



SERVICIO NACIONAL DE ADIESTRAMIENTO EN TRABAJO

INDUSTRIAL

DIRECCION ZONAL AREQUIPA-PUNO

ESCUELA: TECNOLOGÍAS DE LA INFORMACIÓN

INGENIERIA DE SOFTWARE CON INTELIGENCIA ARTIFICIAL

PROYECTO DE MEJORA

“Desarrollo de un Sistema Automatizado para el Proceso de Cotizaciones en KATARI

A&C”

Estudiantes:

Jersson Pelayo Quispe Apaza

Josue Rosell Llatasi Chagua

Elmer Gabriel Quispe Ponce

Instructor:

Juan Carlos Flores Zenteno

Puno – Perú

2024

EPÍGRAFE

En algún lugar, algo increíble está esperando a ser descubierto. El corazón de la ciencia es la apertura a nuevas ideas y a los desafíos a las viejas.

Carl Sagan

DEDICATORIA

Dedico este trabajo a mi familia, por haberme dado la oportunidad de formarme profesionalmente y de que llegue tan lejos. A mi hermano por el apoyo y los consejos, a mis padres que siempre están para mí.

HOJA DE PRESENTACIÓN

ID : 001401239

NOMBRE : JERSSON PELAYO

APELLIDOS : QUISPE APAZA

TELEFONO : 998777712

CORREO : 1401239@senati.pe

ESPECIALIDAD : Ingeniería del Software con Inteligencia Artificial

C.F.P : SENATI - PUNO

BLOQUE : 54PIADS601

SEMESTRE : VI

INGRESO : 2021-20

EMPRESA : 20606248092 - KATARI A & C S.R.L.

HOJA DE PRESENTACIÓN

ID : 001408547

NOMBRE : Elmer Gabriel

APELLIDOS : Quispe Ponce

TELEFONO : 973598070

CORREO : 1408547@senati.pe

ESPECIALIDAD : Ingeniería del Software con Inteligencia Artificial

C.F.P : SENATI - PUNO

BLOQUE : 54PIADS601

SEMESTRE : VI

INGRESO : 2021-20

EMPRESA : 20606248092 - KATARI A & C S.R.L.

HOJA DE PRESENTACIÓN

ID : 001400000

NOMBRE : Josue Rosell

APELLIDOS : Llatasi Chagua

TELEFONO : 929 607 576

CORREO : 1400000@senati.pe

ESPECIALIDAD : Ingeniería del Software con Inteligencia Artificial

C.F.P : SENATI - PUNO

BLOQUE : 54PIADS601

SEMESTRE : VI

INGRESO : 2021-20

EMPRESA : 20606248092 - KATARI A & C S.R.L.

Resumen Ejecutivo Del Proyecto De Mejora.

El presente proyecto propone la implementación de un Sistema de Cotización Automatizado en respuesta a la necesidad de agilizar y optimizar el proceso de cotización de productos y servicios en el ámbito empresarial. Nuestro objetivo es mejorar el sistema de cotizaciones y proformas para realizar el seguimiento en tiempo real de los proyectos y sistemas desarrollados, con el fin de mejorar la eficiencia y la atención a los clientes.

El problema principal radica en la lentitud y la falta de eficiencia en la respuesta a las solicitudes de cotización por parte de los clientes, y el seguimiento de cada proyecto de forma automatizada, lo que impacta negativamente en la satisfacción del cliente y en la competitividad de KATARI A&C.

El análisis de la implementación propuesta revela que el Sistema de Cotización Automatizado permitirá estandarizar el proceso de cotización y generación de proformas, eliminando errores humanos y optimizando los recursos disponibles. Además, proporcionará una plataforma centralizada para gestionar información relevante de los clientes y sus requerimientos.

Se espera que la ejecución del proyecto genere resultados en la reducción de costos, incluyendo los costos operativos asociados al proceso de cotización y el aumento de ingresos, debido a una mayor eficiencia en la conversión de proformas a proyectos concretados, incrementando significativamente la rentabilidad en KATARI A&C.

Concluimos que la implementación del Sistema de Cotización Automatizado representa una oportunidad para la mejora, incrementa la competitividad y la eficiencia en la Empresa KATARI A&C, brindando respuestas rápidas y precisas a los requerimientos de los clientes, permitiendo hacer un seguimiento detallado por cada proyecto, mejorando la gestión de cotizaciones y proyectos.

Índice

Resumen Ejecutivo Del Proyecto De Mejora.	6
Índice.....	7
Capítulo I	11
1.1 Razón Social.	11
1.2 Misión, Visión, Objetivos, Valores de la empresa.	11
1.2.1 Misión	11
1.2.2 Visión	11
1.2.3 Objetivos De La Empresa	12
1.2.4 Valores De La Empresa.....	12
1.3 Productos, mercado, clientes.....	13
1.3.1 Productos.....	13
1.3.2 Mercado	14
1.3.3 Clientes	14
1.4 Estructura de la Organización	16
1.4.1 Organigrama de la empresa.....	16
1.5 Otra información relevante de la empresa donde se desarrolla el proyecto.	17
Capítulo II	18
2.1 Identificación del problema técnico en la empresa.	18
2.1.1 Encuesta y Observación (Anexo 1).....	19

2.2 Objetivos del Proyecto Mejora	20
2.2.1 Objetivo General.....	20
2.2.2 Objetivos Específicos.....	20
2.3 Antecedentes Del Proyecto de Mejora.....	20
2.3.1 Antecedentes Internacionales.....	20
2.3.2 Antecedentes Nacionales	22
2.4 Justificación del Proyecto de Mejora.....	22
2.5 Marco Teórico y Conceptual.....	25
2.5.1 Fundamento teórico del Proyecto de Mejora.....	25
2.5.2 Conceptos Y Términos Utilizados.....	29
Capítulo III.....	35
3.1 Diagrama Del Proceso	35
3.1.1 Diagrama de flujo del proceso actual.....	36
3.1.2 Diagrama de actividades de proceso actual	37
3.1.3 Diagrama de Operaciones del Proceso Actual	38
3.2 Efectos Del Problema En El Área De Trabajo O En Los Resultados De La Empresa.	39
3.3 Análisis De Las Causas Raíz Que Generan El Problema.	40
3.4 Priorización De Causas Raíz.....	41
CAPITULO IV.....	43
4.1 Plan De Acción De La Mejora Propuesta	43

4.1.1 Fase 1: Análisis de Requerimientos y Procesos de Negocio:	43
4.1.2 Fase 2: Diseño del Sistema:	43
4.1.3 Fase 3: Desarrollo del Software:.....	44
4.1.4 Fase 4: Implementación y Capacitación:	44
4.1.5 Fase 5: Evaluación y Mejora Continua:.....	44
4.2 Consideraciones Técnicas, Operativas y Ambientales Para La Implementación De La Mejora	48
4.2.1 Consideraciones Técnicas:.....	48
4.2.2 Consideraciones Operativas:.....	48
4.2.3 Consideraciones Ambientales:	49
4.3. Recursos técnicos para implementar la mejora propuesta	50
4.3.1 Materiales.....	50
4.3.2 Recursos Humanos.....	51
4.3.3 Herramientas y equipos.....	52
4.4 Diagrama del proceso, mapa del flujo, diagrama de operación de la situación mejorada..	53
4.4.1 Diagrama de flujo mejorado	53
4.4.2 Diagrama de actividades del proceso mejorado.....	54
4.4.3 Diagrama de operaciones del proceso mejorado	55
4.5 Cronograma de ejecución de la mejora.....	56
4.6 Aspectos limitantes para la implementación de la mejora	57

	10
CAPITULO V	58
5.1 Costo de materiales.	58
5.2 Costo de mano de obra.....	59
5.3 Costo de herramientas y equipos.	59
5.4 Otros costos de implementación de la Mejora.	60
5.5 Costo total de la implementación de la Mejora.	60
CAPITULO VI.....	61
6.1 Beneficio técnico y/o económico esperado de la Mejora	61
6.1.1 Beneficio Técnico	61
6.1.2. Beneficio Económico.....	62
6.2. Relación Beneficio/Costo.	63
6.2.1. Cálculo del Beneficio Neto Anual:	63
CAPITULO VII	64
7.1 Conclusiones respecto a los objetivos del Proyecto de Mejora.	64
CAPITULO VIII.....	66
8.1 Recomendaciones para la empresa respecto del Proyecto de Mejora.....	66
8.2. Otras recomendaciones	66
REFERENCIAS BIBLIOGRAFICAS.....	67
ANEXOS	71

Capítulo I

Generalidades De La Empresa

1.1 Razón Social.

RUC	: 20606248092
EMPRESA	: “KATARI A & C S.R.L.”
REPRESENTANTE LEGAL	: Edgar Apaza Choque
UBICACIÓN	: Barrio Porteño

1.2 Misión, Visión, Objetivos, Valores de la empresa.

1.2.1 Misión

“Ser una empresa que facilite, mejore y optimice la gestión de las empresas, ofreciendo soluciones tecnológicas y de análisis de datos, que satisfagan las necesidades de nuestros clientes.

Dar soporte en Tecnologías de la Información con base en el análisis de datos a Gobiernos Locales y Regionales, a micro, pequeñas y medianas empresas, con eficacia, eficiencia y efectividad, con un nivel de calidad que supere todas sus expectativas.”

1.2.2 Visión

“Al 2025 ser reconocidos como una empresa solida que ofrece y brinda servicios de calidad, excelencia y con profesionalismo. Apoyar a los clientes a revisar, solucionar y abastecer de equipos e insumos y/o proyectos que requieran.

Convertirnos en aliados estratégicos, generadores de soluciones innovadoras de la mano de los avances tecnológicos.”

1.2.3 Objetivos De La Empresa

Incrementar la base de clientes en un 20% anualmente: El objetivo es atraer y retener más clientes cada año, lo que se logrará mediante la mejora de la calidad del servicio, la implementación de estrategias de marketing efectivas y la oferta de soluciones personalizadas para satisfacer sus necesidades.

Expandir la presencia geográfica a nuevas regiones o mercados: Se busca ingresar a nuevas áreas geográficas o sectores del mercado, estableciendo relaciones con nuevos clientes y socios locales, para diversificar las oportunidades de negocio y aumentar la cuota de mercado de la empresa.

Aumentar los ingresos por ventas de productos y servicios en un 20% cada año: El objetivo es incrementar la facturación anual mediante estrategias de ventas eficaces, la ampliación de la oferta de productos y servicios, y el mejoramiento de la experiencia del cliente para fomentar la fidelidad.

1.2.4 Valores De La Empresa

Katari, empresa de soluciones tecnológicas, se rige por una serie de valores fundamentales que guían su enfoque hacia el trabajo y las relaciones con los empleados y clientes:

Innovación y excelencia: Katari se dedica a ofrecer soluciones tecnológicas innovadoras y de alta calidad para satisfacer las necesidades cambiantes de sus clientes. Esto incluye la búsqueda constante de mejoras en sus productos y servicios para mantenerse a la vanguardia del mercado.

Compromiso con el cliente: La empresa se esfuerza por establecer relaciones duraderas con sus clientes, brindando un servicio excepcional y

personalizado. Katari se compromete a comprender las necesidades de sus clientes y a ofrecer soluciones adaptadas a sus requisitos específicos.

Trabajo en equipo y colaboración: La empresa valora la diversidad de talentos y perspectivas dentro de su equipo, fomentando un ambiente de colaboración y trabajo conjunto para alcanzar objetivos comunes. Katari cree en el poder de la unión y la sinergia entre sus empleados para lograr resultados excepcionales.

1.3 Productos, mercado, clientes

1.3.1 Productos

Katari A&C ofrece una variedad de productos y servicios tecnológicos diseñados para satisfacer las necesidades y desafíos específicos de nuestros clientes. Nuestro portafolio de productos incluye:

Software Personalizado: Desarrollamos software a medida para empresas de diversos sectores, incluyendo aplicaciones empresariales, sistemas de gestión, soluciones de comercio electrónico y aplicaciones móviles.

Aplicaciones Web: Creamos aplicaciones web modernas y escalables utilizando tecnologías como Java, .NET, PHP y Python, ofreciendo funcionalidades personalizadas y una experiencia de usuario intuitiva.

Analítica de Datos: Ofrecemos servicios de análisis y modelado de datos utilizando técnicas de machine learning y análisis estadístico, proporcionando insights valiosos para la toma de decisiones estratégicas.

Infraestructura de Redes: Diseñamos, implementamos y mantenemos redes de datos y sistemas de cableado estructurado para instituciones y hogares, incluyendo instalaciones de cámaras de seguridad y cableado de fibra óptica.

1.3.2 Mercado

Katari A&C opera en un mercado dinámico y competitivo, sirviendo a clientes en diversos sectores y geografías. Nuestros principales segmentos de mercado incluyen:

Empresas y Organizaciones: Proporcionamos soluciones tecnológicas a empresas de todos los tamaños y sectores, ayudándolas a mejorar su eficiencia operativa, aumentar su productividad y alcanzar sus objetivos de negocio.

Gobiernos Locales y Regionales: Ofrecemos soporte en tecnologías de la información y análisis de datos a entidades gubernamentales, ayudándolas a optimizar la prestación de servicios públicos y mejorar la toma de decisiones basada en datos.

Micro, Pequeñas y Medianas Empresas (MIPYME): Atendemos las necesidades tecnológicas de las MIPYMEs, ofreciendo soluciones asequibles y personalizadas que les permitan competir en el mercado actual y adaptarse a las demandas cambiantes de la industria.

1.3.3 Clientes

Katari A&C se enorgullece de contar con una amplia base de clientes satisfechos que confían en nuestros productos y servicios para impulsar su éxito empresarial. Algunos de nuestros clientes destacados incluyen:

Empresas del sector financiero, como bancos y empresas de seguros, que utilizan nuestras soluciones de software para optimizar sus procesos comerciales y mejorar la experiencia del cliente.

Instituciones educativas y académicas que recurren a nuestras aplicaciones web y plataformas de aprendizaje virtual para ofrecer una educación de calidad y accesible a sus estudiantes.

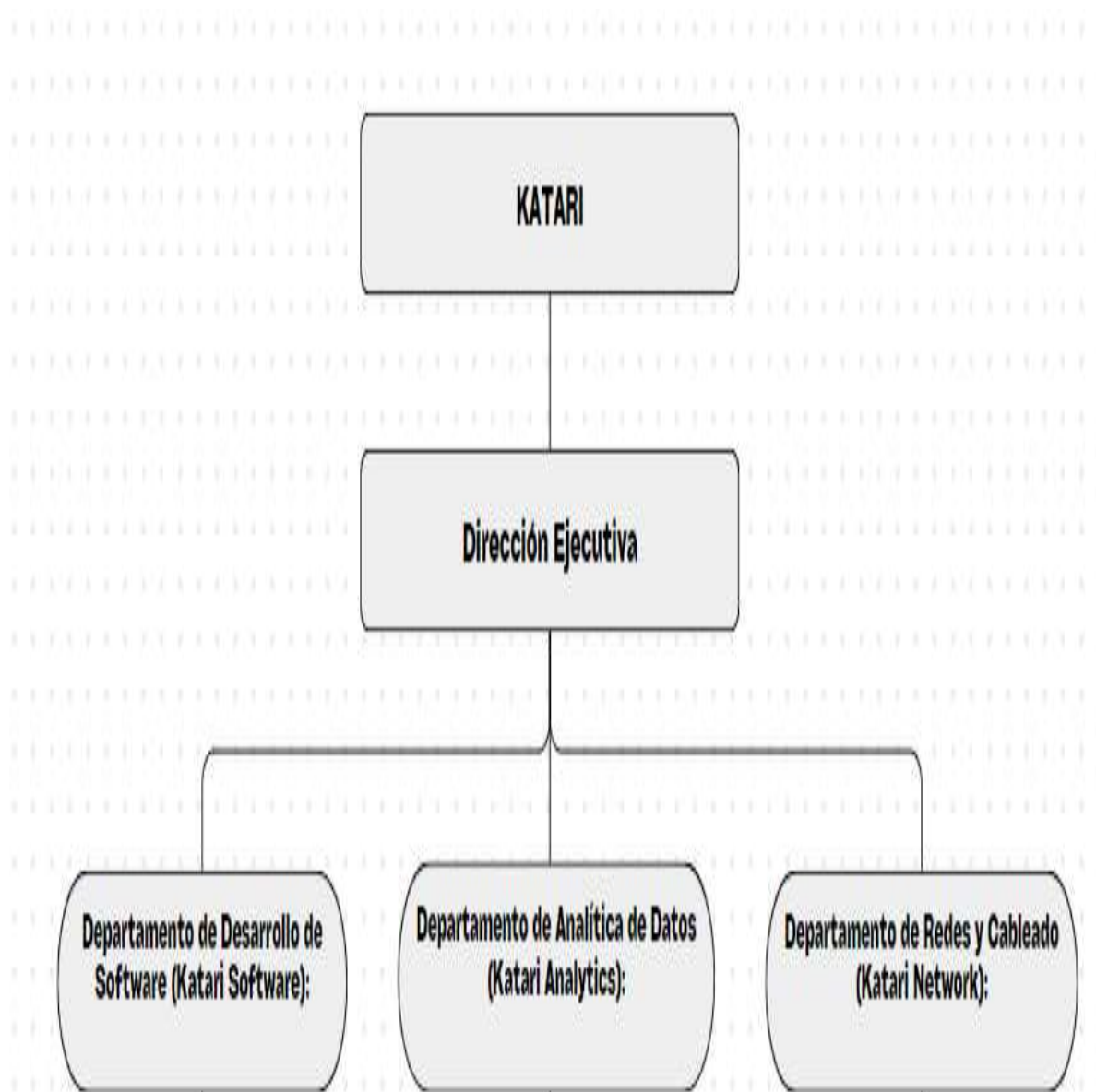
Organizaciones sin fines de lucro y agencias gubernamentales que confían en nuestra experiencia en análisis de datos para obtener perspectivas significativas y mejorar la eficacia de sus programas y servicios.

1.4 Estructura de la Organización

1.4.1 Organigrama de la empresa

Figura 1

Organigrama de la Empresa KATARI A&C



Fuente: Dirección Ejecutiva de KATARI A&C

1.5 Otra información relevante de la empresa donde se desarrolla el proyecto.

La empresa tiene una estructura organizativa bien definida, con departamentos especializados en diferentes áreas de sus operaciones. Estos incluyen desarrollo de software, análisis de datos, redes y cableado. Katari A&C S.R.L. sigue políticas empresariales basadas en la ética y responsabilidad social, buscando siempre satisfacer las necesidades de sus clientes de manera profesional y transparente. Además, la empresa promueve una cultura de innovación continua y mejora constante, con el objetivo de mantenerse a la vanguardia en el sector tecnológico y ofrecer soluciones de alta calidad.

Katari A&C S.R.L. se caracteriza por su capacidad de innovación, lo que le permite desarrollar soluciones tecnológicas personalizadas y adaptadas a las necesidades específicas de sus clientes. Esta habilidad para innovar y crear productos y servicios novedosos le ha permitido ganarse la confianza de una amplia base de clientes y destacarse en el mercado.

Capítulo II

PLAN DEL PROYECTO DE MEJORA.

2.1 Identificación del problema técnico en la empresa.

En toda empresa, la búsqueda de mejora continua en cada uno de sus procesos es fundamental para su sostenibilidad y éxito. En KATARI A&C, con una cartera de clientes establecida desde hace varios años, se brindan servicios en (03) áreas importantes como en el área de estadística, donde se elaboran todo tipo de análisis estadístico para todo tipo de investigaciones, tesis de pre grado y post grado. El otro rubro son el desarrollo de aplicaciones de software y diseño de páginas web y por último el servicio de instalación de redes de computadoras y cableado estructurado para personas naturales y empresas.

Actualmente, la empresa KATARI A&C concede información de costos y proformas a los clientes de manera manual, utilizando lápiz y papel para realizar cálculos de costos de servicios, productos, movilidad, alimentación, materiales, entre otros. Este enfoque anticuado para la elaboración de cotizaciones resulta ineficaz, ya que la falta de automatización ralentiza el proceso y provoca la pérdida de clientes potenciales.

El problema identificado es el procedimiento manual de cotizaciones que es propenso a errores humanos, lo que podría derivar en cálculos inexactos o confusiones en los precios de los productos y servicios. Además, este proceso es lento e insatisfactorio para el cliente, pues genera retrasos en la respuesta y, como consecuencia, el cliente pierde interés en trabajar con la empresa.

Asimismo, la empresa carece de una estandarización clara respecto a los servicios y productos que ofrece a sus clientes, lo que provoca inconsistencias en las cotizaciones y la pérdida de oportunidades de negocio.

2.1.1 Encuesta y Observación (Anexo 1)

Para poder identificar el problema técnico de mayor relevancia para la empresa se hizo una encuesta a todos los empleados, el cual al obtener sus respuestas recopilamos una serie de problemáticas con un mayor o menor impacto negativo.

Se recogieron las respuestas de los trabajadores y miembros de la empresa garantizando la confidencialidad de las respuestas para obtener opiniones sinceras y precisas.

Los datos recopilados fueron analizados para identificar patrones y tendencias. Se examinaron las respuestas para detectar puntos recurrentes de preocupación, así como áreas específicas donde los trabajadores experimentaban problemas.

El problema más recurrente encontrado fue la pérdida de clientes potenciales, el personal y la empresa en general tiene una deficiencia al dar información sobre costos, servicios y productos al cliente, ya que todos los cálculos de los costos se hacen arcaicamente, lo que produce una respuesta de información muy lenta que resulta en pérdidas de clientes molestos por la espera de información.

Determinación del problema clave: Con base en el análisis de los datos y los diagramas elaborados, se llegó a la conclusión de que el problema técnico clave era la lentitud y falta de precisión en el proceso de cotización. Esta problemática afecta tanto la satisfacción del cliente como la eficiencia operativa de la empresa.

2.2 Objetivos del Proyecto Mejora

2.2.1 Objetivo General

Desarrollar un sistema de cotizaciones y proformas para realizar el seguimiento en tiempo real de los proyectos, mejorando la eficiencia y la atención a los clientes.

2.2.2 Objetivos Específicos

- Analizar los requerimientos y requisitos del sistema de cotizaciones y proformas para el seguimiento en tiempo real de proyectos.
- Desarrollar una aplicación con una interfaz intuitiva y sencilla para el uso del usuario.
- Centralizar la gestión de datos, facilitando la actualización y consulta de información por parte del personal.

2.3 Antecedentes Del Proyecto de Mejora.

En el entorno empresarial actual, muchas compañías han refinado sus sistemas de cotización para optimizar sus procesos y mejorar la satisfacción del cliente. Estas mejoras han permitido a las empresas ofrecer respuestas más rápidas y precisas a las solicitudes de cotización, aumentando la eficiencia operativa y la experiencia del cliente.

2.3.1 Antecedentes Internacionales

(Tomala Torres, 2016) en su tesis titulada “Universidad católica de Santiago de Guayaquil (Ecuador) desarrollo e implementación de un sistema automatizado para el proceso de cotizar productos gráficos” para el caso de estudio de la empresa grafica GRAFIMPAC S.A, se concluyó que el sistema implementado ha

mejorado considerablemente la satisfacción de los clientes al hacer una consulta sobre precios y costos a la empresa, dando una rápida y mejor respuesta.

(Llorens Espada, 2022) en su artículo titulado “Aplicaciones informáticas (app) para el registro diario de la jornada laboral. Condiciones de licitud” explica que el estudio se vale de las directrices emanadas de la Agencia Española de Protección de Datos y demás instituciones públicas para analizar los retos y posibles quebradas legales que las aplicaciones informáticas pueden generar sobre los derechos y libertades de los trabajadores, a la vez que se pondera con las oportunidades y beneficios que se generan con ello para todas las partes implicadas. Todo ello se realiza con el apoyo de la específica jurisprudencia en la materia y una mirada al nivel de asunción de esta modalidad telemática de registro por los agentes sociales.

(Zaldívar & Meza, 2023) en su artículo titulado “Estudio de las frecuencias de cotizaciones en el Instituto de Previsión Social (IPS)” explica que gestión del área de preventa en una empresa de telecomunicación ubicada en Lima, Perú, la cual presentaba inconvenientes en los procesos de creación de propuestas económicas y técnicas, generando demoras en las entregas de las cotizaciones y pérdida de oportunidades de ventas y clientes. Mediante la aplicación del método científico, se elaboró instrumentos para poder recolectar datos acerca de las propuestas realizadas por el área de preventa durante los meses de agosto y setiembre, con una población de ocho semanas y un total de 43 propuestas, se observó un incremento del 43% en el índice de propuestas atendidas y un 48% en las propuestas aceptadas por los clientes. Además, los resultados mostraron una

reducción del 117% al tiempo de creación de propuestas y, también una reducción del 48% de las oportunidades perdidas.

2.3.2 Antecedentes Nacionales

(Torres Parodi, 2013) en su tesis titulada “Sistema automatizado de gestión de precios y valorización de cotizaciones para una línea naviera” de la Universidad Peruana de Ciencias Aplicadas (UPC) explica que este proyecto resuelve los problemas que origina la falta de estandarización, cotización y orden en los procesos involucrados. A la vez satisface las necesidades de automatización de las actividades correspondientes a la gestión de precios de Tótem Ocean Trailer Express – TOTE.

(Rios Rojas & Sanchez Alvitez, 2022) en su tesis titulada “Desarrollo de un sistema web para optimizar el proceso de cotizaciones en Respira S.A.C. de UNIVERSIDAD TECNOLÓGICA DEL PERÚ (UTP)” explica que el proyecto que fue implementado para la clínica de Salud Ocupacional Respira S.A.C, los usuarios se encuentran satisfechos, y se evidencia que se logra optimizar el proceso de cotizaciones al implementar el sistema web, además de que existe una correlación positiva entre sistema web y, disminución de tiempos de elaboración e incremento de atenciones a clientes.

2.4 Justificación del Proyecto de Mejora.

En la actualidad, la eficiencia y precisión en la gestión de cotizaciones, pagos y adelantos es crucial para cualquier organización, especialmente en entornos comerciales y académicos. Un sistema innovador que integre estas funcionalidades puede optimizar procesos administrativos, mejorar la transparencia financiera y facilitar la toma de

decisiones. Este proyecto de innovación se centra en desarrollar un sistema automatizado de cotización y registro de pagos y adelantos, proporcionando una solución integral y eficiente.

El problema que se investiga es la falta de eficiencia y la lentitud en el proceso de cotización de productos y servicios en el ámbito industrial de la Empresa KATARI A&C. La recopilación manual de información y la creación de presupuestos personalizados retrasan la respuesta a las solicitudes de los clientes, lo que afecta la satisfacción del cliente y la competitividad de la organización. La teoría de sistemas, propuesta por Ludwig von Bertalanffy, que plantea que las organizaciones pueden ser vistas como sistemas compuestos por múltiples partes interrelacionadas que trabajan juntas para lograr objetivos comunes (Bertalanffy, 1968). En el contexto de este proyecto, un sistema de cotización y registro de pagos es un conjunto de componentes interconectados (módulos de software, bases de datos, interfaces de usuario) que deben funcionar de manera cohesiva para proporcionar una gestión eficiente y precisa.

El impacto directo del problema de estudio en la satisfacción del cliente y la competitividad de la empresa KATARI A&C en el mercado industrial es la "emergencia" del problema de estudio. La falta de eficiencia en el proceso de cotización puede resultar en respuestas lentas a las solicitudes de los clientes, lo que puede resultar en una pérdida de oportunidades comerciales y una percepción de la calidad del servicio por parte de los clientes. Para mantener la competitividad y el éxito empresarial, se requiere una atención inmediata en esta situación. Esta va de la mano de la gestión financiera que es esencial para el éxito de cualquier organización. Según Van Horne y Wachowicz, la gestión financiera implica la planificación, organización, control y monitoreo de los recursos

financieros para alcanzar los objetivos organizacionales (Van Horne & Wachowicz, 2019). Un sistema automatizado puede mejorar estos procesos al proporcionar datos precisos y en tiempo real, reducir errores humanos y aumentar la transparencia en las transacciones financieras.

La investigación es altamente viable debido a la disponibilidad de recursos, el apoyo de la dirección y el potencial impacto positivo en la empresa y sus clientes.

El estudio tiene un impacto positivo tanto en la empresa KATARI A&C como en sus clientes industriales. Mejorar el proceso de cotización aumentará la competitividad de la empresa y la satisfacción del cliente al satisfacer sus necesidades con respuestas más rápidas y precisas. Esto fomenta la eficiencia operativa y la satisfacción del cliente en el mercado industrial, lo que contribuye al desarrollo económico y social.

El estudio no solo ofrece la posibilidad de adquirir habilidades profesionales y personales, sino que también contribuye al avance del conocimiento en las áreas de gestión empresarial y procesos industriales. Esto no solo beneficia a los participantes directos, sino también a la comunidad académica y profesional en general al mejorar las prácticas y técnicas utilizadas en el campo.

El sistema de cotización propuesto representa una innovación significativa al automatizar y centralizar estos procesos. Utilizando tecnologías modernas como bases de datos relacionales, interfaces web amigables y algoritmos de cálculo avanzados, el sistema ofrecerá:

- **Precisión:** Reducción de errores humanos mediante cálculos automáticos y registros detallados.

- **Eficiencia:** Aceleración de procesos administrativos, permitiendo a las organizaciones responder rápidamente a las solicitudes de cotización y gestionar pagos de manera oportuna.
- **Transparencia:** Provisión de registros claros y accesibles de todas las transacciones financieras, facilitando auditorías y mejorando la confianza del cliente.

2.5 Marco Teórico y Conceptual.

2.5.1 Fundamento teórico del Proyecto de Mejora.

El proyecto de implementación de un Sistema de Cotización Automatizado en Katari A&C S.R.L. se basa en varios fundamentos teóricos de la gestión empresarial, la automatización de procesos y las tecnologías de la información. A continuación, se describen los principales fundamentos teóricos que respaldan este proyecto:

- **Arquitectura de Software:** Es la forma en la que se organizan los componentes de un sistema, interactúan y se relacionan entre sí y con el contexto, aplicando normas y principios de diseño y calidad, que fortalezcan y fomenten la usabilidad a la vez que dejan preparado el sistema, para su propia evolución (Bahit, 2011).

El diseño del software se enfoca en cuatro atributos distintos del programa; la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. El proceso de diseño traduce los requisitos en una representación del software con la calidad requerida antes de que comience la codificación (Ble Jurado & Beas, 2010).

- **Metodología del Desarrollo de Software:** Según (García-Peñalvo & García-Holgado, 2018) indica que, desde una perspectiva de Ingeniería de Software, una metodología Describe cómo se organiza un proyecto, establece el orden en el que la mayoría de las actividades tienen que realizarse y los enlaces entre ellas, indica cómo tienen que realizarse algunas tareas proporcionando las herramientas concretas e intelectuales. Con una metodología se intentan cubrir las siguientes necesidades (Piattini Velthuis, 2004): Mejores aplicaciones, Mejor proceso de desarrollo, Establecer un proceso estándar en una organización.
- **Metodología Ágil SCRUM:** Es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

Scrum también se utiliza para resolver situaciones en que no se está entregando al cliente lo que necesita, cuando las entregas se alargan demasiado, los costes se disparan o la calidad no es aceptable, cuando se necesita capacidad de

reacción ante la competencia, cuando la moral de los equipos es baja y la rotación alta, cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo de producto (proyectosagiles.org, s.f.).

- **Tecnología de la Información (TI):** Juega un papel fundamental en la optimización de los procesos de cotización. Su aplicación estratégica permite la integración de sistemas y herramientas tecnológicas para agilizar, automatizar y optimizar las tareas relacionadas con la elaboración de cotizaciones. Esto incluye la utilización de software especializado para la gestión de clientes potenciales, la creación de presupuestos personalizados, la automatización de cálculos y la generación de informes detallados. Además, la implementación de hardware adecuado y la gestión eficiente de la información son aspectos cruciales para garantizar la precisión, confiabilidad y accesibilidad de los datos utilizados en el proceso de cotización (IKUSI velatia, s.f.).
- **Protección de Datos:** Abarca los principios y prácticas para garantizar la seguridad de la información recopilada durante el registro de visitantes. Esto implica medidas de protección de datos personales, cifrado de información sensible y cumplimiento de normativas de privacidad (Normativa de Protección de Datos Personales, s.f.).
- **Teoría de la automatización:** Es un pilar fundamental de la ingeniería de sistemas, cuyo objetivo es optimizar procesos mediante la implementación de tecnología para reducir o eliminar la intervención humana en tareas repetitivas y de bajo valor agregado. Esta automatización se basa en la aplicación de

principios como la retroalimentación, el control y la toma de decisiones autónoma, permitiendo optimizar la eficiencia, precisión y consistencia en la ejecución de tareas (Bolton, 2015).

- **UX (experiencia de usuario) y la UI (interfaz de usuario):** son dos elementos inseparables que trabajan en conjunto para garantizar que los usuarios tengan una interacción positiva y satisfactoria al utilizar un producto digital, como un sitio web o una aplicación.

La UX se enfoca en los aspectos emocionales y cognitivos de la interacción del usuario. Se trata de comprender las necesidades, expectativas y comportamientos del usuario para crear una experiencia fluida, intuitiva y agradable. Esto implica la planificación de la arquitectura de la información, la jerarquía de contenidos, la usabilidad y la accesibilidad.

La UI, por otro lado, se centra en los elementos visuales y la interacción directa del usuario con el producto. Incluye aspectos como el diseño gráfico, la tipografía, la paleta de colores, la disposición de los elementos y la respuesta a las acciones del usuario. Una buena interfaz de usuario debe ser atractiva, clara, consistente y fácil de usar.

En conjunto, la UX y la UI trabajan en armonía para crear productos digitales que sean útiles, agradables y memorables para los usuarios. Un enfoque equilibrado en ambos aspectos es crucial para el éxito de cualquier producto digital en la era actual (Guest, 2019).

- **Biblioteca de programación:** es una colección de código desarrollado previamente que los programadores pueden utilizar para desarrollar software de

manera más ágil. Estas colecciones de código reutilizable suelen resolver problemas o necesidades comunes de desarrollo (KSchool, 2024).

- **Principios de diseño de software:** Los principios como SOLID, DRY (Don't Repeat Yourself) y KISS (Keep It Simple, Stupid) guían la creación de software modular, escalable y fácil de mantener. Estos principios ayudan a estructurar el código de manera que sea comprensible y eviten problemas futuros (Autentia, 2024).
- **Patrones de diseño:** Los patrones de diseño son soluciones reutilizables para problemas comunes en el diseño de software. El uso de un patrón de diseño agiliza la creación del sistema, los más comunes son Factory y Singleton (autentia, 2024).

2.5.2 Conceptos Y Términos Utilizados.

- **PHP:** Acrónimo de "PHP: Hypertext Preprocessor" es un lenguaje de programación de propósito general de uso popular, especialmente adecuado para el desarrollo web. Se puede incrustar en HTML y se utiliza para crear páginas web dinámicas y aplicaciones del lado del servidor (PHP, 2020).
- **MySQL:** Es un sistema de gestión de bases de datos relacional (RDBMS, por sus siglas en inglés) de código abierto que utiliza el lenguaje de consulta estructurada (SQL) para manejar bases de datos. Es ampliamente utilizado para almacenar y organizar datos en aplicaciones web (Oppel & Sheldon, 2010).
- **SQL:** El lenguaje estructurado de consultas (SQL, Structured Query Language) es un lenguaje de programación estándar utilizado para gestionar y manipular bases de datos relacionales. SQL apoya la creación y mantenimiento de bases de datos

relacionales y la gestión de los datos dentro de ellas. Permite realizar diversas operaciones como inserción, consulta, actualización y eliminación de datos, así como la creación y modificación de la estructura de las bases de datos (Oppel & Sheldon, 2010).

- **Foundation:** Es un framework front-end que permite crear interfaces de usuario receptivas y atractivas. Ofrece una colección de componentes y herramientas de diseño que facilitan la construcción de sitios web y aplicaciones móviles adaptables a diferentes dispositivos y tamaños de pantalla (ZURB Foundation, 2018).
- **jQuery:** Es una biblioteca de JavaScript de uso común que facilita la manipulación del DOM, la gestión de eventos, la animación y la interacción con AJAX. Simplifica el desarrollo web al proporcionar una API intuitiva para realizar tareas complejas de forma más sencilla (jQuery foundation, 2020).
- **Apache:** Apache HTTP Server, comúnmente conocido como Apache, es un servidor web de código abierto que permite a los desarrolladores alojar y servir sitios web y aplicaciones. Es uno de los servidores web más populares y ofrece una variedad de características de configuración y seguridad (THE APACHE SOFTWARE FOUNDATION, 2020).
- **HTML:** HyperText Markup Language es un lenguaje de marcado utilizado para crear y estructurar contenido en páginas web. Define la estructura y el contenido de un documento web mediante etiquetas, como párrafos, encabezados, listas, enlaces e imágenes (W3C, 2021).
- **CSS:** Cascading Style Sheets es un lenguaje utilizado para describir la presentación y el diseño de un documento HTML. Permite definir el estilo visual de un sitio web,

incluyendo colores, fuentes, márgenes, alineaciones y disposiciones de elementos (W3C, 2020).

- **JavaScript:** Es un lenguaje de programación de alto nivel que se utiliza principalmente para agregar interactividad y dinamismo a las páginas web. Permite manipular el DOM, gestionar eventos y realizar cálculos del lado del cliente, entre otras tareas (ecma INTERNATIONAL, 2023).
- **UML:** Unified Modeling Language es un lenguaje de modelado visual que se utiliza para representar y documentar la estructura y el comportamiento de un sistema. Incluye diversos tipos de diagramas, como de clases, de secuencia y de casos de uso, entre otros (Object Management Group, 2017).
- **Chart.js:** Es una biblioteca de JavaScript que facilita la creación de gráficos interactivos y atractivos para la visualización de datos. Ofrece varias opciones de gráficos, como barras, líneas, circulares y radiales, con una API simple para la personalización (W3schools, 2020).
- **Fpdf:** Es una clase escrita en PHP que permite generar documentos PDF directamente desde PHP. Es útil para generar documentos en formato PDF a partir de contenido web, facilitando la creación de informes, facturas y otros documentos (FPDF Library, 2020).
- **Ubuntu Linux:** Es una distribución de Linux de código abierto y gratuita basada en Debian. Es una de las distribuciones de Linux más populares y ampliamente utilizadas en todo el mundo debido a su facilidad de uso, estabilidad y soporte integral (Ubuntu 20.04 LTS arrives, 2020).

- **Bases de datos:** Es una colección organizada de datos estructurados, almacenados y gestionados electrónicamente. Las bases de datos utilizan metadatos para definir cómo se almacenan los datos y pueden variar desde simples colecciones de nombres y direcciones hasta complejos sistemas de recuperación de datos (Oppel & Sheldon, 2010).
- **Normalización de Base de Datos:** La parte central de los principios del modelo relacional es el concepto de normalización, una técnica para producir un conjunto de relaciones que poseen un conjunto de ciertas propiedades que minimizan los datos redundantes y preservan la integridad de los datos almacenados tal como se mantienen (añadidos, actualizados y eliminados).

La normalización de bases de datos es un proceso que consiste en designar y aplicar una serie de reglas a las relaciones obtenidas tras el paso del modelo entidad-relación al modelo relacional (KENDALL, 2011).

- **Programación Orientada a Objetos (OOP):** Es un paradigma de programación que utiliza "objetos" para diseñar aplicaciones y sistemas. Los objetos contienen datos y métodos que operan sobre esos datos, facilitando la reutilización y modularidad del código.

Son un conjunto de técnicas que nos permiten incrementar enormemente nuestro proceso de producción de software; aumentando drásticamente nuestra productividad por un lado y permitiéndonos abordar proyectos de mucha mayor envergadura por otro. Usando estas técnicas, nos aseguramos la re-usabilidad de nuestro código, es decir, los objetos que hoy escribimos, si están bien escritos, nos servirán para "siempre".

Pero es que, y esto sí que es innovador, con OOP podemos re-usar ciertos comportamientos de un objeto, ocultando aquellos otros que no nos sirven, o redefinirlos para que los objetos se comporten de acuerdo a las nuevas necesidades (López Román, 2006).

- **MVC:** Modelo-Vista-Controlador es un patrón de arquitectura de software que organiza una aplicación en tres componentes interrelacionados: el modelo (gestión de datos), la vista (interfaz de usuario) y el controlador (gestión de la lógica). Este patrón facilita la separación de preocupaciones y mejora la mantenibilidad del código (Bahit, 2011).
- **Framework:** Es un conjunto de reglas y convenciones que se utilizan para desarrollar software de manera más eficiente y rápida. Proporciona una estructura básica que se puede utilizar como punto de partida para el desarrollo de aplicaciones (Gamma, 1994).
- **Servidor:** El termino servidor tiene dos acepciones en informática:
 - a) Servidor como Hardware
 - b) Servidor como Software.

Según IONOS explica que es un servidor como hardware es “es una máquina física integrada en una red informática en la que, además del sistema operativo, funcionan uno o varios servidores basados en software”, lo que en el argot informático se denomina “host” o anfitrión, aunque debemos aclarar que cada computador puede usarse como host.

Según IONOS está también nos plantea la definición de software como servidor indicando que “es un programa que ofrece un servicio especial que otros

programas denominados clientes (clients) pueden usar a nivel local o a través de una red”, y los servicios son variados de tipo servidor, desde una base de datos, especialmente los basados en el modelo cliente-servidor o intercambio de datos, y los ejemplos son muchos (IONOS, 2023).

- **Sistema operativo:** Los sistemas operativos son programas que gestionan los recursos de hardware y proporcionan servicios a los programas de aplicación. Son fundamentales para el funcionamiento eficiente de los sistemas informáticos (Stallings, 2018).

Capítulo III

Análisis De La Situación Actual

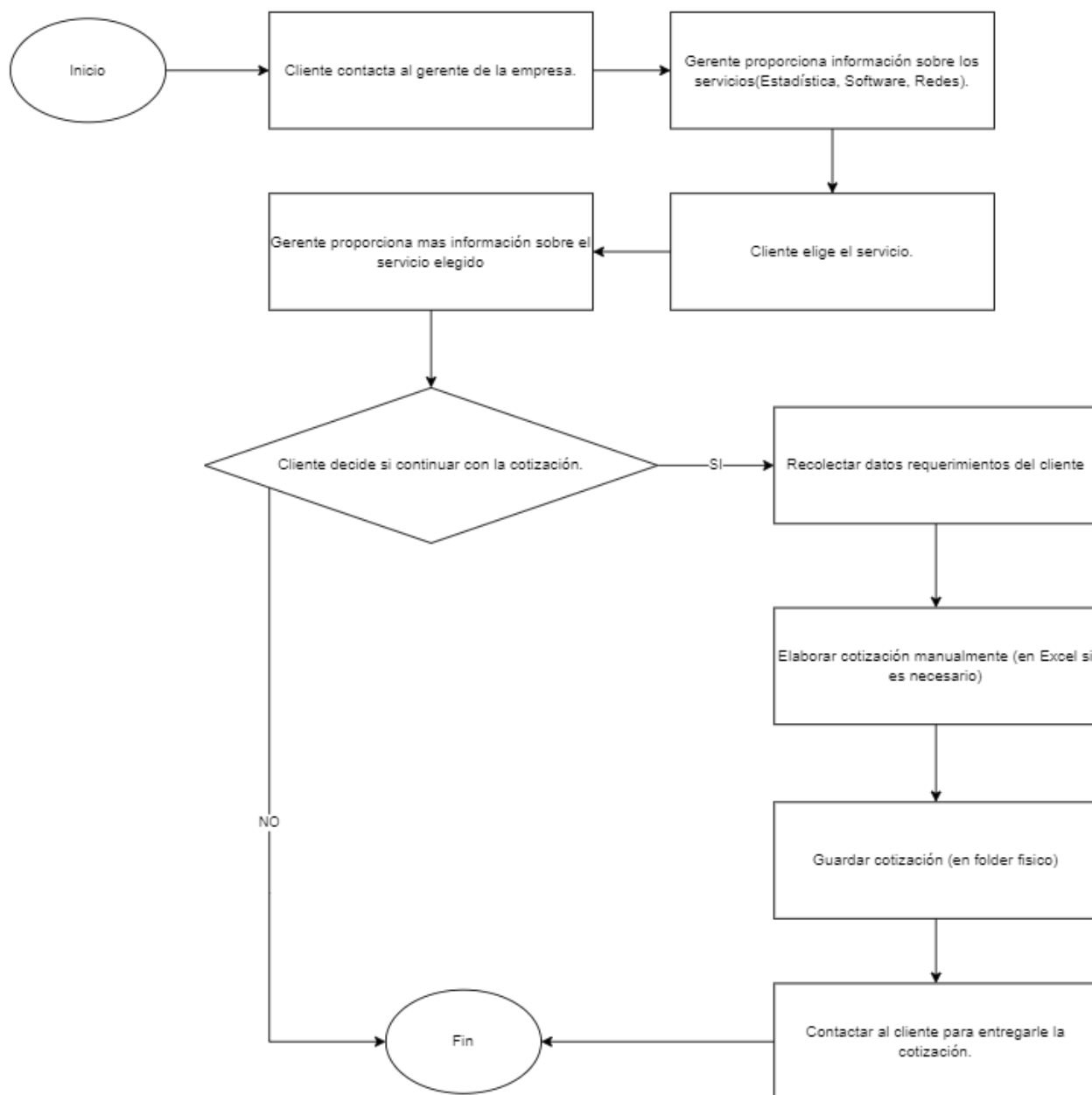
3.1 Diagrama Del Proceso

En esta sección se presentan los diagramas que ilustran el proceso actual de gestión de cotizaciones en la empresa KATARI A&C. Los diagramas proporcionan una visión clara y detallada de las actividades y operaciones involucradas, lo que permite identificar áreas de mejora y optimización. La representación visual del proceso es esencial para comprender los flujos de trabajo actuales y diseñar soluciones eficaces para la automatización y mejora de estos procesos.

3.1.1 Diagrama de flujo del proceso actual

Diagrama 1

Diagrama de Flujo del Proceso Actual



Fuente: Elaboración Propia

3.1.2 Diagrama de actividades de proceso actual

Diagrama 2

Diagrama de análisis del proceso actual (DAP)

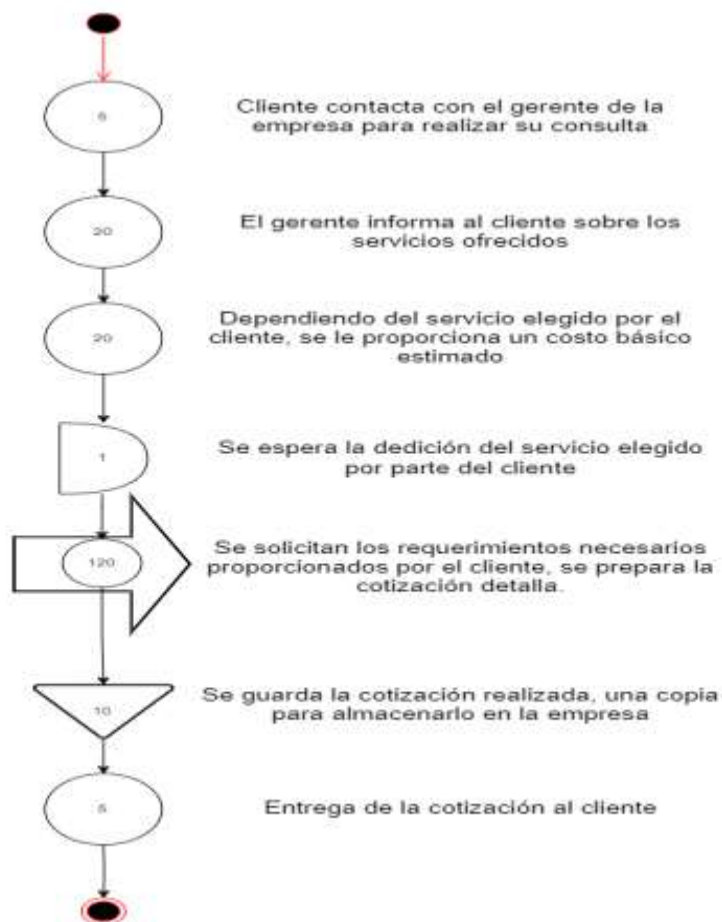
DIAGRAMA DE ANÁLISIS DEL PROCESO									
Objetivo: Cotizar servicios a los clientes		RESUMEN							
		ACTIVIDAD		ACTUAL	PROPUESTO	DIFERENCIA			
Proceso analizado: Cotizaciones para clientes	Método: Actual: X Propuesto	Operación	○	5	3				
		Transporte	⇒	1	0				
		Espera	○	2	0				
Localización: KATARI A&C		Inspección	□	2	0				
		Almacenamiento	▽	1	1				
Operario: Gerente de KATARI		Distancia (m)		-	-				
		Tiempo (min)		181 min	15 min				
		Costo s/.		-	-				
		Total		11	4				
Descripción	Costo	Distancia	Tiempo	Símbolo					Observaciones
				○	⇒	□	▽		
El cliente contacta con el gerente de la empresa para realizar su consulta	-	-							Inicio del proceso
El gerente informa al cliente sobre los servicios que la empresa ofrece, incluyendo Desarrollo de Software, Análisis Estadístico y Redes.	-	-	5 min	X					
Dependiendo del servicio elegido por el cliente, se le proporciona más información.	-	-	20 min	X					
Si el cliente tiene más dudas sobre el servicio elegido y desea conocer el precio según sus necesidades específicas.	-	-	1 min			X			Decisión del cliente
Se solicitan los requerimientos necesarios para el trabajo en cuestión al cliente.	-	-	20 min	X					El cliente brinda sus requerimientos
Con los datos y requerimientos proporcionados por el cliente, se prepara una cotización detallada que incluye el costo de cada elemento del servicio (una cotización completa).	-	-	120 min	X	X		X		Esta acción se realiza manualmente, a lápiz y papel se hacen los requerimientos y la cotización completa, para después digitalizar en Excel.
Se revisa y guarda la cotización hecha manualmente para luego entregársela al cliente.	-	-	10 min			X	X		Se hace una última revisión, se guarda y se hace una espera para entregárselo al cliente personalmente según sea el caso
Se le hace entrega de la cotización al cliente por medio físico o digital.	-	-	5 min	X					Fin del proceso
TOTAL	-	-	181min	5	1	2	2	1	

Fuente: Elaboración Propia

3.1.3 Diagrama de Operaciones del Proceso Actual

Diagrama 3

Diagrama de operaciones del proceso actual



Fuente: Elaboración Propia

Tabla 1

Resumen del Diagrama de operaciones de proceso actual

Tabla Resumen		
Actividad	Cantidad	Tiempo
Operación	5	160
Transporte	1	10
Almacenamiento	1	10
Espera	1	1
Total	7	181 min

Fuente: Elaboración Propia

3.2 Efectos Del Problema En El Área De Trabajo O En Los Resultados De La Empresa.

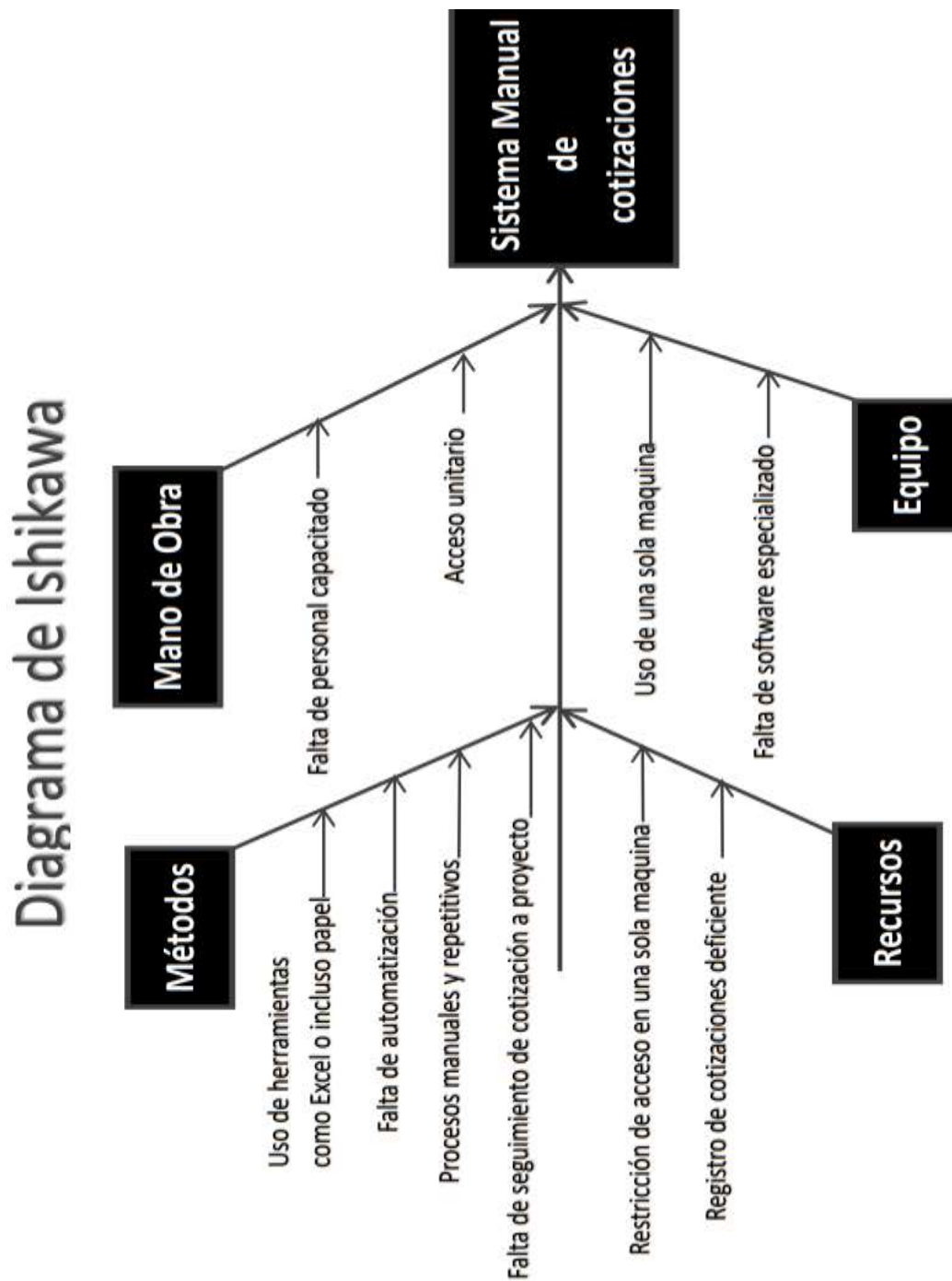
La ausencia de un sistema de cotizaciones en la empresa KATARI A&C, tienen impactos significativos en el área de trabajo y en los resultados de la institución:

- **Déficit en las Cotizaciones:** La falta de un sistema de Cotizaciones hace que se pierdan muchos clientes que hacen sus consultas por los servicios.
- **Falta de seguimiento a los proyectos en curso:** Al no registrar las cotizaciones, se pierde datos de las cotizaciones que pasan a ser proyectos.
- **Falta de Control y Supervisión:** La ausencia de un registro centralizado dificulta la supervisión y el seguimiento de los proyectos, ya que los datos se pierden por el tiempo y por tenerlo físicamente.
- **Impacto en la Imagen y Confianza Institucional:** La incapacidad para controlar y registrar adecuadamente las cotizaciones puede afectar la percepción pública sobre la seriedad, transparencia de la empresa KATARI disminuyendo la confianza de los ciudadanos.

3.3 Análisis De Las Causas Raíz Que Generan El Problema.

Diagrama 4

Diagrama de Ishikawa de identificación de los problemas



Fuente: Elaboración Propia

La combinación de estas problemáticas genera una deficiencia a la empresa, lo que conlleva a la necesidad de un sistema que automatice las cotizaciones de los productos y servicios que se ofrece. Resolver estas causas raíces requiere un enfoque integral, abordando tanto aspectos tecnológicos como culturales y de gestión para implementar soluciones efectivas y sostenibles.

3.4 Priorización De Causas Raíz.

En esta sección se presenta un análisis detallado de las causas raíz que generan las principales deficiencias en el sistema de cotización de la empresa. Para este propósito, se utilizó la herramienta del Diagrama de Pareto, que permite identificar y priorizar las causas más significativas que afectan el rendimiento del proceso.

Tabla 2

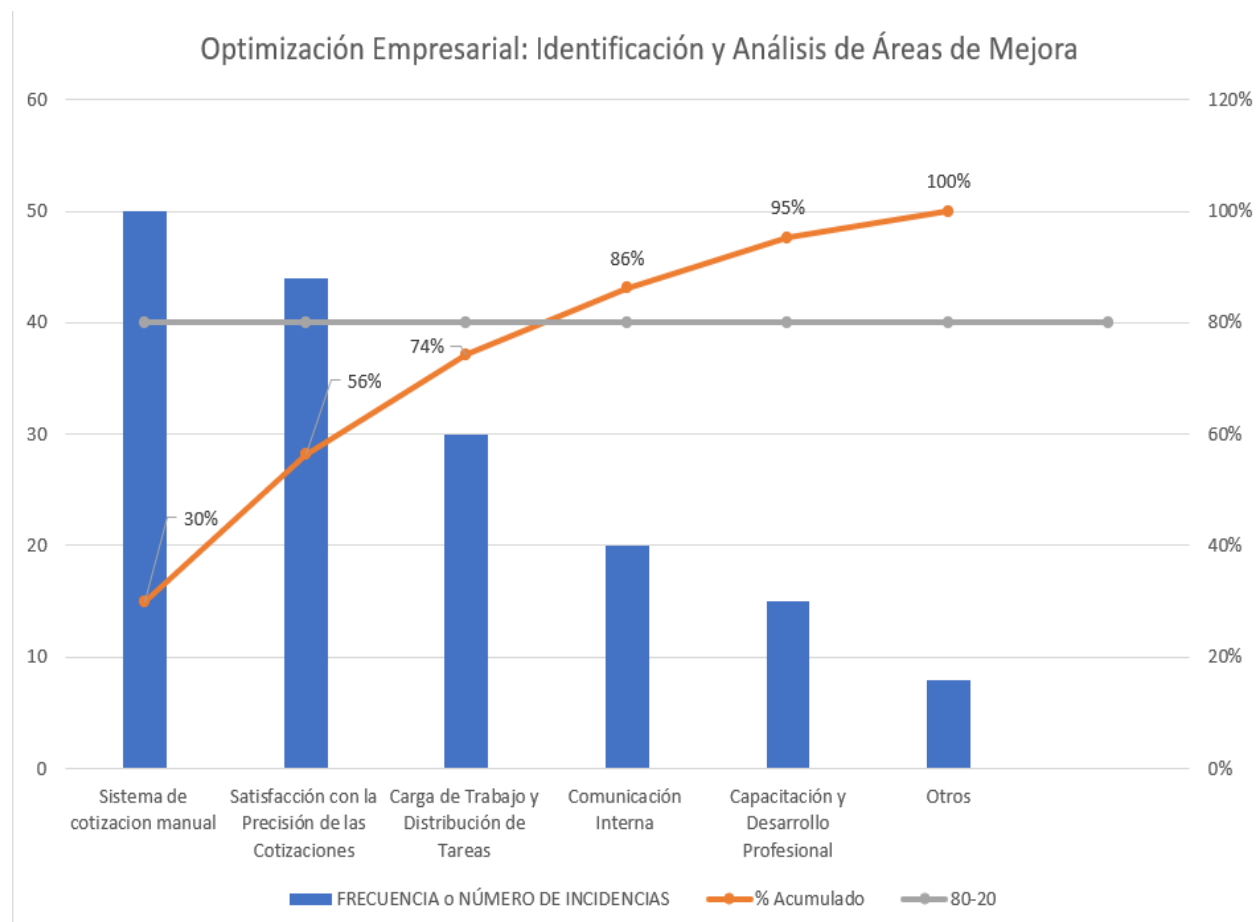
Análisis de causas y frecuencia de incidencias en el sistema de cotización

CAUSAS	FRECUENCIA o NÚMERO DE INCIDENCIAS	TOTAL	% Acumulado	80-20
Sistema de cotización manual	50	30%	30%	80%
Satisfacción con la Precisión de las Cotizaciones	44	26%	56%	80%
Carga de Trabajo y Distribución de Tareas	30	18%	74%	80%
Comunicación Interna	20	12%	86%	80%
Capacitación y Desarrollo Profesional	15	9%	95%	80%
Otros	8	5%	100%	80%
Total	167	100%		80%

Fuente: Elaboración Propia

Diagrama 5

Diagrama de Pareto optimización empresarial



Fuente: Elaboración Propia

Con esta herramienta determinamos que la causa principal de la deficiencia en la empresa es la demora en la respuesta a las consultas que hacen los clientes, este diagrama nos dio una perspectiva más amplia sobre qué problema es el que tiene mayor impacto en la empresa.

CAPITULO IV

PROPUESTA TECNICA DE LA INNOVACIÓN

4.1 Plan De Acción De La Mejora Propuesta

El presente plan de acción de la mejora propuesta tiene como objetivo principal el desarrollo e implementación de un sistema integral de cotización y seguimiento de proyectos para optimizar los procesos internos de la empresa. Con esta iniciativa se busca mejorar la eficiencia operativa, la precisión en la cotización de proyectos y el seguimiento detallado del progreso de cada proyecto, desde su inicio hasta su finalización.

Este plan se estructura en varias fases estratégicas diseñadas para abordar de manera sistemática cada aspecto del proceso de implementación del sistema:

4.1.1 Fase 1: Análisis de Requerimientos y Procesos de Negocio:

- Realizar un análisis detallado de los procesos de cotización y seguimiento de proyectos existentes en la empresa.
- Identificar las necesidades específicas de los diferentes departamentos involucrados en el proceso, como ventas, desarrollo de software, y análisis estadístico.
- Definir los requisitos del sistema, incluyendo funcionalidades clave, flujos de trabajo y necesidades de reportes.

4.1.2 Fase 2: Diseño del Sistema:

- Crear un diseño detallado del sistema, incluyendo la estructura de la base de datos, la interfaz de usuario y los flujos de trabajo
- Definir roles, permisos de acceso y seguridad del sistema.

4.1.3 Fase 3: Desarrollo del Software:

- Implementar el sistema de cotización y seguimiento de proyectos utilizando las tecnologías seleccionadas.
- Desarrollar módulos para la gestión de cotizaciones, seguimiento de proyectos, asignación de recursos, y generación de informes.
- Realizar pruebas exhaustivas para garantizar la funcionalidad, seguridad y usabilidad del sistema.

4.1.4 Fase 4: Implementación y Capacitación:

- Desplegar el sistema en un entorno de producción, asegurando una transición suave desde los procesos anteriores.
- Proporcionar capacitación a los usuarios finales y administradores del sistema sobre cómo utilizar las nuevas funcionalidades y realizar el seguimiento de proyectos de manera efectiva.
- Establecer procedimientos de soporte y mantenimiento para garantizar el funcionamiento continuo del sistema.

4.1.5 Fase 5: Evaluación y Mejora Continua:

- Monitorear el rendimiento del sistema y recopilar comentarios de los usuarios para identificar áreas de mejora.
- Realizar actualizaciones y mejoras periódicas en base a los comentarios y cambios en los requisitos del negocio.
- Mantener una comunicación abierta con los diferentes departamentos para garantizar que el sistema siga siendo relevante y efectivo a medida que evolucionan las necesidades de la empresa.

Diagrama 6

Diagrama de casos de uso

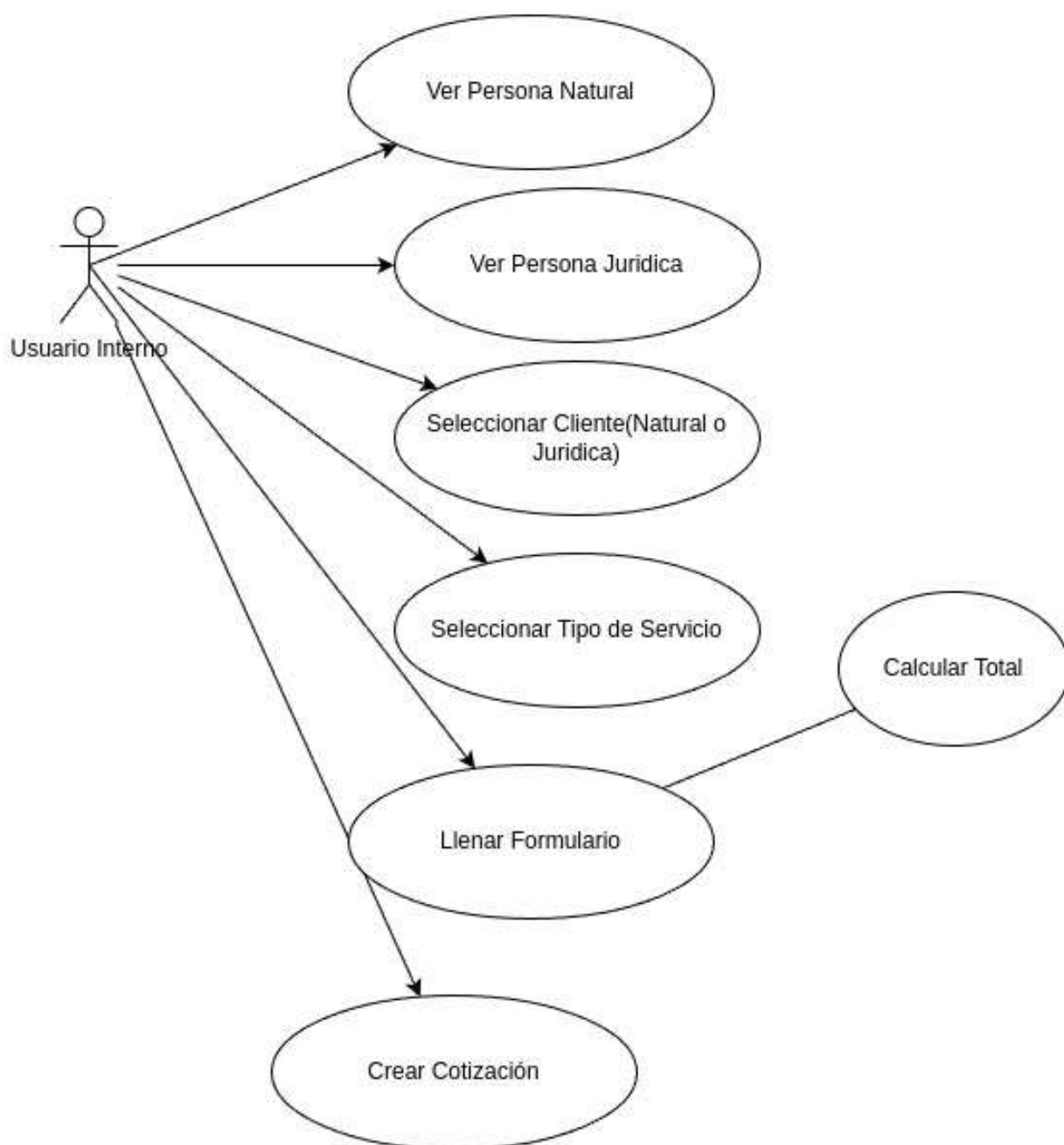
**Fuente:** Elaboración Propia

Diagrama 7

Diagrama de secuencia

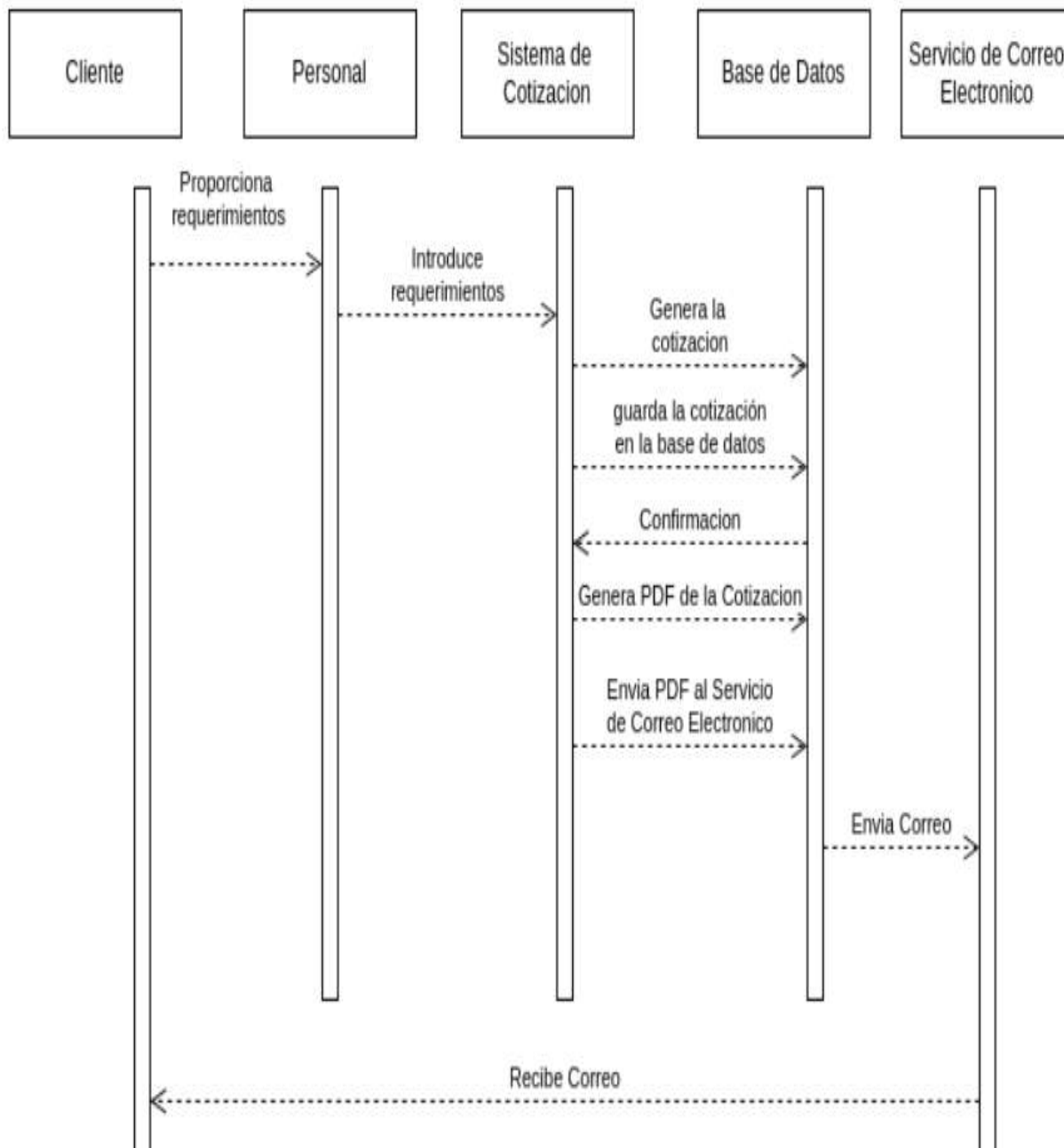
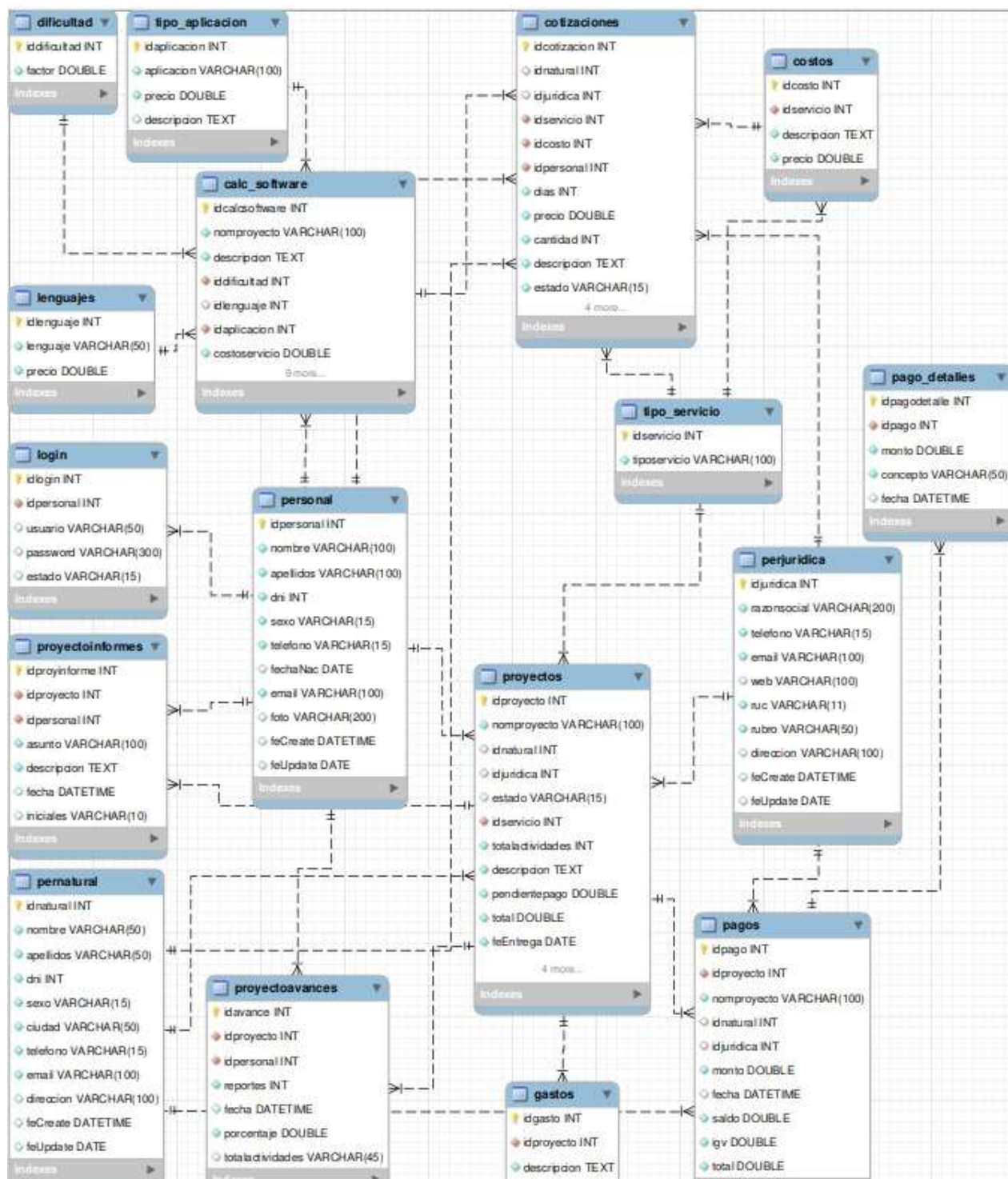
**Fuente:** Elaboración Propia

Diagrama 8

Diagrama de clases



Fuente: Elaboración Propia

4.2 Consideraciones Técnicas, Operativas y Ambientales Para La Implementación De La Mejora

4.2.1 Consideraciones Técnicas:

- **Compatibilidad Tecnológica:** Verifica la compatibilidad del sistema PHP y MySQL con la infraestructura tecnológica existente en la empresa de desarrollo de software. Asegurarse de que los servidores web puedan ejecutar PHP y que la base de datos MySQL esté disponible y sea accesible desde el sistema.
- **Seguridad de Datos:** Implementa medidas de seguridad robustas en el sistema para proteger la integridad y confidencialidad de la información almacenada en la base de datos. Utiliza técnicas de cifrado de datos y establece restricciones de acceso a la base de datos para garantizar que solo usuarios autorizados puedan acceder a la información.
- **Escalabilidad y Rendimiento:** Diseña el sistema de manera que sea escalable y capaz de manejar un gran volumen de cotizaciones y datos de seguimiento de proyectos sin comprometer su rendimiento. Optimiza las consultas a la base de datos y utiliza técnicas de caché para mejorar los tiempos de respuesta del sistema.
- **Respaldo y Recuperación de Datos:** Establece procedimientos de respaldo periódico de la base de datos para garantizar la recuperación de datos en caso de fallos o pérdida de información. Utiliza herramientas de respaldo automatizadas y asegúrate de que los Backups se almacenen en ubicaciones seguras y accesibles.

4.2.2 Consideraciones Operativas:

- **Procedimientos y Protocolos:** Establece procedimientos claros y protocolos de operación para el personal encargado del sistema. Define roles y responsabilidades,

y documenta los pasos necesarios para cotizar proyectos y dar seguimiento a su progreso.

- **Capacitación y Entrenamiento:** Brindar capacitación adecuada al personal sobre el manejo del sistema, haciendo hincapié en la importancia de seguir los protocolos de seguridad y procedimientos establecidos. Proporcionar recursos de capacitación, como manuales y tutoriales, y ofrecer sesiones de entrenamiento práctico.
- **Mantenimiento Preventivo:** Programa actividades de mantenimiento regular para el sistema y sus componentes. Realiza actualizaciones de seguridad y optimización de rendimiento de forma periódica, y asegurarse de que se sigan las mejores prácticas de mantenimiento de sistemas web.

4.2.3 Consideraciones Ambientales:

- **Impacto Ambiental del Hardware:** Considera el impacto ambiental del hardware utilizado en el sistema. Opta por servidores y equipos de bajo consumo energético y asegúrate de que se dispongan adecuadamente los residuos electrónicos al final de su vida útil.
- **Sostenibilidad en la Implementación:** Busca soluciones que minimicen el uso de papel y recursos físicos en el proceso de cotización y seguimiento de proyectos. Utiliza herramientas digitales y electrónicas para reducir la huella ambiental y promueve prácticas sostenibles en toda la empresa.
- **Cumplimiento Normativo Ambiental:** Asegúrate de cumplir con las regulaciones ambientales y disposiciones legales al implementar el sistema. Investiga las normativas locales y regionales relacionadas con el manejo de datos y residuos

electrónicos, y asegúrate de cumplir con todas las obligaciones legales y ambientales pertinentes.

4.3. Recursos técnicos para implementar la mejora propuesta

4.3.1 *Materiales*

- **Sistema Operativo (Linux):** Se optó por un sistema operativo basado en Linux, debido a su velocidad, seguridad y compatibilidad con diversas herramientas de software.
- **Visual Studio Code:** Este es un editor de código fuente desarrollado por Microsoft, pieza fundamental para programar cualquier sistema. Ofrece soporte para múltiples lenguajes de programación y extensiones que facilitan el desarrollo.
- **Foundation:** Es un framework de maquetado en HTML y CSS, el cual ayuda a crear sistemas responsivos y móviles. Proporciona componentes predefinidos que aceleran el diseño y desarrollo de la interfaz de usuario.
- **URL Amigables:** Configuración especial del servidor para trabajar más eficaz y modular de una mejor manera el sistema.
- **GitHub:** Es una plataforma para alojar código fuente utilizando el sistema de control de versiones Git. Permite la colaboración entre desarrolladores, el seguimiento de cambios en el código y la gestión de versiones del proyecto.
- **Servidor Apache:** Es un servidor web de código abierto que proporciona una plataforma robusta y confiable para alojar aplicaciones web. Es compatible con una amplia gama de tecnologías y lenguajes de programación.
- **Canva Premium:** Es una herramienta de diseño gráfico en línea que ofrece recursos avanzados para la creación de elementos visuales atractivos.

4.3.2 Recursos Humanos

Programador junior Backend:

- Responsabilidades: Desarrollo y mantenimiento del lado del servidor de la aplicación, implementación de la lógica de negocio, creación y gestión de API, integración con la base de datos, y aseguramiento de la seguridad y rendimiento del sistema.
- Habilidades Requeridas: Conocimientos en el lenguaje de programación PHP; experiencia con bases de datos SQL; comprensión de arquitectura de software y RESTful APIs.

Programador junior Frontend:

- Responsabilidades: Desarrollo de la interfaz de usuario utilizando tecnologías como HTML, CSS y JavaScript; aseguramiento de que la aplicación sea responsiva y accesible en diferentes dispositivos; colaboración con el desarrollador backend para implementar las funcionalidades del sistema.
- Habilidades Requeridas: Conocimientos en frameworks y bibliotecas de JavaScript como jQuery, Foundation; habilidades en diseño y experiencia de usuario (UX/UI).

Programador junior Data Base:

- Responsabilidades: Diseño, implementación y mantenimiento de las bases de datos utilizadas por la aplicación; optimización de consultas y rendimiento de la base de datos; implementación de procedimientos de backup y recuperación de datos; aseguramiento de la integridad y seguridad de los datos.

- **Habilidades Requeridas:** Conocimientos en sistemas de gestión de bases de datos como MySQL; habilidades en modelado de datos y diseño de bases de datos; comprensión de procedimientos almacenados, triggers y optimización de consultas.

4.3.3 Herramientas y equipos

Laptops: Se necesitarán laptops para cada programador, equipadas con el software necesario para el desarrollo.

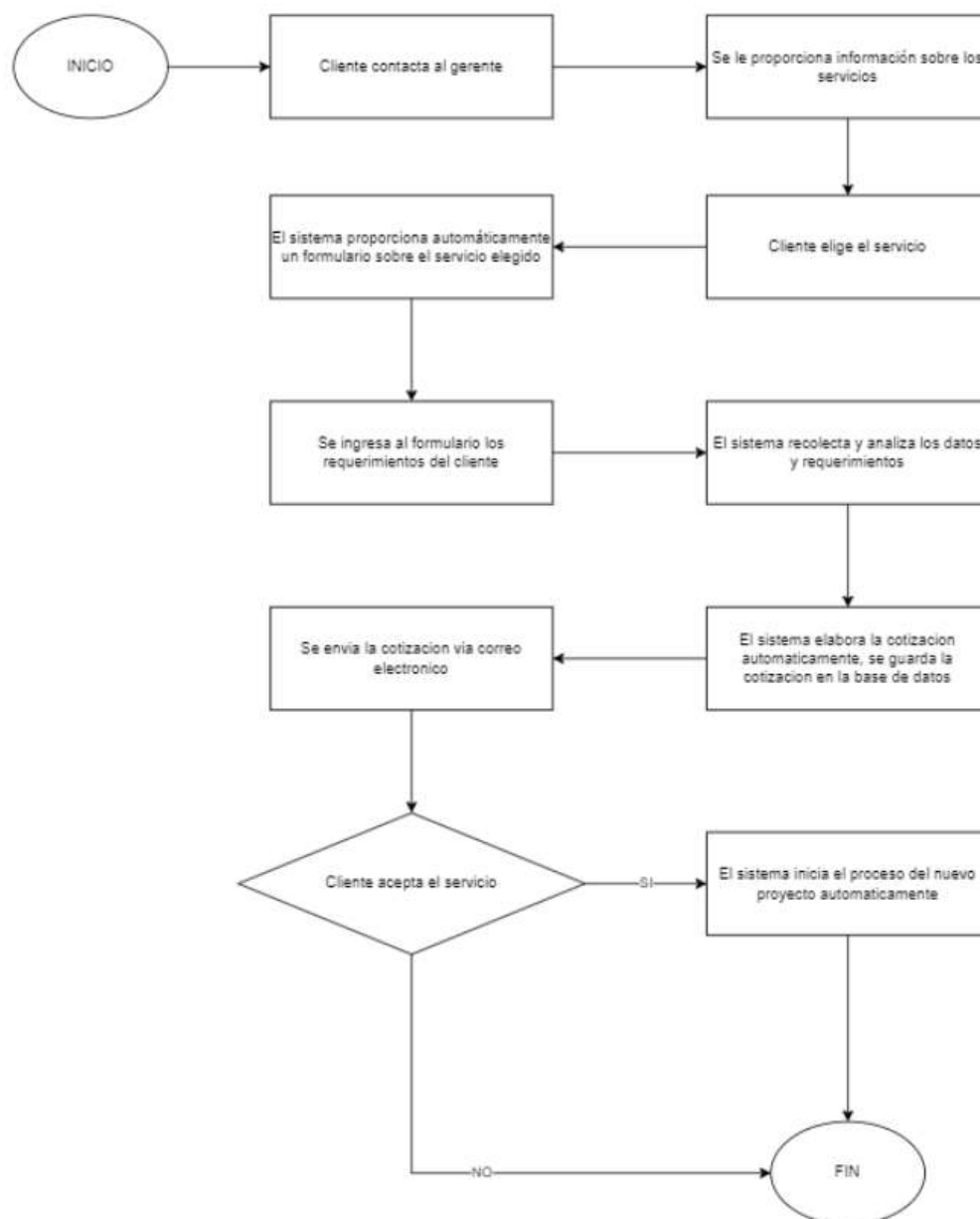
Conexión a internet: Conexión de alta velocidad necesaria para permitir la colaboración en línea y el acceso a recursos de desarrollo.

4.4 Diagrama del proceso, mapa del flujo, diagrama de operación de la situación mejorada

4.4.1 Diagrama de flujo mejorado

Diagrama 9

Diagrama de flujo mejorado



Fuente: Elaboración Propia

4.4.2 Diagrama de actividades del proceso mejorado

Diagrama 10

Diagrama de análisis del proceso mejorado

DIAGRAMA DE ANÁLISIS DEL PROCESO										
Objetivo: Cotizar servicios a los clientes		RESUMEN								
		ACTIVIDAD		ACTUAL	MEJORADO	DIFERENCIA				
		Operación	○	5	3	-2				
Proceso analizado:	Cotizaciones para clientes	Transporte	⇒	1	0	-1				
		Espera	□	2	0	-2				
Método: Actual	Propuesto: X	Inspección	□	2	0	-2				
		Almacenamiento	▽	1	1	0				
Localización: KATARI A&C		Distancia (m)		-	-	-				
		Tiempo (min)		181 min	15 min	166 min				
Operario: Gerente de KATARI		Costo s/.		-	-	-				
		Total		11	4	7				
Descripción		Costo	Distancia	Tiempo	Símbolo					Observaciones
					○	⇒	□	▽		
El cliente contacta con el gerente de la empresa para realizar su consulta		-	-							Inicio del proceso
El gerente informa al cliente sobre los servicios que la empresa ofrece, incluyendo Desarrollo de Software, Análisis Estadístico y Redes.		-	-	5 min	X					
Se le solicita sus requerimientos al cliente		-	-	10 min	X					
Se guarda la cotización realizada		-	-	10 seg					X	
Se le entrega la cotización al cliente por medio digital		-	-	30 seg	X					Fin del proceso
TOTAL		-	-	15:40 min	3	-	-	-	1	

Fuente: Elaboración Propia

4.4.3 Diagrama de operaciones del proceso mejorado

Diagrama 11

Diagrama de operación del proceso mejora



Fuente: Elaboración Propia

Tabla 3

Resumen del diagrama de operación del proceso mejorado

Tabla Resumen		
Actividad	Cantidad	Tiempo
Operación	3	15:30 min
Almacenamiento	1	10 seg
Total	4	15:40 min

Fuente: Elaboración Propia

4.5 Cronograma de ejecución de la mejora

Tabla 4

Cronograma de actividades

CRONOGRAMA														
Actividades	Duración	Marzo		Abril			Mayo				Junio			
		Semana 1	Semana 2	Semana 3	Semana 4	Semana 5	Semana 6	Semana 7	Semana 8	Semana 9	Semana 10	Semana 11	Semana 12	Semana 13
Análisis de Requerimientos y Procesos de Negocio														
Identificación de requerimientos	2 semana													
Análisis de procesos actuales	1 semana													
Redacción de documento de requerimientos	1 semana													
Diseño del Sistema														
Diseño de arquitectura	1 semana													
Diseño de interfaces de usuario	1 semana													
Revisión y aprobación del diseño	1 semana													
Desarrollo del Software														
Configuración del entorno de desarrollo	1 semana													
Programación de funcionalidades	2 semana													
Pruebas unitarias	1 semana													
Implementación														
Instalación del sistema en entorno de producción	2 semana													
Integración y pruebas de sistema	2 semana													
Soporte inicial y ajuste de errores	1 semana													
Evaluación y Mejora														
Evaluación del desempeño del sistema	3 semana													
Recopilación de comentarios de usuarios	2 semana													
Implementación de mejoras	1 semana													

Fuente: Elaboración Propia

4.6 Aspectos limitantes para la implementación de la mejora

Frecuencia de Capacitación: Una de las principales limitaciones es la insuficiente frecuencia de capacitación en nuevas tecnologías y herramientas. La elevada curva de aprendizaje y el tiempo limitado disponible para estas capacitaciones dificultan que el personal adquiera y aplique eficazmente nuevos conocimientos y habilidades. Esto puede retrasar la adopción del nuevo sistema y afectar su uso óptimo.

Conectividad y Latencia de Internet: La conectividad inestable y la alta latencia de internet representan otro desafío importante. Estos problemas afectan negativamente la comunicación y el trabajo en equipo, especialmente durante las reuniones virtuales y la colaboración en tiempo real en el desarrollo de proyectos. La falta de una conexión robusta puede llevar a interrupciones, malentendidos y retrasos en la implementación del sistema.

CAPITULO V

COSTOS DE IMPLEMENTACION DE LA MEJORA

En el marco del proyecto de mejora propuesto, es esencial comprender los costos asociados a su implementación. Estos costos abarcan diversos aspectos, desde la adquisición de materiales hasta la inversión en software y otros gastos necesarios para asegurar el éxito del proyecto. A continuación, se presenta un análisis detallado de los costos de implementación:

5.1 Costo de materiales.

La adecuada provisión de materiales es fundamental para el desarrollo fluido de este proyecto. Aquí se detalla la inversión necesaria para llevar a cabo nuestra iniciativa de mejora:

Tabla 5

Costos de materiales

Nº	Descripción	Unidad de medida	Costo unitario	Cantidad	Costo total
1	Sistema Operativo (Linux)	Licencia	S/. 20.00	3	S/. 60.00
2	Visual Studio Code	Licencia	S/. 00.00	1	S/. 00.00
3	Framework Foundation	Licencia	S/. 00.00	1	S/. 00.00
4	URL Amigables	Licencia	S/. 00.00	1	S/. 00.00
5	GitHub	Licencia	S/. 20.00	3	S/. 60.00
6	API's	Licencia	S/. 30.00	3	S/. 90.00
7	Servidor Apache	Licencia	S/. 00.00	1	S/. 00.00
8	Canva premium (Diseño)	Licencia	S/. 50.00	3	S/. 150.00
TOTAL					S/. 360.00

Fuente: Elaboración Propia

5.2 Costo de mano de obra.

Para asegurar la ejecución eficiente de nuestro proyecto, es fundamental contar con un equipo capacitado y dedicado. A continuación, se presenta un desglose de los costos de mano de obra:

Tabla 6

Costos de mano de obra

ÍTEMS	TÉCNICO	OPERACIÓN	COSTOS x HORA	TIEMPO DE HORAS	TOTAL	SEMANAL
1	Programador junior	Desarrollo Front End	S/. 3	4	S/. 12.00	S/. 84.00
2	Programador junior	Desarrollo Backend	S/. 3	4	S/. 12.00	S/. 84.00
3	Programador junior	Desarrollo de base de datos	S/. 3	4	S/. 12.00	S/. 84.00
TOTAL					S/. 36.00	S/. 252.00

Fuente: Elaboración Propia

5.3 Costo de herramientas y equipos.

La disponibilidad de máquinas, herramientas y equipos adecuados son esenciales para llevar a cabo nuestro proyecto de mejora con éxito.

Tabla 7

Costos de herramientas y equipos

Nº	Descripción	Unidad de medida	Costo unitario	Cantidad	Costo total
1	Alquiler laptop Asus TUF core i5	Unidad	S/. 50.00	1	S/. 50.00
2	Alquiler laptop hp core i5	Unidad	S/. 50.00	1	S/. 50.00
3	Alquiler laptop hp core i5	Unidad	S/. 50.00	1	S/. 50.00
4	Conexión a internet	Unidad	S/. 60.00	1	S/. 60.00
TOTAL					S/. 210.00

Fuente: Elaboración Propia

5.4 Otros costos de implementación de la Mejora.

A continuación, se detallan los costos que se utilizaran para la implementación del sistema:

Tabla 8

Otros costos

Nº	Descripción	Cantidad	Costo por mes	Costo Anual
1	Dominio	1	S/ .09.00	S/. 110
2	Hosting	1	S/ .20.00	S/. 240.00
3	Plan de internet	2	S/ .180.00	S/.2160
4	Dispositivo Portátil	1	S/ .100.00	S/.1200
	Total		S/.309.00	S/. 3710

Fuente: Elaboración Propia

5.5 Costo total de la implementación de la Mejora.

A continuación, se presenta el costo total estimado para la implementación de nuestra mejora. Este cálculo incluye los costos de materiales, mano de obra, máquinas, herramientas, equipos y servicios necesarios para asegurar el éxito del proyecto.

Tabla 9

Costo total de la implementación

COSTO ESTIMADO DE LA IMPLEMENTACIÓN	
Costos	Totales
Costos de materiales	S/. 360.00
Costo de mano de obra	S/. 252.00
Costo de máquinas, herramientas y equipos	S/. 210.00
Otro costo de implementación	S/. 309.00
TOTAL	S/. 1131.00

Fuente: Elaboración Propia

CAPITULO VI

EVALUACION TECNICA Y ECONOMICA DE LA MEJORA

6.1 Beneficio técnico y económico esperado de la Mejora

6.1.1 Beneficio Técnico

Automatización de Procesos:

- La generación automática de cotizaciones y proformas reduce el tiempo necesario para elaborar manualmente estos documentos, acelerando el proceso desde la solicitud del cliente hasta la entrega.
- La automatización minimizara la probabilidad de errores comunes en la elaboración de cotizaciones y cálculos, garantizando una mayor precisión.

Colaboración Mejorada:

- Varios usuarios pueden trabajar simultáneamente en el sistema, evitando conflictos de versiones y facilitando la colaboración.
- El sistema web puede manejar automáticamente el control de versiones, asegurando que todos los usuarios trabajen con la versión más reciente de la información.

Acceso en tiempo real:

- Capacidad de calcular costos y generar proformas en tiempo real asegura que la información proporcionada a los clientes sea siempre la más actualizada y precisa.
- Seguimiento continuo de los proyectos permite a los gestores y equipos tener una visión clara del progreso, detectar desviaciones y tomar decisiones informadas rápidamente.

6.1.2. Beneficio Económico

Reducción de Costos Operativos:

- La automatización del proceso de cotización y la mejora en la gestión de datos reducen la necesidad de personal dedicado a tareas repetitivas y manuales.
- Ahorro significativo en costos laborales y operativos.
- Disminuye la necesidad de trabajo manual, reduciendo costos laborales.
- Reduce errores en cotizaciones que pueden llevar a pérdidas financieras.

Incremento en la Eficiencia y Productividad:

- Acelera el proceso de cotización, permitiendo a los empleados manejar más solicitudes en menos tiempo.
- Optimiza el uso de recursos internos, aumentando la productividad.
- Ofrece cotizaciones más precisas y rápidas, mejorando la satisfacción del cliente y aumentando la probabilidad de cerrar ventas, mejorando la retención y la lealtad del cliente.

Reducción de Errores Costosos:

- Minimiza errores en la generación de cotizaciones, reduciendo el riesgo de pérdidas financieras.
- Asegura que los proyectos se cotizan adecuadamente, evitando subestimaciones.
- La automatización en la generación de cotizaciones reduce los errores humanos que pueden ocurrir en la entrada manual de datos.

6.2. Relación Beneficio/Costo.

Tabla 10

Relación beneficio y costo

Ganancias de la empresa Katari	
Costos	Totales Mensual
Proyecto MTC	S/. 430.00
Proyecto laboratorio UNA-PUNO	S/. 320.00
Trabajos de Estadística	S/. 350.00
TOTAL	S/. 1100.00

Fuente: Elaboración Propia

Para calcular la relación beneficio/costo, consideramos los ingresos adicionales y los costos operativos en soles.

- Ingresos adicionales esperados trimestral: 1100 soles x 3 = 3300
- Costos operativos trimestral del software: 309 soles x 3 = 927

6.2.1. Cálculo del Beneficio Neto Trimestral:

Beneficio Neto Trimestral = Ingresos Adicionales - Costos Operativos

Beneficio Neto Trimestral= 3300 soles - 927 soles = 2373 soles

$$Relacion \frac{B}{C} = \frac{\text{Beneficio Trimestral}}{\text{Total de la inversion}} = \frac{2373}{1131} = 2.09$$

La interpretación que obtuvimos es que tenemos una relación beneficio/costo de 2.09 esto indica que, por cada sol invertido en el desarrollo del software, se espera un retorno de 2.09 soles en beneficios netos trimestral. Esto sugiere que la inversión en el software es económicamente viable y rentable.

CAPITULO VII

CONCLUSIONES

7.1 Conclusiones respecto a los objetivos del Proyecto de Mejora.

7.1.1 La optimización del sistema de cotizaciones y proformas ha permitido realizar un seguimiento en tiempo real de los proyectos, mejorando notablemente la eficiencia y la atención a los clientes. Ahora, los procesos son más rápidos y precisos, lo que ha reducido los tiempos de respuesta y ha aumentado la satisfacción del cliente al recibir cotizaciones más rápidamente.

7.1.2 El desarrollo de una aplicación con una interfaz intuitiva y sencilla ha facilitado el uso del sistema por parte del personal. Esta mejora ha permitido que los usuarios realicen cotizaciones de manera más eficiente y sin complicaciones, reduciendo errores y mejorando la experiencia tanto para los empleados como para los clientes.

7.1.3 La centralización de la gestión de datos ha transformado la manera en que el personal accede y actualiza la información. Con esta mejora, la consulta y actualización de datos se realiza de manera más eficiente, eliminando la dispersión de información y asegurando que todos los datos estén siempre actualizados y disponibles en un único sistema. Esto ha optimizado la coordinación entre departamentos, permitiendo una gestión más integral y precisa de los proyectos y fortaleciendo así la capacidad de respuesta de la empresa.

7.1.4 El análisis exhaustivo de los requerimientos y requisitos del sistema de cotizaciones y proformas para el seguimiento en tiempo real de proyectos fue fundamental para el éxito del proyecto. Durante esta fase, se identificaron y documentaron

minuciosamente las necesidades específicas del negocio y de los usuarios finales, lo que aseguró que el sistema desarrollado cumpliera con todas las expectativas y requerimientos.

CAPITULO VIII

8.1 Recomendaciones para la empresa respecto del Proyecto de Mejora.

- Establecer rutinas de mantenimiento periódico para asegurar el óptimo funcionamiento del sistema, incluyendo actualizaciones de software y hardware.
- Proporcionar capacitación continua al personal encargado del sistema para garantizar su correcto uso y maximizar su eficiencia.
- Realizar encuestas o evaluaciones periódicas para recopilar comentarios y sugerencias de los usuarios, lo que permitirá identificar áreas de mejora.
- Establecer un plan de respaldo de datos efectivo para garantizar la seguridad y disponibilidad de la información almacenada en el sistema.
- Mantener una comunicación efectiva con los empleados sobre el sistema y los logros alcanzados. Destacar los beneficios y logros obtenidos gracias a su uso.

8.2. Otras recomendaciones

- Continuar invirtiendo en tecnología y seguridad para mantenerse al día con las últimas tendencias y garantizar la protección de la información.
- Implementar un programa de seguimiento y evaluación constante para verificar la efectividad del sistema y realizar ajustes o mejoras según sea necesario.
- Continuar con la constante evolución y crecimiento del sistema, implementando nuevas funcionalidades según se necesario para la mejora continua.

REFERENCIAS BIBLIOGRAFICAS

- Autentia. (2024). SOFTWARE DESIGN Principios y patrones del. Obtenido de https://www.autentia.com/wp-content/uploads/libros/SoftwareDesign_PrincipiosyPatrones-Autentia.pdf
- autentia. (2024). Software Design Principios y patrones del desarrollo de software. 30-67. Obtenido de https://www.autentia.com/wp-content/uploads/libros/SoftwareDesign_PrincipiosyPatrones-Autentia.pdf
- Bahit, E. (2011). *El paradigma de la Programación Orientada a Objetos en PHP y el patrón de arquitectura de Software MVC*.
- Bertalanffy, L. v. (1968). *Teoría general de los sistemas: Fundamentos, desarrollo, aplicaciones*.
- Ble Jurado, C., & Beas, J. M. (2010). *ACADEMIA*. Obtenido de https://www.academia.edu/38401326/Diseno_Agil_Con_TDD
- Bolton, D. (2015). *The automation paradox: Why more automation leads to less employment*. Oxford University Press.
- ecma INTERNATIONAL. (2023). Obtenido de ECMA-262, 14th edition Language Specification: <https://262.ecma-international.org>
- FPDF Library. (2020). Obtenido de <http://www.fpdf.org>
- Gamma, E. (1994). *Design patterns: Elements of reusable object-oriented software*.
- García-Peñalvo, F. J., & García-Holgado, A. (2018). Metodologías de Ingeniería de Software. Grupo GRIAL. Obtenido de <http://repositorio.grial.eu/handle/grial/1149>

Guest. (3 de 11 de 2019). *rockcontent*. Obtenido de Diseño UI y UX: ¿descubre cuál es la diferencia entre ambos!: <https://rockcontent.com/es/blog/ui-ux/>

IKUSI velatia. (s.f.). Obtenido de <https://www.ikusi.com/mx/blog/tecnologias-de-la-informacion-en-las-empresas-como-impacta-el-negocio/>

IONOS. (2023). *Digital Guide IONOS*. Obtenido de <https://www.ionos.com/es-us/digitalguide/servidores/know-how/que-es-un-servidor-un-concepto-dos-definiciones/>

jQuery foundation. (2020). Obtenido de <https://contribute.jquery.org/web-sites/>

KENDALL, K. E. (2011). *ANÁLISIS Y DISEÑO*. PEARSON EDUCACIÓN.

KSchool. (2024). *Librerías en programación: ¿qué son y para qué sirven?* Obtenido de <https://kschool.com/blog/programacion/librerias-en-programacion-que-son-y-para-que-sirven/>

Llorens Espada, J. (2022). *Aplicaciones informáticas (app) para el registro diario de la jornada laboral. Condiciones de licitud*. Universidad Pública de Navarra, Navarra. Obtenido de <https://doi.org/10.20318/labos.2022.6853>

López Román, L. (2006). *Metodología de la programación orientada a objetos*. Editorial Alfaomega.

Normativa de Protección de Datos Personales. (s.f.). Obtenido de <https://www.gob.pe/institucion/anpd/colecciones/3482-normativa-de-proteccion-de-datos-personales>

Object Management Group. (2017). Obtenido de <https://www.omg.org/spec/UML/2.5.1/About-UML#spec-versions>

Oppel, A., & Sheldon, R. (2010). *FUNDAMENTOS DE SQL*. McGRAW-HILL

INTERAMERICANA EDITORES, S.A. DE C.V.

PHP. (2020). Obtenido de <https://www.php.net/archive/2020.php>

Piattini Velthuis, M. G. (2004). *Análisis y diseño de aplicaciones informáticas de gestión : una perspectiva de ingeniería del software*. Madrid: RA-MA.

proyectosagiles.org. (s.f.). Obtenido de <https://proyectosagiles.org/que-es-scrum/>

Rios Rojas, E. A., & Sanchez Alvitez, C. G. (2022). *Desarrollo de un sistema web para optimizar el proceso de cotizaciones en Respira S.A.C*. Universidad Tecnológica del Peru, Peru. Obtenido de <https://hdl.handle.net/20.500.12867/7066>

Stallings, W. (2018). *Operating Systems*. Ninth Edition.

THE APACHE SOFTWARE FOUNDATION. (2020). Obtenido de <https://apache.org>

Tomala Torres, N. G. (2016). *DESARROLLO E IMPLEMENTACIÓN DE UN SISTEMA AUTOMATIZADO PARA EL PROCESO DE COTIZAR PRODUCTOS GRÁFICOS EN LA EMPRESA GRÁFICA GRAFIMPAC S.A*. Universidad de Guayaquil Facultad de Ciencias Matemáticas y Físicas Carrera de Ingeniería en Sistemas Computacionales, Guayaquil. Obtenido de https://biblioteca.semisud.org/opac_css/index.php?lvl=author_see&id=208466

Torres Parodi, J. C. (2013). *Sistema automatizado de gestión de precios y valorización de cotizaciones para una línea naviera*. Universidad Peruana de Ciencias Aplicadas (UPC), Peru. Obtenido de <https://repositorioacademico.upc.edu.pe/handle/10757/274134>

Ubuntu 20.04 LTS arrives. (2020). Obtenido de <https://canonical.com/blog/ubuntu-20-04-lts-arrives>

Van Horne, J. C., & Wachowicz, J. M. (2019). *Fundamentos de Administración Financiera (16ª Edición)*. Pearson Educación.

W3C. (2020). Obtenido de CSS Snapshot 2020: <https://www.w3.org/TR/css-2020/>

W3C. (2021). Obtenido de HTML 5.3: <https://www.w3.org/TR/2021/NOTE-html53-20210128/>

W3schools. (2020). Obtenido de Chart.js: https://www.w3schools.com/ai/ai_chartjs.asp

Zaldívar, M. P., & Meza, D. (2023). *Estudio de las frecuencias de cotizaciones en el Instituto de Previsión Social(IPS)*. Universidad Nacional de Asunción, Paraguay, Paraguay. Obtenido de <https://doi.org/10.56216/radee012023jun.a02>

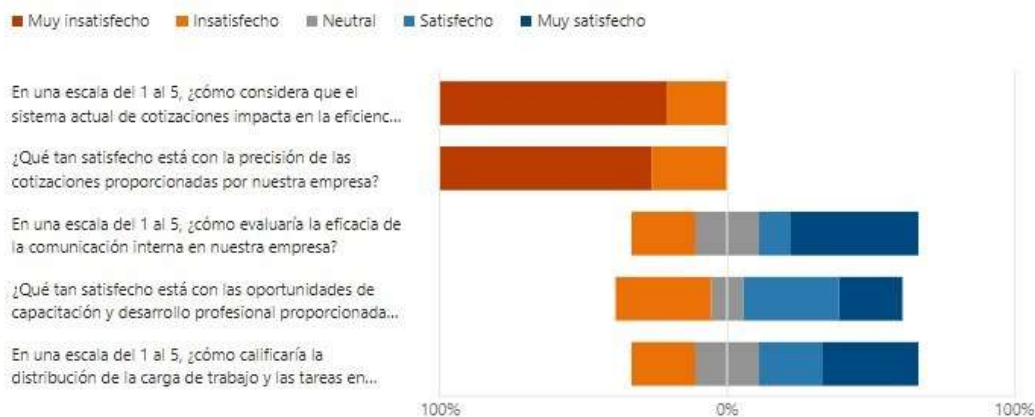
ZURB Foundation. (2018). Obtenido de <https://zurb.com/university/foundation-intro>

ANEXOS

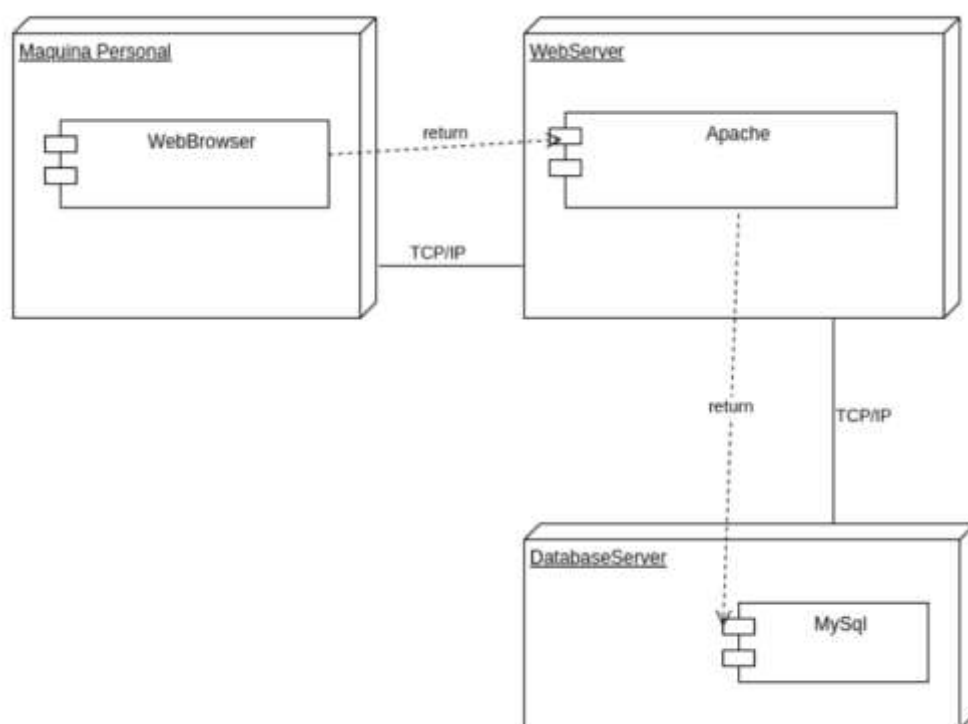
Anexo 1: Encuesta que se realizó a los empleados de KATARI A&C

1. "Queremos conocer tu opinión sobre la experiencia general con nuestros servicios. Por favor, indica en qué medida estás satisfecho con los siguientes aspectos: eficiencia en la prestación de servicios, rapidez en la atención, claridad de la información proporcionada y colaboración entre nuestro equipo. Utiliza una escala del 1 al 5, donde 1 representa un nivel bajo de satisfacción y 5 representa un nivel alto de satisfacción."

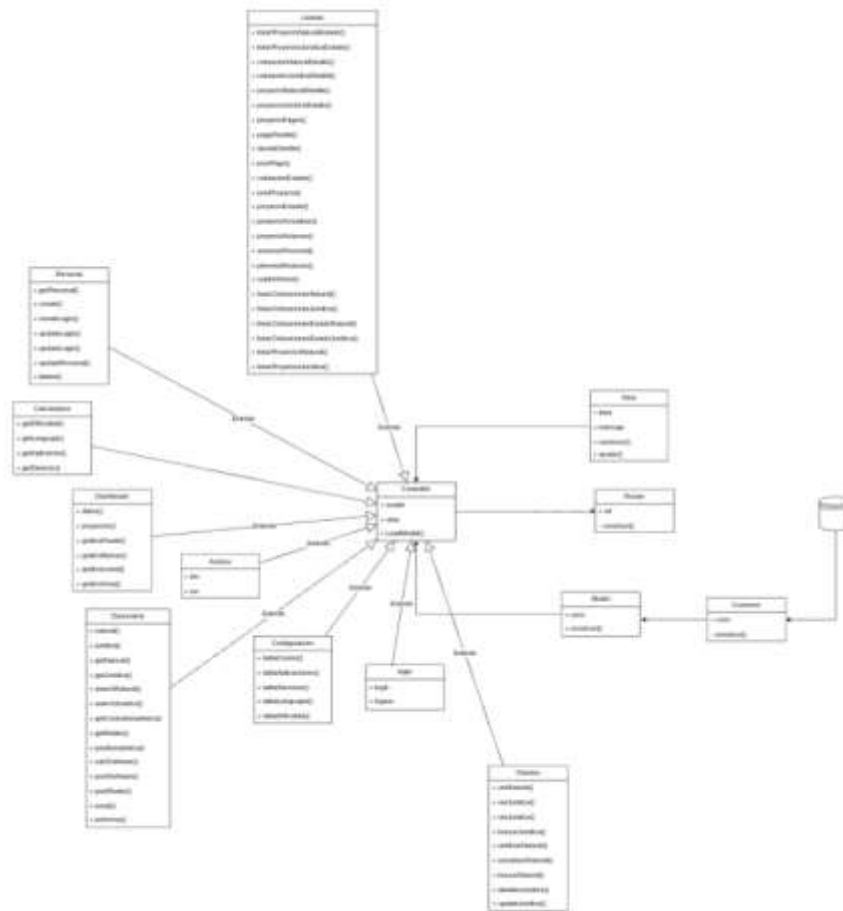
[Más detalles](#)



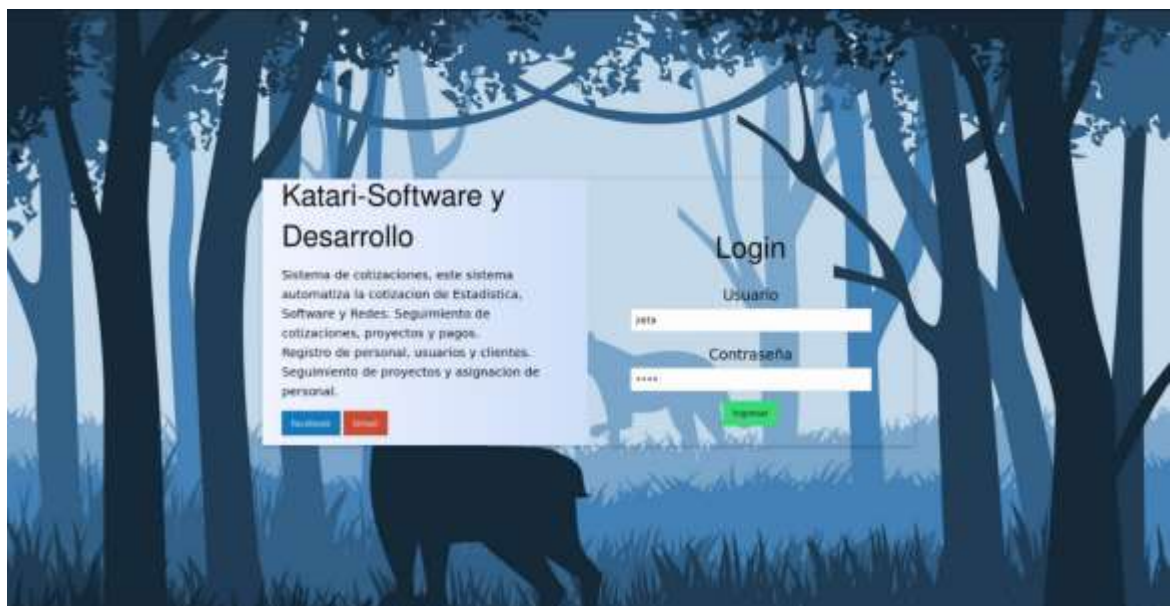
Anexo 2: Diagrama de Implementación



Anexo 3: Diagrama de Clases



Anexo 4: Interfaz del Login



Anexo 5: Interfaz del Dashboard



Anexo 6: Interfaz Cotizar

The 'Cotizar' interface for Personal Natural includes a sidebar with navigation links: Dashboard, Cotizar, Listado, Clientes, Personal, Registros Pagos, Calculadora, Configuración, and SALIR. The main content area features a form for 'Personal Natural' with fields for Nombre (ALEXANDER LUTASI CHAGUA), DNI (70088430), Teléfono (829956157), Email (alexelutasi@gmail.com), Dirección (puno), and Ciudad (puno). Below these are dropdown menus for Servicio (estadística) and Tipo (análisis estadístico), and input fields for Cantidad, Cantidad de la Cotización (90 / mlt / aaaa), and Descripción. A green 'Guardar' button is at the bottom. The footer indicates '© KATARI SUTABASE Copyright Katari Inc. Todos los derechos reservados'.

Anexo 7: Interfaz Listado Cotizaciones

KATARI

Bienvenido: Jersson Pelayo

Listado de Cotizaciones y Proyectos

Seleccione una lista: Cotizaciones

Cotizaciones registradas

Elija el tipo de Persona: Naturales

Elija el estado que busca: Todo

Nombre	Telefono	servicio	Precio	estado	Límite	acciones
Josue Lisset Chagas	834858345	estadística	200	proyecto	2024-07-04	Ver Eliminar
JERSSON PELAYO QUISPE AMAZA	908777712	estadística	300	espera	2024-07-04	Ver Eliminar
Grande Giovanni Cebal Jucapa	970417503	estadística	200	espera	2024-07-03	Ver Eliminar
KEVIN JOSUE GALLEROS CHOQUE	934472638	estadística	830	espera	2024-07-06	Ver Eliminar
YUDAN VICTOR QUISPE AMAZA	932432534	estadística	820	aceptado	2024-07-06	Ver Eliminar

© KATARI SOFTWARE Copyright Katari Inc. Todos los derechos reservados.

Anexo 8: Interfaz Detalles de Cotización

KATARI

Bienvenido: Jersson Pelayo

Detalles de Cotización

Servicio: estadística

Cantidad: 1

Estado: proyecto

Fecha Creación: 2024-07-04

Código aprobado: 150

Descripción: Análisis estadístico sobre la desnutrición en la región de PIURA

Personal Creador: Jersson Pelayo Quispe Ayala

Cambiar de estado:

[Eliminar Cotización](#) [Cotización Aceptada](#)

Datos del Cliente Solicitante

Nombre: Josue Lisset Chagas

Telefono: 834858345

Id: 73649612

Email: joserlissetchagas@gmail.com

Anexo 9: Interfaz Listado Proyectos

KATARI

Bienvenido: Jersson Pelayo

Listado de Cotizaciones y Proyectos

Seleccione una lista:
Proyectos

Proyectos registrados

Elija el tipo de Persona:
Naturales

Elija el estado que busca:
Todo

Nombre Proyecto	Usuario	servicio	estado	pendiente	Total	Fecha Entrega	acciones
Análisis de la rubricación en la región de puno	jessie Listari Chagas	estadística	proceso	180	200	2024-07-08	Verificar Actualizar Eliminar
perfil estadístico de unac	EDGAR ANZA CHOQUE	estadística	proceso	180	210	2024-06-28	Verificar Actualizar Eliminar
Tercer de pregrado	danylo apasol mariani	estadística	proceso	157	200	2024-09-02	Verificar Actualizar Eliminar

© KATARI SOFTWARE Copyright Katari, Inc. Todos los derechos reservados.

Anexo 10: Interfaz Clientes

KATARI

Bienvenido: Jersson Pelayo

Personas Naturales

Mostrar Personas Naturales | Mostrar Personas Jurídicas

Buscar

ID	nombre	dni	ciudad	status
1	Jessie	53425432	Arequipa	Completo
2	Jessie	73609812	Jal	Completo
3	Bryson	88046587	Puno	Completo
4	Elizabeth Guevara	34523453	Arequipa	Completo
5	Danylo Juani	73245154	Puno	Completo

© KATARI SOFTWARE Copyright Katari, Inc. Todos los derechos reservados.

Anexo 11: Interfaz Personal

Personal

Foto	Nombres	Telefono	Email	Acciones
	Jersson Pelayo Quipe Apaza	988777712	jersson.p012@gmail.com	Detalles Login Eliminar
	Johan Victor Quipe Apaza	988717321	johanvic@gmail.com	Detalles Login Eliminar
	Josue Roselli Llatasi Chagas	238764237	jroseg@gmail.com	Detalles Login Eliminar
	Elmer Gabriel Quipe Rosca	676534952	elmer@gmail.com	Detalles Login Eliminar
	Kevin Josue Sotegues Chague	456534754	kavim@gmail.com	Detalles Login Eliminar

© KATARI SOFTWARE Copyright Katari, Inc. Todos los derechos reservados.

Anexo 12: Interfaz Pagos

REGISTRO DE PAGOS POR PROYECTO

Tabla de Pagos

N°	Proyecto	Cliente	Estado	Creacion	Deuda	Total	telefono	Detalles
1	Análisis de la nutrición en la región de puno	Josue Llatasi Chagas	proceso	2024-07-03 09:33:32	130	200	836856245	Detalles
2	análisis estadístico de agua	EDGAR ARIZA CHOQUE	proceso	2024-08-28 10:40:03	180	250	935013466	Detalles
3	Tarea de pregrado	Weyen arpaill maramba	proceso	2024-05-30 21:45:12	157	280	348543875	Detalles

© KATARI SOFTWARE Copyright Katari, Inc. Todos los derechos reservados.

Anexo 13: Interfaz Calculadora

Calculadora de Costos De Software

Nombre Proyecto :

Descripción :

Dificultad : 1 Sprints :

Lenguaje : java Tipo Aplicación : escritorio

Tipo Servicio : software institucional Duración Semanas :

Costo Por Mantenimiento : T. Mantenimiento Meses :

Con Servidor : ☐ Sin Servidor : ☐

© KATARI SOFTWARE Copyright Katari Inc. Todos los derechos reservados.

Anexo 14: Interfaz Configuración

Costos

Servicio	Descripción	Precio	Acciones
estadística	análisis estadístico	550	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
estadística	tesis pregrado	250	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
estadística	tesis maestría	450	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Tipos De Aplicaciones

Aplicación	Descripción	Precio	Acciones
escritorio	aplicación de escritorio	3000	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
módulo	web	2000	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
servicios		2900	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Tipo De Servicios

Código	Servicio	Acciones
100	estadística	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
200	software	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
300	redes	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Lenguaje de Programación

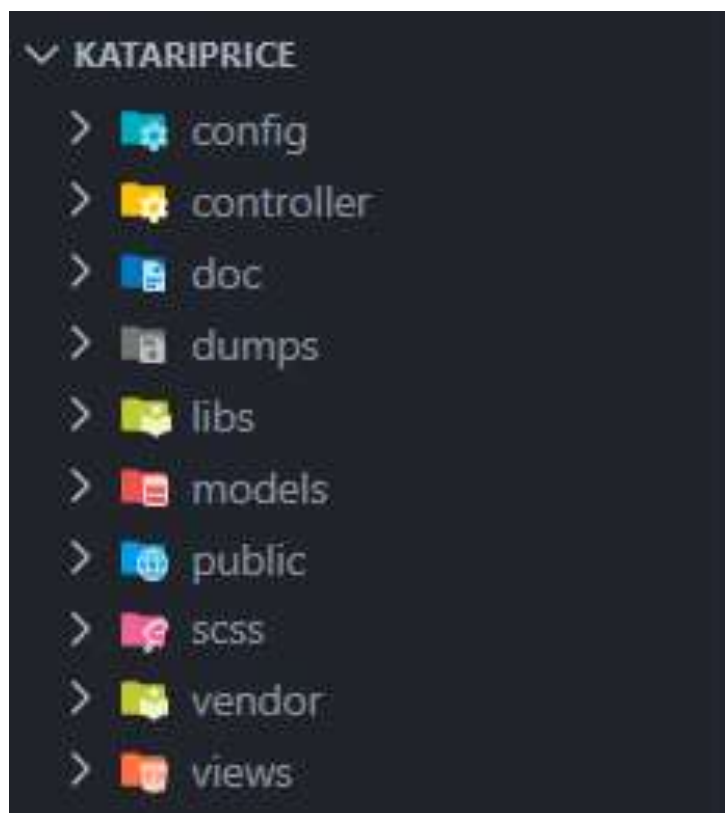
Lenguaje	Precio	Acciones
java	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
c#	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
c++	0	<input type="button" value="Editar"/> <input type="button" value="Eliminar"/>

Dificultad en %

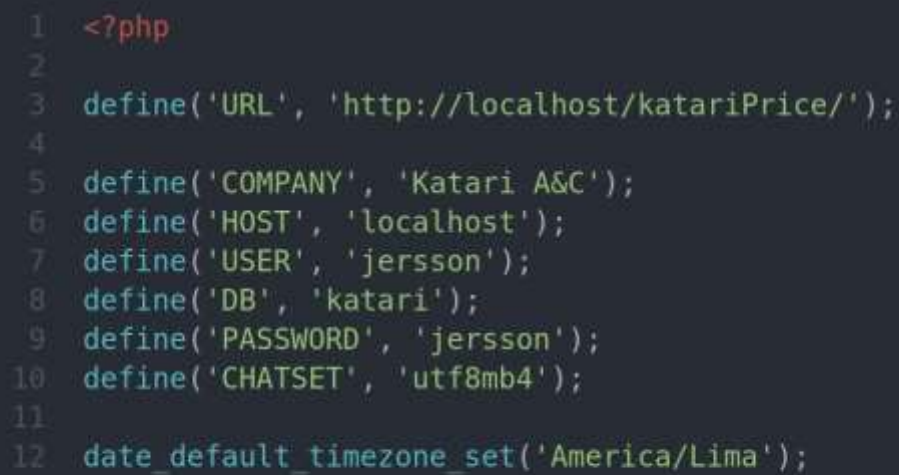
Anexo 15: Cotización a Imprimir



Anexo 16: Arquitectura del sistema de cotizaciones automatizado



Anexo 17: Configuración del sistema de cotizaciones automatizado



```
1  <?php
2
3  define('URL', 'http://localhost/katariPrice/');
4
5  define('COMPANY', 'Katari A&C');
6  define('HOST', 'localhost');
7  define('USER', 'jersson');
8  define('DB', 'katari');
9  define('PASSWORD', 'jersson');
10 define('CHATSET', 'utf8mb4');
11
12 date_default_timezone_set('America/Lima');
```

Anexo 18: Archivo App, router del sistema

```

1  <?php
2  session_start();
3
4  class App
5  {
6      function __construct()
7      {
8          $url = $_GET['url'];
9          $url = rtrim($url, '/');
10         $url = explode('/', $url);
11
12         if ($SESSION['katarl']) {
13             if (empty($url[0])) {
14
15                 $archivoController = "controller/dashboard.php";
16                 require_once $archivoController;
17                 $controller = new Dashboard();
18                 $controller->loadModel('dashboard');
19                 $controller->render();
20                 return false;
21             }
22             #echo "con parametros";
23             $archivoController = "controller/" . $url[0] . ".php";
24             if (file_exists($archivoController)) {
25                 #echo "si existe";
26                 require_once $archivoController;
27                 $controller = new $url[0];
28                 $controller->loadModel($url[0]);
29
30                 # numero de elementos del arreglo
31                 $nparam = sizeof($url);
32
33                 if ($nparam == 1) {
34                     if ($nparam > 2) {
35                         $nparam = [];
36                         for ($i = 2; $i < $nparam; $i++) {
37                             array_push($nparam, $url[$i]);
38                         }
39                         $controller->{$url[1]}($nparam);
40                     } else {
41                         $controller->{$url[1]}();
42                     }
43                 } else {
44                     $controller->render();
45                 }
46             } else {
47                 #echo "no existe";
48                 require "controller/error.php";
49                 $controller = new ErrorGeneral();
50                 $controller->render();
51             }
52         } else {
53             if (empty($url[0])) {
54                 #echo "URL vacia";
55                 $archivoController = "controller/login.php";
56                 require_once $archivoController;
57                 $controller = new Login;
58                 $controller->loadModel('main');
59                 $controller->render();
60                 return false;
61             }
62             #echo "segundo: " . $url[1];
63             if (isset($url[1]) && $url[1] == "login") {
64                 $archivoController = "controller/login.php";
65                 require_once $archivoController;
66                 $controller = new Login;
67                 $controller->loadModel('login');
68                 $controller->login();
69                 $controller->render();
70             } else {
71                 #echo "nada";
72                 header("location: " . constant('URL'));
73             }
74         }
75     }
76 }
77 }
78

```


Anexo 19: Controller Padre



```
1  <?php
2  class Controller
3  {
4      public $view;
5      public $model;
6
7      function __construct()
8      {
9          #echo "<h1>Controlador Base</h1>";
10         $this->view = new View();
11     }
12
13     function loadModel($model)
14     {
15         $url = 'models/' . $model . "model.php";
16
17         if (file_exists($url)) {
18
19             require $url;
20
21             $modelName = $model . 'Model';
22             $this->model = new $modelName();
23         }
24     }
25
```

Anexo 20: Model Padre

```
1  <?php
2
3  class Model
4  {
5      public $conn;
6
7      function __construct()
8      {
9          #echo "<h1>Model Base</h1>";
10         $this->conn = new Conexion();
11     }
12 }
13
```

Anexo 21: Vista Padre

```
1  <?php
2
3  class View
4  {
5      public $mensaje;
6      public $data;
7      public $data2;
8
9      function __construct()
10     {
11         #echo "<h1>View Base</h1>";
12     }
13
14     function Render($nombre)
15     {
16         require 'views/' . $nombre . '.php';
17     }
18 }
19
```

Anexo 22: Controlador Acciones

```

1  <?php
2  class Actions extends Controller
3  {
4      function __construct()
5      {
6          parent::__construct();
7      }
8      function dni()
9      {
10         // Datos
11         $token = 'apis-token-8574.bPsef4wHOYjVwA7bFoDMZqLLrNrAMkiY';
12         $dni = $_POST["dni"];
13
14         // Iniciar llamada a API
15         $curl = curl_init();
16
17         // Buscar dni
18         curl_setopt_array(
19             $curl,
20             array(
21                 CURLOPT_URL => 'https://api.apis.net.pe/v2/reniec/dni?numero=' . $dni,
22                 CURLOPT_SSL_VERIFYPEER => 0,
23                 CURLOPT_RETURNTRANSFER => true,
24                 CURLOPT_SSL_VERIFYPEER => 0,
25                 CURLOPT_ENCODING => '',
26                 CURLOPT_MAXREDIRS => 2,
27                 CURLOPT_TIMEOUT => 0,
28                 CURLOPT_FOLLOWLOCATION => true,
29                 CURLOPT_CUSTOMREQUEST => 'GET',
30                 CURLOPT_HTTPHEADER => array(
31                     'Referer: https://apis.net.pe/consulta-dni-api',
32                     'Authorization: Bearer ' . $token
33                 ),
34             )
35         );
36
37         $response = curl_exec($curl);
38         if (curl_errno($curl)) {
39             echo 'error del scraper: ' . curl_error($curl);
40             exit;
41         }
42
43         curl_close($curl);
44         // Datos listos para usar
45         echo $response;
46
47     }
48
49     function ruc()
50     {
51         $ruc = $_POST["ruc"];
52         $consulta = file_get_contents('https://api.apis.net.pe/v1/ruc?numero=' . $ruc . '');
53         echo $consulta;
54     }
55 }
56

```

Anexo 23: Controlador Calculadora

```

1  <?php
2
3  class Calculadora extends Controller
4  {
5      function __construct()
6      {
7          parent::__construct();
8      }
9      function render()
10     {
11         $this->view->Render('calculadora/index');
12     }
13     public function getDificultad(){
14         $data = $this->model->GetDificultad();
15         $json = array();
16         while($row = mysqli_fetch_assoc($data)){
17             $json[] = array(
18                 "iddificultad"=> $row['iddificultad'],
19                 "factor"=> $row['factor'],
20             );
21         }
22         echo json_encode($json);
23     }
24     public function getlenguaje(){
25         $data = $this->model->Getlenguaje();
26         $json = array();
27         while($row = mysqli_fetch_assoc($data)){
28             $json[] = array(
29                 "idlenguaje"=> $row['idlenguaje'],
30                 "lenguaje"=> $row['lenguaje'],
31                 "precio"=> $row['precio'],
32             );
33         }
34         echo json_encode($json);
35     }
36     public function getAplicacion(){
37         $data = $this->model->GetAplicacion();
38         $json = array();
39         while($row = mysqli_fetch_assoc($data)){
40             $json[] = array(
41                 "idaplicacion"=> $row['idaplicacion'],
42                 "aplicacion"=> $row['aplicacion'],
43                 "precio"=> $row['precio'],
44             );
45         }
46         echo json_encode($json);
47     }
48     public function getServicio(){
49         $data = $this->model->GetServicio();
50         $json = array();
51         while($row = mysqli_fetch_assoc($data)){
52             $json[] = array(
53                 "idcosto"=> $row['idcosto'],
54                 "idservicio"=> $row['idservicio'],
55                 "descripcion"=> $row['descripcion'],
56                 "precio"=> $row['precio'],
57             );
58         }
59         echo json_encode($json);
60     }
61 }

```

Anexo 24: Controlador Clientes

```

controller > clientes.php > PHP Intelephense > Clientes > __construct
1  <?php
2
3  0 references | 0 implementations
4  class Clientes extends Controller
5  {
6      11 references | 0 overrides | prototype
7      function __construct()
8      {
9          parent::__construct();
10     }
11     0 references | 0 overrides
12     function render()
13     {
14         $this->view->Render('clientes/index');
15     }
16     // muestra a la perosa natural
17     0 references | 0 overrides
18     public function verNatural(){ ...
19     }
20     // muestra a la perosa juridica
21     0 references | 0 overrides
22     public function verJuridica(){ ...
23     }
24     // busaca a la persona juridica
25     0 references | 0 overrides
26     public function buscarJuridica(){ ...
27     }
28     0 references | 0 overrides
29     public function verificarNatural(){ ...
30     }
31     0 references | 0 overrides
32     public function actualizarNatural(){ ...
33     }
34     // busca a la persona natural
35     0 references | 0 overrides
36     public function buscarNatural() { ...
37     }
38     //JURIDICA
39     0 references | 0 overrides
40     function detallesJuridica($nparam = null) ...
41     {
42     }
43     0 references | 0 overrides
44     public function updateJuridica(){ ...
45     }
46
47 }

```

Anexo 25: Controlador Configuración

```

controller > configuration.php > ...
1  <?php
2
   0 references | 0 implementations
3  class Configuracion extends Controller
4  {
   11 references | 0 overrides | prototype
5      function __construct()
6      {
7          parent::__construct();
8      }
   0 references | 0 overrides
9      function render()
10     {
11         $this->view->Render('configuracion/index');
12     }
   0 references | 0 overrides
13 > public function tablaCostos(){ ...
25     }
   0 references | 0 overrides
26 > public function tablaAplicaciones(){ ...
38     }
   0 references | 0 overrides
39 > public function tablaServicios(){ ...
49     }

50 > public function tablaLenguajes(){ ...
61     }

62 > public function tablaDificultad(){ ...
74     }
75 }
76 >>

```

Anexo 26: Controlador Cotización

```

controller > cotizacion.php > PHP > Cotizacion
1  <?php
2  use PHPMailer\PHPMailer\PHPMailer;
3  use PHPMailer\PHPMailer\Exception;
4
5  // Cargar Composer's autoloader
6  //require '../vendor/autoload.php';
7  require dirname(__DIR__) . '/vendor/autoload.php';
8  0 references | 0 implementations
9  class Cotizacion extends Controller
10 {
11
12     11 references | 0 overrides | prototype
13     public function __construct() ...
14     }
15
16     0 references | 0 overrides
17     public function render() ...
18     }
19
20     // INSERTAR PERSONA NATURAL
21     0 references | 0 overrides
22     public function natural() ...
23     }
24
25     // INSERTAR PERSONA JURIDICA
26     0 references | 0 overrides
27     public function juridica() ...
28     }
29
30     // OBTENER PERSONA NATURAL
31     0 references | 0 overrides
32     public function getNatural() ...
33     }
34
35     // OBTENER PERSONA JURIDICA
36     0 references | 0 overrides
37     public function getJuridica() ...
38     }
39
40     // BUSCADORES
41     0 references | 0 overrides
42     public function searchNatural() ...
43     }
44
45     0 references | 0 overrides
46     public function searchJuridica() ...
47     }
48
49     0 references | 0 overrides
50     public function getCostosEstadistica() ...
51     }
52
53     // OBTENER REDES
54     0 references | 0 overrides
55     public function getRedes() ...
56     }
57
58     // POST ESTADISTICA
59     0 references | 0 overrides
60     public function postEstadistica() ...
61     }
62
63     // INSERTAR CALCULADOR DE SOFTWARE
64     0 references | 0 overrides
65     public function calcSoftware() ...
66     }
67
68     // POST SOFTWARE
69     0 references | 0 overrides
70     public function postSoftware() ...
71     }
72
73     // POST REDES
74     0 references | 0 overrides
75     public function postRedes() ...
76     }
77
78     6 references | 0 overrides
79     public function email($email,$id=null) ...
80     }
81
82     6 references | 0 overrides
83     public function proforma($nparam=null,$save=null) ...
84     }
85
86     }
87

```

Anexo 27: Controlador Listado

```

controller > listado.php > PHP > Listado
1  <?php
   0 references | 0 implementations
2  class Listado extends Controller
3
   11 references | 0 overrides | prototype
4  function __construct()
5  {
6      parent::__construct();
7  }
   0 references | 0 overrides
8  > function render() ...
11 }
   2 references | 0 overrides
12 > function cotizacion() ...
15 }
   2 references | 0 overrides
16 > function proyecto() ...
19 }
   1 reference | 0 overrides
20 > function pagos() ...
23 }
   1 reference | 0 overrides
24 > function avances() ...
27 }
28 // LISTAR COTIZACIONES *****
   0 references | 0 overrides
29 > public function listarCotizacionesNatural() ...
45 }
   0 references | 0 overrides
46 > public function listarCotizacionesJuridica() ...
63 }
   0 references | 0 overrides
64 > public function listarCotizacionesEstadoNatural() ...
81 }
   0 references | 0 overrides
82 > public function listarCotizacionesEstadoJuridica() ...
99 }
100 // LISTAR COTIZACIONES *END*****
101 // LISTAR PROYECTOS *****
   0 references | 0 overrides
102 > public function listarProyectosNatural() ...
119 }
   0 references | 0 overrides
120 > public function listarProyectosJuridica() ...
137 }
   0 references | 0 overrides
138 > public function listarProyectosNaturalEstado() ...
156 }
   0 references | 0 overrides
157 > public function listarProyectosJuridicaEstado() ...
175 }
176 // LISTAR PROYECTOS END *****
177 // DETALLES COTIZACIONES -----
   0 references | 0 overrides
178 > public function cotizacionNaturalDetalle($nparam = null) ...
186 }
   0 references | 0 overrides
187 > public function cotizacionJuridicaDetalle($nparam = null) ...
195 }
196 // DETALLES COTIZACIONES END-----
197 // DETALLES COTIZACIONES -----
   0 references | 0 overrides
198 > public function proyectoNaturalDetalle($nparam = null) ...
204 }

```


Anexo 28: Controlador Login

```

1  <?php
2
3  class Login extends Controller
4  {
5
6      function __construct()
7      {
8          parent::__construct();
9      }
10
11     function render()
12     {
13         $this->view->Render('login/index');
14     }
15     public function login()
16     {
17         $user = trim(strtolower($_POST['usuario']));
18         $pass = trim(strtolower($_POST['password']));
19
20         $validar = $this->model->Validar($user,$pass);
21
22         if($validar['usuario'] == $user && $validar['password'] == $pass){
23             if($validar['estado'] == 0){
24                 $this->view->mensaje = 'Este usuario no se encuentra activo';
25                 header('location:'.constant('URL'));
26                 return;
27             }
28             $_SESSION['katari'] = 'katariSoftware';
29             $_SESSION['username'] = $validar['nombre'];
30             $_SESSION['nivel'] = $validar['nivel'];
31             $_SESSION['idpersonal'] = $validar['idpersonal'];
32             $this->view->mensaje = 'Credenciales correctas';
33             header('location:'.constant('URL').'.dashboard');
34         }else {
35             $this->view->mensaje = 'Credenciales incorrectas';
36             header('location:'.constant('URL'));
37         }
38     }
39     public function logout()
40     {
41         session_destroy();
42         $_SESSION['katari'] = '';
43         $_SESSION['username'] = '';
44         $_SESSION['nivel'] = '';
45         $_SESSION['idpersonal'] = '';
46         // $this->model->alterTable();
47         $this->view->Render('login/index');
48     }
49 }

```

Anexo 29: Controlador Pagos

```

1  <?php
2
3  class Pagos extends Controller
4  {
5
6      function __construct()
7      {
8          parent::__construct();
9      }
10
11     function render()
12     {
13         $this->view->render('pagos/index');
14     }
15     public function pagos(){
16         $data = $this->model->pagos();
17         $json = array();
18         while ($row = mysqli_fetch_assoc($data)) {
19             $json[] = array(
20                 "idproyecto" => $row['idproyecto'],
21                 "proyecto" => $row['nomproyecto'],
22                 "cliente" => $row['nomres'],
23                 "estado" => $row['estado'],
24                 "deuda" => $row['saldo'],
25                 "total" => $row['total'],
26                 "fecha" => $row['fecha'],
27                 "telefono" => $row['telefono'],
28             );
29         }
30         echo json_encode($json);
31     }
32 }

```

Anexo 30: Controlador Personal

```

Controlador > de personal.php > PHP Integración > de Personal > create
1  <?php
2
3  // Referencia 1 > Implementación
4  class Personal extends Controller
5  {
6      // Referencia 1 > Implementación
7      function __construct()
8      {
9          parent::__construct();
10     }
11
12     // Referencia 1 > Implementación
13     function render()
14     {
15         $this->view->render('personal/index');
16     }
17
18     // Referencia 1 > Implementación
19     function detalles($id = null)
20     {
21         $id = $id ?? 0;
22         $this->view->data = $this->model->getPersonalId($id);
23         $this->view->render('personal/detalles');
24     }
25
26     // Referencia 1 > Implementación
27     function login($usuario = null) {
28     }
29
30     // Referencia 1 > Implementación
31     function nuevo() {
32     }
33
34     // Referencia 1 > Implementación
35     public function getPersonal() {
36     }
37
38     // Referencia 1 > Implementación
39     public function create() {
40     }
41
42     // Referencia 1 > Implementación
43     public function createingie() {
44     }
45
46     // Referencia 1 > Implementación
47     public function updateingie() {
48     }
49
50     // Referencia 1 > Implementación
51     public function updatePersonal() {
52     }
53
54     // Referencia 1 > Implementación
55     public function delete() {
56     }
57 }

```

Anexo 31: Modelo Calculadora

```

models > calculadoramodel.php > PHP Intelephense > CalculadoraModel > GetAplicacion
1  <?php
   0 references | 0 implementations
2  class CalculadoraModel extends Model{
3
   10 references | 0 overrides | prototype
4      function __construct()
5      {
6          parent::__construct();
7      }
   0 references | 0 overrides
8      public function GetDificultad(){
9          $sql = "SELECT * FROM katari.dificultad;";
10         $res = $this->conn->ConsultaCon($sql);
11         return $res;
12     }
   0 references | 0 overrides
13     public function GetLenguaje(){
14         $sql = "SELECT * FROM katari.lenguajes;";
15         $res = $this->conn->ConsultaCon($sql);
16         return $res;
17     }
   0 references | 0 overrides
18     public function GetAplicacion(){
19         $sql = "SELECT * FROM katari.tipo_aplicacion;";
20         $res = $this->conn->ConsultaCon($sql);
21         return $res;
22     }
   0 references | 0 overrides
23     public function GetServicio(){
24         $sql = "SELECT * FROM costos WHERE idservicio = 200;";
25         $res = $this->conn->ConsultaCon($sql);
26         return $res;
27     }
28
29 }
30
31 >>

```

Anexo 32: Modelo Clientes

```

models > clientesmodel.php > ...
1  <?php
   0 references | 0 implementations
2  class ClientesModel extends Model{
3
   10 references | 0 overrides | prototype
4      function __construct()
5      {
6          parent::__construct();
7      }
   0 references | 0 overrides
8  > public function verNatural(){...
12     }
   0 references | 0 overrides
13 > public function verJuridica(){...
17     }
   0 references | 0 overrides
18 > public function BuscarJuridica($nombres,$dni){...
22     }
   0 references | 0 overrides
23 > public function BuscarNatural($query){...
27     }
28
   0 references | 0 overrides
29 > public function verificarNatural($idnatural){...
33     }
34
   0 references | 0 overrides
35 > public function actualizarNatural($idnatural, $nombre, $apellidos, $dni, $sexo, $ciudad, $telefono,
39     $email, $direccion, $feCreate, $feUpdate){...
40     }
   0 references | 0 overrides
40 > public function GetJuridicaId($id){...
44     }
   0 references | 0 overrides
45 > public function UpdateJuridica($razonsocial,$telefono,$email,$web,$ruc,$rubro,$direccion,$feUpdate,
49     $idjuridica){...
50     }
51 }
52 |
53
54 ?>

```

Anexo 33: Modelo Configuración

```

model > configureModelPhp > PHP Integrease > M ConfiguraciónModel
1 <php>
2 <reference> 0 <implementation>
3 class ConfiguraciónModel extends Model{
4
5     <reference> 0 <override> <prototype>
6     function __construct()
7     {
8         parent::__construct();
9     }
10
11     <reference> 0 <override>
12     public function TablaCostos(){
13         $sql = "SELECT costos.*,tipo_servicio.* FROM costos JOIN tipo_servicio ON costos.idservicio =
14             tipo_servicio.idservicio";
15         $data = $this->conn->ConsultaCnn($sql);
16         return $data;
17     }
18
19     <reference> 0 <override>
20     public function TablaAplicaciones(){
21         $sql = "SELECT * FROM tipo_aplicacion";
22         $data = $this->conn->ConsultaCnn($sql);
23         return $data;
24     }
25
26     <reference> 0 <override>
27     public function TablaServicios(){
28         $sql = "SELECT * FROM tipo_servicio";
29         $data = $this->conn->ConsultaCnn($sql);
30         return $data;
31     }
32
33     <reference> 0 <override>
34     public function TablaLenguajes(){
35         $sql = "SELECT * FROM lenguajes";
36         $data = $this->conn->ConsultaCnn($sql);
37         return $data;
38     }
39
40     <reference> 0 <override>
41     public function TablaDificultad(){
42         $sql = "SELECT * FROM dificultad";
43         $data = $this->conn->ConsultaCnn($sql);
44         return $data;
45     }
46 }

```

Anexo 34: Modelo Cotización

[illegible]

Anexo 35: Modelo Dashboard

```

models > dashboardmodel.php > ...
1  <?php
    0 references | 0 implementations
2  class DashboardModel extends Model{
3
4      10 references | 0 overrides | prototype
5      function __construct()
6      {
7          parent::__construct();
8      }
9      0 references | 0 overrides
10 > function TotalCotizaciones(){ ...
12     }
13     0 references | 0 overrides
14 > function TotalProyectos(){ ...
17     }
18     0 references | 0 overrides
19 > function TotalRecaudado(){ ...
22     }
23     0 references | 0 overrides
24 > public function Proyectos($id){ ...
27     }
28     0 references | 0 overrides
29 > public function GraficoPastel(){ ...
32     }
33     0 references | 0 overrides
34 > public function GraficoBarras(){ ...
37     }
38     0 references | 0 overrides
39 > public function GraficoLineal(){ ...
42     }
43     0 references | 0 overrides
44 > public function GraficoArea(){ ...
47     }
48
49
50 }
51
52

```

Anexo 36: Modelo Listado

```

modelo > @ listadomodel.php > _
1 //php
2 // referencia | 0 comentarios
3 class ListadoModel extends Model
4 {
5     // referencia | 0 comentarios | prototype
6     function __construct()
7     {
8         parent::__construct();
9     }
10    // LISTAS COTIZACIONES
11    // referencia | 0 comentarios
12    public function listarCotizacionesNatural()
13    {
14    }
15    // referencia | 0 comentarios
16    public function listarCotizacionesJuridica()
17    {
18    }
19    // referencia | 0 comentarios
20    public function listarCotizacionesEstadoNatural($estado)
21    {
22    }
23    // referencia | 0 comentarios
24    public function listarCotizacionesEstadoJuridica($estado)
25    {
26    }
27    // LISTAS COTIZACIONES END
28    // LISTAS PROYECTOS
29    // referencia | 0 comentarios
30    public function listarProyectosNatural()
31    {
32    }
33    // referencia | 0 comentarios
34    public function listarProyectosJuridica()
35    {
36    }
37    // referencia | 0 comentarios
38    public function listarProyectosNaturalEstado($estado)
39    {
40    }
41    // referencia | 0 comentarios
42    public function listarProyectosJuridicaEstado($estado)
43    {
44    }
45    // LISTAS PROYECTOS END
46    // LISTAS COTIZACIONES DETALLES -----
47    // referencia | 0 comentarios
48    public function cotizacionNaturalDetalle($id)
49    {
50    }
51    // referencia | 0 comentarios
52    public function cotizacionJuridicaDetalle($id)
53    {
54    }
55    // referencia | 0 comentarios
56    public function verificarCotizacion($id)
57    {
58    }
59    // LISTAS COTIZACIONES DETALLES END
60    // LISTAS PROYECTOS DETALLES -----
61    // referencia | 0 comentarios
62    public function proyectoNaturalDetalle($id)
63    {
64    }
65    // referencia | 0 comentarios
66    public function proyectoJuridicaDetalle($id)
67    {
68    }
69    // LISTAS PROYECTOS DETALLES END
70    // CAMBIAR ESTADO DE COTIZACIONES *****
71    // referencia | 0 comentarios
72    public function cotizacionEstado($id, $estado)
73    {
74    }
75    // CAMBIAR ESTADO DE COTIZACIONES END *****
76    // PAGAR A PROYECTO UNA COTIZACION *****
77    // referencia | 0 comentarios
78    public function postProyectoNatural($proyecto, $id, $estado, $servicio, $actividades, $descripcion, $pendiente, $total, $fechaEntrega, $idpersonal, $identificacion, $monto, $saldo, $tipo)
79    {
80    }
81    // referencia | 0 comentarios
82    public function postProyectoJuridica($proyecto, $id, $estado, $servicio, $actividades, $descripcion, $pendiente, $total, $fechaEntrega, $idpersonal, $identificacion, $monto, $saldo, $tipo)
83    {
84    }
85    // PAGAR A PROYECTO UNA COTIZACION END *****
86    // CAMBIAR ESTADO DE PROYECTOS *****
87    // referencia | 0 comentarios
88    public function proyectoEstado($id, $estado)
89    {
90    }
91    // referencia | 0 comentarios
92    public function proyectoActualizar($id, $actividades, $fecha, $descripcion)
93    {
94    }
95    // CAMBIAR ESTADO DE PROYECTOS END *****
96    // *****PAGOS PROYECTO*****
97    // referencia | 0 comentarios
98    public function proyectoPagos($id)
99    {
100   }
101    // referencia | 0 comentarios
102    public function pagoDetalle($id)
103    {
104    }
105    // referencia | 0 comentarios
106    public function deudaDetalle($id)
107    {
108    }
109    // *****PAGOS PROYECTO END*****
110    // GUARDAR PAGO DETALLE
111    // referencia | 0 comentarios
112    public function postPago($idPago, $monto, $concepto)
113    {
114    }
115    // referencia | 0 comentarios
116    public function updatePago($idproyecto, $monto, $saldo)
117    {
118    }
119    // referencia | 0 comentarios
120    public function updateProyecto($idproyecto, $pendiente)
121    {
122    }
123    // esta funcion tambien se utiliza para proyectoAvances
124    // referencia | 0 comentarios
125    public function selectProyecto($idproyecto)
126    {
127    }
128    // referencia | 0 comentarios
129    public function selectPago($idproyecto)
130    {
131    }
132    // GUARDAR PAGO DETALLE END
133    // *****PROYECTO AVANCES*****
134    // referencia | 0 comentarios
135    public function postAvances($idproyecto, $idpersonal, $reportes, $porcentaje, $totalActividades)
136    {
137    }
138    // referencia | 0 comentarios
139    public function selectAvances($idproyecto)
140    {
141    }

```


Anexo 37: Modelo Login

```

1  <?php
2  class LoginModel extends Model{
3
4      function __construct()
5      {
6          parent::__construct();
7      }
8      public function validar($usuario,$password){
9          $sql = "SELECT l.*,p.* FROM login l
10             join personal p
11             on l.idpersonal=p.idpersonal
12             WHERE usuario = '$usuario' AND password = '$password'";
13          $res = $this->conn->ConsultaArray($sql);
14          return $res;
15      }
16
17  }
18  }
19
20
21
22  ?>

```

Anexo 38: Modelo Pagos

```

1  <?php
2  class PagosModel extends Model{
3
4      function __construct()
5      {
6          parent::__construct();
7      }
8      public function pagos(){
9          $sql = "SELECT pa.*, pro.idnatural, pro.idjuridica, pro.nomproyecto,pro.estado,pro.feCreate,
10             COALESCE(concat(n.nombre," ",n.apellidos), j.razonsocial) AS nombres,
11             COALESCE(n.telefono, j.telefono) AS telefono,
12             COALESCE(n.email, j.email) AS email
13             FROM pagos pa
14             JOIN proyectos pro ON pro.idproyecto = pa.idproyecto
15             LEFT JOIN pernatural n ON pro.idnatural = n.idnatural
16             LEFT JOIN perjuridica j ON pro.idjuridica = j.idjuridica
17             ORDER BY pro.feCreate DESC";
18          $data = $this->conn->ConsultaCon($sql);
19          return $data;
20      }
21  }
22
23
24
25  ?>

```


Anexo 39: Modelo Personal

```

models > personalmodel.php >...
1  class PersonalModel extends Model
2      public function getPersonal()
3      {
4      }
5      0 references | 0 overrides
6      public function GetPersonalId($id)
7      {
8          $sql = "SELECT * FROM personal WHERE idpersonal = '$id'";
9          $data = $this->conn->ConsultaArray($sql);
10         return $data;
11     }
12     0 references | 0 overrides
13     public function GetLoginId($id) {
14     }
15     0 references | 0 overrides
16     public function Create($nombre, $apellidos, $dni, $sexo, $telefono, $fechaNac, $email, $foto) {
17     }
18     0 references | 0 overrides
19     public function CreateLogin($idpersonal, $usuario, $password, $estado) {
20     }
21     0 references | 0 overrides
22     public function UpdatePersonal($idpersonal, $nombre, $apellidos, $dni, $telefono, $email, $foto, $fecha) {
23     }
24     0 references | 0 overrides
25     public function UpdateLogin($idpersonal, $usuario, $password, $estado) {
26     }
27     0 references | 0 overrides
28     public function Delete($id) {
29     }
30     }
31     public function DeleteLogin($id) {
32     }
33     }
34     }
35     }
36     }
37     }
38     }
39     }
40     }
41     }
42     }
43     }
44     }
45     }
46     }
47     }
48     }
49     }
50     }
51     }
52     }
53     }
54     }
55     }
56     }
57     }
58     }
59     }
60     }
61     }
62     }
63     }
64     }
65     }
66     }
67     }
68     }
69     }
70     }
71     }
72     }
73     }
74     }
75     }
76     }
77     }
78     }
79     }
80     }
81     }
82     }
83     }
84     }
85     }
86     }
87     }
88     }
89     }
90     }
91     }
92     }
93     }
94     }
95     }
96     }
97     }
98     }
99     }
100    }

```

Anexo 40: Vista Calculadora

[illegible]

Anexo 41: Vista Clientes

[illegible]

Anexo 42: Vista Configuración

[illegible]

Anexo 43: Vista Cotización

[illegible]

Anexo 44: Vista Dashboard

[illegible]

Anexo 49: Script Calculadora

```

calculadora.js X
public > js > calculadora.js
1  $(document).ready(function () {
2      calcularSoftware();
3  });
4
5  > function getDificultad() { ...
22  }
23  > function getLenguaje() { ...
38  }
39  > function getAplicacion() { ...
54  }
55  > function getServicio() { ...
71  }
72
73  > function calcularSoftware() { ...

```

Anexo 50: Script Clientes

```

clientes.js X
public > js > clientes.js
1  $(document).ready(function () {
2      $("#natural-model").show();
3      $("#juridica-model").hide();
4      // $("#contenido-clientes").show();
5      postPersonal();
6      tablaNatural();
7      tablaJuridica();
8      // Manejador de clic para el botón "Natural"
9      $("#natural-boton").click(function () {
10         $("#juridica-model").hide();
11         $("#natural-model").show();
12     });
13
14     // Manejador de clic para el botón "Juridica"
15     $("#juridica-boton").click(function () {
16         $("#natural-model").hide();
17         $("#juridica-model").show();
18     });
19     // tabla natural inicia
20     > $(document).ready(function () {
21     });
22
23     > function tablaNatural() {
24     }
25
26     > function buscarNatural(query) {
27     }
28
29     > function getCurrentDate() {
30     }
31
32     // tabla natural fin
33
34     // tabla juridica
35     > function tablaJuridica() {
36     }
37
38     // Manejadores para los inputs de búsqueda
39
40     > $("#juridica-search-btn").click(function () {
41     });
42
43     > $("#natural-search-btn").click(function () {
44     });
45
46     > function postPersonal() {
47     }

```

Anexo 51: Script Configuración

```

1  $(document).ready(function () {
2      tablaCostos();
3      tablaAplicaciones();
4      tablaServicios();
5      tablaLenguajes();
6      tablaDificultad();
7  });
8  > function tablaCostos() { ...
31 }
32 > function tablaAplicaciones() { ...
55 }
56 > function tablaServicios() { ...
78 }
79 > function tablaLenguajes() { ...
101 }
102 > function tablaDificultad() { ...
124 }
125

```

Anexo 53: Script Dashboard

```

n dashboardjs x
public > js > m dashboardjs
1 1 $(document).ready(function () {
2     datos();
3     proyectos();
4     graficoPastel();
5     graficoBarras();
6     graficoLineal();
7     graficoArea();
8 });
9 // DATOS
10 > function datos() { ...
25 }
26 > function proyectos() { ...
48 }
49 // DATOS END
50
51 // ////////////////////////////////////GRAFICOS////////////////////////////////
52 // BARRAS END*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
53 > function graficoBarras() { ...
91 }
92 // BARRAS END*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
93 // PASTEL -----
94 > function graficoPastel() { ...
125 }
126
127 // PASTEL END-----
128 // LINEAS *****
129 > function graficoLineal() { ...
158 }
159
160
161 // LINEAS END*****
162 // AREA /*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*/*
163 > function graficoArea() { ...
196 }
197

```

Anexo 54: Script Listado

```

n listado x
public > js > m listado
1 // tablas cotizaciones y proyectos
2 $(document).ready() => {
3     //
4     // tablas cotizaciones y proyectos*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
5
6     // cotizaciones segun de NATURALES y JURIDICAS
7     $(document).ready() => {
8         //
9         // ESTADO DE LAS COTIZACIONES- SEGUN NATURALES y JURIDICAS
10     $(document).ready() => {
11         //
12         function getCotizacionesNaturales() {
13             //
14             function getCotizacionesJuridicas() {
15                 //
16                 // TABLA COTIZACIONES POR SU ESTADO, DE LOS NATURALES y JURIDICAS
17             }
18             function getCotizacionesNaturalesEstado(estados) {
19                 //
20             }
21             function getCotizacionesJuridicasEstado(estados) {
22                 //
23             }
24             // COTIZACIONES SEGUN DE NATURALES y JURIDICAS-----
25         }
26         // PROYECTOS -----
27         $(document).ready() => {
28             //
29             // ESTADO DE LOS PROYECTOS- SEGUN NATURALES y JURIDICAS
30         }
31         $(document).ready() => {
32             //
33             function getProyectosNaturales() {
34                 //
35             }
36             function getProyectosJuridicas() {
37                 //
38             }
39             function getProyectosNaturalesEstado(estados) {
40                 //
41             }
42             function getProyectosJuridicasEstado(estados) {
43                 //
44             }
45         }
46         // PROYECTOS END-----
47     }
48 }
49

```


Anexo 55: Script Avances

```

listadoAvances.js X
public > js > listadoAvances.js
1  $(document).ready(function () {
2      $("#modal-informe").hide();
3      personalAvances();
4      selectPersonal();
5      agregarPersonal();
6      subirInforme();
7  });
8  > function personalAvances() { ...
33  }
34
35 > function selectPersonal() { ...
56  }
57
58 > function subirInforme() { ...
84  }
85 > function agregarPersonal() { ...
105  }
106

```

Anexo 56: Script Pagos

```

pagos.js X
public > js > pagos.js
1  $(document).ready(function () {
2      pagoDetalles();
3      deudas();
4      $("#pago-modal").hide();
5      $("#registrar-pago").click(function () {
6          $("#pago-modal").show();
7      });
8      $("#guardar-pago").click(function () {
9          postPago();
10     });
11     $("#cancelar-pago").click(function () {
12         $("#pago-modal").hide();
13     });
14 });
15
16 // DETALLES DE PAGOS DE UN PROYECTO EN ESPECIFICO
17 > function pagoDetalles() { ...
44  }
45
46 > function deudas() { ...
49  }
50
51 // DETALLES DE PAGOS DE UN PROYECTO EN ESPECIFICO END
52
53 // *****GUARDAR PAGO*****
54 > function postPago() { ...
81  }
82
83 // *****GUARDAR PAGO-END*****
84
85 // *****VALIDA LOS INPUTS PARA QUE NO INTRODUZCA DATOS ERRONEOS*****
86 $(document).ready(function () {
87     //numberFloat(selector)
88     //justStrings(selector)
89     //numberLength(selector, maxLength)
90     numberFloat("#monto");
91 });
92 // *****VALIDA LOS INPUTS PARA QUE NO INTRODUZCA DATOS ERRONEOS*****
93

```

Anexo 57: Script Personal

```

personal.js
public > js > personal.js
1 > $(document).ready(function () {
2   $('#modal-login').hide();
3   tablaPersonal();
4   postPersonal();
5   postlogin();
6   updateLogin();
7   updatePersonal();
8   eliminar();
9 });
10
11 > function tablaPersonal() {--
12 }
13 // CREAR NUEVO PERSONAL-> Vista-> PERSONAL NUEVO
14 > function postPersonal() {--
15 }
16
17 > function postlogin() {--
18 }
19
20 > function updateLogin() {--
21 }
22 > function updatePersonal() {--
23 }
24
25 > function eliminar() {--
26 }
27
28 // *****OCULTA LOS INPUTS PARA QUE NO INTRODUCA DATOS E
29
30 > $(document).ready(function () {
31   //numberFloat(selector)
32   //justStrings(selector)
33   //numberLeght(selector, maxLength)
34   //justStrings("Nombre,Apellidos");
35   numberLeght("Edad", 6);
36   numberLeght("Telefono", 9);
37 });
38
39 // *****OCULTA LOS INPUTS PARA QUE NO INTRODUCA DATOS E
40
41

```

Anexo 58: Script Validaciones

```

validation.js
public > js > validation.js
1 > function numberFloat(selector) { ...
9 }
10 > function justStrings(selector) { ...
18 }
19 > function numberLeght(selector, maxLength) { ...
29 }
30
31 // OCULTA LOS MODALES
32 > $(document).ready(function () {
33   $('#modal-confirmacion-global, .modal-error-global').css({ 'display': 'none' });
34 })
35 // CONFIRMACION ,, , 1=EXITO, 0=ERROR
36 > function confirmation(type, mensaje, reload = 0) { ...
63 }
64
65
66 //*****
67 // NUMS = NUMERO DE FILAS
68 // IDTABLE = ID DE LA TABLA SIN EL #
69 // IDPAGINADOR = ID DEL PAGINADOR-> ES UN DIV VACIO CON ID
70 > function initPaginador(nums, idtable, idpaginador) { ...
83 }
84
85 //*****
86

```