

UNIVERSIDAD PÚBLICA DE EL ALTO

CARRERA DE INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

“SISTEMA WEB DE GESTIÓN Y VENTA DE TERRENOS”

CASO: INMOBILIARIA “KOLLITA”

Para Optar al Título de Licenciatura en Ingeniería de Sistemas
Mención: GESTIÓN Y PRODUCCIÓN

Postulante: Cristhian Parraga Olivares
Tutor Metodológico: M. Sc. Lic. Ing. Marisol Arguedas Ballarades
Tutor Especialista: M. Sc. Lic. Ing. Juan Fernando Chambi Guachalla
Tutor Revisor: Lic. Ing. Edwin Mamani Viscarra

EL ALTO – BOLIVIA

2024

Dedicatoria

Dedico el presente proyecto:

A mis padres

Por apoyo incansable y por haberme dado la fuerza y la motivación necesarias para llegar hasta aquí. Gracias por confiar en mí, por enseñarme el valor del esfuerzo y por siempre ser mi mayor inspiración. Este logro es también suyo.

A mi familia

Por ser mi refugio en los momentos de incertidumbre y por sus palabras de aliento que me dieron la energía para continuar cuando todo parecía difícil. Cada uno de ustedes ha dejado una huella profunda en mi corazón y en mi vida...

Gracias a todos por ser parte de este viaje. Este logro es el reflejo de su apoyo, amor y confianza.

A mis tutores

Por su invaluable ayuda y orientación a lo largo de este proyecto.

Cristhian Parraga Olivares

DECLARACIÓN JURADA DE AUTENTICIDAD Y RESPONSABILIDAD

Yo **Cristhian Parraga Olivares** estudiante con **C.I. 8283962 LP** mediante la presente **declaro** de manera pública que la propuesta del **PROYECTO DE GRADO** titulada **“SISTEMA WEB DE GESTION Y VENTA DE TERRENOS” CASO: INMOBILIARIA KOLLITA** es original, siendo resultado de mi trabajo personal y no constituye una copia o replica de trabajos similares elaborados,

Autorizo la publicación del resumen de mi propuesta en internet y me comprometo a responder a todos los cuestionamientos que se desprenden de su lectura.

Asimismo, me hago responsable ante la universidad o terceros, de cualquiera irregularidad o daño que pudiera ocasionar, por el incumplimiento de lo declarado.

De identificarse falsificación, plagio, fraude, o que el **TRABAJO DE GRADO** haya sido publicado anteriormente; asumo las consecuencias y sanciones que de mi acción se deriven, responsabilizándome por todas las cargas legales que se deriven de ello sometiéndome a las normas establecidas y vigentes de la Carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

El Alto, noviembre de 2024

Cristhian Parraga Olivares

C.I. 8283962 L.P.

crisparraoliv@gmail.com

Agradecimientos

A mis padres, quienes han sido mi pilar fundamental, por su amor incondicional, apoyo constante y por enseñarme el valor de la perseverancia, su aliento y sacrificio.

A mis hermanos, por su compañía, sus palabras de ánimo y su paciencia en los momentos de estrés.

A mis tutores y todo el personal académico que ha formado parte de este camino. Gracias por compartir su conocimiento, por su paciencia y por inspirarme a ser mejor cada día. Ya que no solo han sido guías, sino que también han dejado una huella en mi forma de pensar, actuar y ver el mundo.

A mis amigos, que durante todo este tiempo se han convertido en una extensión de mi familia. Gracias por estar siempre a mi lado, por apoyarme en los días más difíciles y por celebrar los momentos felices. La amistad que hemos cultivado es uno de los mayores tesoros que me llevo de esta etapa.

Cristhian Parraga Olivares

ÍNDICE DE CONTENIDO

ÍNDICE DE TABLAS.....	viii
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE ECUACIONES	xii
CAPITULO I.....	1
1 MARCO PRELIMINAR.....	1
1.1 INTRODUCCIÓN.....	1
1.2 ANTECEDENTES.....	2
1.2.1 Antecedentes Institucionales	2
1.2.2 Antecedentes afines al proyecto de grado	3
1.3 PLANTEAMIENTO DEL PROBLEMA	5
1.3.1 Problema Principal.....	5
1.3.2 Problemas Secundarios.....	6
1.4 OBJETIVOS	6
1.4.1 Objetivo General.....	6
1.4.2 Objetivos Específicos	6
1.4.3 Justificación Técnica.....	7
1.4.4 Justificación Económica	7
1.4.5 Justificación Social	7
1.5 METODOLOGÍA.....	8
1.5.1 Metodología UWE (UML-BASED WEB ENGINEERING)	8

1.5.2	Métricas De Calidad	8
1.5.3	Costo.....	9
1.5.4	Seguridad	10
1.5.5	Pruebas De Software.....	10
1.6	MÉTODOS DE RECOLECCIÓN DE DATOS.....	11
1.6.1	Entrevista	11
1.6.2	Observación	11
1.6.3	Documentos y registros	12
1.7	HERRAMIENTAS	12
1.7.1	HTML5.....	12
1.7.2	CSS3.....	12
1.7.3	Apache HTTP Server.....	13
1.7.4	MySQL	13
1.7.5	Javascript	13
1.7.6	PHP	13
1.7.7	Laravel.....	14
1.8	LIMITES Y ALCANCES	14
1.8.1	Limites.....	14
1.8.2	Alcances.....	14
1.9	APORTES	15
	CAPITULO II.....	16

2	MARCO TEÓRICO	16
2.1	INTRODUCCIÓN.....	16
2.2	SISTEMA.....	16
2.2.1	Tipos De Sistema	17
2.2.2	Según su Organización:	17
1.6.2.	Según su Finalidad:	17
2.3	WEB.....	17
2.3	SISTEMA WEB.....	18
2.3.1	Front-end (Interfaz de usuario):	18
2.3.2	Back-end (Lógica y procesamiento del sistema):.....	18
2.3.3	Base de datos:.....	19
2.3.4	Tipos de sistemas web	19
2.3.5	Sistemas ERP (Enterprise Resource Planning)	20
2.3.6	Componentes de un ERP	20
2.3.7	Importancia de los sistemas ERP en las organizaciones	21
2.4	GESTIÓN	21
2.4.1	Elementos de la Gestión.....	21
2.4.2	Gestión De Ventas.....	22
2.4.3	Tipos de Gestión de Ventas.....	23
2.5	METODOLOGÍA UWE (UML-based Web Engineering)	24
2.5.1	Características de la metodología UWE	24

2.5.2	Fases de la metodología UWE	25
2.5.3	Ventajas de la metodología UWE	26
2.5.4	Aplicación de UWE en el desarrollo de sistemas web.....	27
2.6	UML (Lenguaje Unificado De Modelado)	28
2.6.1	Diagramas UML.....	28
2.7	MÉTRICAS DE CALIDAD.....	29
2.7.1	ISO/IEC 25010	29
2.8	ESTIMACIÓN DE COSTOS	30
2.8.1	Método de estimación de COCOMO	32
2.9	MÉTRICAS DE SEGURIDAD	34
2.9.1	Métrica de Seguridad 27001	35
2.10	Modelo en V	36
2.11	PRUEBAS DE FUNCIONAMIENTO	37
2.11.1	Pruebas de Caja Negra	37
2.11.2	Pruebas de la Caja Blanca	37
2.11.3	Prueba de Stress.....	38
2.11.4	Pruebas de accesibilidad.....	40
2.12	HERRAMIENTAS.....	40
2.12.1	HTML5.....	41
2.12.2	CSS3.....	41
2.12.3	Bootstrap.....	41

2.12.4	JavaScript.....	41
2.12.5	Dompdf.....	42
2.12.6	Jetstream.....	42
2.12.7	Laravel.....	42
2.12.8	PHP.....	43
2.12.9	MYSQL.....	44
2.12.10	Visual Studio Code.....	44
CAPITULO III.....		46
3	MARCO APLICATIVO	46
3.1	INTRODUCCIÓN.....	46
3.2	ANÁLISIS DE LA SITUACIÓN ACTUAL	46
3.3	ESQUEMA DEL SISTEMA WEB	47
3.4	ANÁLISIS DE LOS REQUERIMIENTOS	47
3.4.1	Obtención de Requerimientos	48
3.4.2	Requerimientos No Funcionales	49
3.5	DISEÑO.....	49
3.5.1	Metodología UWE- construcción e implementación del sistema	49
3.5.2	Actores	50
3.5.3	Detalles de Actores de Caso de Uso	50
3.5.4	Diagrama de Casos de Uso del Sistema	50
3.6	Diseño conceptual.	54

3.6.1	Modelo Entidad Relacional.....	54
3.7	FASE DE CONSTRUCCIÓN.....	55
3.7.1	Despliegue del Sistema web.....	56
3.7.2	Capacitación del Usuario.....	56
CAPÍTULO IV.....		58
4	INTRODUCCIÓN.....	58
4.1	MÉTRICAS DE CALIDAD, COSTO Y SEGURIDAD.....	58
4.2	MÉTRICAS DE CALIDAD.....	58
4.2.1	Funcionalidad.....	59
4.2.2	Confiabilidad.....	60
4.2.3	Usabilidad.....	61
4.2.4	Mantenibilidad.....	63
4.2.5	Portabilidad.....	64
4.3	ESTIMACIÓN DE COSTO.....	66
4.3.1	Método De Estimación COCOMO.....	66
4.3.2	Costo Total del Proyecto.....	68
4.4	SEGURIDAD.....	68
4.4.1	Seguridad de Información ISO - 27001.....	68
4.4.2	Seguridad del Sistema.....	69
4.4.3	Seguridad a Nivel de Base de Datos.....	69
4.4.4	Implementación y Beneficios.....	69

4.5	PRUEBAS AL SOFTWARE	69
4.5.1	Pruebas de Caja Blanca	70
4.5.2	Pruebas de la Caja Negra.....	71
4.5.3	Pruebas de Estrés	74
4.5.4	Compatibilidad.....	75
CAPITULO V		77
5	CONCLUSIONES Y RECOMENDACIONES	77
5.1	CONCLUSIONES.....	77
6	RECOMENDACIONES.....	79
6.1	Revisión y actualización constante de los requerimientos.....	79
6.2	Implementación de mecanismos de sincronización de datos	79
6.3	Fortalecimiento de la seguridad de la base de datos	79
6.4	Ampliación de funcionalidades del sistema.....	79
6.5	Capacitación a usuarios finales	79
6.6	Pruebas de rendimiento y escalabilidad.....	80
6.7	Estudio de experiencias del usuario	80
BIBLIOGRAFIA.....		81
ANEXOS.....		84

ÍNDICE DE TABLAS

Tabla 1. Sistema Web Para las Ventas de Empresa Internacional	3
Tabla 2. Sistema Web Para las Ventas de Empresa Nacional	4
Tabla 3. Ejecución del modelo intermedio	32
Tabla 4. Ecuaciones de Modelo COCOMO	33
Tabla 5. Cálculo de Atributos FAE.....	33
Tabla 6. Ecuaciones de Modelo COCOMO	48
Tabla 7. Planilla de caso de Uso	49
Tabla 8. Definición de Actores.....	50
Tabla 9. Planillas de caso de Uso Gestionar Clientes	52
Tabla 10. Planillas de caso de Uso Gestionar Usuarios	53
Tabla 11. Planillas de caso de Registrar Venta	54
Tabla 12. Funciones de Entradas Externas.....	59
Tabla 13. Funciones de Salidas Externas	59
Tabla 14. Escala de Valores de las Preguntas	60
Tabla 15. Escala de Valores de las Preguntas	62
Tabla 16. Preguntas para Determinar la Usabilidad del Sistema	62

Tabla 17. Valores para Determinar la Mantenibilidad	64
Tabla 18. Aplicación del Modelo Intermedio	66
Tabla 19. Ecuación de Modelo COCOMO	67
Tabla 20. Pruebas funcionales del Sistema.....	70
Tabla 21. Evaluaciones Preliminares.....	70
Tabla 22. Tabla de Conclusiones	78

ÍNDICE DE FIGURAS

Figura 1 Organigrama de Institución	2
Figura 2 Estructura de Sistema Web	19
Figura 3 Proceso de Gestión de Ventas	23
Figura 4 Facetas de la Metodología	27
Figura 5 Diagramas UML	28
Figura 6 Modelo ISO/IEC	30
Figura 7 Cálculo de Atributos	33
Figura 8 Modelo en V	36
Figura 9 Tipos de Prueba de Funcionamiento	38
Figura 10 Prueba con JMeter	39
Figura 11 Arquitectura Sistema Web	47
Figura 12 Caso de Uso General del Sistema.....	51
Figura 13 Caso de Uso Clientes.....	51
Figura 14 Caso de Uso Gestionar UsuariosCaso de Uso Gestionar Usuarios.....	52
Figura 15 Caso de Uso Registrar Ventas	53
Figura 16 Modelo Entidad Relación de las Tablas.....	55

Figura 17 Modelo de Calidad de Producto de Software.....	58
Figura 18 Prueba de Inicio al Sistema.....	72
Figura 19 Prueba de Acceso a la Plataforma	72
Figura 20 Prueba de Visualización	73
Figura 21 Prueba de Funcionalidad.....	74
Figura 22 Prueba de con JMeter	75
Figura 23 Compatibilidad con Google Chrome	75
Figura 24 Compatibilidad con Opera GX	76
Figura 25 Compatibilidad con Microsoft Edge	76

ÍNDICE DE ECUACIONES

(1) Ecuación de Tiempo de Funcionamiento	60
(2) Ecuación de Confiabilidad.....	61
(3) Ecuación de Facilidad de Uso.....	62
(4) Ecuación de Mantenibilidad	63
(5) Ecuación Grado de Portabilidad.....	65

RESUMEN

El presente trabajo aborda el diseño e implementación de un sistema de gestión y venta de terrenos. El proyecto surge como respuesta a la creciente demanda de herramientas digitales que optimicen la comercialización y agilicen los procesos administrativos mejorando la experiencia del cliente, la gestión eficiente de inventarios de terrenos, la automatización de procesos de compra y la integración de planes de pago. Además, incorpora módulos para el seguimiento de clientes, generación de reportes y análisis de datos en tiempo real, lo que facilita la toma de decisiones estratégicas. La metodología empleada combina el enfoque de desarrollo ágil y el diseño centrado en el usuario, garantizando un producto funcional y adaptable a las necesidades del mercado. El sistema se desarrolló utilizando PHP y JavaScript, junto con Bootstrap para un diseño responsivo y atractivo. La base de datos MySQL se utiliza para el almacenamiento de información así mismo ofrece a la inmobiliaria “KOLLITA” una herramienta operativa, eficiente y proporciona una excelente experiencia de compra a sus clientes.

Palabras clave: *Gestión, venta de terrenos, inmobiliaria, desarrollo.*

ABSTRACT

This work addresses the design and implementation of a land management and sale system. The project arises in response to the growing demand for digital tools that optimize marketing and streamline administrative processes, improving the customer experience, the efficient management of land inventories, the automation of purchasing processes and the integration of payment plans. In addition, it incorporates modules for customer monitoring, report generation and real-time data analysis, which facilitates strategic decision making. The methodology used combines the agile development approach and user-centered design, guaranteeing a functional product adaptable to market needs. The system was developed using PHP and JavaScript, along with Bootstrap for a responsive and attractive design. The MySQL database is used for information storage and offers the real estate agency "KOLLITA" an efficient, operational tool and provides an excellent purchasing experience to its clients.

Keywords: *Management, land sale, real estate, development.*

CAPITULO I

MARCO PRELIMINAR

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PUBLICA DE EL ALTO



CAPITULO I

1 MARCO PRELIMINAR

1.1 INTRODUCCIÓN

Las tecnologías empresariales y de marketing, avanzan tan rápidamente que en la actualidad, estar a la vanguardia y ser competitivo, significa invertir en los avances tecnológicos más recientes, con el fin de mejorar en todos los aspectos de un negocio por ello contar con un sistema web de ventas, es más que necesaria, debido a que proporciona innumerables beneficios, tales como la funcionalidad, la agilidad en los procesos administrativos, la facilidad al momento de usarlo, la generación de información detallada, entre otras.

La productividad y consumo de bienes materiales impulsa a los habitantes a comprar todo tipo de productos, particularmente terrenos para la construcción de sus hogares que en anteriores años la compra era un lujo, sin embargo, con el paso del tiempo ya se encuentra al alcance de muchos.

Uno de tipos de sistemas de información son los Sistemas denominados ERP (Enterprise Resource Planning – Planificación de Recursos Empresariales) cuyo propósito es permitir la integración de ciertas operaciones de una empresa, especialmente las que tienen que ver con la producción, la logística, el inventario, ventas, clientes. (Chiesa, 2004, pág. 17).

El presente proyecto tiene por objeto desarrollar un Sistema ERP para la gestión y ventas de terrenos en la empresa inmobiliaria KOLLITA. Para el desarrollo de este proyecto se utilizará la metodología UWE, para las tecnologías que se utilizaran en la parte de Frontend son HTML, CSS, JavaScript y para el Backend, PHP, MySQL.

1.2 ANTECEDENTES

1.2.1 Antecedentes Institucionales

INMOBILIARIA KOLLITA empresa de bienes raíces dedicada a la compra y venta de casas, terrenos al contado y crédito directo, con terrenos en la Carretera a Copacabana y Laja, fundado el 12 de septiembre del 2017 en la ciudad de El Alto ubicado en la avenida Juan Pablo II, frente UPEA (El Alto) Edif. Los Ángeles 1er piso Of.4, La Paz, Bolivia.

1.2.1.1 Misión

Brindar un servicio profesional, responsable e idóneo en la compra, venta y crédito, siempre queriendo satisfacer al cliente dando solución oportuna y eficaz, a todas sus necesidades inmobiliarias específicas, dando plus valía a todas sus inversiones.

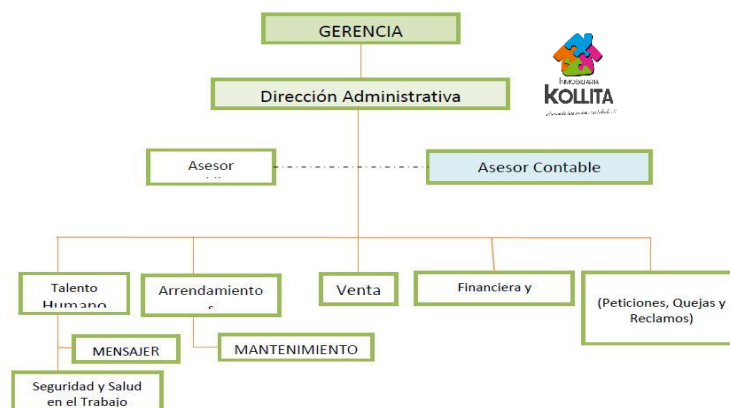
1.2.1.2 Visión

Ser reconocidos como una empresa líder en la región por gestión inmobiliaria satisfaciendo a nuestros clientes ayudando solucionar su necesidad de encontrar su lugar adecuado.

1.2.1.3 Organigrama

Figura 1

Organigrama de Institución



1.2.2 Antecedentes afines al proyecto de grado

1.2.2.1 Antecedentes Internacionales

Tabla 1

Sistema web de ventas internacional

Título de la publicación	Implementación de un sistema web para las ventas en la empresa One to One Contact Solutions
Lugar	País: Peru
Autor/es	Año 2018 Ronald Alexander Guillermo Navarro
Palabras claves	Metodología Ágil, Aplicativo Web, Canal de Ventas, Necesidad de la Empresa, Tecnología de la Información, Bases de Datos.
Dirección electrónica específica	https://repositorio.usil.edu.pe/handle/usil/2764
Tópicos Relevantes	
Formulación del problema o preguntas de investigación	En la actualidad la Tecnología de la Información aporta un valor agregado a toda empresa que busca eficiencia en sus procesos, o alguna oportunidad de destacar frente a otras empresas, la necesidad de implementar soluciones informáticas hoy en día se ha convertido en una prioridad en toda empresa que quiere mantenerse en su rubro
Objetivo de la investigación	El presente proyecto aborda la implementación de un sistema web como respuesta a una necesidad de la empresa One To One Contact Solutions en el área de ventas, la cual está posicionada entre los mejores canales de ventas asociados a Scotiabank Perú
Hipótesis/Idea a defender	Existirá una diferencia significativa en la cantidad promedio de llamadas realizadas por gestión del cliente antes de implementar el sistema web (pre prueba) y la cantidad promedio después de implementar el sistema web (post prueba).
Metodología de la investigación o desarrollo.	Haciendo uso de la Metodología Ágil (Scrum y XP), se pretende desarrollar un aplicativo web basado en lenguaje de programación PHP, con un motor de base de datos MYSQL y servidor Web Apache 2.2.
Resultados de la investigación	Existe diferencia significativa en la cantidad promedio de llamadas realizadas por gestión del cliente antes y después de implementar el sistema web por tanto la implementación de un sistema web para el área de ventas de la empresa One To One Contact Solutions ha tenido un impacto positivo en el proceso de ventas.
Análisis Del Trabajo Observaciones	Se debe incluir datos recientes, para medir la efectividad del sistema web en la actualidad.

Nota. Datos extraídos del perfil de trabajo de grado de (Ronald Alexander Guillermo Navarro, 2018)

1.2.2.2 Antecedentes Nacionales

Tabla 2

Sistema Web Para las Ventas de Empresa Nacional

Título de la publicación	Sistema Web de administración de ventas basado en la gestión de procesos de negocios CASO: Empresa de Muebles Artemobile, Departamento de Ventas
Lugar	País: Bolivia
Autor/es	Liz Andrea Ramos Huarachi
Palabras claves	Sistema web, Administración de ventas
Dirección electrónica específica	https://repositorio.umsa.bo/handle/123456789/7806
Tópicos Relevantes	
Formulación del problema o preguntas de investigación	Actualmente la información en una empresa cualquiera sea esta, es de vital importancia con lo que se llegó a comprobar que es un factor clave en la empresa, y con ello una clave de la gestión empresarial, que puede llegar a determinar el éxito o el fracaso de un negocio, con lo que llega a ser el activo intangible más importante de dicha empresa, por lo cual es imposible que una empresa no haga uso de la información para el desarrollo de sus actividades diarias.
Objetivo de la investigación	El presente Proyecto de Grado titulado “Sistema Web De Administración De Ventas, Basado En La Gestión De Procesos De Negocios” ha sido desarrollado en la empresa ARTEMOBILE, empresa dedicada al desarrollo y venta.
Hipótesis/Idea a defender	Sin hipótesis.
Metodología de la investigación o desarrollo.	Para el desarrollo del proyecto se utilizó la metodología Ágil SCRUM, que propone un modelo de proceso incremental, basado en iteraciones y revisiones continuas. También se utilizó en cada iteración la metodología web WEBML, que se especializa en el diseño de Aplicaciones Web.
Resultados de la investigación	Luego de concluir satisfactoriamente el proyecto de grado “sistema web de administración de ventas, basado en la gestión de procesos de negocios caso: empresa de muebles ARTEMOBILE, departamento de ventas”, logrando el cumplimiento de todos los requisitos especificados por el cliente, se puede mencionar que uno de los objetivos fue lograr la construcción de la base de datos para poder almacenar toda la información necesaria de la empresa.
Análisis Del Trabajo Observaciones	En el siguiente proyecto se recomienda realizar la actualización y mantenimiento del sistema implantado, esto para un correcto funcionamiento y evitar sorpresivas fallas en el futuro.

Nota. Datos extraídos del perfil de trabajo de grado de (Liz Andrea Ramos Huarachi, 2013)

1.3 PLANTEAMIENTO DEL PROBLEMA

En muchas empresas, la falta de un manejo adecuado y organizado de la información impide tomar decisiones eficientes. Esto afecta áreas clave como los productos que ofrecen, las ventas realizadas y la gestión de los clientes.

Los clientes buscan un servicio rápido y sencillo. Por lo tanto, brindar una atención de calidad es esencial para lograr su satisfacción y asegurar una posición competitiva en el mercado.

En este caso la inmobiliaria KOLLITA presenta cierto déficit en el registro de ventas, planes de pago y reportes de terrenos disponibles, lo que es consecuencia directa de la pérdida o deserción de clientes tanto potenciales como no potenciales.

El problema en el proceso de venta radica en que los créditos para terrenos se gestionan de forma manual. Esto a menudo genera confusiones en los terrenos disponibles y planes de pago tanto al contado como a crédito especialmente, en cuanto a la cantidad del monto económico para la venta. Como resultado, los clientes pierdan el interés y se formen una mala impresión sobre la atención y el servicio de la empresa.

1.3.1 Problema Principal

Actualmente la empresa INMOVILIARIA KOLLITA, la mayor parte de los procesos y actividades que realizan la hacen de forma manual, todos los registros que están distribuidos en diferentes equipos de computación, en archivos Word y Excel no actualizados simultáneamente produciendo dificultad en el control de todos los procesos concernientes de las ventas y planes de crédito de terrenos a los clientes.

1.3.2 Problemas Secundarios

- La atención al cliente en KOLLITA es manual, lo que retrasa el tiempo de respuesta y genera insatisfacción debido a la búsqueda de información en múltiples documentos.
- El control y administración de los terrenos y planes de pago no son eficientes, lo que con lleva a errores al momento de ofrecer terrenos al cliente.
- Riesgo de pérdida de información de ventas debido a errores del personal, extravío de documentos y a la desorganización en la gestión de la información.
- La obtención de información por gestión es difícil ya que se encuentra dispersa, lo que ocasiona la toma de decisiones no acertadas.

1.4 OBJETIVOS

En el presente proyecto se establecen los siguientes objetivos general y específicos:

1.4.1 Objetivo General

Desarrollar un Sistema Web de Gestión y Venta de terrenos para optimizar el control para la empresa inmobiliaria KOLLITA, con el propósito de mejorar los procesos de atención a sus clientes y así generar información confiable y oportuna.

1.4.2 Objetivos Específicos

- Analizar y clasificar los requerimientos necesarios para el desarrollo y funcionamiento del sistema.
- Permitir el acceso rápido y actualizado a datos de terrenos disponibles, características, ubicaciones, y planes de pago.

- Centralizar la información en una base de datos que permita realizar consultas en línea.
- Generar información confiable sobre los terrenos y su rentabilidad para facilitar una toma de decisiones óptima.

1.4.3 Justificación Técnica

El proyecto se justifica técnicamente ya que la inmobiliaria tiene a disposición herramientas de diseño, programación, aplicaciones, así también se justifica porque en el desarrollo se utilizarán tecnologías como ser Backend y Frontend, además de servidores.

1.4.4 Justificación Económica

El desarrollo del Sistema reducirá los gastos en materiales (reportes, planillas información en libros de manera manual) con lo que se ahorrará costos.

El proyecto se desarrollará con el uso de software libre lo que evitará a la asociación pagos por licencias y por ende se constituye en la reducción de gastos. Además de contar con los equipos de cómputo necesarios.

1.4.5 Justificación Social

El desarrollo del Software para el control de ventas principalmente porque no se cuenta con información automatizada dentro de la institución que coadyuve con la toma de decisiones del personal a cargo. Con la ejecución del software se ofrece una mejor eficiencia al personal encargado de manera rápida, segura y agilizada.

1.5 METODOLOGÍA

En relación con la metodología de este proyecto, se ha optado por emplear una serie de enfoques y procedimientos que permitirán alcanzar los objetivos propuestos de manera eficiente y efectiva. A continuación, se detallan los siguientes puntos:

1.5.1 Metodología UWE (UML-BASED WEB ENGINEERING)

Es una herramienta para modelar aplicaciones web, utilizada en la ingeniería web, prestando especial atención en sistematización y personalización (sistemas adaptativos).

UWE es una propuesta está formada por una notación para especificar el dominio (basada en UML) y un modelo para llevar a cabo el desarrollo del proceso de modelado (Luis Acevedo & Andrés Felipe, 2017).

- Las fases o etapas a utilizar son:
- Análisis de requerimiento.
- Diseño del sistema
- Codificación del software
- Pruebas
- La Instalación o fase de implementación.
- El Mantenimiento.

1.5.2 Métricas De Calidad

Evaluación de Calidad de Productos de Software (SquaRE). Constituyen una serie de normas basadas en la ISO 9126, y su objetivo principal es guiar el desarrollo de los productos

de software con la especificación y evaluación de requisitos de calidad. La familia ISO 25000 está orientada al producto software, permitiendo definir el modelo de calidad y el proceso a seguir para evaluar dicho producto (Pantaleo Guillermo, 2016).

- Las métricas de la ISO son:
- Funcionalidad.
- Eficiencia.
- Compatibilidad.
- Usabilidad.
- Seguridad

1.5.3 Costo

COCOMO (COConstructive COst MOdel), es un modelo de costo objetivo para la planificación e implementación de proyectos de software. Es un componente esencial para la gestión de proyectos de software o negocios de software. El modelo de costos proporciona un marco para el intercambio de decisiones comerciales entre las partes interesadas en el desarrollo de software. COCOMO II incluye una serie de mejoras probadas en el campo que amplían la aplicabilidad y mejoran la precisión de las estimaciones para los métodos modernos de desarrollo de software. El primero es un marco para describir un proyecto de software, incluidos los modelos de proceso, la cultura, las partes interesadas, los métodos, las herramientas y el tamaño/complejidad del software. (Sharma, 2011)

1.5.4 Seguridad

Según ISO 27001, se refiere a la confidencialidad, la integridad y la disponibilidad de la información y los datos importantes para la organización, independientemente del formato que tengan, estos pueden ser:

- Electrónicos
- En papel, etc.

1.5.5 Pruebas De Software

Para las pruebas de software se utiliza el método de pruebas de caja negra el cual evalúa las entradas introducidas por los usuarios y analiza el resultado devuelto por el sistema además de la prueba de funcionalidad.

1.5.5.1 Caja Negra

Las pruebas de caja negra o también conocidas como pruebas de comportamiento se centran en los requisitos funcionales del software. Para realizar la prueba de caja negra se realiza las pruebas a la interfaz

1.5.5.2 Caja Blanca

Esta prueba se orienta al cálculo de las regiones que deben ser consideradas como partes independientes del sistema, y estableciendo las entradas y se ejecutan cada una de las regiones, asegurando así que cada región se ejecuta al menos una vez.

1.5.5.3 Pruebas De Estrés

Las pruebas de estrés validan el comportamiento de la aplicación bajo condiciones de carga máxima, es decir se lleva al sistema a los límites que esta pueda soportar, de esta forma identificaremos el comportamiento del sistema bajo distintos escenarios.

1.5.5.4 Pruebas de accesibilidad

Son un conjunto de evaluaciones realizadas en un sitio web, aplicación o sistema digital para asegurar que sea accesible para personas con discapacidades. El objetivo es garantizar que todas las personas, independientemente de sus capacidades o discapacidades, puedan interactuar y utilizar correctamente los contenidos y funcionalidades del sitio. Estas pruebas se basan en directrices y normativas internacionales, como las WCAG (Web Content Accessibility Guidelines), para evaluar diversos aspectos de accesibilidad.

1.6 MÉTODOS DE RECOLECCIÓN DE DATOS

1.6.1 Entrevista

La entrevista es, en esencia, una conversación bien planificada. En ella, el investigador plantea una serie de preguntas o temas de debate a una o varias personas, con el fin de obtener información específica.

1.6.2 Observación

La observación es una técnica que consiste precisamente en observar el desarrollo del fenómeno que se desea analizar. Este método puede usarse para obtener información cualitativa o cuantitativa de acuerdo con el modo en que se realiza.

1.6.3 Documentos y registros

Esta técnica consiste en examinar los datos presentes en documentos ya existentes, como bases de datos, actas, informes, registros de asistencia, etc.

Por lo tanto, lo más importante para este método es la habilidad para encontrar, seleccionar y analizar la información disponible. Es necesario tener en cuenta que la información recopilada puede dar información inexacta o incompleta. Por este motivo, debe ser analizada en relación con otros datos para que pueda ser útil a la investigación (GRUPO EDITORIAL, 2021).

1.7 HERRAMIENTAS

1.7.1 HTML5

(Lenguaje de Marcado de Hipertexto) es un lenguaje compuesto por entidades que ayudan a estructurar y dar sentido a las diferentes partes de un documento. Cada una de estas entidades, comúnmente denominadas elementos o etiquetas, consta de contenido y cero, uno o más atributos. Cada uno de los atributos tiene una función y puede o no estar asociado a un comportamiento específico. (Casado, 2023)

1.7.2 CSS3

(Hojas de Estilo en Cascada) es un lenguaje de programación muy similar a HTML que le permite aplicar estilos a diferentes elementos de las páginas web para que los encabezados, las listas y los párrafos se vean iguales en todas las páginas. (A. C. Luna, 2019)

1.7.3 Apache HTTP Server

Es un software de servidor web gratuito y de código abierto para plataformas Unix con el cual se ejecutan el 46% de los sitios web de todo el mundo. Es mantenido y desarrollado por la Apache Software Foundation (Gustavo B., 2022).

1.7.4 MySQL

Es un sistema de gestión de bases de datos relacional (RDBMS, por sus siglas en inglés) que utiliza el lenguaje **SQL** (Structured Query Language) para gestionar y manipular los datos almacenados en bases de datos. Es uno de los sistemas de bases de datos más populares y ampliamente utilizados en aplicaciones web y otras plataformas, por lo tanto, se distribuye de distintas maneras. Hay una versión con licencia pública general de GNU (la denominada Community Edition) y otras orientadas a empresas que brindan servicios y productos adicionales. Cabe destacar que Facebook, YouTube y Twitter, entre otros de los sitios web más visitados del mundo, trabajan con MySQL(Pérez Porto & Gardey J., 2019).

1.7.5 Javascript

Diseñado para añadir efectos y animaciones a los sitios web, muchos sitios Frontend ,Backend, aplicaciones isomórficas convirtiéndose en el lenguaje de programación del presente(Azaustre Carlos, 2016).

1.7.6 PHP

Lenguaje de Programación de código abierto del lado del servidor que se utiliza principalmente para crear páginas web dinámicas. La abreviatura nació originariamente de “Personal Home Page Tools”, aunque hoy en día se ha convertido en el acrónimo recursivo para “PHP:Hypertext Preprocessor”.(digital guide IONOS, 2020).

1.7.7 Laravel

Es un framework de desarrollo web **de código abierto** basado en PHP que proporciona una estructura robusta para el desarrollo de aplicaciones web, facilitando tareas comunes como la autenticación, las rutas, la base de datos y el manejo de sesiones. Laravel promueve el uso de un patrón de diseño MVC (Modelo-Vista-Controlador), lo que ayuda a organizar y estructurar las aplicaciones de manera más eficiente (Otwell T., (2011).

1.8 LIMITES Y ALCANCES

1.8.1 Limites

Dado que el principal requerimiento del proyecto es la integración de información bibliográfica y administración de las mismas, el proyecto se limitará solo a estos procesos mencionados.

1.8.2 Alcances

Con la implementación del proyecto, se alcanzará lo siguiente:

- Módulo de búsqueda global de los usuarios registrados en el sistema.
- Módulo de urbanizaciones y sus terrenos.
- Módulo de las cotizaciones de los diferentes precios crediticios que ofrece la inmobiliaria.
- Módulo de los terrenos y sus características para su respectivo precio.
- Módulo de terrenos vendidos.
- Módulo de cotizaciones de terreno.

1.9 APORTES

El aporte para la empresa inmobiliaria KOLLITA es el desarrollo e implementación del Sistema Web para la Gestión y Venta de Terrenos. Proponiendo mejorar un control confiable y coherente de las mismas, brindando información para ayudar a la inmobiliaria venta de sus inmuebles.

Asimismo, ayudando en agilizar los procesos rutinarios para minimizar y optimizar tiempos de ejecución generando información que coadyuvé a la buena toma de decisiones

Con el desarrollo del sistema mejorará:

- Automatización de procesos y control de los servicios que ofrece la inmobiliaria.
- Precisión en la información de los terrenos.
- Gestión eficiente de créditos y ventas.
- Mejorará la experiencia del cliente en las consultas .

CAPITULO II

MARCO TEORICO

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PUBLICA DE EL ALTO



CAPITULO II

2 MARCO TEÓRICO

2.1 INTRODUCCIÓN

En la elaboración de un proyecto de grado, se siguen diversas etapas que requieren una atención minuciosa, especialmente en cuanto a su descripción. El marco teórico constituye la base fundamental del proyecto. En este capítulo se desarrollarán los conceptos y principios que sustentan el presente trabajo, tiene como objetivo proporcionar una sólida base conceptual para comprender y evaluar la arquitectura y el funcionamiento de los sistemas de información web. De esta manera, se complementa la parte teórica para un mejor entendimiento de los distintos conceptos empleados en el proyecto.

2.2 SISTEMA

Un *sistema* es un conjunto de elementos interrelacionados que actúan de manera coordinada para alcanzar un objetivo o cumplir una función específica. Según varios autores, este concepto puede ser entendido desde diversas perspectivas.

Los sistemas se encuentran en diversos campos de estudio y son fundamentales para comprender procesos complejos, desde fenómenos naturales hasta aplicaciones tecnológicas

. En el ámbito de la tecnología, por ejemplo, un sistema informático se compone de hardware, software, usuarios y procedimientos que trabajan juntos para procesar y gestionar información de manera eficiente (Morales & García, 2021).

En este contexto, el presente proyecto considera un sistema como una herramienta estructurada para la venta y gestión de terrenos, integrando componentes tecnológicos y organizativos que optimizan los procesos administrativos y comerciales en el sector inmobiliario.

2.2.1 Tipos De Sistema

2.2.2 Según su Organización:

Sistemas Organizados: Tienen una estructura interna bien definida y ordenada, con una jerarquía o distribución clara de funciones. Ejemplos incluyen una organización empresarial o un organismo vivo.

Sistemas Desorganizados: No poseen una estructura interna definida y sus elementos no siguen un orden específico. En física, se puede considerar como ejemplo un gas en un recipiente cerrado.

1.6.2. Según su Finalidad:

Sistemas de Control: Son sistemas diseñados para regular y controlar otros sistemas o procesos mediante la retroalimentación. Un ejemplo típico es un termostato que regula la temperatura.

Sistemas Productivos: Son sistemas destinados a transformar insumos en productos o servicios, como una fábrica o una empresa de servicios.

2.3 WEB

La web o World Wide Web (WWW) es un sistema de información que permite el acceso a una gran cantidad de contenidos y servicios en Internet. Consiste en una colección de documentos y recursos interconectados mediante enlaces o "hipervínculos".

Estos documentos están almacenados en servidores y pueden ser consultados por personas desde dispositivos conectados a la red, como computadoras, teléfonos o tablets.

2.3 SISTEMA WEB

Un sistema web es una aplicación o plataforma informática diseñada para operar en un entorno en línea, utilizando tecnologías de Internet para su funcionamiento y acceso. Estos sistemas se alojan en servidores y son accesibles desde cualquier dispositivo conectado a la red a través de navegadores web (González & Pérez, 2020).

Los sistemas web son herramientas clave para la gestión empresarial, ya que permiten la automatización de procesos, el acceso remoto a la información y una mayor eficiencia operativa. Ejemplos comunes incluyen plataformas de comercio electrónico, sistemas de gestión de contenidos (CMS) y aplicaciones de gestión empresarial (López 2018).

Los sistemas web están compuestos por tres elementos principales:

2.3.1 *Front-end (Interfaz de usuario):*

- Es la parte visible con la que el usuario interactúa.
- Incluye elementos gráficos, texto, botones y formularios.
- Se desarrolla generalmente usando tecnologías como HTML, CSS y JavaScript para hacer la experiencia atractiva e interactiva.

2.3.2 *Back-end (Lógica y procesamiento del sistema):*

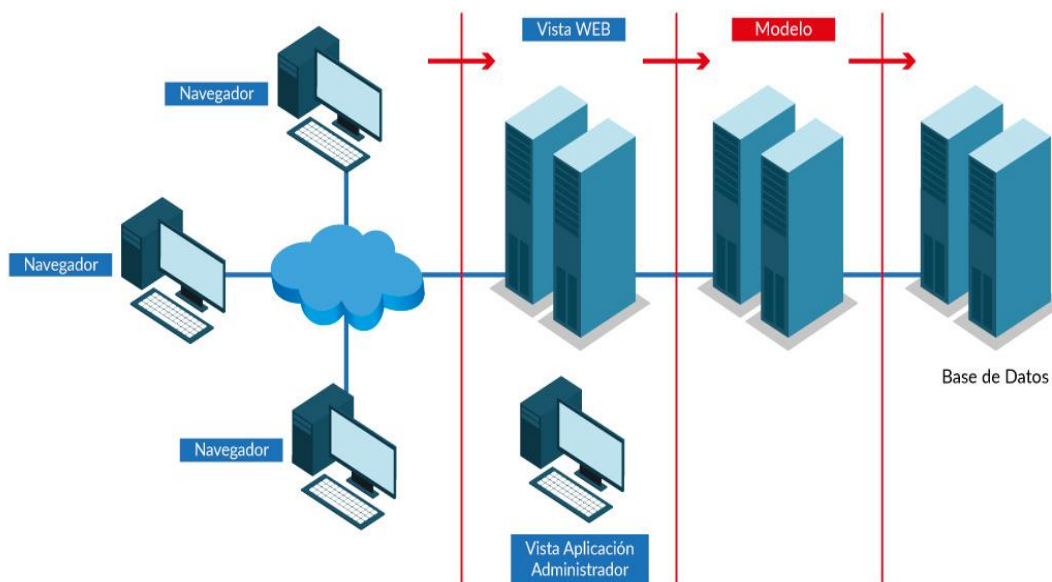
- Es la parte del sistema que no es visible para el usuario y se encarga de procesar y gestionar la información.
- Incluye el servidor de aplicaciones y la lógica del negocio.
- Se desarrolla con tecnologías como Python, Java, PHP, Node.js, Ruby, entre otras.

2.3.3 Base de datos:

- Almacena y gestiona los datos necesarios para el funcionamiento del sistema (por ejemplo, información de usuarios, transacciones, inventarios).
- Los sistemas de bases de datos más comunes incluyen MySQL, PostgreSQL, MongoDB, entre otros.

Figura 2

Estructura de Sistema Web



2.3.4 Tipos de sistemas web

Sistemas de información: Organizan y muestran información, como directorios o catálogos.

Gestión empresarial: Sistemas ERP o CRM que ayudan a gestionar recursos empresariales o relaciones con clientes.

2.3.5 Sistemas ERP (Enterprise Resource Planning)

Los sistemas de planificación de recursos empresariales, conocidos como **ERP** (por sus siglas en inglés, *Enterprise Resource Planning*), son plataformas de software diseñadas para integrar y gestionar todos los procesos y áreas de una organización. Según Laudon y Laudon (2016), un ERP es un "sistema que permite centralizar la información y automatizar los procesos en distintas áreas de una empresa, mejorando la comunicación y la toma de decisiones al unificar los datos en una única base" (*Management Information Systems: Managing the Digital Firm*).

Los sistemas ERP se define como "herramientas que facilitan el flujo de información entre funciones empresariales y permiten una visión integral del negocio" O'Brien y Marakas (2011).

2.3.6 Componentes de un ERP

Un sistema ERP está compuesto por varios módulos funcionales que permiten gestionar diferentes áreas de la organización, entre los que se destacan (Monk y Wagner, 2008):

Finanzas y contabilidad: Administra el presupuesto, las cuentas por cobrar y pagar, y el control financiero.

Gestión de inventarios: Permite el seguimiento de inventarios y el control del flujo de suministros y productos.

Ventas y marketing: Facilita la gestión de pedidos, la facturación y el seguimiento de ventas.

Recursos humanos: Controla la información de los empleados, incluyendo la nómina, el reclutamiento y la evaluación del desempeño.

Cada módulo se conecta con los demás, asegurando una visión integral y actualizada de la empresa, lo cual facilita la toma de decisiones en tiempo real (Davenport, 1998).

2.3.7 Importancia de los sistemas ERP en las organizaciones

La implementación de un sistema ERP aporta numerosos beneficios a las organizaciones, tales como la reducción de costos operativos, el aumento de la productividad, la mejora de la comunicación interdepartamental y la centralización de datos en una única base (Escobar, Vélez y López, 2013).

2.4 GESTIÓN

La gestión es el proceso de planificar, organizar, dirigir y controlar los recursos de una organización de manera efectiva y eficiente para lograr sus objetivos. Esto implica la toma de decisiones y la asignación óptima de recursos, como el capital humano, financiero, físico y tecnológico.

2.4.1 Elementos de la Gestión

Los principales elementos de la gestión son:

Planificación: Establezca objetivos claros y objetivos a largo plazo, teniendo en cuenta las fortalezas de la organización.

Organización: Discutir las tareas necesarias para lograr objetivos y distribuir completamente las tareas entre los miembros de la organización.

Dirección: Dirige los esfuerzos de los empleados para lograr las metas planificadas a través de la motivación, la influencia y la comunicación efectiva.

Monitoreo: monitorear el desempeño organizacional frente a objetivos previamente establecidos y tomar acciones correctivas si es necesario.

2.4.2 Gestión De Ventas

Es el proceso de desarrollar la fuerza de ventas mediante la coordinación de las estrategias y tácticas de ventas más efectivas para lograr los objetivos comerciales. Aunque su empresa esté generando ingresos hoy en día, siempre hay una mejor manera de garantizar operaciones comerciales fluidas y aumentar la productividad. Además, una buena gestión de ventas creará un equipo de ventas más comprometido y centrado en lograr resultados. Lo mejor de la gestión de ventas es que no es necesario ser una empresa de talla mundial. Las pequeñas y medianas empresas pueden beneficiarse de este nivel de trabajo. Incluso animamos a las empresas a dominar la gestión de ventas para lograr un crecimiento sostenible, independientemente de su tamaño o sector. (Palacios, 2023).

La gestión de ventas es el proceso de coordinar un grupo de personas y recursos para lograr los objetivos de ventas de una empresa. Contiene un conjunto de políticas y procedimientos que brindan orientación para una amplia gama de actividades. El departamento de ventas es el responsable de gestionar las ventas. Una adecuada gestión de ventas debe identificar nuevas oportunidades, realizar previsiones de ventas y gestionar informes y métodos. Todas estas tareas ayudarán a los participantes a alcanzar sus objetivos. Sin embargo, la gestión de ventas debe incluir una serie de procesos y actividades. Si se hace de forma eficaz, ayudará a la empresa a crecer y obtener más beneficios. (Quiroa, Gestión de ventas, 2021)

La gestión de ventas es la actividad comercial de intercambiar bienes o servicios por dinero u otros artículos de valor. Generalmente, es un proceso o serie de pasos lógicos y ordenados que se siguen para lograr el objetivo final de ambas partes (comprador y vendedor).

2.4.3 Tipos de Gestión de Ventas.

Venta Directa: consiste en la venta personalizada por parte del vendedor directamente al comprador, sin intermediarios. Ejemplos de este tipo de ventas incluyen demostraciones de productos en el hogar o ventas de productos en el hogar.

Figura 3

Proceso de Gestión de Ventas



2.4.4 Terrenos

Los terrenos son porciones delimitadas de la superficie terrestre que presentan características físicas, químicas y biológicas específicas, las cuales determinan su capacidad de uso y aprovechamiento. Factores como la calidad del suelo, la topografía, el clima y la ubicación geográfica influyen directamente en su aptitud para actividades agrícolas, urbanísticas, industriales o de conservación ambiental (Food and Agriculture Organization [FAO], 2017). Además, los terrenos son elementos clave en el desarrollo sostenible, ya que su manejo adecuado permite maximizar su potencial mientras se minimiza el impacto ambiental (United Nations, 2020).

2.5 METODOLOGÍA UWE (UML-based Web Engineering)

La metodología UWE es un enfoque estructurado para el desarrollo de aplicaciones web, que utiliza el Lenguaje de Modelado Unificado (UML) para describir, especificar y documentar los distintos aspectos de un sistema web. UWE facilita el desarrollo de aplicaciones web complejas al proporcionar un marco formal que cubre todo el ciclo de vida del sistema, desde la concepción y análisis de requerimientos hasta la implementación y mantenimiento (Koch & Kraus, 2002).

En conclusión, UWE es una metodología eficaz para el desarrollo de sistemas web complejos, proporcionando una estructura clara y formal que facilita la gestión de la complejidad y asegura que los productos desarrollados cumplan con los requerimientos de calidad y funcionalidad.

2.5.1 Características de la metodología UWE

Entre las principales características de la metodología UWE se destacan:

Basada en UML: UWE emplea UML para todos los modelos, lo cual estandariza la representación visual de los elementos del sistema, haciendo que el desarrollo sea más comprensible y manejable.

Orientación hacia la navegabilidad y usabilidad: La metodología incluye elementos para modelar la estructura de navegación y la interacción con el usuario, lo cual permite construir sistemas web intuitivos y fáciles de usar.

Extensibilidad: Los modelos de UWE son escalables y modulares, lo cual permite que el sistema pueda ampliarse en el futuro si es necesario.

2.5.2 Fases de la metodología UWE

UWE sigue un conjunto de fases organizadas que cubren todo el ciclo de vida del desarrollo de un sistema web, desde el análisis de requisitos hasta la implementación. Estas fases son (Baumeister, Knapp, Koch, & Zhang, 2005):

2.5.2.1 Modelado de requisitos

En esta fase se identifican y analizan los requisitos funcionales y no funcionales del sistema. Se utilizan diagramas de casos de uso en UML para representar las interacciones principales de los usuarios con el sistema.

2.5.2.2 Modelado conceptual

El objetivo de esta fase es desarrollar un modelo conceptual del sistema, incluyendo las clases y relaciones que representan los datos y las entidades que el sistema manejará. Este modelo conceptual se representa mediante diagramas de clases.

2.5.2.3 Modelado de navegación

La navegación es un aspecto crucial en las aplicaciones web. En esta fase se modelan las rutas de navegación que los usuarios pueden seguir, asegurando que la estructura de navegación sea clara y adecuada para el flujo de trabajo de los usuarios.

2.5.2.4 Modelado de presentación

El enfoque aquí es el diseño de la interfaz de usuario. Se crean diagramas de interacción y secuencia para representar la presentación y los elementos de interacción de la aplicación, asegurando que la experiencia de usuario sea coherente y funcional.

2.5.2.5 Implementación

En esta última fase se convierte el diseño en un sistema funcional, utilizando los modelos generados en las fases anteriores como guía para el desarrollo. Las tecnologías y lenguajes específicos pueden variar dependiendo de los requisitos del sistema.

2.5.3 Ventajas de la metodología UWE

La metodología UWE presenta varias ventajas para el desarrollo de sistemas web, entre las que destacan:

Claridad en el diseño: Al estar basada en UML, UWE proporciona una representación clara y visual del sistema, facilitando la comprensión del diseño.

Facilidad de mantenimiento: La estructura modular de los modelos permite realizar modificaciones de manera ágil, facilitando el mantenimiento y la actualización del sistema.

Enfoque en la experiencia del usuario: El modelado de navegación y presentación asegura que el sistema sea fácil de usar y de entender para los usuarios finales.

2.5.4 Aplicación de UWE en el desarrollo de sistemas web

La metodología UWE es particularmente útil para proyectos de sistemas web complejos que requieren una arquitectura bien definida y un diseño centrado en el usuario. Debido a su estructura modular y su uso de UML, es adecuada para proyectos que involucren equipos de desarrollo distribuidos y que requieran una documentación completa y clara (Escalona & Koch, 2004).

Figura 4

Fases de la Metodología



2.6 UML (Lenguaje Unificado De Modelado)

Este es un conjunto de 13 tipos diferentes de diagramas para modelar sistemas de software. Surgió de una investigación sobre modelado orientado a objetos en la década de 1990, cuando se integraron anotaciones orientadas a objetos similares para crear el lenguaje UML. La versión completa (UML 2) se completó en 2004. UML es ampliamente aceptado como enfoque estándar para crear modelos de sistemas de software. Se proponen opciones de modelado de sistemas más generales. (Sommerville, 2011)

2.6.1 Diagramas UML

2.6.1.1 Diagramas de Clase.

En este diagrama, cada clase se representa como un rectángulo con tres secciones: la primera sección muestra el nombre de la clase, la segunda sección muestra los atributos de la clase y la tercera sección muestra los métodos de la clase.

Figura 5

Diagramas UML



2.7 MÉTRICAS DE CALIDAD

Son indicadores cuantitativos utilizados para medir y evaluar el desempeño, la eficiencia y la efectividad de un producto, proceso o sistema, con el fin de asegurar que cumpla con los requisitos establecidos y las expectativas de los usuarios. Estas métricas son esenciales en el ámbito de la ingeniería de software, pues permiten realizar una evaluación objetiva y continua durante todas las fases del desarrollo (Pressman, 2020).

2.7.1 ISO/IEC 25010

La norma ISO/IEC 25010 define los estándares de calidad del software, abarcando tanto sus características estáticas como dinámicas en los sistemas informáticos. Este modelo se aplica tanto a los sistemas informáticos en general como al software específico.

Esta norma establece ocho características clave, que son:

Funcionalidad: Se refiere a la capacidad del software para cumplir con los requisitos funcionales y llevar a cabo las tareas esperadas de forma precisa y completa.

Desempeño: Es la habilidad del software para realizar las tareas requeridas de manera eficiente, optimizando el uso de los recursos disponibles.

Compatibilidad: Hace referencia a la capacidad del software para interactuar y funcionar correctamente con otros sistemas, entornos o componentes, asegurando la integración adecuada.

Usabilidad: Se refiere a la facilidad con la que los usuarios pueden interactuar con el software, comprendiendo su funcionamiento, incluyendo la interfaz de usuario, la navegación y la facilidad de aprendizaje.

Confiabilidad: Indica la capacidad del software para mantener un rendimiento estable y evitar errores o fallos durante su operación.

Seguridad: Consiste en proteger la información y los datos del software contra accesos no autorizados, garantizando su confidencialidad, integridad y disponibilidad.

Mantenibilidad: Es la facilidad con la que el software puede ser modificado, corregido o mejorado, asegurando que pueda evolucionar y adaptarse a futuras necesidades.

Portabilidad: Se refiere a la capacidad del software de ser trasladado o utilizado en diferentes entornos o plataformas sin perder funcionalidad o rendimiento.

Figura 6

Modelo ISO/IEC



2.8 ESTIMACIÓN DE COSTOS

Este proceso es esencial para la planificación adecuada del proyecto, ya que permite a los interesados (como los desarrolladores, gerentes y clientes) tener una visión clara de los

costos asociados a la creación y operación del sistema a lo largo de su ciclo de vida (Bozman & Huff, 2019).

Costos de desarrollo: Estos incluyen los gastos asociados a la creación del software, como los salarios del equipo de desarrollo, costos de herramientas y tecnologías utilizadas (por ejemplo, servidores, licencias de software), y costos indirectos como la capacitación del personal. Según Sommerville (2019), este es el componente más grande de los costos y debe considerar tanto el tiempo de los desarrolladores como los recursos utilizados en las fases de diseño, codificación y prueba del sistema.

Costos de implementación y despliegue: Una vez que el sistema ha sido desarrollado, se requieren recursos adicionales para su implementación y despliegue en los servidores de producción. Estos costos incluyen la infraestructura necesaria, como servidores, bases de datos y otros recursos de hardware y software, así como los gastos asociados a la migración de datos y la formación de usuarios (Pressman, 2020).

Costos de mantenimiento: Después de la implementación, el sistema web debe mantenerse para asegurar su funcionamiento continuo y actualizado. Los costos de mantenimiento incluyen el soporte técnico, las actualizaciones de seguridad, la corrección de errores, así como la mejora continua del sistema para adaptarse a cambios en los requisitos del negocio o las tecnologías (Bozman & Huff, 2019).

Costos operacionales: Estos son los costos relacionados con la operación continua del sistema, como el alojamiento web, la gestión de bases de datos y los costos de red. También se incluyen los costos asociados con la supervisión del sistema, la implementación de medidas de seguridad y la gestión de la infraestructura (Pressman, 2020).

2.8.1 Método de estimación de COCOMO

(Modelo de Costo Constructivo, 1981) es un modelo utilizado para estimar los costos de proyectos de desarrollo de software, con el objetivo de ayudar en la planificación y ejecución de dichos proyectos. Fue propuesto por Barry Boehm en 1981 y más tarde se actualizó a la versión COCOMO 2.0.

2.8.1.1 COCOMO Intermedio.

Este modelo calcula el esfuerzo necesario en función del tamaño estimado del proyecto, considerando también varios "factores de costo". Estos factores evalúan características específicas del producto, el equipo de trabajo, el personal y el diseño del proyecto. La fórmula asociada a este modelo es la siguiente:

$$KLDC = (LDC)/100$$

Donde:

KLDC = Kilo de líneas de código

LDC = Líneas de código

Tabla 2

Ejecución del modelo intermedio

PROYECTO DE SOFTWARE		A	B	C	D
Orgánico	2,4	1.5	2.5	0.38	
Semicopado	3.0	1.12	2.5	0.35	
Empotrado	2.8	1.20	2.5	0.32	

A continuación, presentaremos las ecuaciones que permitirán realizar el costo total del software.

Tabla 3

Ecuaciones de Modelo COCOMO

VARIABLE	ECUACIÓN	TIPO/UNIDAD
Esfuerzo requerido	$E=a*(KLDC)^b*FAE$	Personas/mes
Tiempo requerido por el Proyecto	$T=c(E)^n$	Meses
Número de personas requeridas para el proyecto	$NP=E/T$	Personas
Costo Total	$CT=Sueldo\ Mes * NP*T$	Dólares

Para hallar los valores de la FAE, se realiza la tabla de atributos multiplicadores.

Figura 7

Calculo de Atributos

CONDUCTORES DE COSTE	VALORACIÓN					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extr. alto
Fiabilidad requerida del software	0,75	0,88		1,15	1,40	-
Tamaño de la base de datos	-	0,94	1,00	1,08	1,16	-
Complejidad del producto	0,70	0,85	1,00	1,15	1,30	1,65
Restricciones del tiempo de ejecución	-	-	1,00	1,11	1,30	1,66
Restricciones del almacenamiento principal	-	-	1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual	-	0,87	1,00	1,15	1,30	-
Tiempo de respuesta del ordenador	-	0,87	1,00	1,07	1,15	-
Capacidad del analista	1,46	1,19	1,00	0,86	0,71	-
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	-
Capacidad de los programadores	1,42	1,17	1,00	0,86	0,70	-
Experiencia en S.O. utilizado	1,21	1,10	1,00	0,90	-	-
Experiencia en el lenguaje de programación	1,14	1,07	1,00	0,95	-	-
Prácticas de programación modernas	1,24	1,10	1,00	0,91	0,82	-
Utilización de herramientas software	1,24	1,10	1,00	0,91	0,83	-
Limitaciones de planificación del proyecto	1,23	1,08	1,00	1,04	1,10	-

Aplicando y las fórmulas correspondientes y la tabla anterior para encontrar el valor de FAE tendremos el resultado final que estará expresando en dólares americanos, posteriormente haremos una conversión a moneda boliviana y obtendremos el costo total del software.

2.9 MÉTRICAS DE SEGURIDAD

Cuentan con indicadores cuantitativos que permiten evaluar la efectividad de las medidas de seguridad implementadas en un sistema web. Estas métricas son esenciales para garantizar que el sistema esté protegido contra amenazas externas e internas, y que cumpla con los estándares de confidencialidad, integridad y disponibilidad de la información (González & López, 2018).

En un sistema web de gestión y venta, las métricas de seguridad son particularmente críticas debido a la naturaleza de las transacciones involucradas, que manejan datos sensibles de los usuarios, como información personal y detalles de pagos.

Las métricas de seguridad se agrupan en varias categorías, que incluyen:

Métricas de autenticación y autorización: Estas métricas miden la efectividad de los procesos de autenticación (verificación de la identidad del usuario) y autorización (control del acceso a diferentes recursos del sistema). Ejemplos de estas métricas incluyen la tasa de fallos de inicio de sesión, la complejidad de las contraseñas y el número de usuarios con privilegios elevados (Mayer, 2020).

Métricas de vulnerabilidades: Evalúan la cantidad y severidad de las vulnerabilidades presentes en el sistema. Estas métricas incluyen el número de vulnerabilidades críticas sin resolver, el tiempo promedio para corregir vulnerabilidades y la frecuencia con que se realizan auditorías de seguridad. El uso de herramientas automáticas

para escanear vulnerabilidades puede proporcionar una medición objetiva de la seguridad de la infraestructura web (Smith, 2019).

Métricas de protección de datos: Estas métricas se centran en la efectividad de las políticas de privacidad y seguridad de la información. Incluyen el porcentaje de datos cifrados en tránsito y reposo, así como la cantidad de incidentes de fuga de datos. En el contexto de un sistema de gestión y venta de terrenos, la protección de los datos personales y las transacciones financieras es clave para mantener la confianza de los usuarios y evitar riesgos legales (González & López, 2018).

Métricas de resiliencia y recuperación ante desastres: Miden la capacidad del sistema para recuperarse de ataques cibernéticos, fallos de hardware o desastres naturales. Esto puede incluir el tiempo de inactividad debido a un incidente de seguridad y el tiempo promedio de recuperación de los sistemas afectados (Mayer, 2020).

2.9.1 Métrica de Seguridad 27001

La **ISO27001** desarrolla los conceptos de “calidad de la seguridad” y “madurez” como principales parámetros que dan la definición de indicadores y métricas. Los pasos que tienen que seguir para adoptar un conjunto de métricas son:

Con ello, obtendremos una pareja de valores que nos dará la información del nivel de evolución del control y de la calidad con la que se ha llevado a cabo la implementación. Unos valores que pueden que sean una imagen real del nivel de seguridad de la empresa.

2.9.1.1 Protocolos según la Norma ISO 27001

Control de acceso: Restringe el acceso a la información únicamente a las personas que tienen autorización para ello.

Seguridad física: Medidas para proteger los activos de información física, como dispositivos de almacenamiento, oficinas, entre otros.

Seguridad lógica: Se centra en proteger la información y los sistemas informáticos mediante el uso de diversos controles.

Respaldo y recuperación: Establece un plan de copias de seguridad periódicas para asegurar la disponibilidad de la información en caso de desastres.

Monitoreo y auditoría: Realiza un seguimiento y revisión constante de los sistemas y registros de seguridad para identificar posibles vulnerabilidades y actividades inusuales.

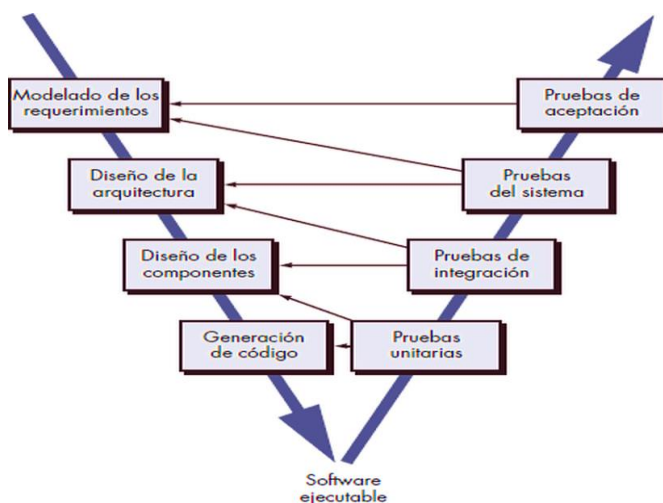
(GlobalSuite Solutions, 2023)

2.10 Modelo en V

El modelo en V es una metodología de desarrollo de software que estructura las etapas de desarrollo y pruebas en una disposición que forma una "V". En este enfoque, cada etapa del desarrollo tiene su correspondiente fase de prueba, permitiendo que las pruebas se lleven a cabo de manera simultánea con el proceso de construcción del software.

Figura 8

Modelo en V



2.11 PRUEBAS DE FUNCIONAMIENTO

Todo sistema desarrollado debe ser sometido a una etapa de pruebas para asegurar el funcionamiento del sistema web. El uso de las pruebas funcionales es para asegurar EL correcto trabajo de entrada de datos, la navegación en el sistema, procedimientos y obtención de resultados.

2.11.1 Pruebas de Caja Negra

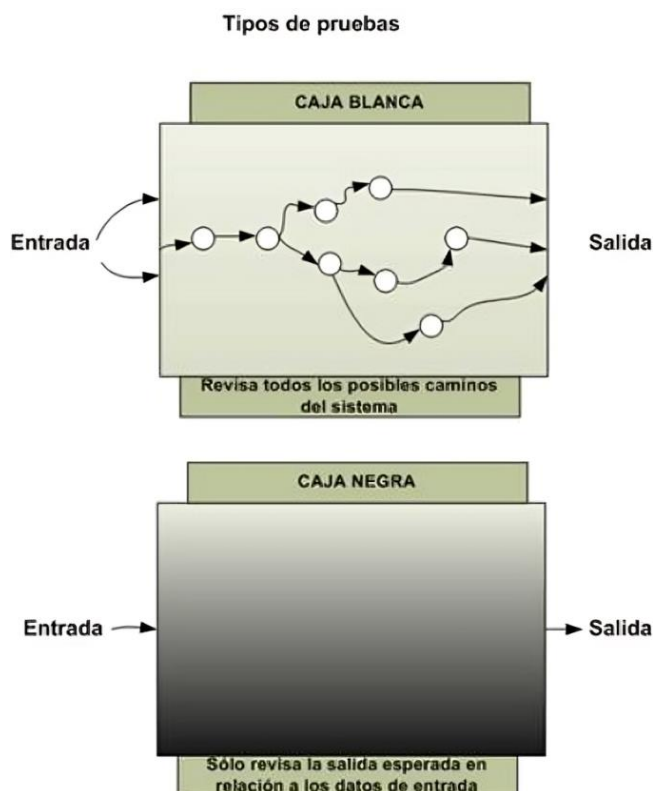
La prueba de caja negra es una técnica de prueba de software en la que el tester evalúa el funcionamiento del sistema sin conocer su estructura interna o el código fuente. En lugar de centrarse en cómo se implementa el sistema, la prueba de caja negra se enfoca en verificar si el sistema produce los resultados correctos a partir de las entradas proporcionadas. Este tipo de prueba es particularmente útil en sistemas web de gestión y venta, donde el objetivo principal es asegurarse de que las funcionalidades del sistema, tales como la compra, la venta, la gestión de pagos y la visualización de terrenos, se ejecuten correctamente desde la perspectiva del usuario final (Black, 2020).

2.11.2 Pruebas de la Caja Blanca

Las pruebas de caja blanca, también conocidas como pruebas estructurales o pruebas basadas en la lógica interna de un programa, se centran en evaluar el código fuente interno de una aplicación. A diferencia de las pruebas de caja negra, donde el tester no tiene conocimiento interno del código, en las pruebas de caja blanca se examinan las estructuras, la lógica y las rutas del código.

Figura 9

Tipos de Prueba de Funcionamiento



2.11.3 Prueba de Stress

Realizar pruebas de estrés en un sistema web de comercio electrónico es crucial para asegurar que el sitio pueda manejar grandes volúmenes de tráfico y transacciones sin fallar.

JMeter: es una herramienta de código abierto diseñada para realizar pruebas de rendimiento y carga en aplicaciones web y otros servicios, como bases de datos, servidores FTP, servicios de correo, entre otros. Fue desarrollada inicialmente por Apache Software Foundation y es utilizada principalmente para evaluar la capacidad de los sistemas bajo condiciones de carga y estrés (Apache JMeter, 2020).

JMeter permite realizar pruebas de carga, rendimiento y funcionalidad de manera eficiente, simulando múltiples usuarios simultáneos para medir la respuesta del sistema en términos de tiempo de respuesta, rendimiento y estabilidad. Su versatilidad y capacidad de personalización lo han convertido en una de las herramientas más populares para pruebas de rendimiento en el ámbito del desarrollo de software (Davis, 2015).

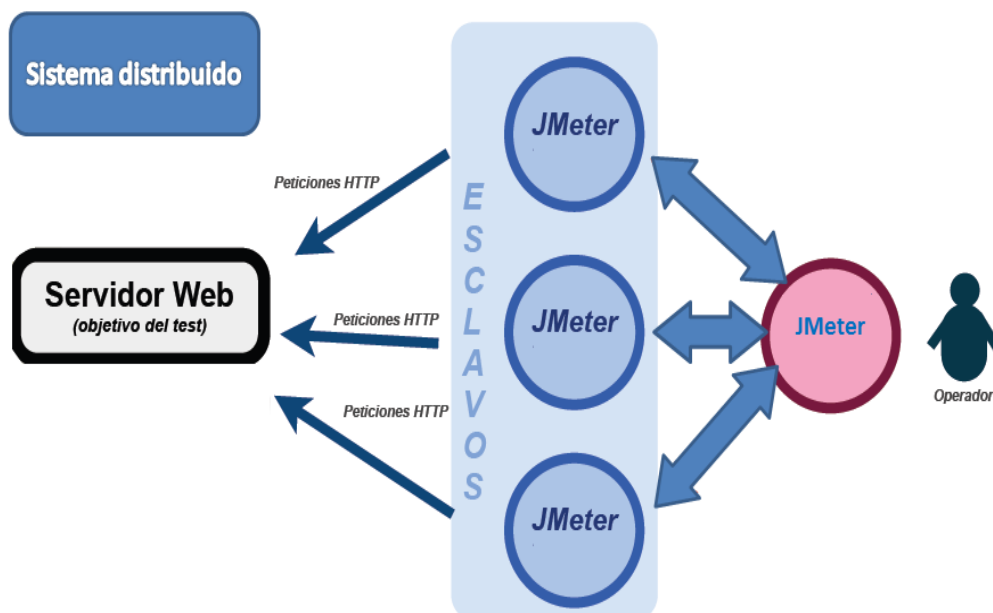
2.11.3.1 Aplicaciones de JMeter

JMeter es utilizado en una amplia variedad de aplicaciones, entre las que se incluyen:

Pruebas de estrés: JMeter puede generar una carga extremadamente alta para evaluar la robustez y estabilidad de un sistema bajo condiciones extremas.

Figura 10

Prueba con JMeter



2.11.4 Pruebas de accesibilidad

Son procesos de evaluación diseñados para garantizar que un producto, servicio o sistema (como sitios web, aplicaciones móviles o dispositivos) sea utilizable por personas con diversas discapacidades, incluyendo aquellas relacionadas con la visión, audición, movilidad o cognición. Estas pruebas verifican el cumplimiento de estándares como las Pautas de Accesibilidad para el Contenido Web (WCAG, por sus siglas en inglés) y promueven la igualdad de acceso y la inclusión digital. (W3C, 2018).

Las pruebas se basan en los siguientes principios:

Perceptible: la información y los componentes de la interfaz de usuario deben ser presentados a los usuarios de modo que ellos puedan percibirlos.

Operable: los componentes de la interfaz de usuario y la navegación deben ser operables.

Comprensible: la información y el manejo de la interfaz de usuario deben ser comprensibles.

Robusto: el contenido debe ser suficientemente robusto como para ser interpretado de forma fiable por una amplia variedad de aplicaciones de usuario, incluyendo las ayudas técnicas.

2.12 HERRAMIENTAS.

Para el desarrollo del proyecto que se señala muchos son las herramientas en las cuales se tomara apoyo.

2.12.1 HTML5

HTML5 es la última versión del Lenguaje de Marcado de Hipertexto (HTML), utilizado para estructurar el contenido de las páginas web. Introduce nuevas etiquetas y características que mejoran la semántica, accesibilidad y compatibilidad multimedia, optimizando la experiencia del usuario en diferentes dispositivos y navegadores (W3C, 2014).

2.12.2 CSS3

CSS3, la tercera versión de las Hojas de Estilo en Cascada (CSS), es un lenguaje utilizado para describir la presentación visual de un sitio web. Permite diseñar estilos avanzados, como animaciones, transiciones y diseño adaptable, facilitando la personalización del aspecto de los elementos HTML (W3C, 2018).

2.12.3 Bootstrap

Bootstrap es un marco de desarrollo de código abierto que combina HTML, CSS y JavaScript para facilitar la creación de interfaces web responsivas y modernas. Ofrece componentes prediseñados, como botones, menús y formularios, junto con una estructura de diseño adaptable basada en un sistema de cuadrícula (Bootstrap, 2023).

2.12.4 JavaScript

JavaScript es un lenguaje de programación interpretado que permite crear contenido dinámico e interactivo en páginas web. Se utiliza para mejorar la experiencia del usuario al añadir funcionalidades como validación de formularios, efectos visuales y comunicación en tiempo real con servidores (Mozilla Developer Network, 2023).

2.12.5 Dompdf

Dompdf es una biblioteca PHP que permite generar archivos PDF a partir de contenido HTML y CSS. Es ampliamente utilizada para crear reportes, facturas y otros documentos personalizables de forma dinámica en aplicaciones web (Dompdf Team, 2023).

2.12.6 Jetstream

Jetstream es un paquete oficial de Laravel que proporciona una solución de inicio para la creación de aplicaciones web, integrando características como autenticación, administración de sesiones y soporte para equipos de usuarios. Ofrece plantillas preconfiguradas utilizando frameworks modernos como Livewire y Inertia.js (Laravel, 2023).

2.12.7 Laravel

Laravel es un framework de desarrollo web de código abierto basado en PHP, diseñado para facilitar la creación de aplicaciones web robustas y escalables. Creado por Taylor Otwell en 2011, Laravel se ha convertido en uno de los frameworks más populares en el desarrollo de aplicaciones web debido a su sintaxis elegante y a la simplicidad que ofrece para resolver tareas comunes (Otwell, 2011).

Según Otwell (2012), “Laravel está diseñado para facilitar el desarrollo de aplicaciones web con sintaxis expresiva y eficiente, permitiendo a los desarrolladores enfocarse en lo esencial sin preocuparse por tareas complejas de configuración y construcción”. Esto lo convierte en una herramienta ideal tanto para principiantes como para desarrolladores experimentados.

2.12.7.1 Características de Laravel

Laravel cuenta con una serie de características que lo diferencian de otros frameworks de PHP y facilitan el desarrollo web ágil y eficiente:

Estructura MVC (Modelo-Vista-Controlador): Laravel utiliza el patrón de arquitectura MVC, separando la lógica de la aplicación en componentes (modelo, vista y controlador) que mejoran la organización del código y su mantenimiento (Fowler, 2003).

Sistema de migración de bases de datos: Laravel incluye herramientas para la migración y gestión de bases de datos, permitiendo modificar la estructura de las tablas de manera controlada y sencilla.

2.12.8 PHP

PHP (acrónimo de Hypertext Preprocessor, originalmente Personal Home Page) es un lenguaje de programación de código abierto y propósito general, especialmente diseñado para el desarrollo web. Fue creado en 1994 por Rasmus Lerdorf y se caracteriza por su capacidad de incrustarse en el HTML de las páginas web, lo cual facilita la creación dinámica de contenido en servidores web (Lerdorf, 1995). PHP es actualmente uno de los lenguajes más utilizados para desarrollar aplicaciones y sitios web, y su popularidad se debe a su simplicidad y efectividad en entornos de desarrollo web (Atkinson & Holscher, 2018).

2.12.8.1 Características de PHP

Simplicidad y facilidad de aprendizaje: PHP es fácil de aprender para los principiantes y, al mismo tiempo, es lo suficientemente potente para desarrollar aplicaciones complejas.

Integración con HTML y bases de datos: PHP se integra fácilmente con HTML, lo que permite generar contenido dinámico de forma rápida. También ofrece soporte para una amplia gama de bases de datos, como MySQL, PostgreSQL y SQLite.

Amplia portabilidad: Es compatible con múltiples sistemas operativos (Windows, Linux, macOS) y se puede ejecutar en la mayoría de servidores web (Apache, Nginx), lo cual asegura su portabilidad.

Manejo de sesiones y cookies: PHP facilita el almacenamiento y la recuperación de datos de sesión y cookies, lo cual es fundamental en aplicaciones web que requieren autenticación de usuarios y personalización de contenidos.

2.12.9 MYSQL

El software MySQL es un motor de base de datos relacional de código abierto creado en 1994 por la empresa sueca MYSQL AB. En 2008, A finales de 2010, Oracle adquirió MySQL.

Está programado en C/C++, es un motor de base de datos conectado a MySQL y se ejecuta a través de un cliente/servidor. Un servidor MySQL escucha conexiones, y un cliente (o varias clientes) se están conectando a él. Utiliza SQL como lenguaje de consulta que nos permite insertar y recuperar datos de la base de datos. (Urriaga, 2020)

2.12.10 Visual Studio Code

Visual Studio Code (VS Code) es un editor de código fuente desarrollado por Microsoft que se utiliza ampliamente para el desarrollo de aplicaciones web, de escritorio y móviles. A diferencia de un entorno de desarrollo integrado (IDE) completo, VS Code es un editor de texto ligero pero potente, con una amplia gama de extensiones que le permiten adaptarse a

diferentes lenguajes de programación y tecnologías. Fue lanzado por primera vez en 2015 y ha crecido rápidamente en popularidad debido a su flexibilidad, velocidad y facilidad de uso (Microsoft, 2015).

Según su documentación oficial, “Visual Studio Code es un editor de código fuente que ofrece soporte para depuración, control de versiones, autocompletado de código y una amplia variedad de extensiones, todo integrado en un entorno sencillo y accesible” (Visual Studio Code Documentation, 2020). Este enfoque lo hace adecuado tanto para desarrolladores novatos como para programadores experimentados.

2.12.10.1 Características de Visual Studio Code

Ligereza y rapidez: A pesar de ser un editor de texto potente, VS Code es ligero y rápido en su rendimiento, lo que lo hace ideal para proyectos pequeños y grandes sin afectar el rendimiento del sistema.

Extensiones y personalización: VS Code cuenta con un mercado de extensiones que permite personalizar el editor para trabajar con distintos lenguajes de programación como JavaScript, Python, PHP, C++, entre otros. Esto le otorga una gran versatilidad, permitiendo que los desarrolladores adapten el entorno de trabajo a sus necesidades.

Terminal integrada: VS Code incluye una terminal que permite ejecutar comandos directamente desde el editor, lo cual facilita la ejecución de scripts y la gestión de proyectos sin tener que alternar entre ventanas o aplicaciones.

CAPITULO III

MARCO APLICATIVO

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



CAPITULO III

3 MARCO APLICATIVO

En el presente capítulo se realiza: el análisis, desarrollo implementación del sistema web, aplicando la metodología UWE, normas y técnicas mencionadas en el marco teórico que permite evolucionar adecuadamente en el diseño del sistema.

3.1 INTRODUCCIÓN

En el presente capítulo se realiza: el análisis, desarrollo e implementación del sistema web de gestión y venta de terrenos, aplicando la metodología UWE, normas y técnicas mencionadas en el marco teórico del capítulo II, que permite evolucionar adecuadamente en el diseño del sistema.

Al estar basado en la web, este tipo de sistema es accesible desde cualquier dispositivo con conexión a Internet, lo que permite a los usuarios gestionar las ventas de manera remota y en tiempo real.

3.2 ANÁLISIS DE LA SITUACIÓN ACTUAL

Inmobiliaria KOLLITA es una empresa dedicada y comprometida en el sector de bienes raíces, conocida por su trato personalizado y enfoque en la satisfacción del cliente. Su equipo de agentes inmobiliarios ofrece una atención profesional y detallada a cada cliente, acompañándolos desde la búsqueda del terreno ideal hasta el proceso de cierre de la venta. Sin embargo, con el auge de la tecnología y la necesidad de procesos más eficientes, KOLLITA tiene una gran oportunidad para dar el siguiente paso: implementar un sistema web que potencie su gestión y expanda su alcance.

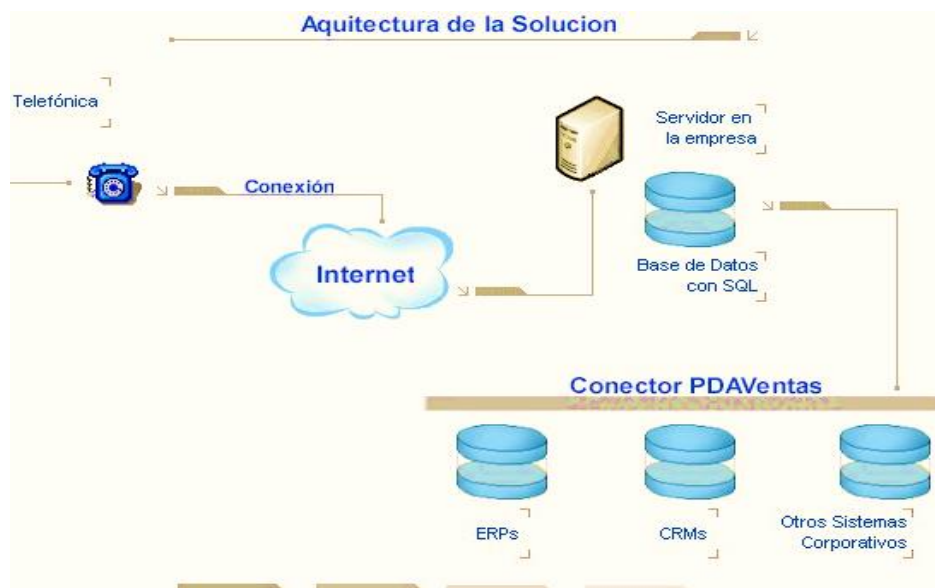
Con un sistema web, Inmobiliaria KOLLITA podría digitalizar y agilizar varios de sus procesos, ofreciendo a sus clientes una experiencia mucho más accesible y rápida. Una

plataforma en línea le permitiría a los clientes buscar terrenos, ver detalles específicos, y hasta coordinar visitas de manera rápida y cómoda desde cualquier dispositivo. Además, el sistema facilitaría la gestión de documentos y contratos de venta, permitiendo a los agentes trabajar de forma más organizada y eficiente, mientras los administradores tienen acceso a reportes e información actualizada en tiempo real.

3.3 ESQUEMA DEL SISTEMA WEB

Figura 11

Arquitectura Sistema Web



3.4 ANÁLISIS DE LOS REQUERIMIENTOS

En este punto, se detallan los requerimientos funcionales y no funcionales claves que definen el comportamiento y que funciones debe cumplir el sistema para satisfacer las necesidades del usuario.

3.4.1 Obtención de Requerimientos

3.4.1.1 Requerimientos funcionales

En los requisitos funcionales se muestran las características que necesita el sistema a partir de la información obtenida como parte de las tareas de obtención de requisitos. Estas se describen a continuación en la siguiente tabla.

Tabla 4

Ecuaciones de Modelo COCOMO

Nro. REQ	FUNCIÓN	DESCRIPCIÓN	EVIDENTE
RF-01	Gestionar Usuarios	Permite al administrador registrar, modificar y eliminar usuarios.	Evidente
RF-02	Iniciar Sesión	Permite a los usuarios iniciar sesión en el sistema.	Evidente
RF-03	Gestionar Terrenos	Permite registrar, modificar y eliminar terrenos al sistema.	Evidente
RF-04	Gestionar Ventas	Permite a los usuarios registrar ventas y generar recibos.	Evidente
RF-05	Gestionar Ubicaciones	Permite registrar ubicaciones de los lotes	Evidente
RF-06	Generar Reporte de Ventas	Genera informes detallados de ventas en un período específico.	Evidente
RF-07	Gestionar Clientes	Permite registrar, editar y eliminar información de los clientes.	Evidente
RF-08	Configurar Perfil	Permite al usuario personalizar su perfil.	Evidente
RF-09	Cerrar Sesión	Permite al usuario cerrar su sesión actual y regresar a la página de inicio.	Evidente

3.4.2 Requerimientos No Funcionales

En tabla que se verá a continuación, se describen los requerimientos no funcionales para el sistema:

Tabla 5

Planilla de Uso de Caso - Gestionar Usuarios

Nro. REQ	FUNCIÓN	EVIDENTE
El sistema debe ser compatible con los navegadores más utilizados:		
RNF-01	Google Chrome, Mozilla Firefox, Microsoft Edge, Brave, Opera y Safari.	Evidente
RNF-02	El sistema debe cargar páginas en un tiempo máximo de 5 segundos.	Evidente
RNF-03	El sistema debe contar con información cifrada de las contraseñas de los usuarios.	Evidente
RNF-04	La interfaz de usuario debe ser intuitiva y fácil de navegar para los usuarios.	Evidente
RNF-05	El sistema debe contar con autenticación para garantizar el acceso a la información.	Evidente

3.5 DISEÑO

3.5.1 Metodología UWE- construcción e implementación del sistema

UWE ofrece un conjunto de herramientas y técnicas para representar los elementos clave en sistemas web mediante diagramas UML, permitiendo una comprensión clara de la funcionalidad y la arquitectura del sistema.

3.5.2 Actores



Los actores representan una función que puede realizar alguien o algo que interactúa o quiere comunicarse con el sistema, pero que está fuera del sistema.

3.5.3 Detalles de Actores de Caso de Uso

Para verificar los actores se nos permitió conocer a las personas encargadas en el proceso de control de los terrenos en la inmobiliaria “KOLLITA” con el propósito de formar casos de uso como se describe en la siguiente tabla.

Tabla 6

Actores

TIPO DE ACTOR	DETALLE
Administrador 	Es el usuario con amplios privilegios y responsabilidades en la gestión y supervisión del sistema. Su función principal es garantizar el correcto funcionamiento, la seguridad y el mantenimiento de la plataforma
Cliente 	Es el usuario final o consumidor de un producto o servicio proporcionado por el sistema.

3.5.4 Diagrama de Casos de Uso del Sistema

En el diagrama de caso de uso general siguiente se detallan las acciones llevadas a cabo. Además, se presentan los actores involucrados en cada transacción de venta, incluyendo los clientes que participan en estas interacciones.

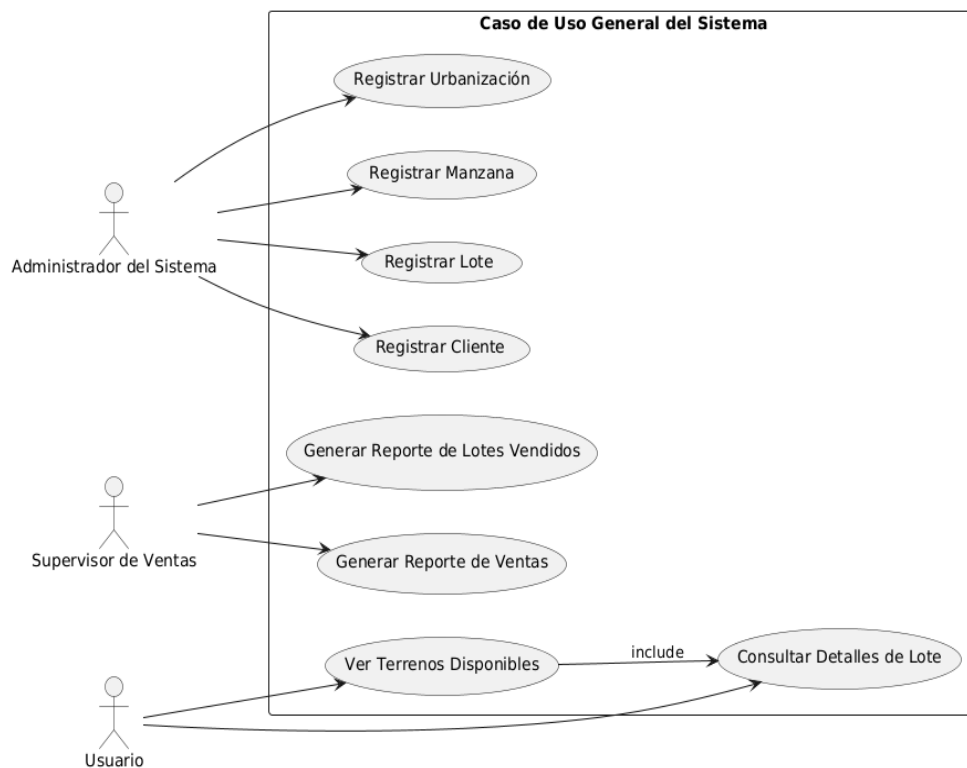
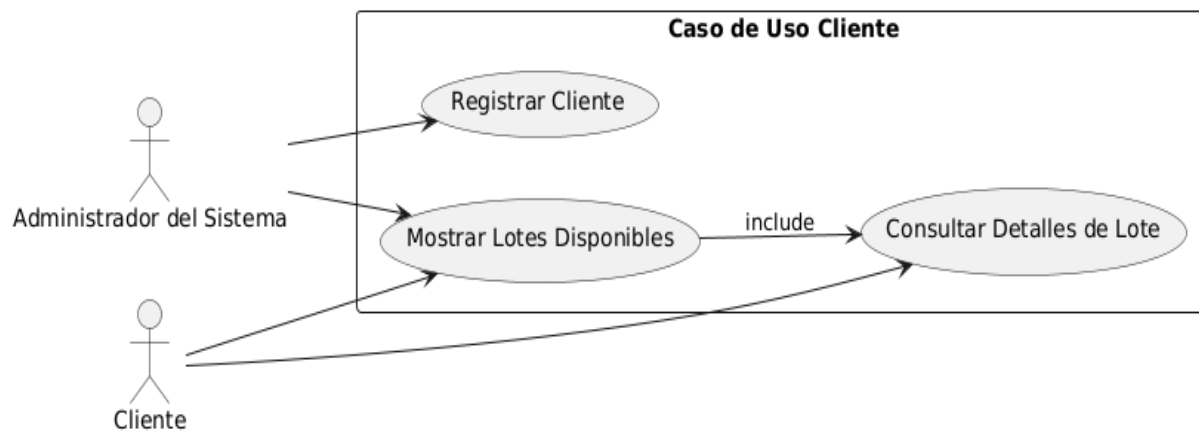
Figura 12*Caso de Uso General del Sistema***Figura 13***Caso de Uso Clientes*

Tabla 7

Plantilla de Caso de Uso Gestionar Clientes

Caso de Uso	Gestionar Clientes
Actores:	Administrador y Supervisor de Ventas
Propósito:	Tiene como propósito permitir a los actores gestionar la información de Clientes en el sistema.
Resumen:	El administrador y supervisor de ventas pueden llevar a cabo acciones relacionadas con la gestión de clientes. Esto incluye la creación, modificación y eliminación de registros de clientes, así como la actualización de sus datos personales, preferencias y necesidades.
Tipo:	Primario (Supervisor de Ventas) Secundario (Administrador) Esencial

Figura 14

Caso de Uso Gestionar Usuarios Caso de Uso Gestionar Usuarios

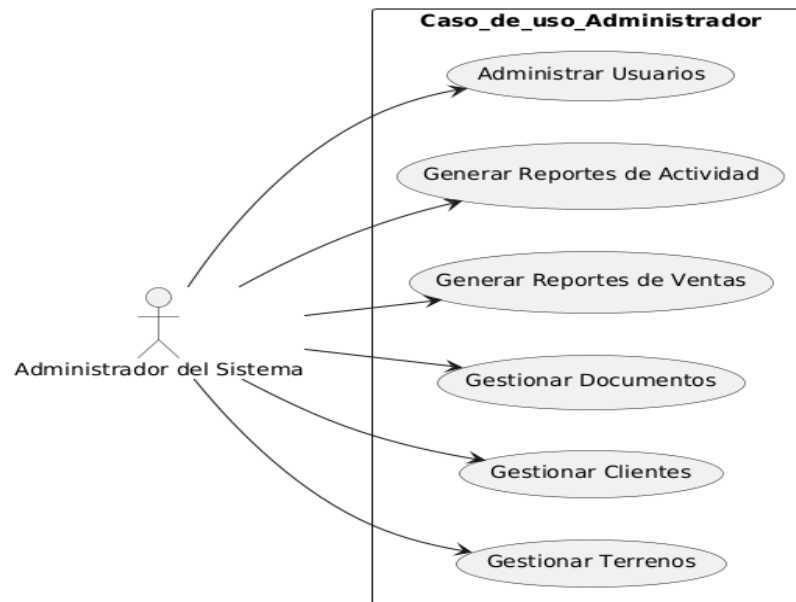


Tabla 8

Plantilla de Caso de Uso Gestionar Usuarios

Caso de Uso	Gestionar Usuarios
Actores:	Administrador
Propósito:	Permite a los administradores del sistema gestionar las cuentas de los usuarios en el sistema. Concede a los administradores realizar acciones relacionadas con la gestión de cuentas de usuario, como crear, editar, eliminar, restablecer contraseñas, asignar roles y visualizar la lista de usuarios registrados.
Resumen:	registrados.
Tipo:	Primario Esencial

Figura 15

Caso de Uso Registrar Ventas

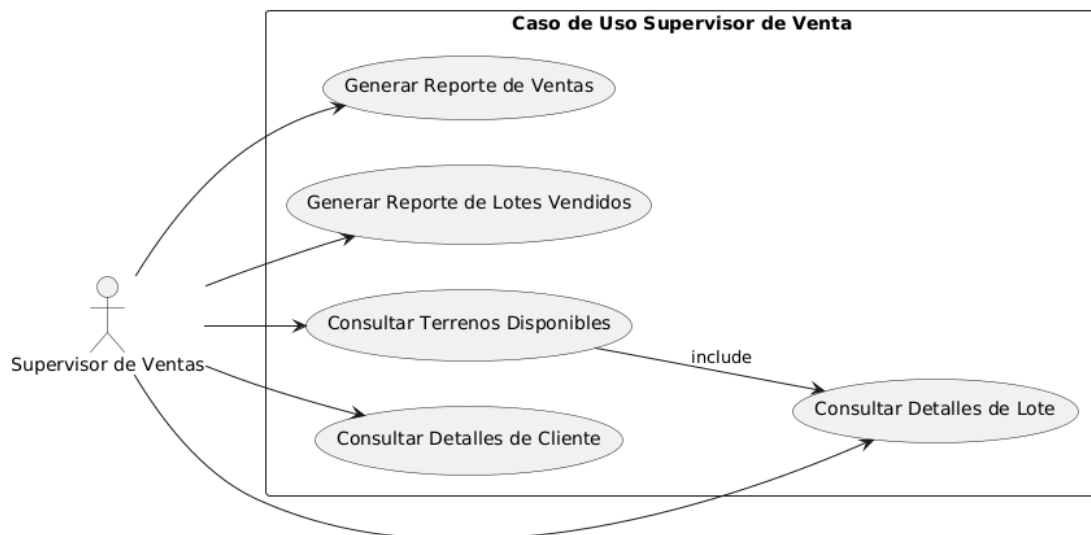


Tabla 9*Plantilla de Caso de Uso Registrar Venta*

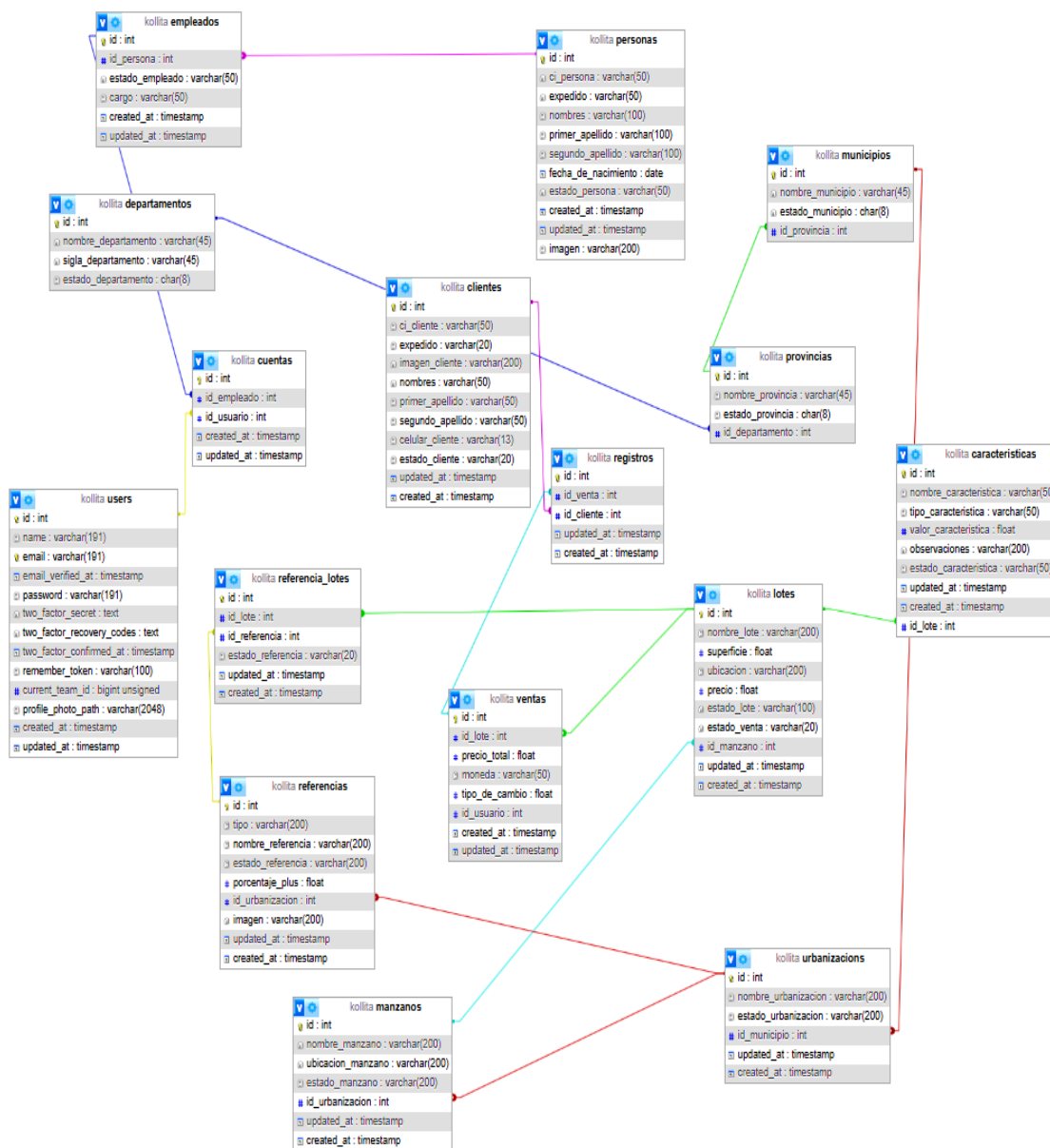
Caso de Uso	Registrar Ventas
Actores:	Supervisor de Ventas
Propósito:	Este caso de uso tiene como propósito permitir a los vendedores registrar ventas en el sistema.
Resumen:	Los supervisores de venta pueden llevar a cabo la acción principal de registrar ventas. Esto incluye la selección de lotes vendidos, la asignación de clientes correspondientes, el registro, detalles de transacción, y la generación de recibos de venta.
Tipo:	Primario Esencial

3.6 Diseño conceptual.

3.6.1 Modelo Entidad Relacional.

Figura 16

Modelo Entidad Relación de las Tablas



3.7 FASE DE CONSTRUCCIÓN

En esta etapa se presenta el desarrollo de la interfaz de usuario y sus componentes construidos. Para el sistema, se muestra la siguiente secuencia de pantallas básicas.

3.7.1 Despliegue del Sistema web

- El proceso de implementación del sistema implica una serie de pasos cruciales:
- Configuración de servidores: Instalar y ajustar los servidores necesarios, como Apache, para respaldar el sistema.
- Configuración de la base de datos: Establecer el motor de base de datos, como MySQL, y diseñar su estructura para satisfacer las necesidades del sistema.
- Despliegue del sistema: Implementar el sistema en el servidor de la pyme, garantizando su disponibilidad para su uso.
- Gestión de accesos: Establecer credenciales de acceso a la base de datos para garantizar la seguridad de la información.
- Capacitación de usuarios: Proporcionar formación a los usuarios finales para que puedan utilizar eficazmente el sistema.

3.7.2 Capacitación del Usuario

El objetivo de esta capacitación es dotar a los usuarios de las habilidades y conocimientos necesarios para utilizar eficazmente el Sistema Web de Gestión de la inmobiliaria "KOLLITA", optimizando la gestión, ventas, vistas, usuarios y perfiles. Además, busca familiarizar a los usuarios con la interfaz y funcionalidades del sistema para que puedan gestionar sus actividades diarias de manera eficiente.

La capacitación se realizará en un taller práctico con una duración máxima de una hora. Este tiempo es suficiente para cubrir los aspectos esenciales del sistema, permitiendo a los

participantes aplicar lo aprendido en tiempo real sin sentirse abrumados, a continuación, se describe el contenido de la capacitación:

Acceso al Sistema: Enseñar a los usuarios cómo iniciar sesión y acceder al sistema.

Gestión de Terrenos: enseñar a los usuarios a agregar, editar y eliminar productos.

Gestión de Ventas: Mostrar a los usuarios cómo manejar las ventas dentro del sistema.

Gestión de Usuarios: Enseñar la administración de usuarios dentro del sistema.

Gestión de Perfiles: Capacitar en la configuración y administración de perfiles de usuario.

La Metodología de la Capacitación será con:

Demostraciones en Vivo: El capacitador mostrará en tiempo real cómo realizar cada tarea dentro del sistema.

Prácticas Guiadas: Los participantes realizarán actividades prácticas bajo la supervisión del capacitador.

Sesión de Preguntas y Respuestas: Al final del taller, se dedicará tiempo para responder preguntas y resolver dudas.

Recursos Necesarios

- Computadoras con acceso al Sistema Web
- Manuales de usuario y guías rápidas.
- Proyector y pantalla para las demostraciones.

CAPITULO IV

METRICAS DE CALIDAD, COSTO Y SEGURIDAD

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PUBLICA DE EL ALTO



CAPÍTULO IV

4 INTRODUCCIÓN

En el desarrollo de cualquier sistema las métricas de calidad y de costo de modelo son indicadores clave que permiten evaluar y comparar el desempeño y la eficiencia de un modelo o sistema en desarrollo. En el contexto del presente proyecto, se presentan en detalle las pruebas exhaustivas a las que ha sido sometido el sistema, evaluando su rendimiento, su usabilidad y su capacidad de satisfacer las necesidades de los usuarios finales.

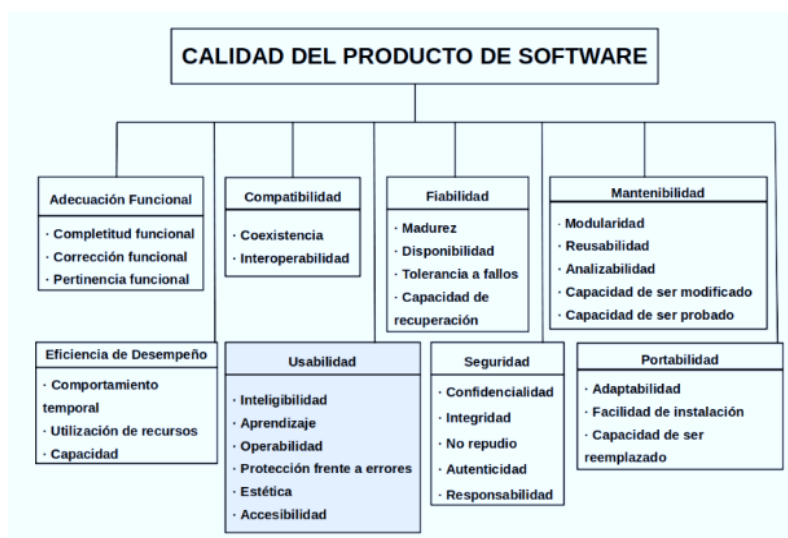
4.1 MÉTRICAS DE CALIDAD, COSTO Y SEGURIDAD

En este capítulo se describirá la calidad, el costo y la seguridad de un sistema de software mediante el uso de diversas métricas. Estas métricas son esenciales para evaluar y asegurar que el sistema cumple con los requisitos y expectativas establecidos. A continuación, se detallarán las métricas correspondientes a cada uno de estos aspectos.

4.2 MÉTRICAS DE CALIDAD

Figura 17

Modelo de Calidad de Producto de Software



4.2.1 Funcionalidad

La Funcionalidad del presente este proyecto se centra en ofrecer una amplia gama de capacidades destinadas a satisfacer las necesidades y requerimientos de nuestros usuarios finales. Esto incluye, pero no se limita a, la implementación de las siguientes funcionalidades clave:

4.2.1.1 Funciones de tipo Entrada Externa (EE).

Tabla 10

Funciones de Entradas Externas

Nro.	Descripción	Ponderación
1	Generación de Recibos de Venta	10
2	Generación de Proformas	10
3	Generación de Reporte de Ventas	6
4	Generación de Reporte de Terrenos	8
5	Generación de Reporte de Urbanizaciones	6
6	Exportación de Datos en PDF	5
Total		42

Tabla 11

Funciones de Salida Externa

Nro.	Descripción	Ponderación
1	Generación de Recibos de Venta	10
2	Generación de Proformas	10
3	Generación de Reporte de Ventas	6
4	Generación de Reporte de Terrenos	8
5	Generación de Reporte de Urbanizaciones	6
6	Exportación de Datos en PDF	5
Total		42

4.2.1.2 Funciones de Tipo Consulta Externa (EQ).

Tabla 12

Escala de Valores de las Preguntas

Nro.	Descripción	Ponderación
1	Consulta de Terrenos Disponibles	8
2	Consulta de Terrenos Vendidos	8
3	Búsqueda de Recibos de Ventas	6
4	Consulta de Información de Clientes	7
5	Consulta de Información de Ventas	7
7	Consulta de Información de Usuarios	5
8	Consulta de Información de Ubicación	7
Total		48

4.2.2 Confiabilidad

Para realizar el cálculo del nivel de confiabilidad en el sistema se deben tomar en cuenta la cantidad de fallas que se producen en un tiempo determinado de funcionamiento $F(t)$, para hallar $F(t)$ se hace uso del valor hallado durante el cálculo de la funcionalidad del sistema realizado y se utiliza en la siguiente fórmula:

Ecuación de Tiempo de Funcionamiento

$$F(t) = f * e^{-u*t}$$

Considerando un periodo de 15 días como tiempo de prueba donde se obtiene que en cada 10 ejecuciones y tareas resultan en 1 falla.

Se obtienen los siguientes datos:

f : Funcionalidad del sistema = 0.837

u: probabilidad de errores en el sistema = $1/10 = 0.1$

t: tiempo de prueba que se le realizo al sistema = 15 días

calculando y reemplazando en la fórmula de ecuación de tiempo de funcionamiento:

$$F(t) = 0.837 * e^{-0.1*15}$$

$$F(t) = 0.837 * e^{-1.5}$$

$$F(t) = \mathbf{0.186}$$

Tenemos que el tiempo de funcionamiento es de un **0.186**

Ya que logramos conseguir el tiempo de funcionamiento posteriormente se calcula la confiabilidad reemplazando el valor hallado en la fórmula:

Ecuación de Confiabilidad

$$P(T \leq t) = \mathbf{1 - F(t)}$$

$$P(T \leq t) = 1 - 0.186$$

$$P(T \leq t) = 1 - 0.186 = \mathbf{80\%}$$

Para finalizar la confiabilidad del sistema es de un 80% en el tiempo de pruebas de 15 días.

4.2.3 Usabilidad

A continuación, se realiza el cálculo de la usabilidad del sistema tomando en cuenta la facilidad de uso (FU) para los usuarios y su valoración hacia el sistema, esta se determina usando la fórmula:

Ecuación de Facilidad de Uso

$$FU = \sum \frac{Xi}{n} * 100$$

Donde:

Xi: Es la sumatoria de Valores

n: Indica el número de preguntas

Las preguntas contienen una escala de valor definidas por la siguiente tabla:

Tabla 13

Escala de Valores de las Preguntas

N°	Escala	Valor
1	Muy Bueno	5
2	Bueno	4
3	Regular	3
4	Malo	1

Tabla 14

Preguntas para Determinar la Usabilidad del Sistema

Nro.	Preguntas	SI	NO	Evaluación
1	¿El sistema brinda la confiabilidad de uso?	4	1	80%
2	¿Las operaciones del sistema son controladas?	4	1	80%
3	¿El sistema permite la retroalimentación de la Información?	4	1	80%
4	¿El sistema cuenta con una interfaz de usuario amigable y agradable a la vista?	5	0	100%
5	¿Las respuestas del sistema son satisfactorias?	5	0	80%
6	¿Las funciones del sistema le parecen complicadas?	3	2	50%
7	¿El sistema facilita el trabajo de los usuarios?	5	0	100%
8	¿Durante el proceso de funcionamiento, el sistema produjo errores?	0	5	90%
TOTAL, USABILIDAD				85%

En la tabla se muestra el resultado de las preguntas realizadas a 3 integrantes de la Inmobiliaria KOLLITA para medir el nivel de usabilidad del sistema.

Con los datos obtenidos, calculamos la usabilidad del sistema:

$$FU = \sum \frac{Xi}{n} * 100$$

$$FU = \sum \frac{660}{8} * 100$$

$$FU = 83\%$$

Realizando un análisis del resultado obtenido, existe un 85 % de comprensión o entendimiento de los usuarios con respecto a la capacidad del sistema.

4.2.4 Mantenibilidad

El mantenimiento de un sistema se realiza con el fin de cambiar o mejorar funcionalidades, en respuesta a los nuevos requerimientos que puedan surgir.

Para obtener los datos de madurez del sistema (IMS), se emplea la siguiente ecuación.

El IMS del software se puede calcular haciendo uso de la fórmula:

Ecuación de Mantenibilidad

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

Tabla 15*Valores para Determinar la Mantenibilidad*

Descripción	Valor
Mt: Número de módulos de la versión actual	13
Fc: Número de módulos en la versión actual que se han modificado	1
Fa: Número de módulos en la versión actual que se han añadido	0
Fd: Número de módulos de la versión anterior que se han borrado	0

Reemplazando en la fórmula:

$$IMS = \frac{13 - (0 + 1 + 0)}{13}$$

$$IMS = 0.9230$$

Por lo tanto, el sistema tiene un índice de estabilidad de 93.75% de facilidad de mantenimiento, y el restante 6.25% es el margen de error correspondiente a los cambios y las modificaciones.

4.2.5 Portabilidad

La portabilidad define la facilidad de llevar el sistema de un entorno a otro sin mucho esfuerzo y sin ningún problema. El presente proyecto está diseñado en un entorno de acceso vía Web y mide su portabilidad en el lado del Servidor y en el lado del cliente, para este tipo de sistema se toma en cuenta 3 aspectos:

- Acceso a internet.

- El equipo debe contar con un servidor Apache que ejecute PHP.
- Computadora con sistema operativo Windows o Linux, con base de datos MySQL.

Aplicando la fórmula:

Ecuación Grado de Portabilidad

$$GP = 1 - \frac{ET}{ER}$$

$$GP = 1 - \frac{5}{100}$$

$$GP = 0.95$$

Por tratarse de un sistema Web, el proyecto presenta una gran portabilidad, ya que los requerimientos para la implementación y ejecución del software son mínimos, por ellos a criterio propio se brinda un 95% de Portabilidad.

4.2.5.1 Resultados finales de Calidad

Tabla 2

Resultados Finales

ELEMENTO	PONDERACIÓN
Funcionalidad	87%
Confiabilidad	80%
Mantenibilidad	87%
Usabilidad	85%
Portabilidad	95%
Estimación Total	88.15%

4.3 ESTIMACIÓN DE COSTO

La utilización de COCOMO en el presente proyecto tiene como propósito fundamental estimar el esfuerzo, el tiempo, el costo y personas requeridas para el desarrollo del sistema.

4.3.1 Método De Estimación COCOMO

La estimación de costos para el presente sistema se ha desarrollado utilizando la métrica KLDC (kilo-líneas de código), como se detalla a continuación. Para calcular el esfuerzo requerido, es necesario determinar la variable KLDC. Este proyecto comprende 8093 líneas de código implementadas en el lenguaje PHP.

Aplicando las conversiones se tiene:

$$\text{KLDC} = (\text{LDC})/1000$$

$$\text{KLDC} = 9812/1000$$

$$\text{KLDC} = 9.8 \text{ KLDC}$$

Es un modelo intermedio y sistema orgánico la evaluación del sistema ha sido considerada bajo 9812 KLDC los coeficientes que se usaran serán los valores que detallen en las siguientes tablas:

Tabla 16

Aplicación del Modelo Intermedio

PROYECTO SOFTWARE	A	B	C	D
Orgánico	3,2	1,05	2,5	0,38
Semiacoplado	3,0	1,12	2,5	0,35
Empotrado	2,8	1,20	2,5	0,32

A continuación, se presentarán las ecuaciones necesarias para calcular el costo total del software.

Tabla 17

Ecuación de Modelo COCOMO

VARIABLE	ECUACIÓN	TIPO/UNIDAD
Esfuerzo requerido por el proyecto	$E=a*(KLDC)^b*FAE$	Personas/mes
Tiempo requerido por el proyecto	$T=c*(E)^n$	Meses
Número de personas requeridas para el proyecto.	$NP=E/T$	Personas

Para hallar los valores de la FAE, se realiza se utilizará la tabla de atributos multiplicadores. (ver página 33)

Se tiene un total $FAE=0,72$

Aplicando y remplazando a los valores de la fórmula de esfuerzo, se tiene:

$$E= a*(KLDC)^b * FAE$$

$$E= 3,2*(9.8)^{1.05}*0,72$$

$$E= 25.3(\text{Personas/Mes})$$

Cálculo del Tiempo

$$T=c*Esfuerzo^d$$

$$T=2,5*25^{0,38}$$

$$T=8.5(\text{Meses})$$

Cálculo de la Productividad

$$\text{PR} = \text{LCD/ESFUERZO}$$

$$\text{PR} = 9812/25$$

$$\text{PR}=392,48(\text{LCD/Personas Mes})$$

Cálculo del personal requerido

$$\text{P}=\text{E/T}$$

$$\text{P}=25,3/8,5$$

$$\text{P}=25,3/8,5$$

P=2,97 (**Personas**) equivalente a 3 personas

4.3.2 Costo Total del Proyecto

$$(\text{Coste Mes}) = \text{P} * \text{Salario medio de programadores}$$

$$\text{Costo Persona Mes } 350\$$$

$$\text{Coste Mes} = 3*350 = 1050 \text{ \$us en Bs.- } 7245$$

$$\text{Costo Total} = 1050*9 = 9400 \text{ \$us en Bs.-} 64860$$

En resumen, los costos que se obtiene estimado que 3 personas a un trabajo de 8 meses resultan un costo total de 8400\$us equivalente en bolivianos 58800 Bs.

4.4 SEGURIDAD

4.4.1 Seguridad de Información ISO - 27001

ISO 27001 es un estándar internacional que define los requisitos para establecer, implementar, mantener y mejorar continuamente un Sistema de Gestión de Seguridad de la Información (SGSI). Su objetivo principal es proteger la confidencialidad, integridad y disponibilidad de la información dentro de una organización. Implementar esta norma ayuda a gestionar los riesgos de seguridad de manera sistemática, garantizando que la información

está protegida contra accesos no autorizados, alteraciones y pérdidas (Normas ISO) (Software ISO) (Rextie).

4.4.2 Seguridad del Sistema

Se implementó un módulo de control de acceso para garantizar que solo usuarios autorizados puedan ingresar al sistema. Este módulo valida y permite el acceso mediante un nombre de usuario y una contraseña asignados por el administrador. Asimismo, registra la identidad de los usuarios que realizan cambios en la base de datos, almacenando su identificador en cada tabla al actualizar la información.

4.4.3 Seguridad a Nivel de Base de Datos

Con el objetivo de garantizar que únicamente los usuarios autorizados puedan acceder a la base de datos, se aplican estrictas políticas de control de acceso. Adicionalmente, se llevan a cabo respaldos periódicos y se diseñan estrategias de recuperación ante desastres para reducir al mínimo el riesgo de pérdida de información en caso de imprevistos.

4.4.4 Implementación y Beneficios

La implementación de ISO 27001 proporciona una estructura sólida para gestionar y proteger la información, asegurando los datos de los clientes estén seguros y cumplan con los estándares internacionales. (NIBNORCA, 2024)

4.5 PRUEBAS AL SOFTWARE

Una vez finalizado el ciclo de vida del desarrollo, se llevan a cabo pruebas exhaustivas para garantizar el correcto funcionamiento del sistema web, basándose en casos de uso representativos del sistema. Estas pruebas funcionales tienen como objetivo asegurar que la

entrada de datos, la navegación por el sistema, los procedimientos y la obtención de resultados se realicen correctamente.

Tabla 18

Pruebas funcionales del Sistema

Procedimientos	Descripción	Valor
Prueba previa requerida Usuario	Registro de usuarios Administrador	Si

Tabla 21

Evaluaciones preliminares del sistema antes de su puesta en marcha

Procedimientos	Resultados Esperados	Calificación de Funcionalidad
Ingresar al Sistema con email contraseña	Validación de credenciales el sistema el ingreso	Si
Fallas encontradas	Descripción	Gravedad
Ninguna	Ninguna	
Pasos de Prueba	Resultados esperados	Acceso del Sistema

4.5.1 Pruebas de Caja Blanca

Para llevar a cabo las pruebas de la caja blanca, también conocidas como pruebas de la organización lógica, se analiza la estructura interna del código, sin considerar los detalles de las reincorporaciones y salidas. Para ello, se sigue el flujo del programa para verificar la lógica del diseño desde una perspectiva aritmética

$V(G) = \text{Numero de condiciones} + 1$

Dado que:

$V(G) = 2$

$V(G) = 2$, se procede a determinar el número de caminos independientes que deben seguirse para realizar los casos de prueba.

Camino: 1,2,3,11,12. Caso de prueba del camino 1: el usuario que accede al sistema web debe familiarizarse con el funcionamiento de la interfaz del sistema web dado que su manejo es sencillo y fácil de usar.

Camino: 1,2,3,4,5,6,7,8,9,10,12. Caso de prueba del camino 2: El usuario inicia sesión exitosamente en el sistema web y obtiene privilegios para gestionar sistema web, lo que incluye la capacidad de administrar y actualizar lotes y precios. La validación de cada etapa realizada garantiza que se alcance el objetivo previsto y que el sistema web cumpla con su propósito diseñado.

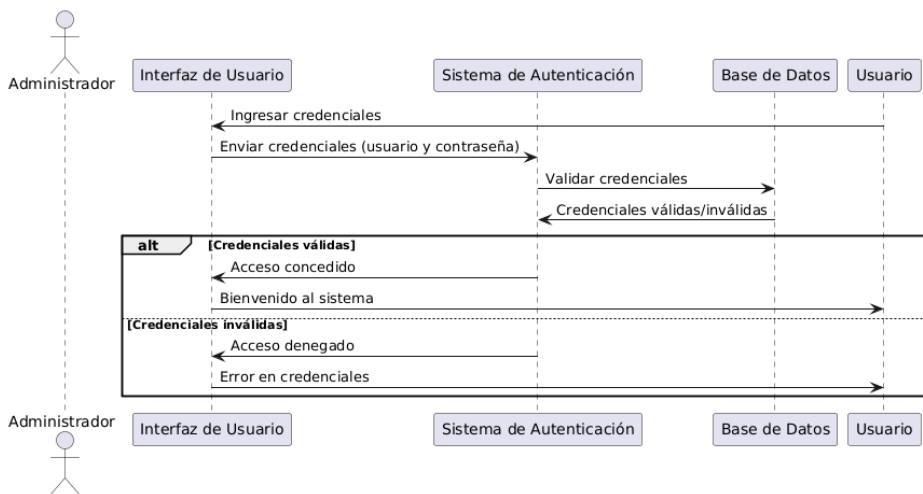
4.5.2 Pruebas de la Caja Negra

Para las pruebas de caja negra de este proyecto, se deberá evaluar la funcionalidad del sistema sin necesidad de conocer su estructura interna. Aquí hay algunas pruebas que se realizaron:

Prueba 1. Prueba de Inicio de Sesión: Verificar que los usuarios puedan iniciar sesión correctamente en el sistema utilizando credenciales válidas.

Figura 18

Prueba de Inicio al Sistema

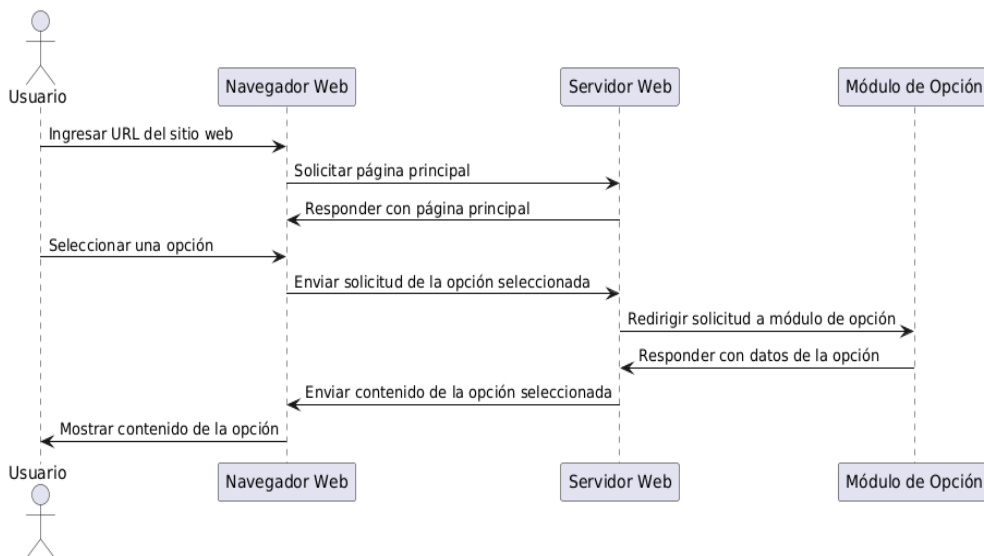


Nota: Elaboración propia según el desarrollo del software

Prueba 2. Verificar que los usuarios pueden acceder a diferentes opciones del menú de navegación que presenta la página principal del sistema web.

Figura 19

Prueba de Acceso a la Plataforma

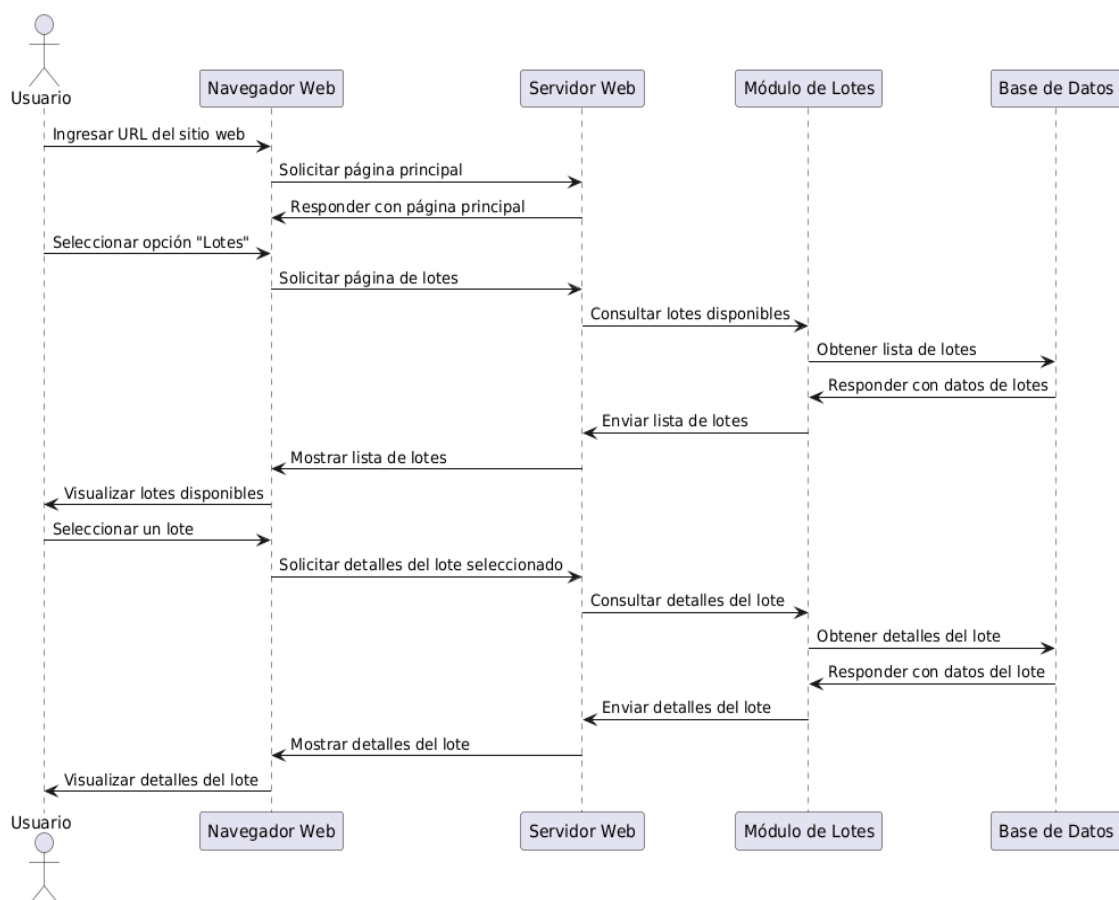


Nota: Elaboración propia según el desarrollo del software

Prueba 3. En la opción de ver categorías de productos, se pueden consultar los productos y sus precios para obtener más información sobre ellos.

Figura 20

Prueba de Visualización

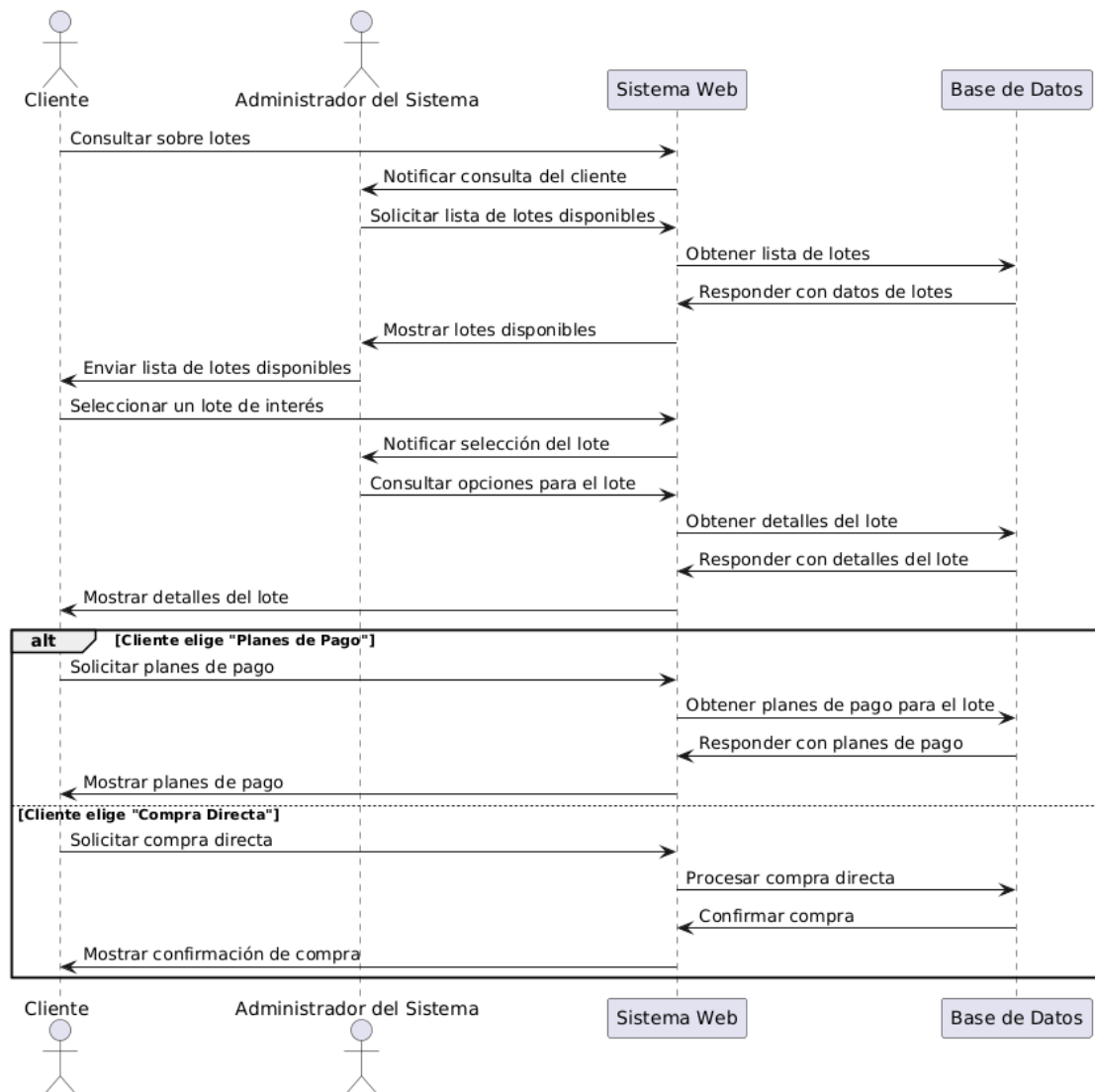


Nota: Elaboración propia según el desarrollo del software

Prueba 4. Probar la capacidad del sistema para permitir a los usuarios agregar productos al carrito, gestionar cantidades y finalizar la compra de manera exitosa.

Figura 21

Prueba de Funcionalidad



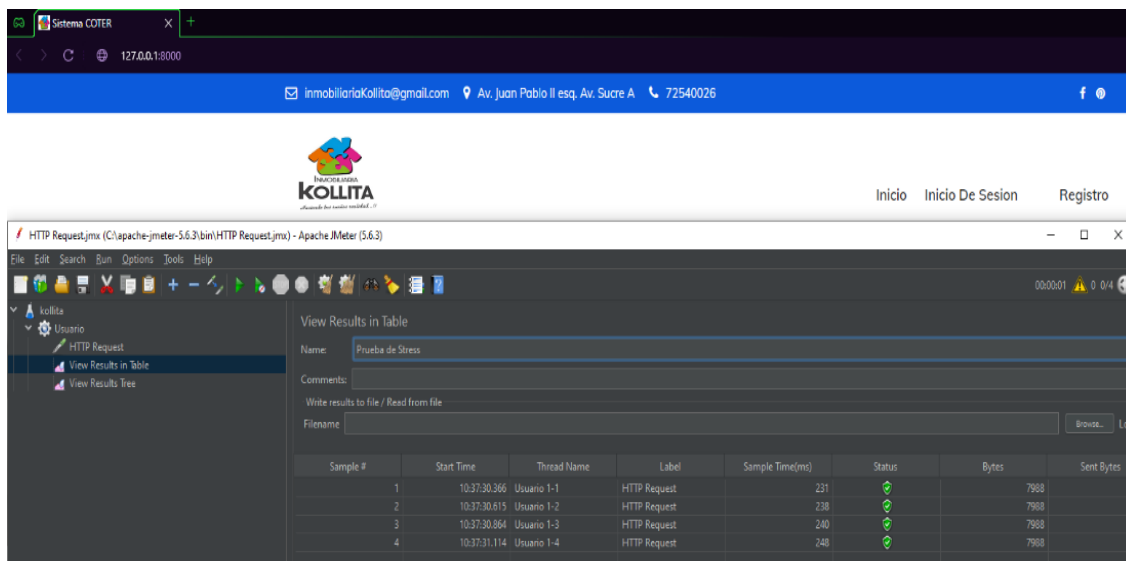
Nota: Elaboración propia según el desarrollo del software

4.5.3 Pruebas de Estrés

El propósito de las pruebas de estrés es someter al sistema web a una carga extrema hasta alcanzar su límite de capacidad. Para llevar a cabo estas pruebas, se empleó la herramienta Apache JMeter. Los resultados obtenidos se presentan a continuación. Prueba de estrés de la página de inicio de los usuarios lectores.

Figura 22

Prueba de con JMeter



4.5.4 Compatibilidad

El proceso de la compatibilidad, se realiza en diferentes entornos, dispositivos y navegadores, para garantizar el funcionamiento correcto y adecuado del sistema.

Figura 23

Compatibilidad con Google Chrome

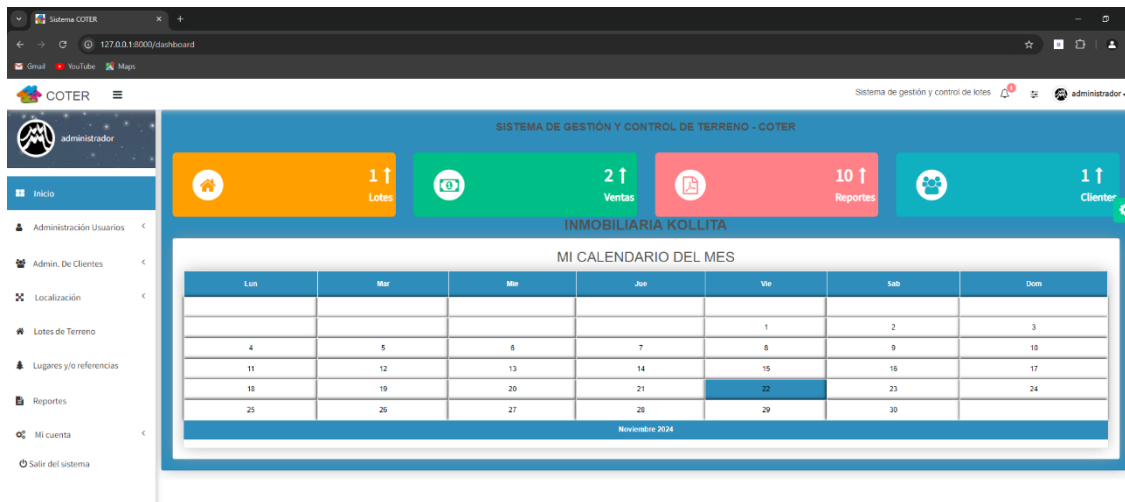


Figura 24

Compatibilidad con Opera GX

Lun	Mar	Mie	Jue	Vie	Sab	Dom
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

Figura 25

Compatibilidad con Microsoft Edge

Lun	Mar	Mie	Jue	Vie	Sab	Dom
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

CAPITULO V

CONCLUSIONES

Y

RECOMENDACIONES

**INGENIERÍA
DE SISTEMAS**
UNIVERSIDAD PÚBLICA DE EL ALTO



CAPITULO V

5 CONCLUSIONES Y RECOMENDACIONES

Se ha llegado a cumplir de manera exitosa con los objetivos planteados en el presente proyecto.

5.1 CONCLUSIONES

El proyecto desarrollado cumplió con los objetivos propuestos al abordar de manera integral la gestión de información sobre terrenos.

En primer lugar, se logró analizar y clasificar de manera efectiva los requerimientos necesarios para el desarrollo y funcionamiento del sistema, asegurando que este cubriera las necesidades operativas y técnicas establecidas desde el inicio.

Asimismo, se implementó una plataforma que permite el acceso rápido y actualizado a datos relacionados con terrenos disponibles, sus características, ubicaciones y opciones de planes de pago, facilitando a los usuarios la obtención de información de manera ágil y confiable.

La centralización de la información en una base de datos robusta y accesible en línea se consolidó como uno de los logros más significativos del proyecto, permitiendo consultas eficientes y garantizando la integridad de los datos.

Finalmente, se generó información confiable y estructurada sobre la rentabilidad de los terrenos, proporcionando a los usuarios herramientas clave para una toma de decisiones informada y estratégica.

En conjunto, el sistema desarrollado no solo optimiza los procesos de gestión y consulta de información, sino que también contribuye a una mejor planificación y análisis de inversiones en el sector inmobiliario.

Tabla 22

Tabla de Conclusiones

Objetivo / Problema	Descripción del Problema	Solución Propuesta en el Sistema Web	Cumplimiento del Objetivo
Atención al cliente manual	La atención al cliente en KOLLITA es manual, lo que retrasa el tiempo de respuesta y genera insatisfacción debido a la búsqueda de información en múltiples documentos.	El sistema web automatiza la atención al cliente, ofreciendo acceso rápido a información actualizada sobre terrenos, disponibilidad, precios y planes de pago a través de una plataforma en línea.	Cumplido: El sistema mejora significativamente los tiempos de respuesta, eliminando la necesidad de buscar información en documentos físicos. Los clientes pueden obtener respuestas rápidas y detalladas.
Control y administración de terrenos y planes de pago ineficientes	El control y administración de los terrenos y planes de pago no son eficientes, lo que conlleva a errores al momento de ofrecer terrenos al cliente.	El sistema web centraliza la información sobre los terrenos, su disponibilidad, precios y planes de pago en una base de datos unificada, accesible desde cualquier dispositivo.	Cumplido: La centralización de datos reduce los errores, mejora la precisión de la información disponible y permite ofrecer terrenos sin riesgo de confusión o falta de disponibilidad.
Riesgo de pérdida de información	Riesgo de pérdida de información de ventas debido a errores del personal, extravío de documentos y desorganización en la gestión de la información.	El sistema web almacena todos los datos de ventas y transacciones electrónicamente en bases de datos seguras y realiza copias de seguridad periódicas para evitar la pérdida de información.	Cumplido: Al eliminar el uso de documentos físicos y almacenar la información en un sistema centralizado y protegido, el riesgo de pérdida de datos se reduce considerablemente.
Dificultad en la obtención de información	La obtención de información por gestión es difícil ya que se encuentra dispersa, lo que ocasiona la toma de decisiones no acertadas.	El sistema web permite la consulta rápida de información relevante en tiempo real, facilitando la toma de decisiones basada en datos precisos y actualizados.	Cumplido: La centralización y accesibilidad de la información en el sistema permite una toma de decisiones más ágil y precisa, evitando la dispersión y desorganización previa.

6 RECOMENDACIONES

6.1 Revisión y actualización constante de los requerimientos

Se sugiere realizar revisiones periódicas de los requerimientos analizados y clasificados para garantizar que el sistema continúe cumpliendo con las necesidades operativas y técnicas, considerando posibles cambios en el mercado inmobiliario o en las normativas aplicables.

6.2 Implementación de mecanismos de sincronización de datos

Para mantener el acceso rápido y actualizado a la información de terrenos, se recomienda implementar procesos automáticos de sincronización de datos, integrando fuentes externas, como registros catastrales o bases de datos públicas, cuando sea posible.

6.3 Fortalecimiento de la seguridad de la base de datos

Dado que la información centralizada es crítica, se debe priorizar la implementación de medidas de seguridad avanzadas para proteger los datos de accesos no autorizados, pérdida o manipulación indebida. Esto incluye el uso de cifrado, autenticación de múltiples factores y políticas de respaldo periódico.

6.4 Ampliación de funcionalidades del sistema

Se recomienda incluir herramientas analíticas avanzadas que permitan realizar proyecciones y simulaciones sobre la rentabilidad de los terrenos. Estas funcionalidades podrían incluir gráficos interactivos, reportes automatizados y análisis de tendencias.

6.5 Capacitación a usuarios finales

Para maximizar el uso del sistema, se sugiere desarrollar programas de capacitación para los usuarios finales, enfocados en el manejo de la plataforma y en la interpretación de la información generada para la toma de decisiones.

6.6 Pruebas de rendimiento y escalabilidad

Se aconseja realizar pruebas regulares para garantizar que el sistema pueda manejar un volumen creciente de datos y usuarios sin perder rendimiento, asegurando su escalabilidad para futuras expansiones.

6.7 Estudio de experiencias del usuario

Finalmente, se recomienda llevar a cabo encuestas y estudios de usabilidad para identificar oportunidades de mejora en la interfaz del sistema y optimizar la experiencia del usuario final.

BIBLIOGRAFIA

Ramos Huarachi, Liz Andrea (2013) Sistema Web de administración de ventas basado en la gestión de procesos de negocios CASO: Empresa de Muebles Artemobile, Departamento de Ventas

<https://repositorio.umsa.bo/handle/123456789/7806>

Luis Acevedo, & Andres felipe. (2017, November 16). Metodología UWE. Universidad Centro Superior.

<https://view.genial.ly/5a0bbf8021cf372478d5b94e/interactive-content-metodologia-uwe>

Marianela393. (2016, April 29). Cocomo

<https://es.slideshare.net/marianela0393/cocomo-ii-61519624>

©OpenWebinars S.L. (17 de abril de 2024). OpenWebinars.

<https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>

©Rock Content. (15 de Abril de 2024). rockcontent bloq. rockcontent bloq:

<https://rockcontent.com/es/blog/bootstrap/>

Alberto, L. G. (2018). rep-umsa. rep-umsa:

<https://repositorio.umsa.bo/handle/123456789/1804>

BLOG. (28 de Febrero de 2024). Guía de las Normas ISO en español.

<https://normasiso.org/norma-iso-9126/>

Chungas, E. (2012). SISTEMA DE GESTIÓN DE VENTAS PARA REDUCIR EL TIEMPO DE ATENCIÓN AL CLIENTE Y AUMENTAR EL MARGEN DE UTILIDAD EN LA EMPRESA.

<https://repositorio.upn.edu.pe/bitstream/handle/11537/3164/Chugnas%20Morales%20C%20Jaynor%20Eduardo.pdf?sequence=1&isAllowed=y>

Concepto. (14 de Febrero de 2024). © 2013-2024 Enciclopedia Concepto.

<https://concepto.de/sistema/>

Condori, R. (2015). SISTEMA WEB DE CONTROL DE COMPRA VENTA E INVENTARIOS.

<https://doi.org/https://repositorio.umsa.bo/bitstream/handle/123456789/8184/T.3033.pdf?sequence=1&isAllowed=y>

Copyright © Arsys. (8 de Abril de 2024). arsys Blog. <https://www.arsys.es/blog/mysql>

Crea System. (15 de Mayo de 2024). © 2024 Crea System.

<https://www.creasystem.net/posts/que-es-un-sistema-web>

Blogger. (14 de Febrero de 2024). Con la tecnología de Blogger.

<http://datosinformaticadiadia.blogspot.com/2017/08/metodo-openup.html>
digital/#:~:text=El%20Marketing%20Digital%20o%20Marketing%20Online%20es%20el%20c
onjunto%20de,y%20en%20el%20momento%20adecuado.

© 2023 Urudata Softwar. (17 de Enero de 2024). QFLOW.

[https://qflowbpm.com/es/compras/beneficios-sistema-
compras/#:~:text=Implementar%20un%20sistema%20de%20gesti3n%20de%20comp
ras%20puede%20conducir%20a,salud%20financiera%20general%20del%20negocio](https://qflowbpm.com/es/compras/beneficios-sistema-compras/#:~:text=Implementar%20un%20sistema%20de%20gesti3n%20de%20comp
ras%20puede%20conducir%20a,salud%20financiera%20general%20del%20negocio).

© 2023 Urudata Software. (17 de Enero de 2024). QFLOW.

[https://qflowbpm.com/es/compras/beneficios-sistema-
compras/#:~:text=Implementar%20un%20sistema%20de%20gesti3n%20de%20comp
ras%20puede%20conducir%20a,salud%20financiera%20general%20del%20negocio](https://qflowbpm.com/es/compras/beneficios-sistema-compras/#:~:text=Implementar%20un%20sistema%20de%20gesti3n%20de%20comp
ras%20puede%20conducir%20a,salud%20financiera%20general%20del%20negocio).

© Copyright 1996 - 2024. (17 de ENERO de 2024). VISA. [https://www.visa.com.bo/dirija-su-
negocio/pequenas-medianas-empresas/notas-y-recursos/tecnologia/que-es-
ecommerce-o-comercio-electronico.html](https://www.visa.com.bo/dirija-su-negocio/pequenas-medianas-empresas/notas-y-recursos/tecnologia/que-es-ecommerce-o-comercio-electronico.html)

©Amazon Web Services. (08 de abril de 2024). aws. [https://aws.amazon.com/es/what-
is/javascript/](https://aws.amazon.com/es/what-is/javascript/)

©Banco Santander. (28 de Febrero de 2024). Santander.

[https://www.santanderopenacademy.com/es/blog/metodologias-desarrollo-
software.html](https://www.santanderopenacademy.com/es/blog/metodologias-desarrollo-software.html)

Garrido, S. (9 de 10 de 2021). IEBS. IEBS: [https://www.iebschool.com/blog/que-son-
metodologias-agiles-agile-scrum/](https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/)

González, L. E. (23 de Febrero de 2014). METODOLOGÍA OPEN UP.

<http://openup3.blogspot.com/2014/02/metodologia-open-up.html>

HubSpot, Inc. (09 de abril de 2024). HubSpot. <https://blog.hubspot.es/website/que-es-jquery>

IBM Documentation Help. (09 de ABRIL de 2024). IBM.

[https://www.ibm.com/docs/es/rational-soft-arch/9.6.1?topic=page-asynchronous-
javascript-xml-ajax-overview](https://www.ibm.com/docs/es/rational-soft-arch/9.6.1?topic=page-asynchronous-javascript-xml-ajax-overview)

Jacqueline, C. C. (2018). rip-umsa. rip-umsa: <https://repositorio.umsa.bo/xmlui/ldap-login>

LIFE.ART.TECH. (29 de Febrero de 2024). START A BLOG .

<https://lifeartech.wordpress.com/tag/metodo-open-up/>

NIBNORCA. (14 de 05 de 2024). NIBNORCA. NIBNORCA

NORMAS ISO.ORG. (14 de Mayo de 2024). NORMAS ISO 25000. NORMAS ISO 25000:

<https://normasiso.org/norma-iso-2500/>

Osmosis Latina. (15 de Abril de 2024). *Osmosis Latina*. *Osmosis Latina*:

<https://www.osmosislatina.com/jmeter/basico.htm>

Pressman, R. (2010). *Ingeniería del software(enfoque practico)*. McGraw Hill.

<https://doi.org/http://cotana.informatica.edu.bo/downloads/Id->

[Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF](https://doi.org/http://cotana.informatica.edu.bo/downloads/Id-Ingenieria.de.software.enfoque.practico.7ed.Pressman.PDF)

requerimientos, I. d. (s.f). *Ingeniería de requerimientos*.

[https://doi.org/http://sedici.unlp.edu.ar/bitstream/handle/10915/4057/2_-_](https://doi.org/http://sedici.unlp.edu.ar/bitstream/handle/10915/4057/2_-_Ingenier%C3%ADa_de_requerimientos.pdf?sequence=4)

[_Ingenier%C3%ADa_de_requerimientos.pdf?sequence=4](https://doi.org/http://sedici.unlp.edu.ar/bitstream/handle/10915/4057/2_-_Ingenier%C3%ADa_de_requerimientos.pdf?sequence=4)

Sommerville, I. (2005). *Ingeniería del Software*. Pearson education S.A.

https://doi.org/http://zeus.inf.ucv.cl/~bcrawford/AULA_ICI_3242/Ingenieria%20del%20

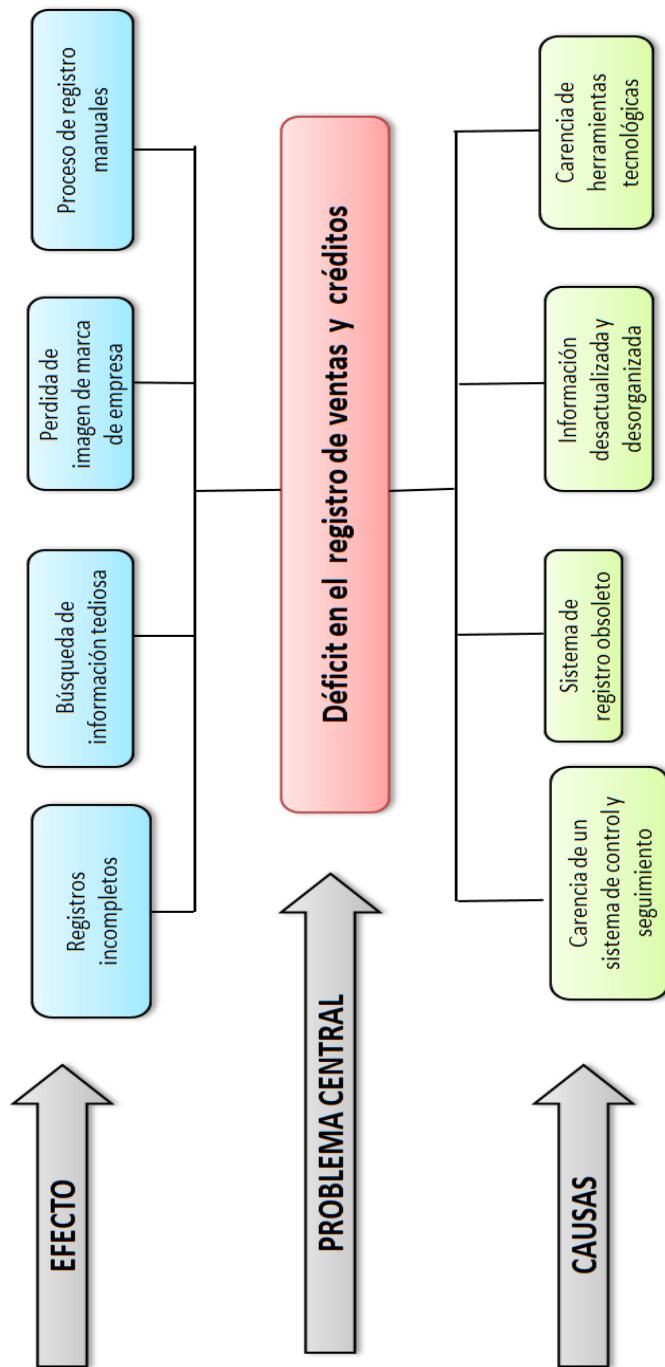
[Software%207ma.%20Ed.%20-%20lan%20Sommerville.pdf](https://doi.org/http://zeus.inf.ucv.cl/~bcrawford/AULA_ICI_3242/Ingenieria%20del%20Software%207ma.%20Ed.%20-%20lan%20Sommerville.pdf)

ZAPTEST. (07 de 03 de 2024). Copyright 2024. [https://www.zaptest.com/es/pruebas-de-](https://www.zaptest.com/es/pruebas-de-caja-negra-que-son-tipos-procesos-enfoques-herramientas-y-mucho-mas)

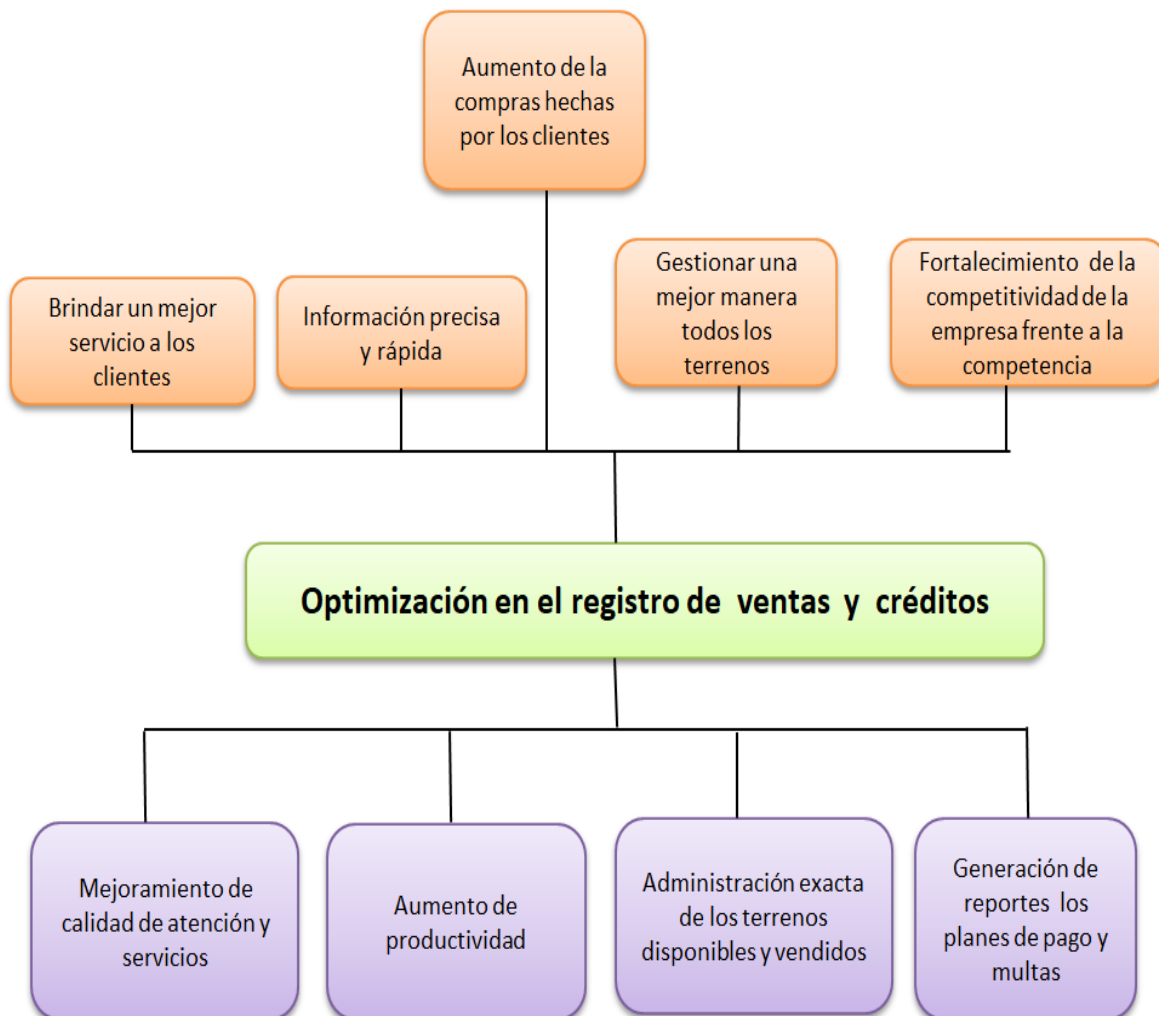
[caja-negra-que-son-tipos-procesos-enfoques-herramientas-y-mucho-mas](https://www.zaptest.com/es/pruebas-de-caja-negra-que-son-tipos-procesos-enfoques-herramientas-y-mucho-mas)

ANEXOS

ARBOL DE PROBLEMAS



ARBOL DE SOLUCIONES





MANUAL DE USUARIO

SISTEMA WEB DE GESTIÓN
Y VENTA DE TERRENOS



Manual de Usuario

SISTEMA WEB DE GESTIÓN Y VENTA DE TERRENOS

CASO: INMOBILIARIA “KOLLITA”

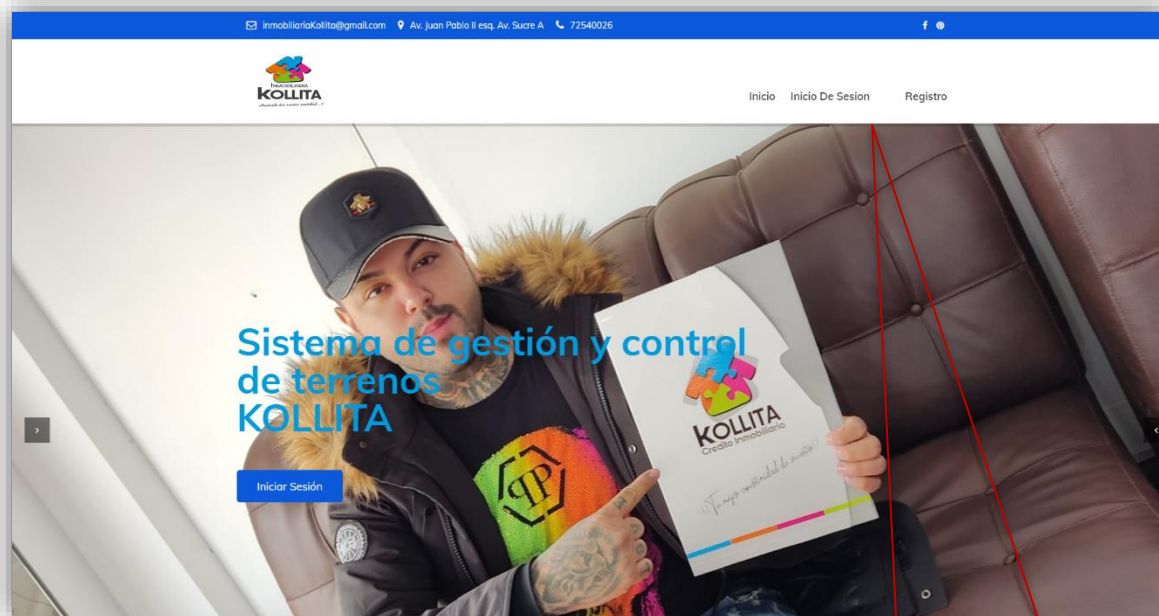
Contenido

1. Introducción
2. Registro de Usuario
3. Registro de nuevos clientes
4. Registro de nuevas Urbanizaciones
5. Registro de nuevo Manzano de Urbanización
6. Registro de nuevo Lote de Terreno
7. Características de Lote De Terreno
8. Vista del plan de pago a crédito (Proforma
9. Generación de proforma
10. Vista de detalles de lote de terreno – Ubicación
11. Vista de detalles de lote de terreno – Precio
12. Generación de reportes de venta – Listado de lotes
13. Generación de reportes de venta – Listado de lotes por Ubicacion
14. Generación de reportes de venta – Listado de lotes por rango de fecha

1. Introducción

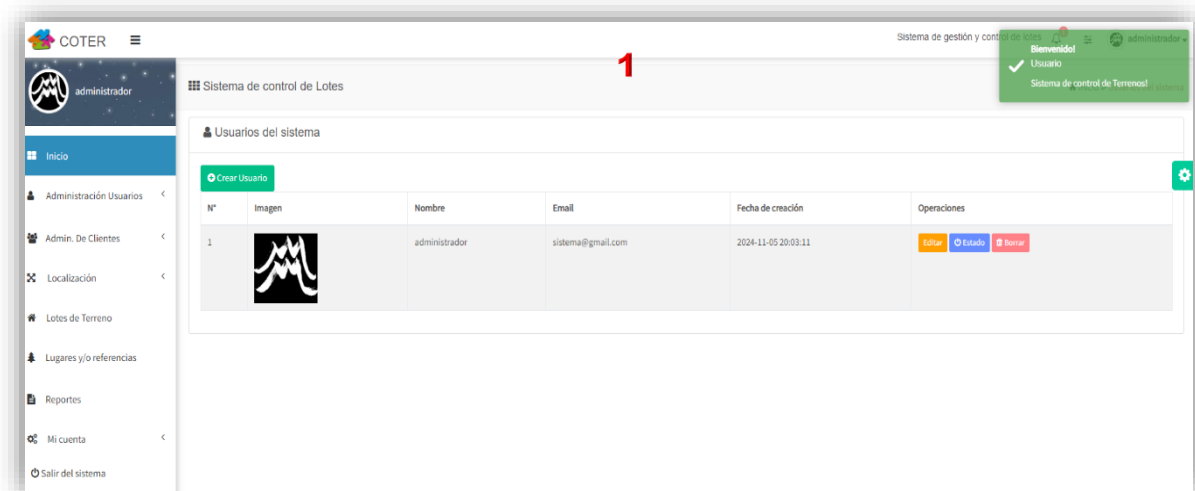
En el presente manual se explicará lo necesario para poder utilizar el sistema de control de entradas y salidas y la generación de planillas mediante tarjetas RFID

Ingreso al sistema

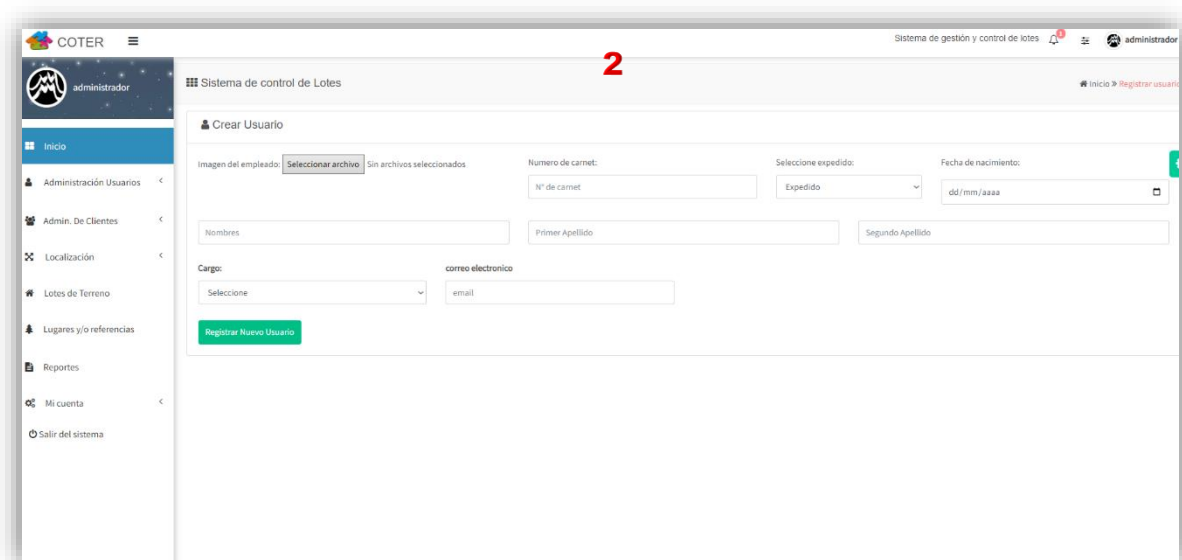


- 1) Ingresar usuario
- 2) Ingresar contraseña
- 3) Ingresar el texto CAPCHA
- 4) Ingreso al sistema

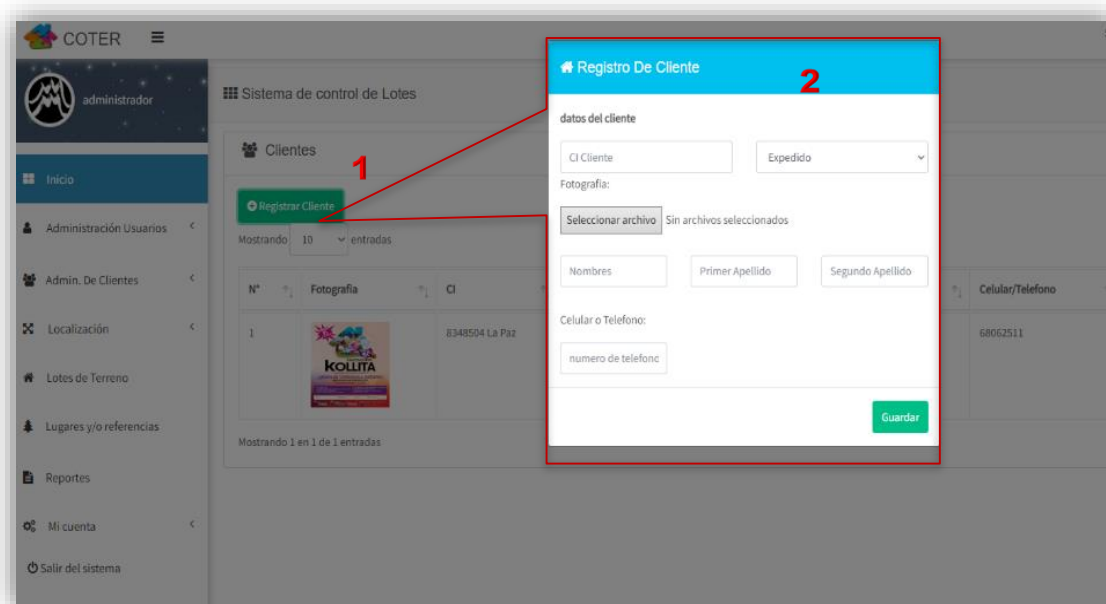
2. Registro de Usuario



- 1) Botón de registro de nuevo usuario
- 2) Registro de usuario según su cargo

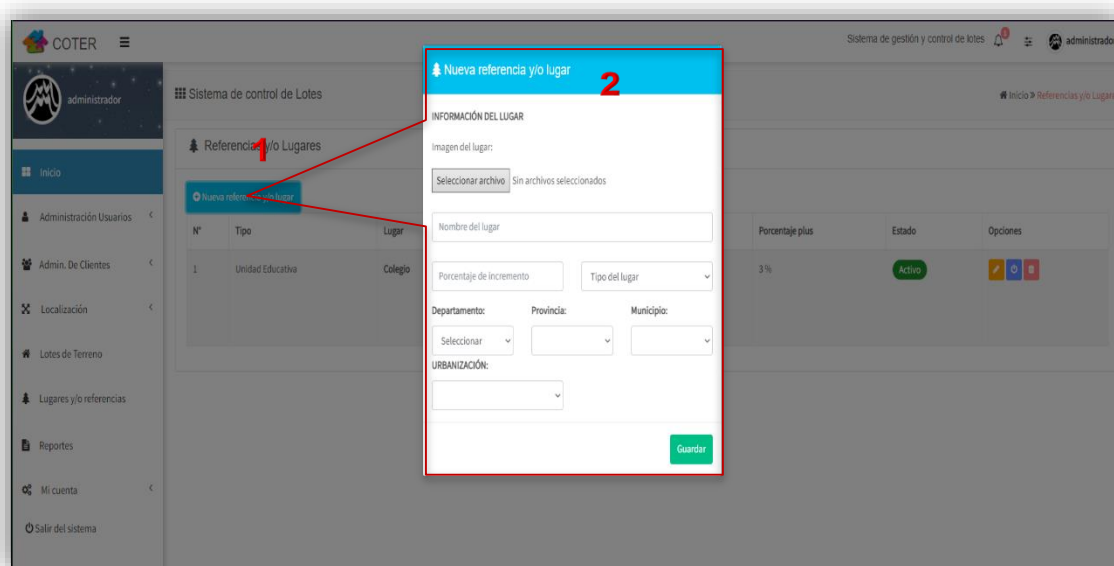


3. Registro de nuevos clientes

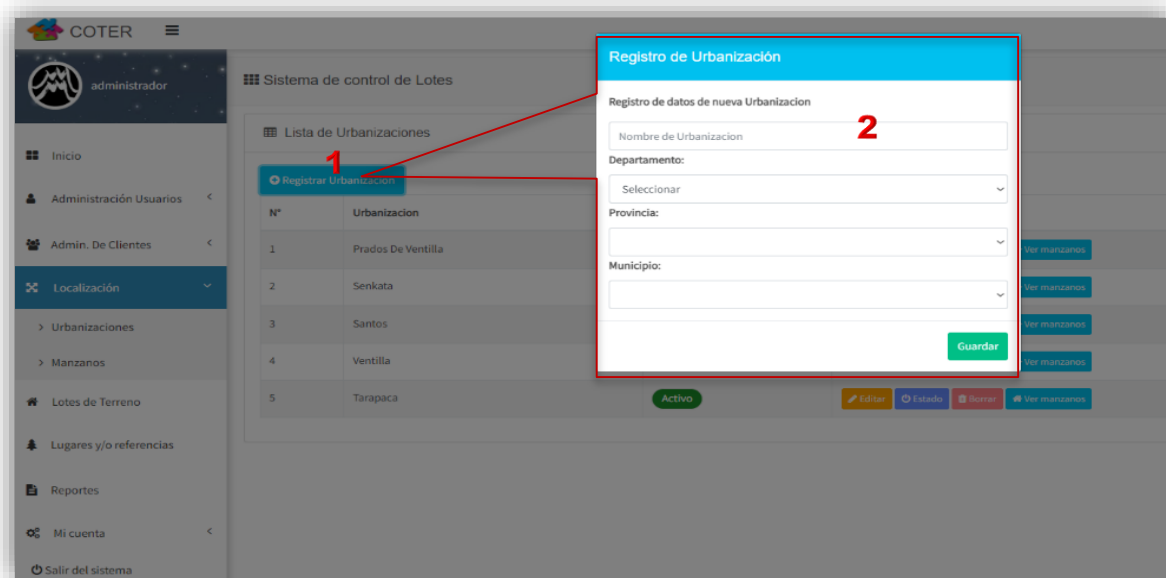


- 1) Botón de registro de nuevo
- 2) Ventana emergente de registro

Registro de Ubicaciones referentes al terreno

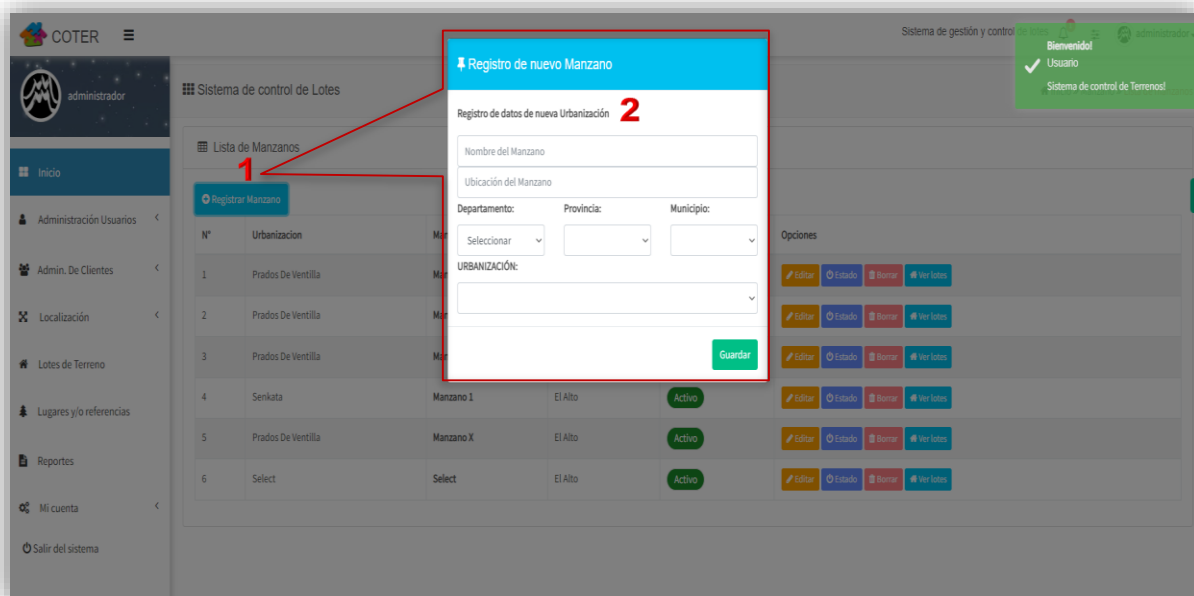


4. Registro de nuevas Urbanizaciones

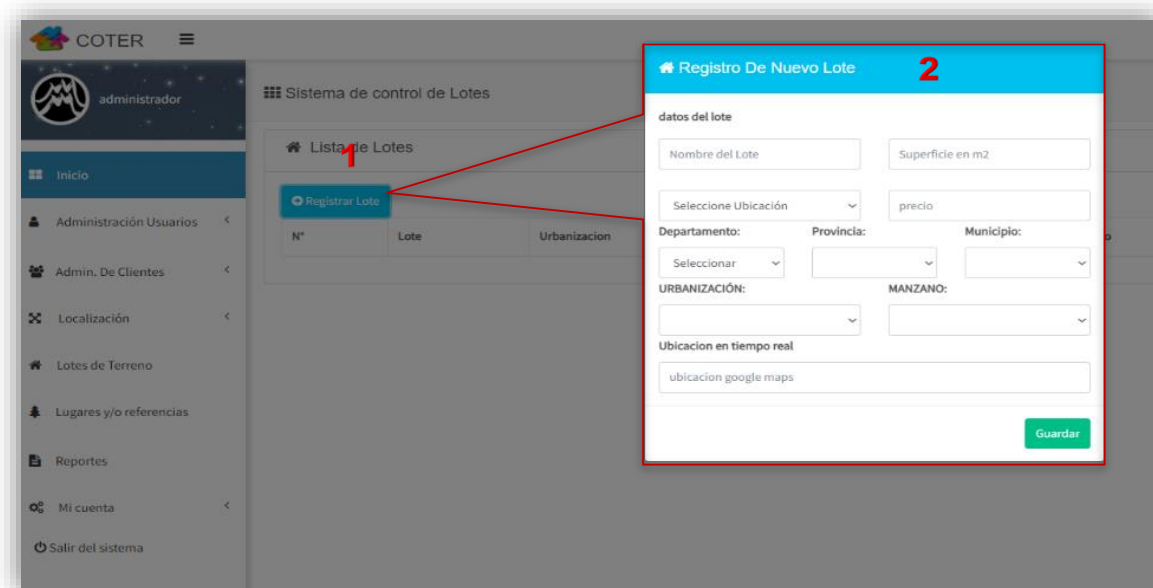


- 1) Botón de registro de nuevo
- 2) Ventana emergente de registro

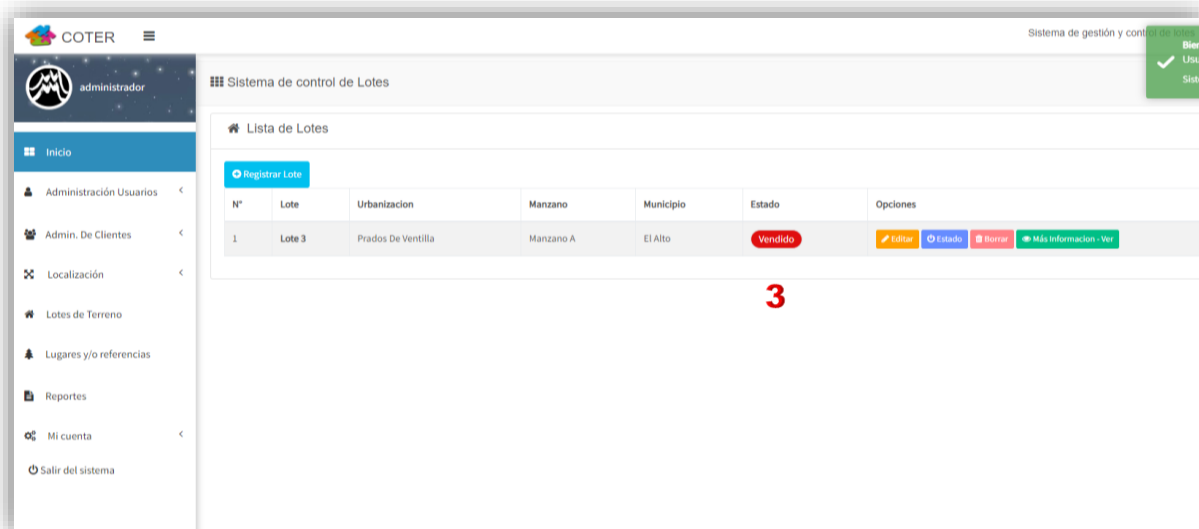
5. Registro de nuevo Manzano de Urbanización



6. Registro de nuevo Lote de Terreno



- 1) Botón de vista de lote de terreno
- 2) Ventana emergente de los datos
- 3) Estado de Terreno (Disponible-Vendida)



7. Características de Lote De Terreno

The image displays two screenshots of the COTER system interface. The top screenshot shows the 'Información Del Lote' page for 'Lote 3'. The page includes a sidebar with navigation options and a main content area with a table of lot details and a table for 'Lugares y Referencias'. A red '1' is placed over the 'Añadir Características Especiales' button. The bottom screenshot shows a modal window titled 'Añadir un lugar de referencia al lote' with a red '2' and a dropdown menu for 'Referencia'.

1

Lote	Lote 3	Departamento	LA PAZ
Superficie	200 m2	Provincia	MURILLO
Ubicación	Avenida	Municipio	El Alto
Precio Base	4000	Urbanización	Prados De Ventilla
Estado	disponible	Manzano	Manzano A

Precio al contado Bs. 27880 Bs.
Precio al contado \$ 4000 \$

2

Añadir un lugar de referencia al lote

Seleccionar el Lugar de referencia

Referencia:

Seleccionar

Guardar

- 1) Botón de despliegue para añadir características
- 2) Ventana emergente de los datos

8. Vista del plan de pago a crédito (Proforma)

colización en 5 años

Plazo	Costo en \$	Costo en Bs.	Cuota por mes en \$	Cuota por mes en Bs.	Opciones
1 año	4500	31365	3755	2613.7	Generar proforma
2 años	5000	34850	416.75	2904.2	Generar proforma
3 años	5500	38335	458.35	3194.6	Generar proforma
4 años	6000	41820	5005	3485	Generar proforma
5 años	6500	45305	541.75	3775.4	Generar proforma

Definir Lapsos de tiempo

Cantidad de años

- 5 años
- 4 años
- 3 años
- 2 años
- 1 año

1) Botón de generación de proforma

2) Archivo PDF de proforma

9. Generación de proforma

Cotización en 4 años Cambiar periodo de tiempo

Plazos de Pago

Plazo	Costo en \$	Costo en Bs.	Cuota por mes en \$	Cuota por mes en Bs.	operacion
1 año	4500	31365	375 \$	2613.7 Bs	Generar proforma
2 años	5000	34850	416.7 \$	2904.2 Bs	Generar proforma
3 años	5500	38335	458.3 \$	3194.6 Bs	Generar proforma
4 años	6000	41820	500 \$	3485 Bs	Generar proforma

PROFORMA

INMOBILIARIA "KOLLITA"


URBANIZACIÓN	Prados De Ventilla	SUPERFICIE	UBICACIÓN
MANZANO	Manzano A	200m2	Avenida
LOTE	Lote 3		

Precio al contado Su\$

Precio al contado Bs.


4000 Dolares


27840 Bs.



KOLLITA
INMOBILIARIA
Asesando las mejores realidades...

Plazo	Costo en \$	Costo en Bs.	Cuota por mes en \$	Cuota por mes en Bs.
1 año	4500 \$	31365 Bs.	375 \$	2613.7 Bs.
2 años	5000 \$	34850 Bs.	416.7 \$	2904.2 Bs.
3 años	5500 \$	38335 Bs.	458.3 \$	3194.6 Bs.
4 años	6000 \$	41820 Bs.	500 \$	3485 Bs.

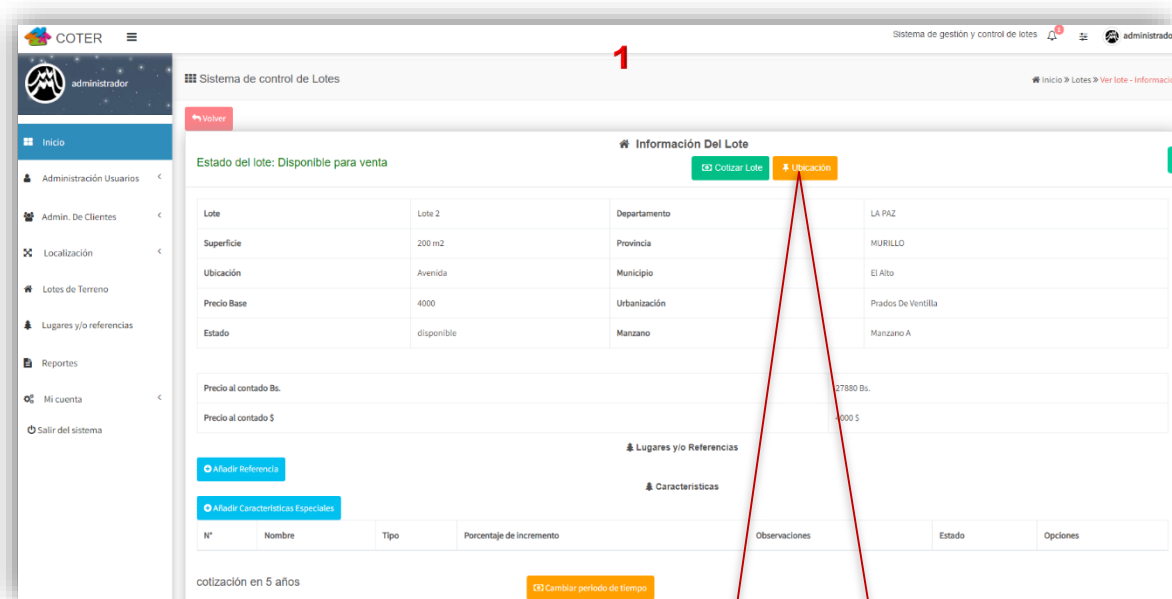




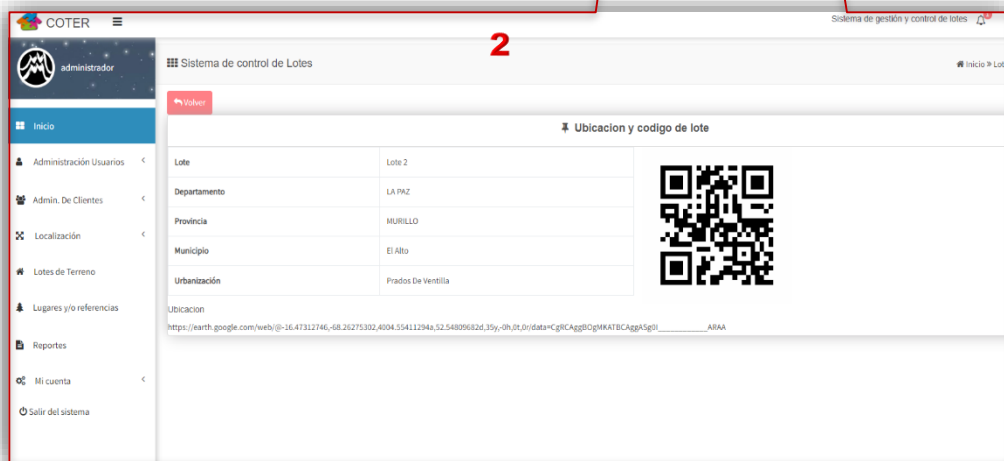
7757148 - 75289449 - 65161285

Edif. Bv. Nueva Ventilla # 2445 - Ciudad Comercial La Reina - Caracas - Venezuela
Finca. G.T. - San. Pádo. C.O. - A. Estrada de acceso del la LUCCA.

10. Vista de detalles de lote de terreno – Ubicación



- 1) Botón de despliegue para añadir características
- 2) Ventana emergente de los datos



11. Vista de detalles de lote de terreno – Precio

Sistema de gestión y control de lotes

administrador

Sistema de control de Lotes

Volver

Estado del lote: Disponible para venta

Información Del Lote

Cotizar Lote Ubicación

Lote	Lote 2	Departamento	LA PAZ
Superficie	200 m2	Provincia	MURILLO
Ubicación	Avenida	Municipio	El Alto
Precio Base	4000	Urbanización	Prados De Ventilla
Estado	disponible	Manzano	Manzano A

Precio al contado Bs. 2780 Bs.

Precio al contado S 4000 S

Añadir Referencia

Añadir Características Especiales

Lugares y/o Referencias

Características

N°	Nombre	Tipo	Porcentaje de Incremento	Observaciones	Estado	Opciones
cotización en 5 años						

Cambiar periodo de tiempo

- 1) Botón de despliegue para añadir características
- 2) Ventana emergente de los datos

Sistema de gestión y control de lotes

administrador

Sistema de control de Lotes

Cotización Del Lote

Lote	Lote 2	Departamento	LA PAZ
Superficie	200 m2	Provincia	MURILLO
Ubicación	Avenida	Municipio	El Alto
Precio Base	4000	Urbanización	Prados De Ventilla
Estado	disponible	Manzano	Manzano A

Incremento de Lugares y/o Referencias

N°	Nombre	Porcentaje	Total Incremento
Total			0

Incremento de Características

N°	Nombre	Porcentaje	Total Incremento
Total			0

\$ Total Costo:4000

Realizar Venta

12. Generación de reportes de venta – Listado de lotes

The screenshot displays the COTER system interface. The top navigation bar includes the COTER logo, a user profile for 'administrador', and a welcome message: 'Bienvenido Usuario Sistema de control de Terrenos'. The main content area is titled 'Sistema de control de Lotes' and features a section for 'Reportes disponibles'. Three report options are visible:

- Listado de lotes** (Red card): 'Lista de todos los lotes registrados en el sistema' with a 'Crear PDF' button.
- Listado de lotes por urbanización** (Blue card): 'Lista de todos los lotes, según una urbanización' with a 'Crear PDF' button.
- Listado de ventas de lotes** (Green card): 'Lista de todas las ventas con los clientes' with a 'Crear PDF' button.

A sidebar on the left contains navigation links: Inicio, Administración Usuarios, Admin. De Clientes, Localización, Lotes de Terreno, Lugares y/o referencias, Reportes, Mi cuenta, and Salir del sistema.

13. Generación de reportes de venta – Listado de lotes por Ubicacion

This screenshot shows the same dashboard as above, but with a modal window open over the 'Listado de lotes por urbanización' report card. The modal is titled 'Datos para generar el reporte' and contains the following fields:

- Selección de urbanización:**
 - Departamento: LA PAZ (dropdown)
 - Provincia: MURILLO (dropdown)
 - Municipio: El Alto (dropdown)
 - URBANIZACIÓN: Prados De Ventilla (dropdown)
- Buscar** (button)

A red arrow points from the 'Crear PDF' button on the report card to the modal window.

14. Generación de reportes de venta – Listado de lotes por rango de fecha

The screenshot displays the COTER system interface. The top navigation bar includes the COTER logo, the user role 'administrador', and a welcome message: 'Bienvenido Usuario Sistema de control de Terrenos'. The main content area is titled 'Sistema de control de Lotes' and shows 'Reportes disponibles' (Available Reports). Three report options are visible:

- Listado de lotes** (List of lots): 'Lista de todos los lotes registrados en el sistema' (List of all lots registered in the system). A red box highlights this option with a red '1'.
- Listado de lotes por urbanización** (List of lots by urbanization): 'Lista de todos los lotes, segun una urbanización' (List of all lots, according to an urbanization). A blue box highlights this option with a red '1'.
- Listado de ventas de lotes** (List of lot sales): 'Lista de todas las ventas con los clientes' (List of all sales with clients). A green box highlights this option with a red '1'.

A red arrow points from the 'Listado de ventas de lotes' option to a dialog box titled 'Datos para generar el reporte' (Data to generate the report). This dialog box contains the following fields and elements:

- Header: 'Datos para generar el reporte' (Data to generate the report).
- Section: 'Seleccione el rango de fecha' (Select the date range).
- Fields: 'Desde:' (From) and 'Hasta:' (To), both with date input fields showing 'dd/mm/aaaa' and a calendar icon.
- Button: 'Buscar' (Search).

A red box highlights the dialog box with a red '2' next to it.



MANUAL TECNICO

**SISTEMA WEB DE GESTIÓN Y
VENTA DE TERRENOS**

Caso; Inmobiliaria "KOLLITA"

GESTIÓN -2024

ÍNDICE DE CONTENIDO

1.	Introducción	1
2.	Requisitos Previos mínimos	1
2.1	Requisitos Generales.....	1
2.2	Requisitos De Hardware mínimos.....	1
3.	Instalación De WAMPSEVER.....	1
3.1	Descargar WAMPSEVER.....	1
3.2	Instalar WAMPSEVER	1
3.3	Configurar WAMPSEVER	1
4.	Instalación De Visual Studio Code	2
4.1	Descargar Visual Studio Code	2
4.2	Instalar Visual Studio Code.....	2
5.	Instalación de Composer	3
5.1	Descargar Composer.....	3
5.2	Instalar Composer.....	3
5.3	Verificar La Instalación De Composer	3
6.	Instalación De Laravel 11.....	4
6.1	Crear Un Proyecto Laravel.....	4
6.2	Instalar Livewire	4
6.3	Configurar El Servidor Local	4

1. Introducción

Este manual técnico proporciona instrucciones detalladas para la instalación y configuración de un entorno del desarrollo web completo utilizando WAMP SERVER, Visual Studio Code, Laravel, MySQL, y Composer. Este entorno es ideal para el desarrollo de aplicaciones web modernas con PHP y MySQL.

2. Requisitos Previos y Mínimos

Antes de todo, asegurarse de contar con:

1. Requisitos Generales

- Un sistema operativo: Windows 10 o superior.
- Conexión a Internet
- Permisos de administrador en tu computadora
- Disco Sólido
- Espacio en Disco: Al menos 5 GB de espacio libre.

2. Requisitos de Hardware Mínimos

- Procesador: Intel Core i3 o RYZEN 3
- Memoria RAM: 4 GB (8 GB recomendado).
- Espacio en Disco: 5 GB de espacio libre (10 GB recomendado).
- Instalación de WAMP SERVER

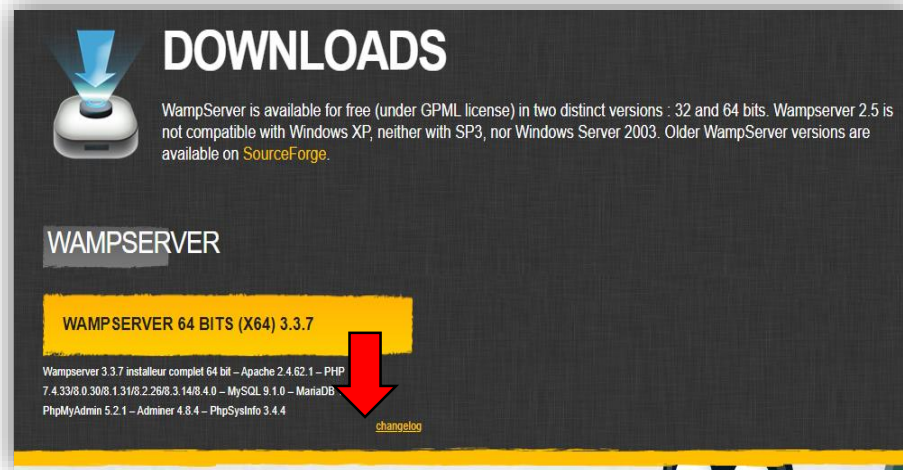
- WAMPSEVER es una distribución de Apache
que incluye PHP, MySQL .

Pasos de Instalación:

a. Descargar WAMPSEVER

- Visita el sitio web oficial de WAMPSEVER

<https://www.wampserver.com/en/download-wampserver-64bits/>



- Descarga el instalador para Windows.
- Instalar WAMPSEVER
- Ejecuta el instalador descargado.
- Sigue las instrucciones del asistente de instalación.
- Selecciona los componentes a instalar (por defecto están seleccionados Apache,MySQL, PHP).

- Elige la carpeta de destino para la instalación.
- Completa la instalación y abre el panel de control de WAMPSERVER.

b. Configurar WAMPSERVER

- Inicia los servicios de Apache y MySQL desde el panel de control de WAMPSERVER.
- Verifica la instalación abriendo un navegador web e ingresando <http://localhost>. Deberías ver la página de bienvenida de WAMPSERVE.



3. Instalación de Visual Studio Code

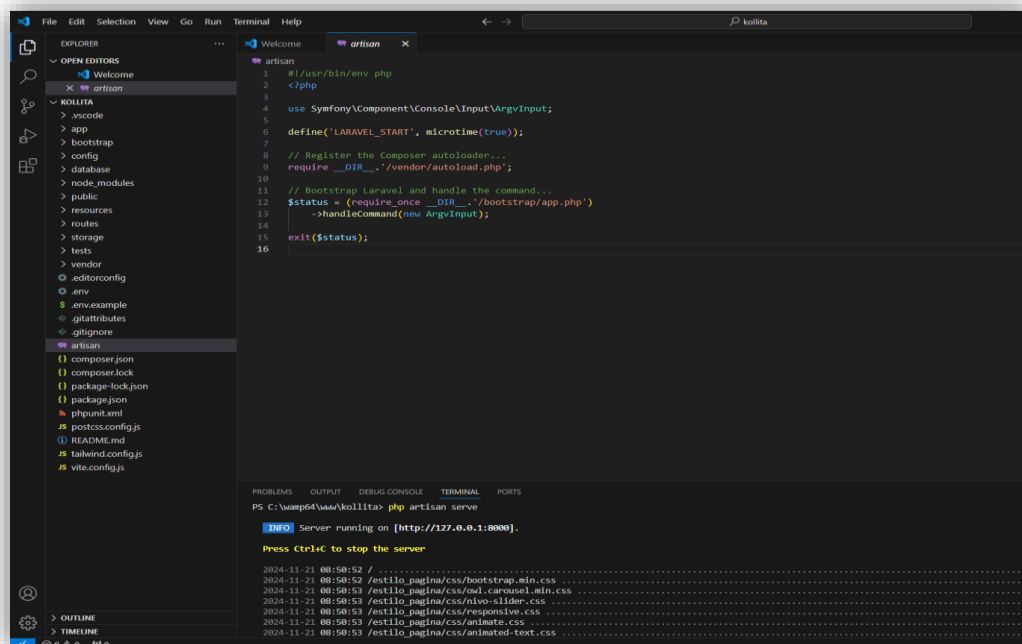
Visual Studio Code (VS Code) es un editor de código fuente desarrollado por Microsoft. Pasos para la Instalación:

a. Descargar Visual Studio Code

- Visita el sitio web oficial de Visual Studio Code: <https://code.visualstudio.com/>
- Descarga el instalador adecuado para tu sistema operativo.

b. Instalar Visual Studio Code

- Ejecuta el instalador descargado.
- Sigue las instrucciones del asistente de instalación.
- Selecciona las opciones adicionales que desees
- Completa la instalación y abre Visual Studio Code.



7 Instalación de Composer

Composer es un gestor de dependencias para PHP. Pasos de Instalación:

a. Descargar Composer

- Visita el sitio web oficial de Composer: <https://getcomposer.org/>
- Descarga el instalador adecuado para tu sistema operativo.

b. Instalar Composer

- Ejecuta el instalador descargado.
- Sigue las instrucciones del asistente de instalación.
- Durante la instalación, asegúrate de que el instalador detecte correctamente el ejecutable de PHP (usualmente en la carpeta de XAMPP).

c. Completa la instalación.

- Abre una terminal o línea de comandos.
- Ejecuta el comando `composer --version`.

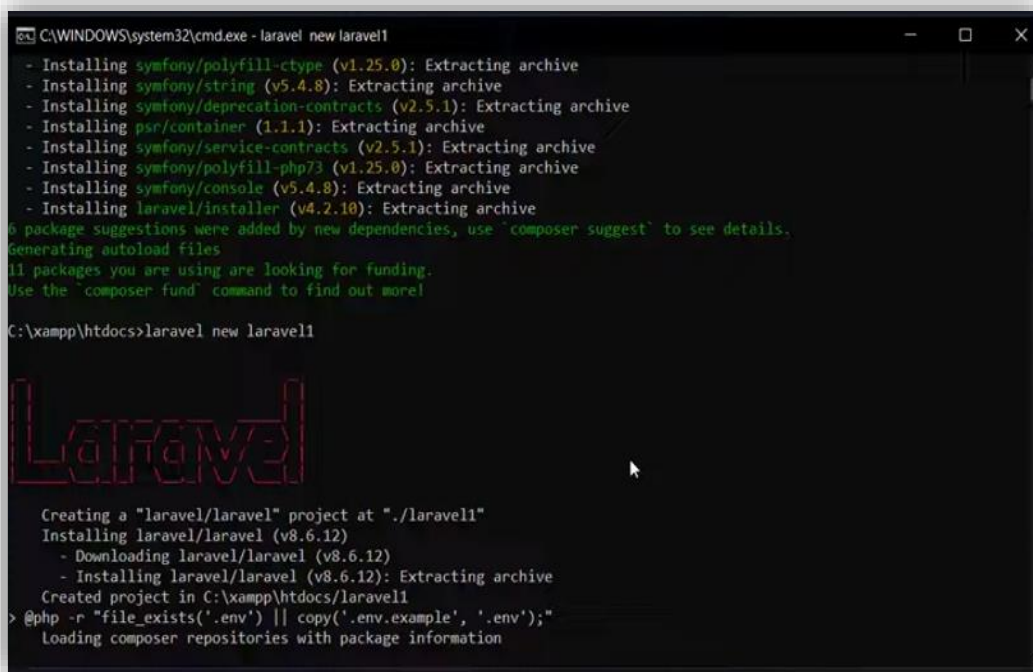
Deberías ver la versión de Composer instalada.



4. Instalación de Laravel

Laravel es un framework PHP para el desarrollo de aplicaciones

web. Pasos de Instalación:



```
C:\WINDOWS\system32\cmd.exe - laravel new laravel1
- Installing symfony/polyfill-ctype (v1.25.0): Extracting archive
- Installing symfony/string (v5.4.8): Extracting archive
- Installing symfony/deprecation-contracts (v2.5.1): Extracting archive
- Installing psr/container (1.1.1): Extracting archive
- Installing symfony/service-contracts (v2.5.1): Extracting archive
- Installing symfony/polyfill-php73 (v1.25.0): Extracting archive
- Installing symfony/console (v5.4.8): Extracting archive
- Installing laravel/installer (v4.2.10): Extracting archive
Some package suggestions were added by new dependencies, use 'composer suggest' to see details.
Generating autoload files
11 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!

C:\xampp\htdocs>laravel new laravel1

Laravel

Creating a "laravel/laravel" project at "./laravel1"
Installing laravel/laravel (v8.6.12)
- Downloading laravel/laravel (v8.6.12)
- Installing laravel/laravel (v8.6.12): Extracting archive
Created project in C:\xampp\htdocs\laravel1
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
```

a. Crear un Proyecto Laravel

- Abre una terminal o línea de comandos.
- Navega al directorio donde deseas crear tu proyecto Laravel.
- Ejecuta el comando “composer create-project

--prefer-dist laravel/laravel nombre_del_proyecto”

```
Selecionar C:\WINDOWS\system32\cmd.exe - php artisan serve
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
  Discovered Package: facade/ignition
  Discovered Package: fruitcake/laravel-cors
  Discovered Package: laravel/sail
  Discovered Package: laravel/sanctum
  Discovered Package: laravel/tinker
  Discovered Package: nesbot/carbon
  Discovered Package: nunomaduro/collision
  Package manifest generated successfully.
  76 packages you are using are looking for funding.
  Use the `composer fund` command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force
  No publishable resources for tag [laravel-assets].
  Publishing complete.
> @php artisan key:generate --ansi
  Application key set successfully.

Application ready! Build something amazing.

C:\xampp\htdocs>cd laravell

C:\xampp\htdocs\laravell>php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Wed May 25 12:43:11 2022] 127.0.0.1:2620 [200]: /favicon.ico
```

b. Configurar el Servidor Local

- Navega al directorio de tu proyecto Laravel.
- Ejecuta el comando “php artisan serve”.

```
Starting Laravel development server: http://127.0.0.1:8000
[Wed May 25 12:43:11 2022] 127.0.0.1:2620 [200]: /favicon.ico
```

- Abre un navegador web e ingresa <http://localhost:8000>

para ver tu aplicación