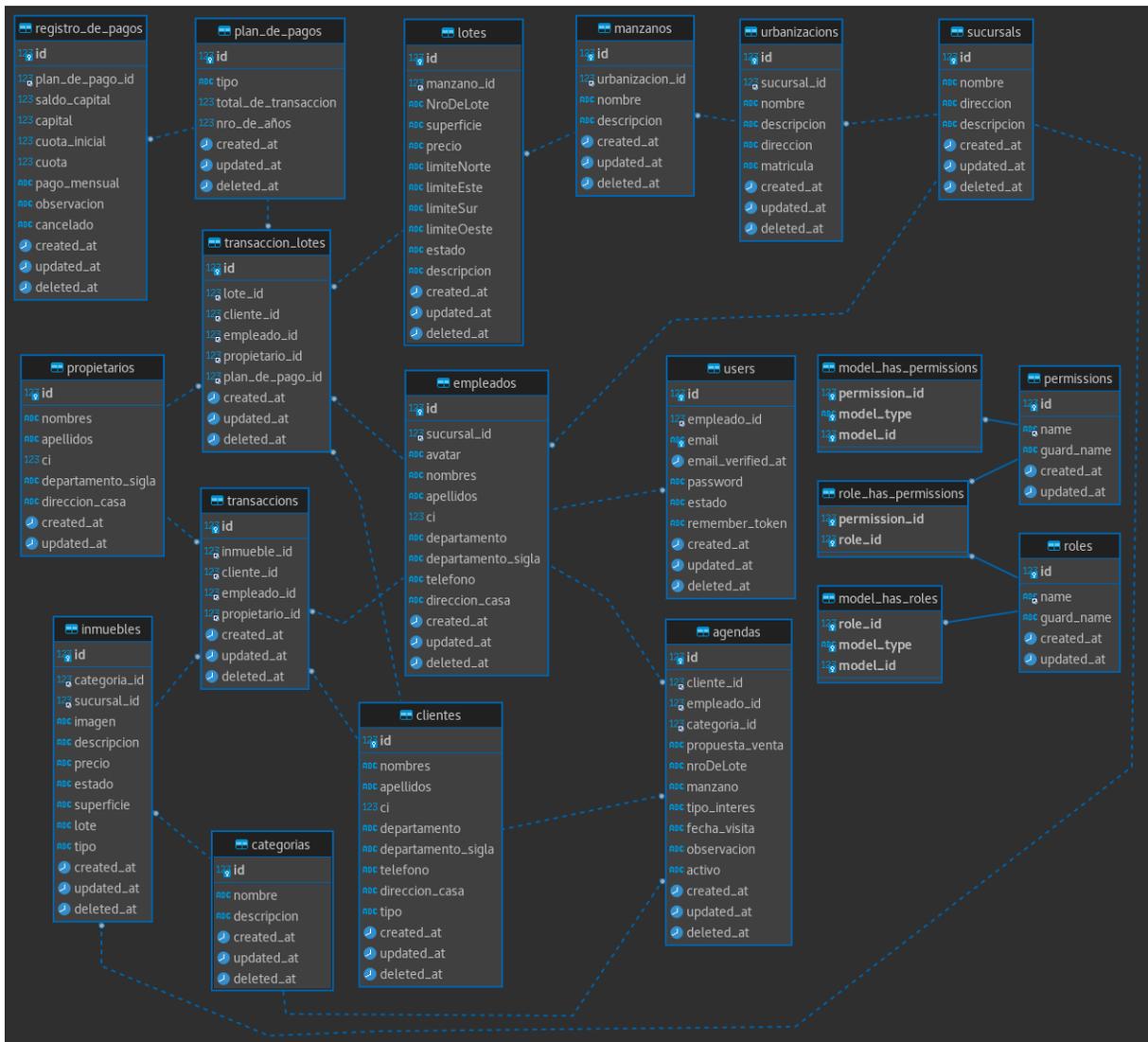
Fuente: [Elaboración propia, 2022]

3.1.2.4. Diseño conceptual

❖ Diagrama de clases

Figura 39

Diagrama de clases



Fuente: [Elaboración propia, 2022]

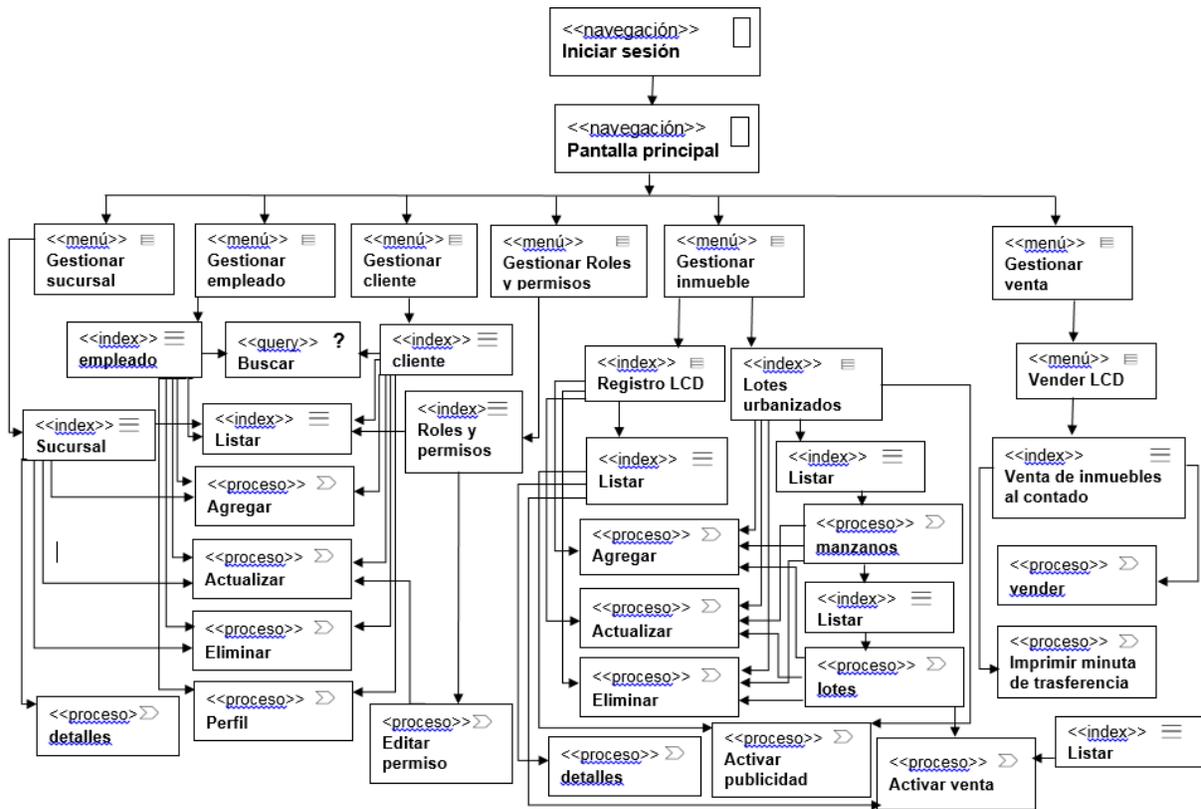
3.1.2.5. Modelo de Navegación

A continuación, se tiene la interacción de los usuarios con el software a través de la navegación por el sistema.

❖ Modelo de navegación – Gerente general

Figura 40

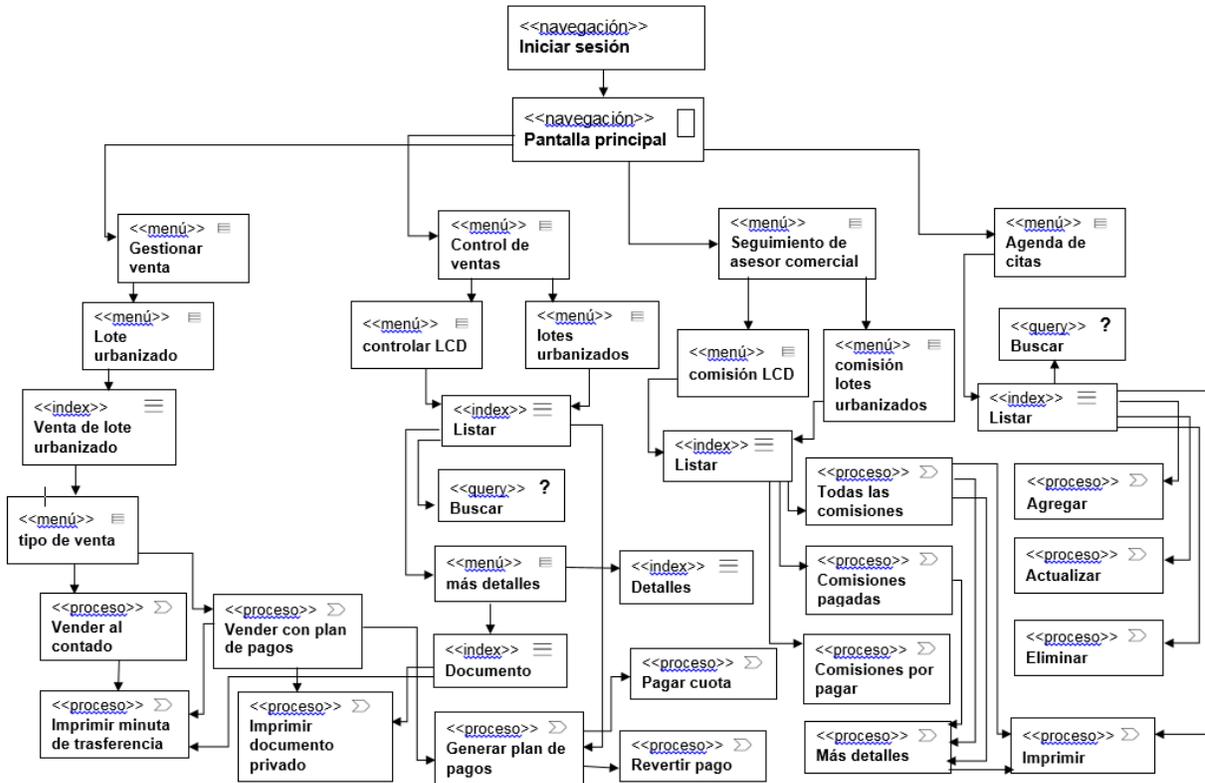
Modelo de navegación - Gerente general, A



Fuente: [Elaboración propia, 2022]

Figura 41

Modelo de navegación - Gerente general, B

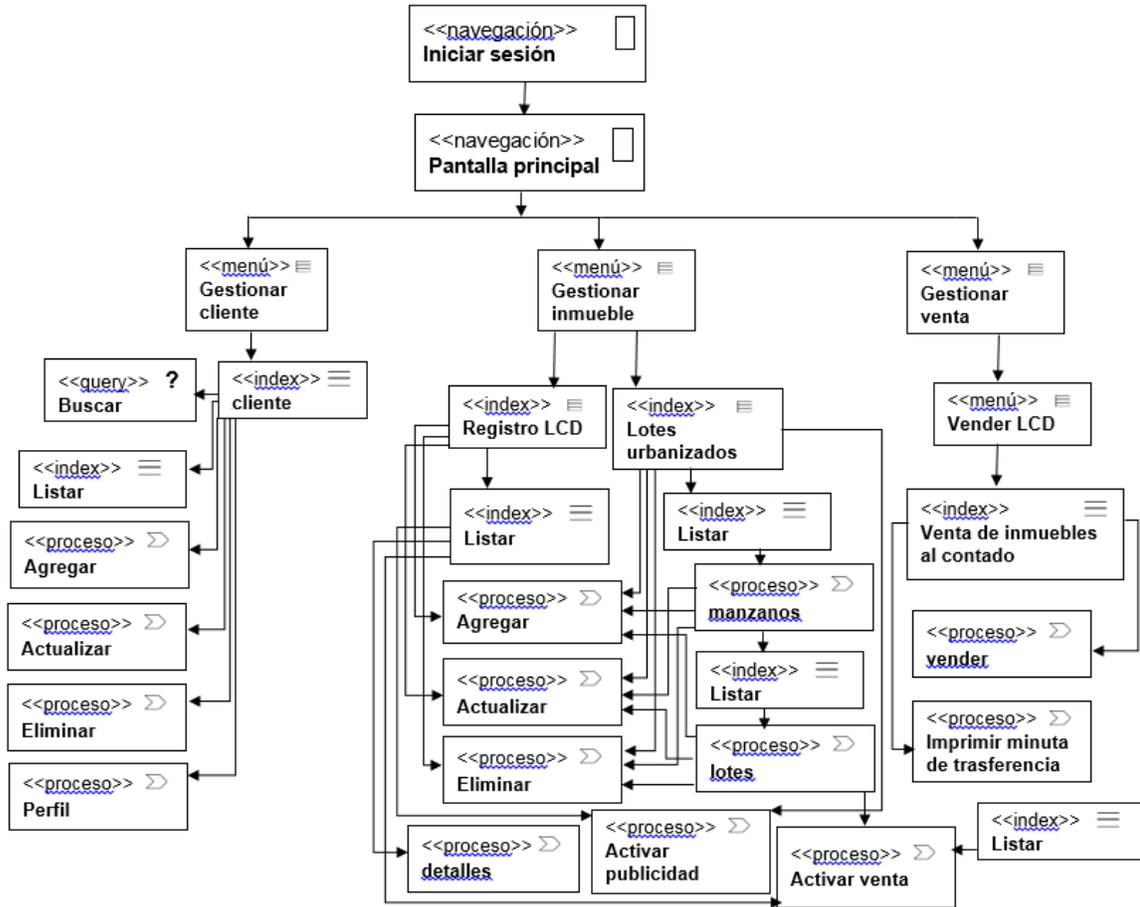


Fuente: [Elaboración propia, 2022]

❖ Modelo de navegación – Administrador

Figura 42

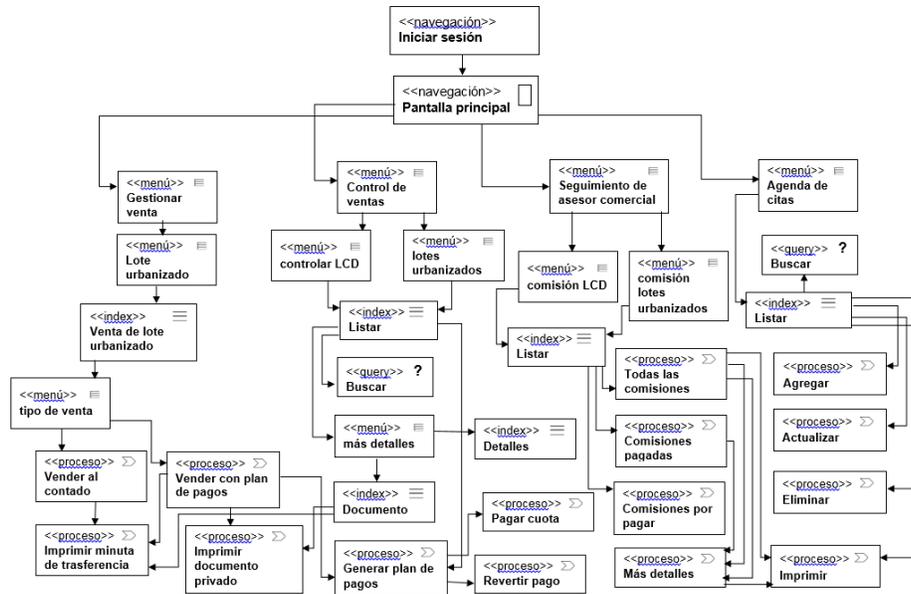
Modelo de navegación - Administrador, A



Fuente: [Elaboración propia, 2022]

Figura 43

Modelo de navegación - Administrador, B

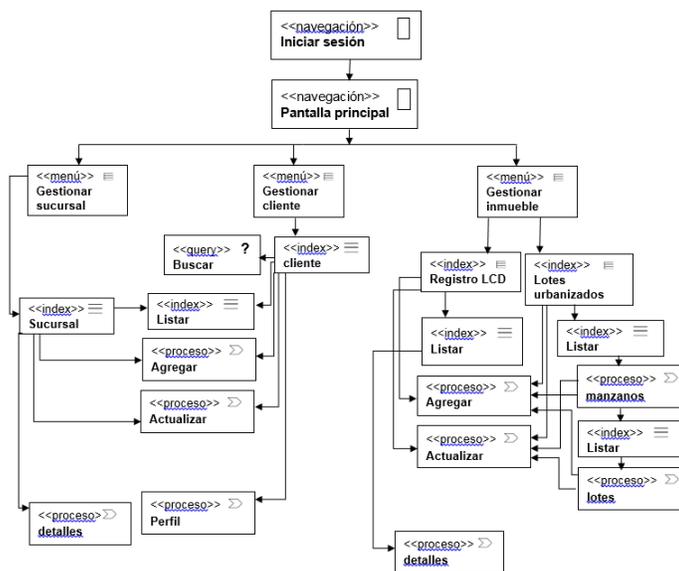


Fuente: [Elaboración propia, 2022]

❖ **Modelo de navegación – Jefe de agencias**

Figura 44

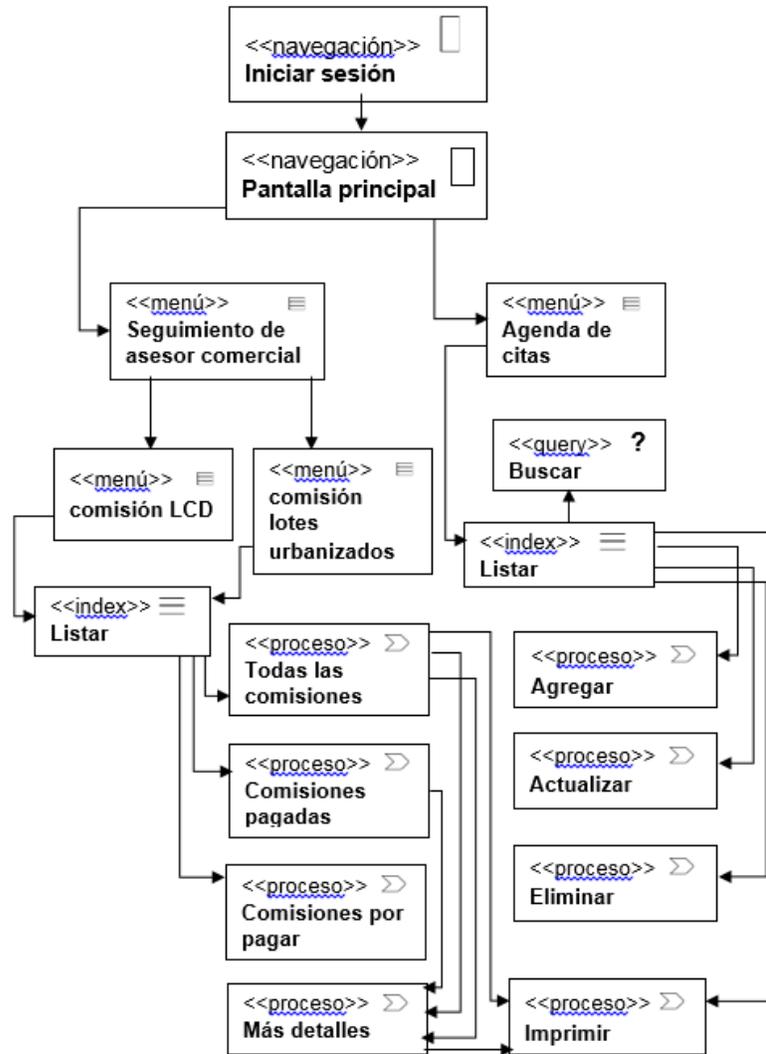
Modelo de navegación - Jefe de agencias, A



Fuente: [Elaboración propia, 2022]

Figura 45

Modelo de navegación - Jefe de agencias, B

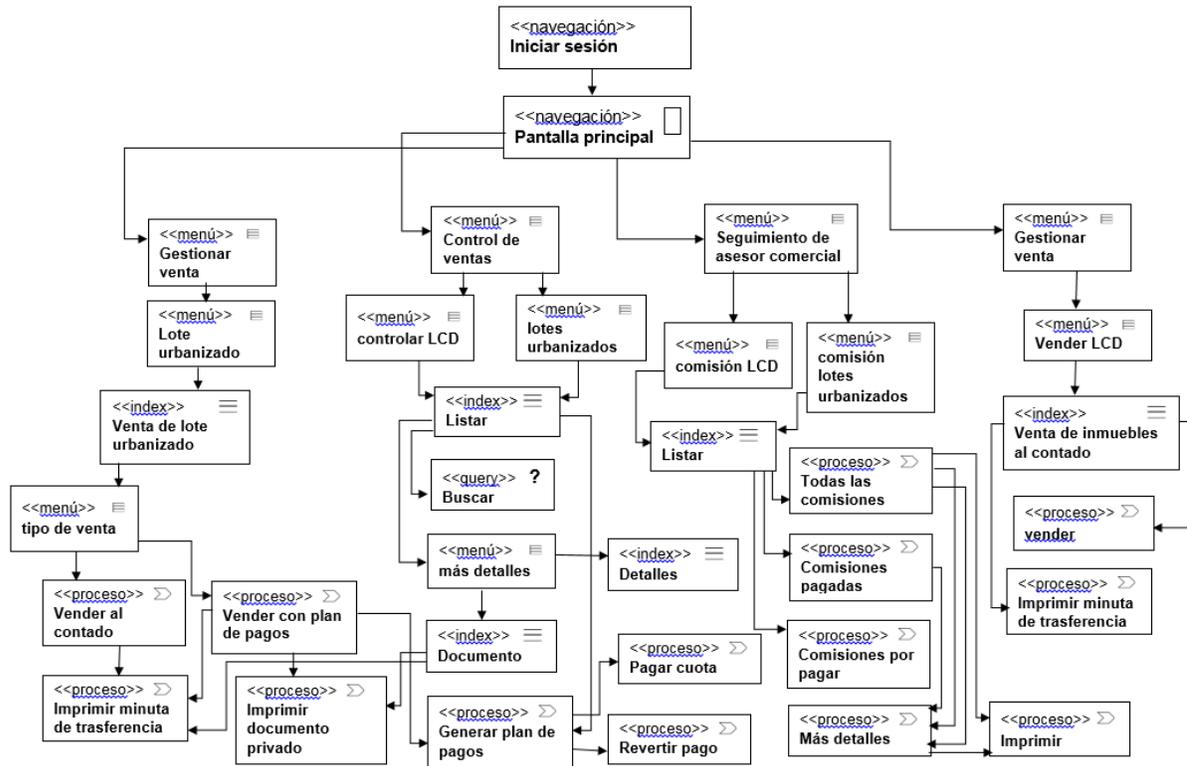


Fuente: [Elaboración propia, 2022]

❖ Modelo de navegación – Contador general

Figura 46

Modelo de navegación - Contador general

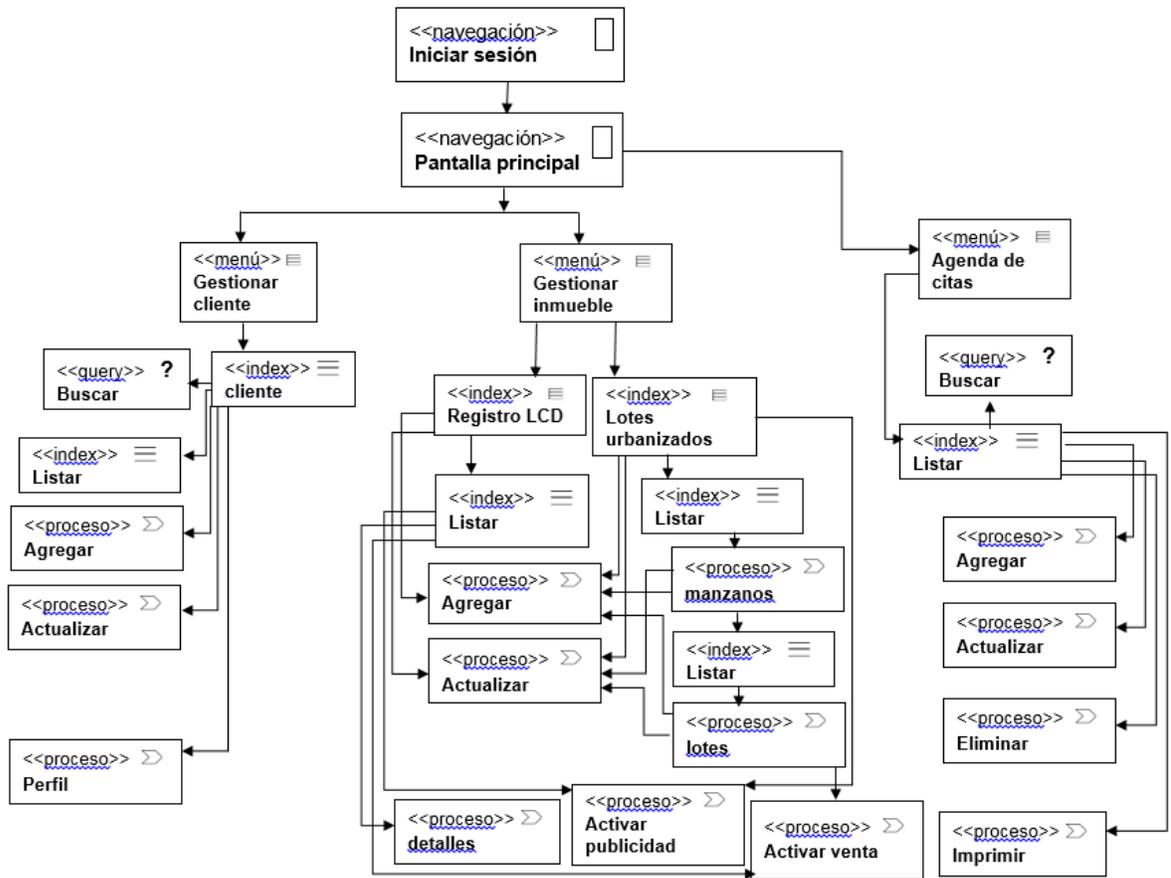


Fuente: [Elaboración propia, 2022]

❖ Modelo de navegación – Asesor comercial

Figura 47

Modelo de navegación - Asesor comercial

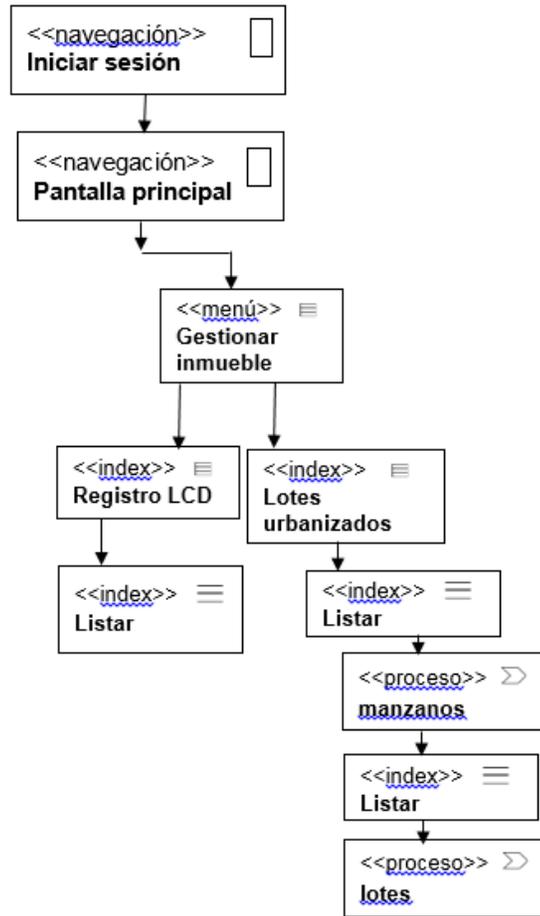


Fuente: [Elaboración propia, 2022]

❖ Modelo de navegación – Agente inmobiliario

Figura 48

Modelo de navegación - Agente inmobiliario

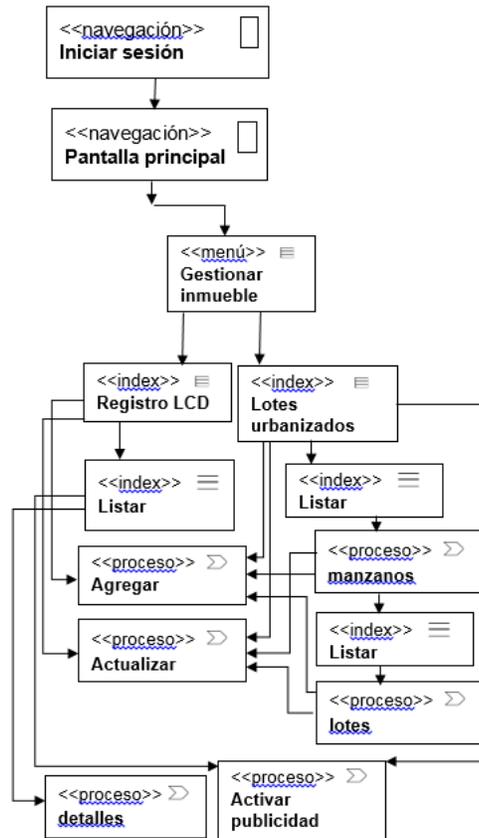


Fuente: [Elaboración propia, 2022]

❖ Modelo de navegación – Administrador de páginas web

Figura 49

Modelo de navegación - Administrador de aginas web



Fuente: [Elaboración propia, 2022]

3.1.2.6. Modelo de presentación

❖ Modelo de presentación - Inicio de sesión

Figura 50

Modelo de presentación - Inicio de sesión

Diagrama de la interfaz de inicio de sesión de JALHISA LTDA. El formulario contiene los siguientes elementos:

- Encabezado: JALHISA LTDA
- Título: Iniciar sesión
- Campo de texto: Usuario
- Campo de texto: Contraseña
- Botón: Iniciar sesión

Fuente: [Elaboración propia, 2022]

❖ Modelo de presentación – Sucursal

Figura 51

Modelo de presentación - Sucursal

Diagrama de la interfaz de gestión de sucursales de JALHISA LTDA. El formulario contiene los siguientes elementos:

- Encabezado: JALHISA LTDA
- Encabezado de la tabla: SUCURSAL
- Botón: agregar
- Tabla de sucursales:

			eliminar	detalles
			eliminar	detalles
			eliminar	detalles

Fuente: [Elaboración propia, 2022]

❖ **Modelo de presentación – Gestionar empleado**

Figura 52

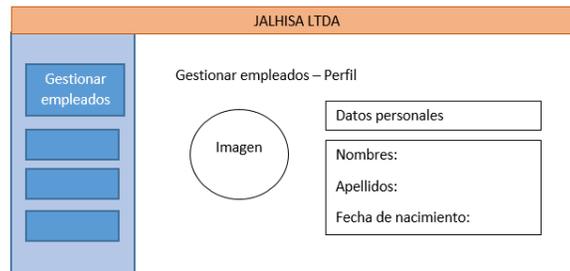
Modelo de presentación - Gestionar empleados



Fuente: [Elaboración propia, 2022]

Figura 53

Modelo de presentación - Gestionar empleado, perfil



Fuente: [Elaboración propia, 2022]

❖ **Modelo de presentación – Roles y Permisos**

Figura 54

Modelo de presentación - Roles y permisos



Fuente: [Elaboración propia, 2022]

❖ Modelo de presentación – Gestionar cliente

Figura 55

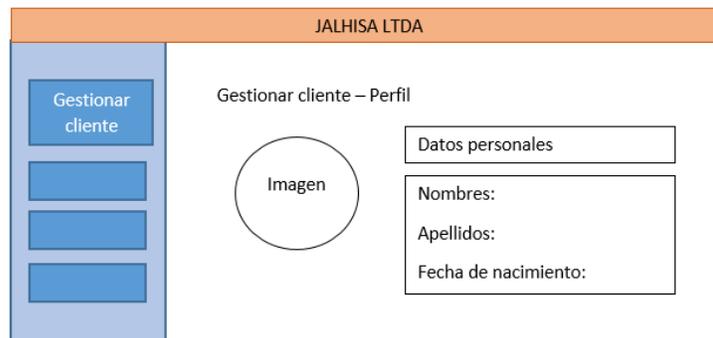
Modelo de presentación - Gestionar cliente



Fuente: [Elaboración propia, 2022]

Figura 56

Modelo de presentación - Gestionar cliente, perfil



Fuente: [Elaboración propia, 2022]

❖ **Modelo de presentación – Gestionar inmuebles**

Figura 57

Modelo de presentación - Gestionar inmueble, Registro LCD

JALHISA LTDA				
Gestionar inmuebles Registro LCD	Registro de LCD – lote, casa y edificio agregar			
	Todos los registros disponible Para la publicidad			
	Activar venta	Activar publicidad	eliminar	detalles
Activar venta	Activar publicidad	eliminar	detalles	
Activar venta	Activar publicidad	eliminar	detalles	

Fuente: [Elaboración propia, 2022]

Figura 58

Modelo de presentación -Gestionar inmuebles, lotes urbanizados

JALHISA LTDA				
Gestionar inmuebles Lotes urbanizados	Lotes urbanizados agregar			
	Activar publicidad	Manzano	eliminar	detalles
Activar publicidad	Manzano	eliminar	detalles	
Activar publicidad	Manzano	eliminar	detalles	

Fuente: [Elaboración propia, 2022]

Figura 59

Modelo de presentación - Gestionar inmuebles, lotes urbanizados, manzanos

JALHISA LTDA				
Gestionar inmuebles Lotes urbanizados	Lotes urbanizados / Manzano			agregar
		Lotes	eliminar	detalles
		Lotes	eliminar	detalles
	Lotes	eliminar	detalles	

Fuente: [Elaboración propia, 2022]

Figura 60

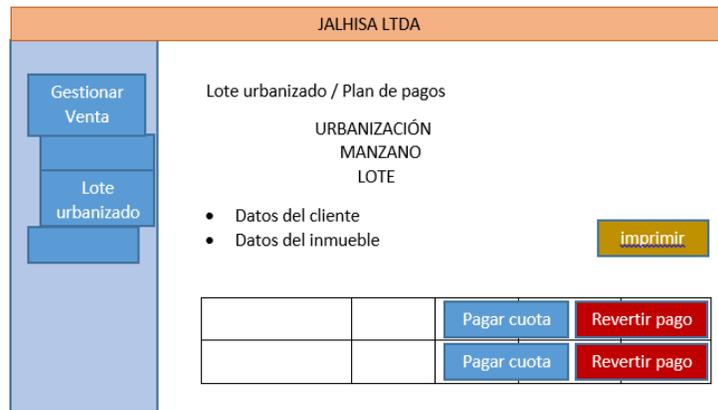
Modelo de presentación - Gestionar inmuebles, lotes urbanizados, manzanos, lotes

JALHISA LTDA				
Gestionar inmuebles Lotes urbanizados	Lotes urbanizados / Manzano / lotes			agregar
		Activar venta	eliminar	detalles
		Activar venta	eliminar	detalles
	Activar venta	eliminar	detalles	

Fuente: [Elaboración propia, 2022]

Figura 63

Modelo de presentación - Gestionar venta, lote urbanizado, plan de pago

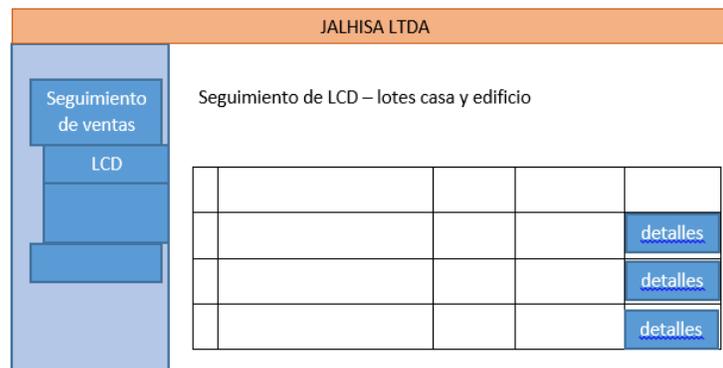


Fuente: [Elaboración propia, 2022]

❖ **Modelo de presentación – Seguimiento de Ventas**

Figura 64

Modelo de presentación - Seguimiento de ventas, LCD



Fuente: [Elaboración propia, 2022]

Figura 65

Modelo de presentación - Seguimiento de ventas, lotes urbanizados

JALHISA LTDA				
Seguimiento de ventas Lotes urbanizados	Control de venta – lotes urbanizados			
			Plan de pagos	detalles
			Plan de pagos	detalles
			Plan de pagos	detalles

Fuente: [Elaboración propia, 2022]

❖ **Modelo de presentación – Seguimiento de asesores comerciales**

Figura 66

Modelo de presentación - Seguimiento de asesores c., comisión LCD

JALHISA LTDA				
Seguimiento de asesores C Comisión LCD	Comisión LCD – lote, casa edificio			
	Todos las comisiones		Pagadas	Por pagar
				detalles
				detalles

Fuente: [Elaboración propia, 2022]

Figura 67

Modelo de presentación - Seguimiento de asesor comercial, Comisión lotes

Urbanizados

JALHISA LTDA					
Seguimiento de asesores C Comisión lotes Urb.	Comisión lotes Urbanizados				
	Todos las comisiones		Pagadas	Por pagar	
					detalles
					detalles
					detalles

Fuente: [Elaboración propia, 2022]

❖ **Modelo de presentación – Agenda de citas**

Figura 68

Modelo de presentación - Agenda de citas

JALHISA LTDA					
Agenda de citas	Agenda				
	Agendas de citas		Mi agenda de citas	agregar	
				eliminar	detalles
				eliminar	detalles

Fuente: [Elaboración propia, 2022]

3.1.3. CODIFICACIÓN

En esta etapa se realiza la tarea de programación; que consiste, esencialmente, en llevar a código fuente, todo lo diseñado en la fase anterior.

Figura 69

Codificación - Página principal del software

```
jalhisa > resources > views > jalhisa.blade.php
You, a few seconds ago | 1 author (You)
1 <!doctype html>
2 <html lang="es">
3 <head>
4 <<<<meta charset="UTF-8">
5 <<<<meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <<<<meta http-equiv="X-UA-Compatible" content="ie=edge">
7 <<<<title>Jalhisa L.T.D.A.</title>
8 <<<<link rel="manifest" href="/favicon/manifest.json">
9 <<<<link rel="icon" type="image/png" sizes="32x32" href="/favicon/jalhisa-icon3.png">
10 <<<<meta name="msapplication-TileColor" content="#ffffff">
11 <<<<meta name="msapplication-TileImage" content="/favicon/jalhisa-icon3.png">
12 <<<<meta name="theme-color" content="#ffffff">
13 <<<<link href="{{ asset('css/app.css') }}" type="text/css" rel="stylesheet" />
14 </head>
15 <body>
16 <<<<div id="app">
17 <<<<<< <app />
18 <<<</div>
19 <<<<<<<script src="{{ asset('js/manifest.js') }}"></script>
20 <<<<<<<script src="{{ asset('js/app.js') }}"></script>
21 </body>
22 </html>
```

Fuente: [Elaboración propia, 2022]

Figura 70

Codificaron - Controlador empleado

```
jalhisa > app > Http > Controllers > Empleado > EmpleadoController.php > ...
14 class EmpleadoController extends ApiController
15 {
16
17 <<<< public function index()
18 <<<< {
19 <<<<<<< $empleados = Empleado::all();
20 <<<<<<<
21 <<<<<<< return $this->showAll($empleados);
22 <<<<<<< }
23
24 <<<< public function store(Request $request)
25 <<<< {
26 <<<<<<< $data = $request->all();
27 <<<<<<< $data['avatar'] = $request->avatar->store('/empleadq');
28
29 <<<<<<<< if ($request->has('departamento sigla')) {
30 <<<<<<<<<< $data['departamento_sigla'] = $request->departamento_sigla;
31 <<<<<<<<<< switch ($request->departamento_sigla) {
32 <<<<<<<<<<<< case 'LP':
33 <<<<<<<<<<<<<<< $data['departamento'] = 'La Paz';
34 <<<<<<<<<<<<<<< break;
35 <<<<<<<<<<<<<<< case 'CH':
36 <<<<<<<<<<<<<<<<<< $data['departamento'] = 'Chuquisaca';
37 <<<<<<<<<<<<<<<<<< break;
38 <<<<<<<<<<<<<<<<<< case 'CB':
39 <<<<<<<<<<<<<<<<<<< $data['departamento'] = 'Cochabamba';
40 <<<<<<<<<<<<<<<<<<< break;
41 <<<<<<<<<<<<<<<<<<<<<< case 'OR':
42 <<<<<<<<<<<<<<<<<<<<<<<< $data['departamento'] = 'Oruro';
43 <<<<<<<<<<<<<<<<<<<<<<<<< break;
44 <<<<<<<<<<<<<<<<<<<<<<<<<<<< case 'PT':
```

Fuente: [Elaboración propia, 2022]

3.1.4. IMPLEMENTACIÓN DEL SISTEMA

En esta fase se tiene ya el producto final que viene a ser el software, con las diferentes interfaces graficas amigables. Para la interacción con el usuario final “el cliente”. El cual está listo para su uso

3.2. CALIDAD DEL SOFTWARE ISO-9126

3.2.1. Funcionalidad

No se puede medir directamente porque corresponde derivar medidas directas, como el punto función que cuantifica el tamaño y la complejidad del sistema en términos de las funciones del usuario.

Para la medición de las funciones y propiedades del sistema, se tomar como referencia las siguientes tablas y formulas.

- ❖ **Número de entradas de usuario:** es la cantidad de entradas de usuarios. Se tiene 55.
- ❖ **Número de salidas de usuario:** proporciona al usuario la información elaborada por el software para ser mostrada en pantalla. Se tiene 27.
- ❖ **Número de archivos:** son las tablas de la base de datos del software. Se tiene 20.
- ❖ **Número interfaces externas:** todas las interfaces legibles por el ordenador para ser utilizados en transmitir información. Se tiene 1.

Tabla 24*Calculo de la cuenta total con factores de ponderación*

Factores de ponderación					
Parámetros de medición	Cuenta	Simple	Medio	Complejo	Resultado
Número de entradas de usuario	55	3	4	6	220
Número de salidas de usuario	27	4	5	7	135
Número de archivos	20	7	10	15	200
Número de interfaces externas	1	5	7	10	7
Cuenta total					562

Fuente: [Elaboración propia, 2022]

Por lo tanto, la **cuenta total** con factores de ponderación es 562.

A continuación, calculo el factor de ajuste:

Tabla 25*Calculo de los valores de ajuste de complejidad*

IMPORTANCIA		0%	20%	40%	60%	80%	100%	
Escala	No influencia	Incidencias	Moderado	Medio	Significativo	Esencial		Nivel de influencia
Factor	0	1	2	3	4	5		
¿Requiere el sistema copias de						X		5

respaldo y recuperación?							
¿Se requiere comunicación de datos?				X			3
¿Existen funciones de procesamiento distribuido?			X				2
¿Es crítico el rendimiento?		X					1
¿Se ejecutara el software en un entorno operativo existente y frecuentemente utilizado?				X			3
¿Requiere el sistema entrada de datos en línea?					X		4
¿Requiere la entrada de datos interactiva que las transacciones de entrada se lleven a cabo sobre múltiples pantallas o variadas operaciones?			X				2
¿Se actualizara archivos maestros de forma interactiva?				X			3
¿Son complejas las entradas, salidas, archivos o peticiones?			X				2
¿Es complejo el procesamiento interno?						X	5
¿Se diseña el código para ser reutilizable?				X			3
¿Están incluidas en el diseño la conversión y la instalación?			X				2
¿Se ha diseñado el sistema para soportar múltiples instalaciones?						X	5
¿Se ha diseñado para facilitar cambios y ser fácilmente utilizada por el usuario?					X		4
Total $\sum(F_i)$							44

Fuente: [Elaboración propia, 2022]

◆ Factor de ajuste

$$\text{Factor de ajuste} = \left(\sum (Fi) * 0,01 \right) + 0,65$$

$$\text{Factor de ajuste} = (44 * 0,01) + 0,65$$

$$\text{Factor de ajuste} = 1,09$$

◆ Punto de Función (PF)

Donde, **cuenta total = 452**

$$PF = \text{cuenta total} * \left(\left(\sum (Fi) * 0,01 \right) + 0,65 \right)$$

$$PF = 562 * 1,09$$

$$PF = 612,58$$

◆ Punto de Función máximo (PF')

Donde el PF', alcanza $\sum(Fi) = 70$

$$PF' = \text{cuenta total} * \left(\left(\sum (Fi) * 0,01 \right) + 0,65 \right)$$

$$PF' = \text{cuenta total} * ((70 * 0,01) + 0,65)$$

$$PF' = 562 * 1,35$$

$$PF' = 758,7$$

◆ Funcionalidad del software

$$\text{Funcionalidad} = \left(\frac{PF}{PF'} \right) * 100$$

$$\text{Funcionalidad} = \left(\frac{612,58}{758,7} \right) * 100$$

$$\text{Funcionalidad} = 80,7\%$$

Por lo tanto, la funcionabilidad del software es de 80,7%

➤ **Fiabilidad:** o confiabilidad del software.

Para calcular el índice de error de Lambda (λ) = $(1/10) = 0,1$; ya que se define de cada 10 ejecuciones 1 fue fallida en un periodo de una semana, durante un 2 meses. Donde $t = 2$ meses que equivale a $t = 8$ semanas y $PF = 0.807$

Para encontrar la fiabilidad del sistema se utilizó las siguientes formulas:

Probabilidad de fallos

$$\begin{aligned}P(T \leq t) &= F(t) \\F(t) &= PF * e^{-\lambda * t} \\F(t) &= 0.807 * e^{-(0.1 * 8)} \\F(t) &= 0,36 * 100 \\F(t) &= 36\%\end{aligned}$$

Probabilidad de éxito

$$\begin{aligned}P(T \leq t) &= 1 - F(t) \\F(t) &= 1 - (PF * e^{-\lambda * t}) \\F(t) &= 1 - (0.36) \\F(t) &= 0,64 * 100 \\F(t) &= 64\%\end{aligned}$$

Por lo tanto, se tiene una confiabilidad del 64% por parte del software. Lo que nos indica una buena estabilidad y funcionamiento.

➤ **Usabilidad:** Para determinar la usabilidad del software se realizó un cuestionario relacionado con el uso del software a 6 personas, mediante una serie de preguntas y se obtuvo los siguientes resultados.

Tabla 26*Encuesta sobre la usabilidad*

	Preguntas	SI	NO	R
1	¿Puede utilizar con facilidad el sistema?	6	0	1
2	¿La interfaz gráfica es de su agrado?	6	0	1
3	¿El sistema realizo las solicitudes de manera rápida?	5	1	0.86
4	¿El sistema le ayuda a realizar su trabajo de una manera más rápida?	6	0	1
5	¿Puede fácilmente navegar por las distintas opciones que tiene el sistema?	6	0	1
6	¿El sistema cuenta con interfaces agradables a la vista?	4	2	0.667
7	¿La respuesta del sistema es satisfactorio?	5	1	0.86
8	¿La operaciones que realizo son fáciles?	5	1	0.86
USABILIDAD				0.90

Fuente: [Elaboración propia, 2022]

$$U = \left(\frac{\sum xi}{n} \right) * 100$$

$$U = \left(\frac{7,25}{8} \right) * 100$$

$$U = 90\%$$

Por lo tanto, se tiene una usabilidad del 90%

- **Facilidad de mantenibilidad:** Para hallar la mantenibilidad tomemos las siguientes variables y formula:

Mt: cantidad de módulos en la versión actual.

Fa: cantidad de módulos que se ha agregado en la versión actual.

Fc: cantidad de módulos que se ha cambiado en la versión actual.

Fd: cantidad de módulos que se ha eliminado en la versión actual.

Se obtuvo los siguientes valores: **Mt = 8, Fc = 0, Fa = 0, Fd = 0**

$$M = \frac{(Mt - (Fa + Fc + Fd))}{Mt}$$
$$M = \frac{(8 - (0 + 0 + 0))}{8}$$
$$M = 1 * 100$$
$$M = 100\%$$

Por lo tanto, el software tiene una facilidad de mantenibilidad del 100%.

3.3. SEGURIDAD DEL SOFTWARE ISO – 27001

En el proceso del desarrollo del software se implementó seguridad informática tanto para la parte del cliente (FrontEnd) y el servidor (BackEnd). Con el propósito de resguardar los datos e información.

➤ **Del lado del cliente:** se aplicó medidas de seguridad en el FrontEnd, tales como:

- Acceso al sistema por parte del usuario, según roles y permisos.
- Validación de datos en los formularios que contiene el sistema.
- Autenticación de la página web HTML a través de token.

➤ **Del lado del servidor:** se aplicó medidas de seguridad en el BackEnd, tales como:

- Encriptación de la contraseña en la base de datos, a través de **Bcrypt**¹⁷.

¹⁷ Bcrypt: Método para encriptar un determinado dato

- Para transmitir información de forma segura entre el cliente y servidor se implementó **JWT**¹⁸.
- Autenticación por token, cada vez que el usuario inicia sesión, se genera una llave única para acceder a los datos e información del sistema de forma segura.
- El acceso a las rutas del sistema están delimitadas por los roles y permisos del usuario.

3.4. MEDICIÓN Y ESTIMACIÓN DEL SOFTWARE MÉTODO COSMIC

Para la medición y estimación de costo del software con el método COSMIC, se determinara a través de:

- **Requerimientos funcionales:** proporciona los Puntos de Función COSMIC (CFP), el sistema cuenta con 27 CFP. Los cuales se describen en la Tabla 6.
- **Costo del equipo de trabajo:** para tener el costo de desarrollo de una unidad de medición del tamaño del software, tomemos como referencia organizaciones que se dedican al desarrollo de software.

Organizaciones dedicadas al desarrollo de software como CyberGlobalNet, Fidooo Engineering y AGETIC tienen un rango de sueldo entre Bs.- 3.000 a 13.500 para un desarrollador de software o un programador de software.

Para la medición y estimación de costo del software, tomemos como referencia las siguientes formulas:

- **Costo por Punto de Función COSMIC**

- ◆ CFP: Punto de Función COSMIC = 27 CFP

¹⁸ JWT: JSON Web Token,

- Costo mes del equipo de trabajo: sueldo promedio del desarrollador de software = Bs.- 3.000
- CFP mes: Punto de Función COSMIC por mes en un año

$$\text{CFP mes} = \frac{\text{CFP}}{12} = \frac{27}{12} = 2,25$$

$$\text{Costo por CFP} = \frac{\text{Costo mes del equipo de trabajo}}{\text{CFP mes}} = \frac{3.000\text{Bs}}{2,25} = 1.333,33 \text{ Bs.}$$

➤ **Tamaño del software:**

- Tamaño del software: se determina por el número de Puntos de Función COSMIC (CFP) = 27

$$\begin{aligned} \text{Consto del proyecto de software} &= \text{tamaño del software} * \text{Costo por CFP} \\ \text{Consto del proyecto de software} &= 27 * 1.333,33 \text{ Bs.} = 35.999,91 \text{ Bs.} \end{aligned}$$

➤ **Tiempo de duración del proyecto en meses:**

$$\text{Duración del proyecto} = \frac{\text{CFP}}{\text{CFP mes}} = \frac{27}{2,25} = 12 \text{ meses}$$

Por lo tanto el costo del software equivale a Bs.- 35.999,91, con una duración de 12 meses.

CAPITULO IV

CONCLUSIÓN Y RECOMENDACIÓN

4.1. CONCLUSIÓN

Se logró realizar exitosamente el objetivo del proyecto de grado con el desarrollo del software titulado “Sistema de Gestión Inmobiliaria”.

- Se obtuvo los requerimientos funcionales y no funcionales en el diagnóstico de la estructura de la Inmobiliaria.
- Se realizó la integración de la información en una base de datos.
- Se sistematizó el proceso de seguimiento de asesores comerciales.
- Se automatizó el control y seguimiento de la venta y servicio de publicidad de inmuebles.
- Se sistematizó el proceso de plan de pagos.
- Se automatizó la agenda de citas.
- Se realizó la Integración de los datos e información de los clientes e inmuebles.
- Se realizó la página de presentación de publicidad.

Es de esta manera que el software realiza de manera eficiente los diferentes procesos en la gestión de venta, alquiler y anticrético de inmuebles, ya que el modelo de negocio de la empresa ha sido automatizado y los datos e información se encuentran en una base de datos, lo cual permite a la parte administrativa de la empresa realizar los procesos con una mayor facilidad. A continuación, los beneficios que se han logrado realizar:

- Se cuenta con un software escalable, ya que es posible aumentar o modificar los módulos.

- Se cuenta con un software que tiene la capacidad de estar en funcionamiento 24 horas del día.
- Se cuenta con un software que es un Sistema de Información, el cual permite el acceso a los datos e información de los inmuebles de la empresa, de manera rápida y precisa.
- Se cuenta con un software que genera reportes, facilitando la salida de información en cualquier momento.
- El software cuenta con mecanismos de seguridad y autenticación de usuarios. Para el resguardo de la información y los datos de intrusos que quieran acceder al sistema.

Es de esta manera que el software ayuda en los procesos de venta, alquiler, anticrético y publicidad de inmuebles, en un periodo de tiempo mínimo.

4.2. RECOMENDACIONES

El software en la parte del BACK-END tiene la posibilidad de interactuar con otras aplicaciones web y móviles, ya que es una API-REST y puede ser consumido sus recursos. Por otras aplicaciones. Se recomienda realizar:

- El mantenimiento del software en determinados periodos de tiempo, tomando en cuenta el reglamento interno de la empresa.
- Los datos e información que sean introducidos al sistema. Sean datos fiables para que los reportes generados y la información dentro del sistema sean confiables para la administración y toma de decisiones.
- Es importante realizar las copias de seguridad cada mes o en su defecto dependiendo del reglamento interno de la empresa, con el propósito de evitar la pérdida de información.

BIBLIOGRAFÍA

- [Kendall y Kendall, 2011] Kendall, Kenneth y Kendall, Julie. Análisis y Diseño de Sistemas. Camden, New Jersey. USA.
- [Pressman, 2010] Pressman Roger S. Ingeniería de software, un enfoque práctico. USA.
- [López, 2020] López Q. José. Laravel. España
- [Bahit, 2012] Bahit Eugenia. El lenguaje PHP. Argentina
- [Gauchat, 2012] Gauchat Juan D. El gran libro de HTML5, CSS3 y Javascript. España.
- [Dongil, 2018] Dongil S. Jose A. Desarrolla aplicaciones con Vuejs. España
- [ISOTools, 2013]. La norma ISO 27001. Colombia, Bogotá.
- [American, 2020] American Psychological Association. Guía resumen del Estilo APA séptima edición.
- [Bermudez, 2008] Bermudez, C. Nataly. Sistema de control inmobiliario. Colombia, Bogotá.
- [Rincón, 2013] Rincón D. Rafael, Sistema de gestión de calidad basado en la norma NTC-ISO9001:2008 para la empresa inmobiliaria e ingeniería AJC LTDA. Colombia.
- [Pulido, 2018] Pulido S. Cristian. Desarrollo e implementación de un sistema de información para la gestión de los procesos de la inmobiliaria REX/MAX PLATINUM. Perú.
- [Sarco, 2017] Sarco M. Mónica. Sistema de control de compra, venta e inventarios. La Paz, Bolivia.

- [Forra, 2020] Forra L. Armin R. Sistema web para la venta de ladrillos y control de personal. La Paz, Bolivia.
- [Cuellar, 2016] Cuellar R. Elías O. Diseño de un sistema de gestión para la administración y control de proyectos de construcción basado en la metodología del PMBOK. Sucre, Bolivia.
- [Guanolema, 2019] Guanolema Ch. Luis. Desarrollo de un sistema web para automatizar el proceso de compra y venta en la microempresa RAZA. Ecuador.
- [Latorre y Castañeda 2016] Latorre V. Axel, Castañeda C. Jorge L. Desarrollo e implementación de sistema de administración de ventas para Humane Escuela de Negocios para optimizar los procesos de gestión de ventas de servicios académicos. Ecuador.
- [Rivera, Rodríguez, Sanchez y Sanchez, 2011] Rivera V. Luis de Jesus, Rodriguez M. Vladimir, Sanchez G. Diana Carolina y Sanchez M. Fatima Patricia. Sistema informático para la administración académica.
- [Montano, Camargo, Martínez, Parra, Prada, Suárez, 1995] Cabuya M. Luis, Camargo C. Bernardo, Martínez F. Carlos, Parra R. Bernardo, Prada B. Luis, Suárez E. Rafael. ORGANIZACIÓN Y GESTIÓN DE EMPRESAS. TÉCNICAS MODERNAS DE GERENCIA. Santa Fe de Bogotá.
- [García, García y Vázquez, 2020] García P Francisco, García H. Alicia y Vázquez I. Andrea. INGENIERÍA DE SOFTWARE I. Universidad Salamanca
- [Vallejos, Larios y Sandoval, 2016] Vallejos M. Santiago, Larios C. Jerald y Sandoval H. Álvaro, Servicios web. Nicaragua.
- [Cobo, Gómez, Pérez y Rocha, 2005] Cobo Ángel, Gómez Patricia, Pérez Daniel y Rocha Roció. PHP y MySQL. España.
- [Eguiluz, 2009] Eguiluz P. Javier. Introducción a JavaScript.

- [Kenneth, 2012] Kenneth C. Laudon. Sistema de información gerencial.
- [Gonzalez, 2015] Francisco M. Gonzalez L. Introducción a los Sistemas de información.
- [AcensTechnologies, 2015] AcensTechnologies. Framework para el desarrollo ágil de aplicaciones.
- [Stack Overflow Contributors, 2015] Stack Overflow Contributors. Aprendiendo Vue.js.
- [Javier Eguíluz Pérez, 2008] Eguíluz P. Javier. Introducción a CSS.
- [Scalone, 2006] Scalone Fernanda. Estudio comparativo de los modelos y estándares de calidad del software. Argentina.
- [Vazquez, 2015] Vazquez Carlos E. Estimaciones de software con COSMIC. Brasil.
- [Cadme y Duque, 2012] Cadme R. Christian y Duque P. Diego. Auditoria de seguridad informática ISO 27001. Ecuador.
- [Amodeo, 2013] Enrique Amodeo. CA Tecnologías. Principios de diseño de API REST. USA.
- [García, Lozano, Mendoza, 2016] García G. Laura, Lozano S. Jorge Luis, Mendoza L. Nicolás. Servicios web – aplicaciones empresariales. Colombia
- [Bonilla, 2021] Bonilla A. Alexander D. Aplicación web SPA para la gestión de fichas médicas en el hospital universitario andino utilizando servicios REST. Ecuador.
- [Platero y Soñez, 2019] Platero Leandro, Soñez Mailin. API REST con acceso por JWT. Argentina.
- [Rossainz, 2018] Rossainz L. Mario. UWE (UML-Based Web Engineering)

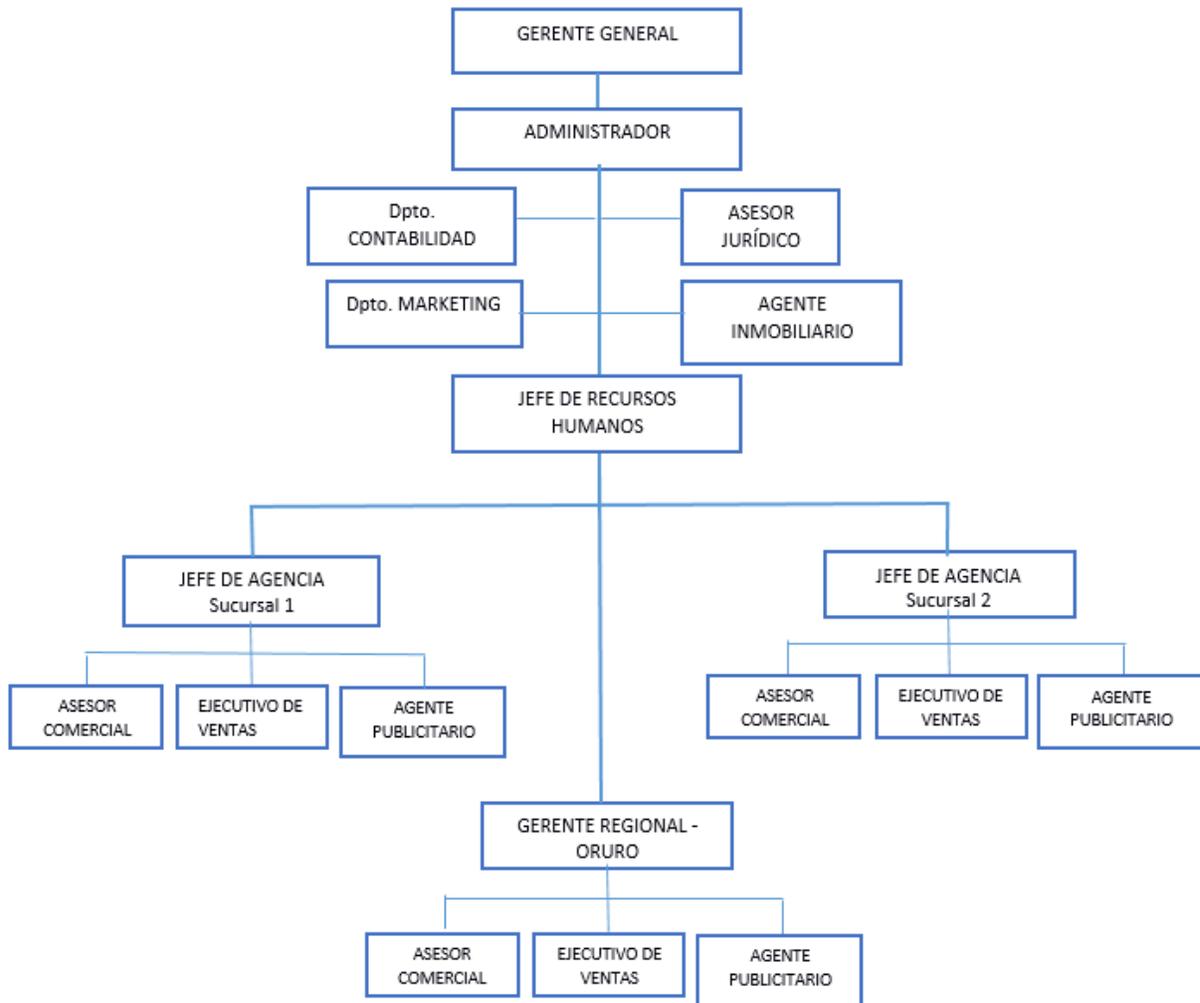
- [Arias, 2020] Arias F. Jesus. Protocolo HTTP - Aplicación web. España.

ANEXOS

ANEXO A: ORGANIGRAMA

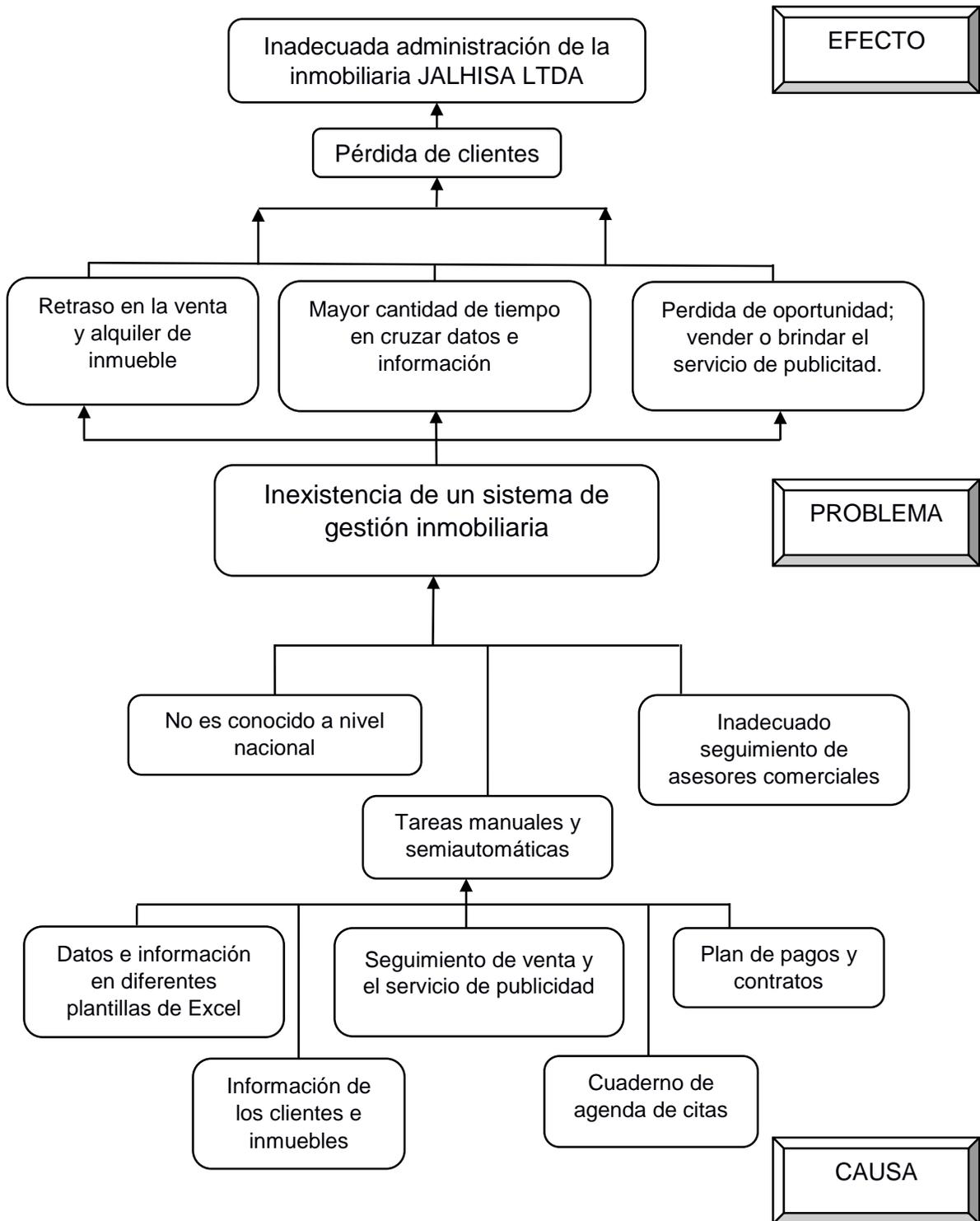
Figura 71

Organigrama de la empresa JALHISA LTDA.

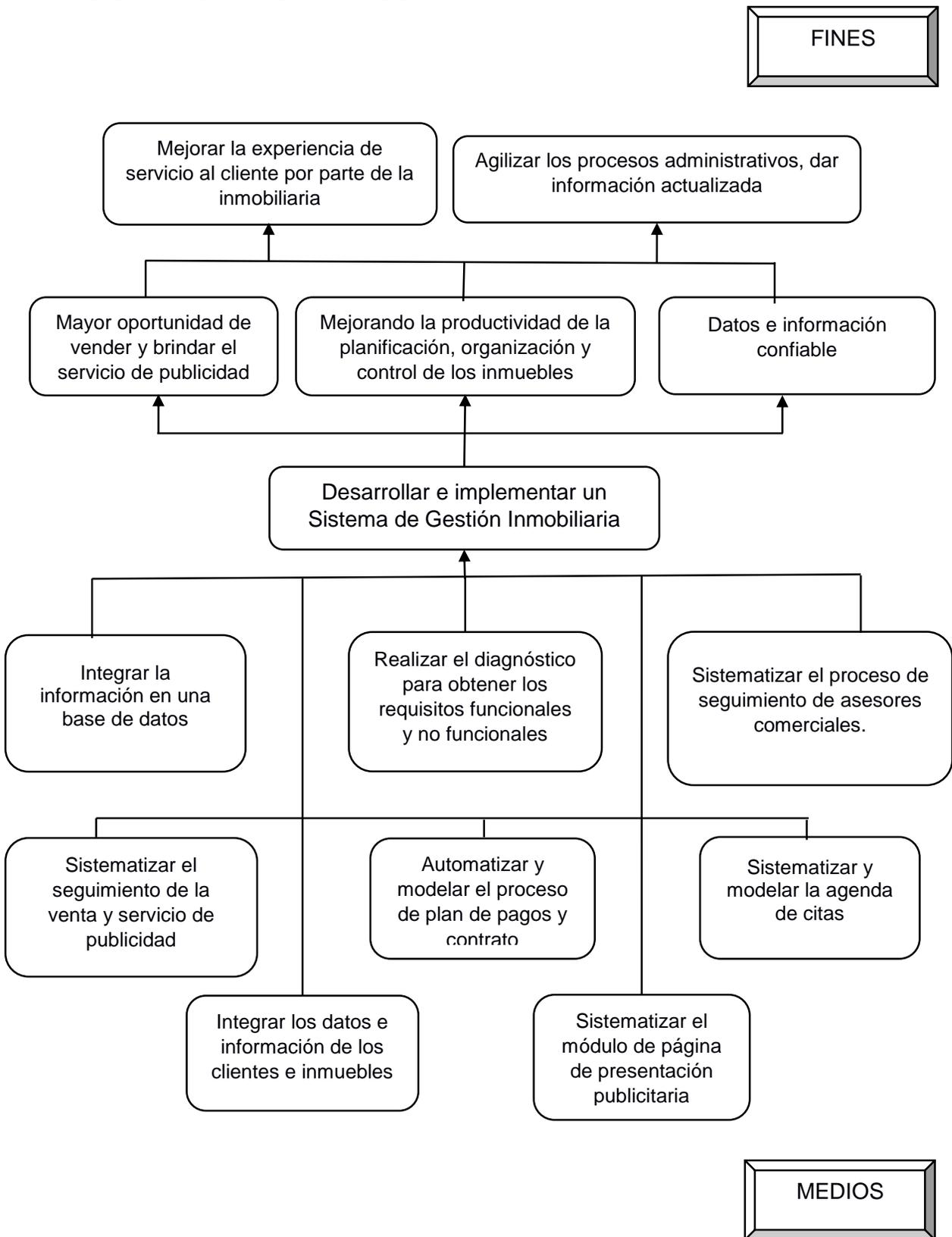


Fuente: [JALHISA LTDA.]

ANEXO B: ÁRBOL DE PROBLEMAS



ANEXO C. ÁRBOL DE OBJETIVOS



ANEXO D. ENTREVISTA

ENTREVISTA

1. ¿A qué se dedica la empresa JALHISA LTDA.?

R.-

2. ¿Cuáles son las actividades importantes de la empresa JALHISA LTDA.?

R.-

3. ¿Dónde se encuentra ubicado la empresa JALHISA LTDA.?

R.-

4. ¿Tiene sucursales la empresa JALHISA LTDA.?

R.-

5. ¿Cuál es la MISIÓN de la empresa JALHISA LTDA.?

R.-

6. ¿Cuál es la VISIÓN de la empresa JALHISA LTDA.?

R.-

8. ¿Cuál es el rol del Gerente General, dentro de la empresa JALHISA LTDA.?

R.-

9. ¿Cuál es su función el Gerente General, dentro de la empresa JALHISA LTDA.?

R.-

10. ¿Cuál es su función el Administrador, dentro de la empresa JALHISA LTDA.?

R.-

11. ¿Cuál es su función el Jefe de agencia, dentro de la empresa JALHISA LTDA.?

R.-

12. ¿Cuál es su función el Contador general, dentro de la empresa JALHISA LTDA.?

R.-

13. ¿Cuál es su función el Asesor comercial, dentro de la empresa JALHISA LTDA.?

R.-

14. ¿Cuál es su función el Administrador de paginas web y redes sociales, dentro de la empresa JALHISA LTDA.?

R.-

15. ¿Cuál es su función el Agente inmobiliario, dentro de la empresa JALHISA LTDA.?

R.-

16. ¿Cómo se realiza el registro de los inmuebles?

R.-

17. ¿Cómo se realiza el plan de pagos de un inmueble?

R.-

ANEXO E. ENCUESTAS

ENCUESTA

La encuesta se realizó al personal de la empresa JALHISA LTDA, como ser: Gerente general, Administrador, Jefe de agencia, Contador general, Asesor Comercial, Administrador de páginas web y redes sociales y Agente inmobiliario.

1. ¿Es necesario automatizar la información, para tener una mejor administración de la información?

- R.- Si, el 85%
No, el 10%
No sabe no responde, el 5%

2. ¿Los datos e información se encuentran más seguro en el sistema de información “software”?

- R.- Si, el 80%
No, el 18%
No sabe no responde, el 2%

3. ¿Le ayudaría a su trabajo que el sistema de información “software”, genera reportes e informes?

- R.- Si, el 95%
No, el 4%
No sabe no responde, el 1%

4. ¿Le ayudaría a su trabajo que el sistema de información “software”, le proporcione datos e información en el momento que lo requiera?

- R.- Si, el 97%
No, el 2%
No sabe no responde, el 1%

ANEXO F. DOCUMENTOS DE LA EMPRESA



DOCUMENTO PRIVADO

Conste por el presente Documento privado **DE RESERVA DE DERECHO PROPIETARIO POR LA COMPRA VENTA DE UN LOTE DE TERRENO**, de conformidad a los artículos 519, 521, 1297, 1298 del Código Civil Boliviano, que surtirá los efectos legales de instrumento público a sólo reconocimiento de firmas, las siguientes cláusulas: -----

Primera. - (PARTES). Suscriben el presente Documento: -----

1) Señor **TOMÁS LENZ BUSTINZA** con Cedula de Identidad número 250494 expedido en La Paz y Señora **WALLIA ROSARIO TELLERÍA DE LENZ**, con Cedula de Identidad número 237463 ambos domiciliados en la calle Héctor Ormachea No 5115, Zona Obrajes, La Paz, se constituyen como **VENEDORES**. -----

2) Señor **LUIS RODRIGO CUSSI CHIQUE** con Cedula de Identidad número 8346742 expedido en La Paz, domiciliado en la Av. Panorámica No. 38, zona Final Rosas Pampa, se constituye como **COMPRADOR**. -----

Segunda. - (DERECHO DE PROPIEDAD). Los **VENEDORES** son propietarios del lote de terreno N° 11, manzano "E" de la Urbanización "El Cruce", situada en el Distrito 10 del Municipio de El Alto, con una superficie de 300 m² (TRESCIENTOS METROS CUADRADOS), que limita al Norte con Lote "10", al Este con la Avenida "F", al Sur con la Calle 12, al Oeste con el lote N° 6, de dicha Urbanización; inscrito en los **Registros de Derechos Reales con matrícula No 2.01.4.01.0239340**. -----

Tercera. - (OBJETO). Los **VENEDORES** en su condición de propietarios de dicho terreno, otorgan en calidad de Reserva de Derecho Propietario, el mencionado lote de terreno a los **COMPRADORES**. -----

Cuarta. - (PRECIO y FORMA DE PAGO). El precio acordado entre partes es de \$us 9.400.- (NUEVE MIL CUATROCIENTOS 00/100 DÓLARES AMERICANOS) y se establece un anticipo de \$us. 1.000 (MIL 00/100 DOLARES AMERICANOS) que los **COMPRADORES** entregan a los **VENEDORES** a la firma del presente Documento. El saldo es de \$us 8.400.- (OCHO MIL CUATROCIENTOS 00/100 DÓLARES AMERICANOS) suma de dinero que el comprador se compromete a cancelar en 48(cuarenta y ocho) cuotas mensuales de \$us 175.- (CIENTO SETENTA Y CINCO 00/100 DOLARES AMERICANOS, computables a partir de la firma del presente Documento, hasta terminar de cancelar recesivamente según plan de pagos. -----

Quinta. - (OBLIGACIONES). EL **COMPRADOR** asumen la obligación de honrar el pago de las cuotas señaladas en la cláusula anterior, en las fechas establecidas. Y cubrir los impuestos municipales de las gestiones a partir del año 2021. Los **VENEDORES** tienen la obligación de cubrir dichos impuestos hasta la gestión 2020; y realizar la transferencia legal y de posesión definitiva del lote de terreno objeto del Contrato al **COMPRADOR**, una vez cancelado el total del saldo establecido. -----

Sexta. - (RECISIÓN). En caso de incumplimiento a la amortización de las cuotas por parte del comprador, se disipará un 50% del monto entregado por los **COMPRADOR** a favor de

MANUAL

MANUAL DE USUARIO

En esta sección veremos como el usuario puede interactuar con el aplicativo, dando énfasis en el manejo; navegando por las diferentes interfaces gráficas.

❖ **Inicio de sesión:** acceso al sistema



Para ingresar al sistema el usuario debe introducir dos datos muy importantes, usuario y contraseña

- ❖ **Usuario:** el usuario consta de solo el número de cedula de identidad, ejemplo: 9921457
- ❖ **Contraseña:** la es la fecha de nacimiento del usuario en el formato día-mes-año, veamos el siguiente ejemplo: 02-06-2022

Después de introducir el usuario y la contraseña presione el botón “iniciar sesión” para acceder al sistema. También es posible presionar la tecla **Enter**, después de haber escrito la contraseña.

A continuación, la interfaz panel de control.

❖ **Panel de control:** pantalla principal del sistema



- a) Barra de menú: se encuentran los módulos del sistema.
- b) Opciones: enlaces de navegación.
- c) Accesos directos: son enlaces que nos mandan a una determinada opción de la barra de menú.
- d) Bitácora: nos muestra donde nos encontramos en el aplicativo.
- e) Usuario: usuario activo en el aplicativo.
- f) Pestaña de navegación: aparecen cada vez que se accede a una opción de la barra de menú; como acceso directo.
- g) Pantalla completa: nos permite ampliar la pantalla de manera completa.

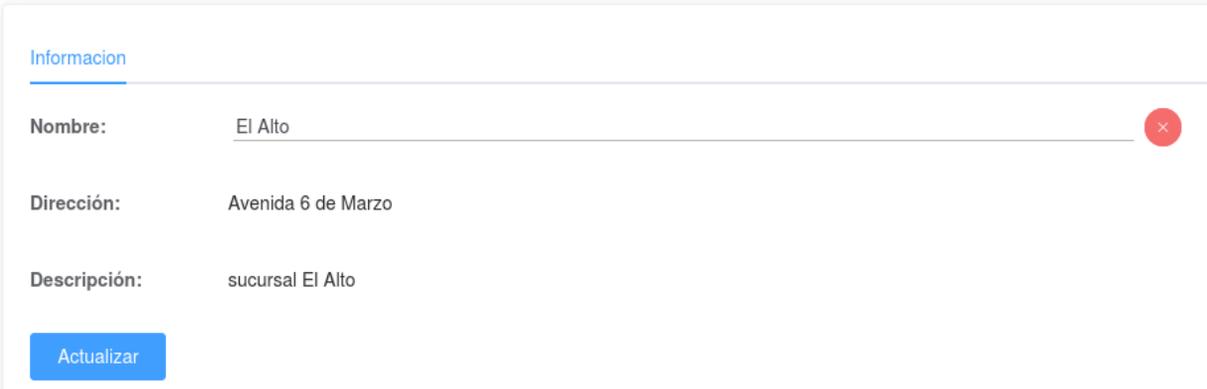
❖ Crear un registro: sucursal

Para crear un registro en Sucursal u otro modulo del aplicativo, son los mismos pasos que se siguen.

- a) Click en el botón “nueva sucursal”
- b) Automáticamente nos aparecerá un formulario que debemos llenar.
- c) Click en el botón crear.

❖ **Actualizar un registro:** sucursal

Detalle de la sucursal



Información

Nombre: El Alto

Dirección: Avenida 6 de Marzo

Descripción: sucursal El Alto

Actualizar

Para actualizar un registro en Sucursal u otro modulo del aplicativo, son los mismos pasos que se siguen.

- a) Click en el botón “detalles”
- b) Doble click encima de dato a modificar, se habilitara la edición.
- c) Click en botón actualizar

❖ **Activar venta de inmueble:** para activar una venta dirígete a: Gestionar inmueble->registro de LCD. Estos pasos también se aplican a lotes urbanizados.

Gestionar Inmuebles

Todos los registros **Disponibles** Para la publicidad

N°	Categoría	Sucursal	Superficie	Precio	Estado	tipo		
1	Casa	El Alto	300 mts2	9000 dorales	disponible	para la venta	activar venta	perfil
2	Casa	Oruro	250 mts2	11000 dorales	disponible	para la venta	activar venta	perfil

a) Click en la pestaña “Disponibles”.

b) Click en el botón “activar venta”.

❖ **Activar publicidad inmuebles:** para activar publicidad, dirígete a: gestionar inmueble -> LCD. Este mismo proceso también se aplica para lotes urbanizados

Gestionar Inmuebles

Todos los registros Disponibles **Para la publicidad**

N°	Categoría	Sucursal	Superficie	Precio	Estado	tipo		
1	Edificio	El Alto	200 mts2	9800 dorales	disponible	para la publicidad	desactivar publicidad	perfil
2	Lote	El Alto	150 mts2	9000 dorales	disponible	para la publicidad	activar publicidad	perfil
3	Edificio	Oruro	500 mts2	11000 dorales	disponible	para la publicidad	activar publicidad	perfil

a) Click en la pestaña “Para publicidad”.

b) Click en el botón “activar venta”.

❖ **Realizar venta de inmueble:** dirígete a vender LCD, aplicable también para venta de lotes urbanizados.

The screenshot shows a web application interface for 'Venta de inmuebles al contado'. On the left is a dark sidebar with navigation options: 'Sucursal', 'Gestionar empleado', 'Gestionar cliente', 'Gestionar inmueble', 'Gestionar Venta', 'vender LCD' (highlighted), 'lote urbanizado', 'Control de ventas', 'Gestionar comision', and 'Agenda de citas'. The main content area is titled 'Venta de inmuebles al contado' and contains a table 'Mi lista de inmuebles para ventar' with one row: '1 Casa El Alto 300 mts2 9000 dorales'. Below the table is a 'Formulario de venta al contado' with sections for 'Datos del inmueble' (fields for 'Categoria', 'Superficie', 'Sucursal', 'Precio') and 'Datos del cliente' (with a 'seleccionar cliente' button).

Venta de inmuebles al contado

Mi lista de inmuebles para ventar

N°	Categoría	Sucursal	Superficie	Precio
1	Casa	El Alto	300 mts2	9000 dorales

Formulario de venta al contado

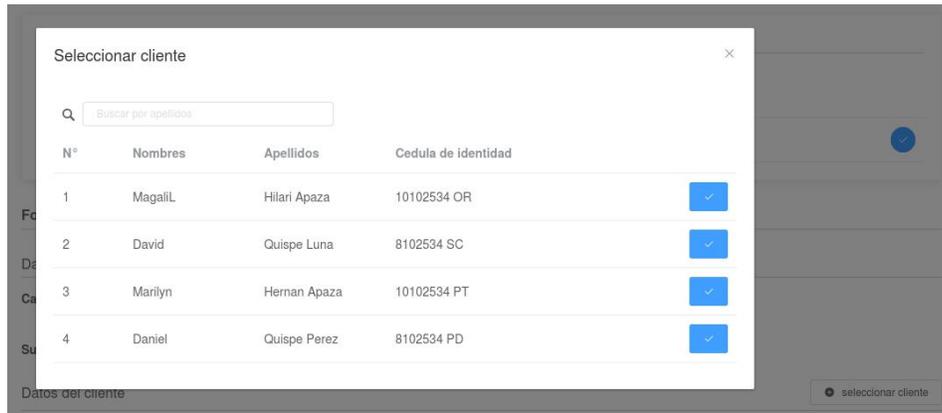
Datos del inmueble

Categoria: _____ **Sucursal:** _____

Superficie: _____ **Precio:** _____

Datos del cliente seleccionar cliente

- a) Llena el formulario.
- b) En “mi lista de inmuebles para vender”, seleccionar el inmueble a vender
- c) Click en seleccionar cliente, aparecerá un formulario, selecciona el cliente.
Realiza la misma operación con “propietario”



- d) Por último click en el botón vender.

Formulario de venta al contado

Datos del inmueble

Categoría: casa Sucursal: El Alto

Superficie: 300 mt2 Precio: 9000 dolares

Datos del cliente

Nombre: David Apellidos: Quispe Luna

C.I.: 8102534 SC Telefono: 2995364

Datos del propietario

Nombre: Wallia Rosario Apellidos: Telleria de Lenz

C.I.: 237463 LP

❖ **Seguimiento de venta de inmueble:** para realizar esta acción dirigente ha:

The screenshot shows a sidebar menu on the left with options like 'Gestionar empleado', 'Gestionar cliente', 'Gestionar inmueble', 'Gestionar Venta', 'Seguimiento ventas', 'controlar LCD', and 'lotes urbanizados'. The main content area is titled 'Seguimiento de venta' and 'Lotes Urbanizados'. It features a search bar and a table with columns: N°, Urbanización/Manzano/lote, Superficie, Precio, Tipo de Venta, and Comprado. Two rows are visible: one for 'La Lomas / A / 4' with a price of 9400 \$us and type 'al contado', and another for 'El cruce / E / 14' with a price of 9400 \$us and type 'plan de pagos'. Each row has a 'mas detalles' button.

- Los registros, muestran las ventas.
- Click en el botón “más detalles” para conocer mas datos de un determinado registro.
- Si la venta se realizó mediante un plan de pagos, Click en el botón “plan de pagos”.

👤 **Plan de pagos**

The screenshot shows the 'Plan de pagos' interface. At the top, it displays: 'Urbanización - El cruce', 'Manzano: E', 'Lote N°: 14', and 'Numero de matricula: 2.01.4.01.02.39343'. Below this, it lists: 'Cliente: MagaliL, Hilari Apaza', 'Moneda: Dolares Americanos', 'Plazo: 48 Meses', and 'Precio total de la transaccion: \$us.- 9400'. The main part is a table with columns: N°, Saldo Cap., Capital, C/Inicial, Cuota, Cancelado, OBS, and two action buttons: 'pagar cuota' and 'revertir pago'. The table contains four rows of payment data.

N°	Saldo Cap.	Capital	C/Inicial	Cuota	Cancelado	OBS		
1	8400	0	1000	0	SI	sin ob	pagar cuota	revertir pago
2	8225	175	0	175	SI	sin ob	pagar cuota	revertir pago
3	8050	175	0	175	NO	sin ob	pagar cuota	revertir pago
4	7875	175	0	175	NO	sin ob	pagar cuota	revertir pago

- Click en el botón “pagar cuota” para realizar el pago de una determinada cuota.
- Para revertir la acción click en el botón “revertir pago”.

❖ **Gestionar asesores comerciales:** para el seguimiento de asesores comerciales, dirige a la siguiente sección del aplicativo.

Gestionar Comision de lotes urbanizados

Todas las comisiones Comisiones pagadas Comisiones por pagar

Buscar por apellidos

N°	Empleado	Inmueble vendido	Comision	Estado de comision	
1	Fernando, Lobos	N° de lote 4	28	pago pendiente	pagar comision + detalles
Total			28		

a) A continuación se observa los registros de los inmuebles que el asesor comercial llegó a vender.

b) Para pagar su comisión click en el botón “pagar comisión”.

❖ **Crear una cita:** para crear una cita dirígete a la opción “agenda de citas” que se encuentra en la barra de menú.

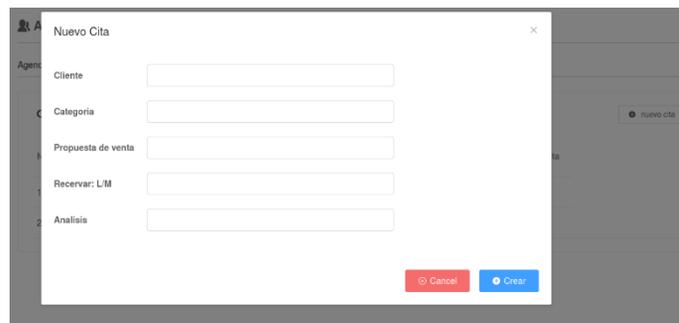
Agenda de citas

Agendas de citas Mi agenda de citas

Buscar cliente por apellidos nuevo cita

N° cliente	Categoría	Propuesta de venta	Reservar: L/M	Analisis al cliente	Fecha de visita
1 Magalil, Hilari Apaza	casa	plan de pagos	10 / E	interesado	30/06/2022
2 Magalil, Hilari Apaza	casa	al contado	30 / A	muy interesado	31/05/2022

c) Click en el botón “nueva cita”



Nuevo Cita

Agencia

Cliente

Categoria

Propuesta de venta

Reservar: L/M

Analisis

Cancel Crear

d) Llenar el formulario.

e) Click en botón crear.