



FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y SISTEMAS

**IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN DE
SERVICIO AL CLIENTE PARA LAS MIPYMES DEL SECTOR
METALMECÁNICO DEL PERÚ**

PRESENTADA POR
**CLAUDIO EMILIO ROJAS JARA
JOSE ANTONIO TAFUR UMERES**

TESIS

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

LIMA – PERÚ

2016



**Reconocimiento - No comercial - Sin obra derivada
CC BY-NC-ND**

Los autores permiten que se pueda descargar esta obra y compartirla con otras personas, siempre que se reconozca su autoría, pero no se puede cambiar de ninguna manera ni se puede utilizar comercialmente.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



USMP
UNIVERSIDAD DE
SAN MARTÍN DE PORRES

**FACULTAD DE
INGENIERÍA Y ARQUITECTURA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y
SISTEMAS**

**IMPLEMENTACIÓN DE UN SISTEMA DE INFORMACIÓN DE
SERVICIO AL CLIENTE PARA LAS MIPYMES DEL SECTOR
METALMECÁNICO DEL PERÚ**

TESIS

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

PRESENTADA POR

**ROJAS JARA, CLAUDIO EMILIO
TAFUR UMERES, JOSE ANTONIO**

LIMA – PERÚ

2016

A Dios, por inspirarnos y guiarnos diariamente en el sendero de la vida. A nuestros queridos padres y hermanos, por su apoyo incondicional.

Agradecemos al Dr. Augusto Mellado Méndez por su invaluable apoyo y consejos y a nuestro asesor, Ing. Gustavo Castillo Sini por su guía en el logro del presente trabajo de investigación.

ÍNDICE

	Página
RESUMEN	ix
ABSTRACT	x
INTRODUCCIÓN	xi
CAPÍTULO I MARCO TEÓRICO	1
1.1 Antecedentes	1
1.2 Bases teóricas	3
1.3 Glosario técnico	45
CAPÍTULO II METODOLOGÍA	50
2.1 Material y métodos	50
2.2 Desarrollo de proyecto	59
CAPÍTULO III PRUEBAS Y RESULTADOS	90
3.1 Plan de pruebas del software	90
3.3 Bondades y características del producto	103
CAPÍTULO IV DISCUSIÓN Y APLICACIONES	105
4.1 Discusión	105
4.2 Logros realizados	118
4.3 Campo de aplicación	118
CONCLUSIONES	120
RECOMENDACIONES	121
FUENTES DE INFORMACIÓN	123
ANEXOS	129

ÍNDICE DE FIGURAS

	Página
Figura 1 Uso de herramientas informáticas según actividad manufacturera	2
Figura 2 Clasificación de una empresa de acuerdo a sus ventas anuales (en UIT)	4
Figura 3 Evolución de las Mipymes formales, 2009-13	5
Figura 4 Mipymes formales en el sector manufacturero, 2013	9
Figura 5 Porcentaje de la utilización de los sistemas en el mercado	11
Figura 6 Ejemplo de material design	13
Figura 7 Tipos de desarrollo móvil	15
Figura 8 Proveedores de IaaS	20
Figura 9 Proveedores de PaaS	22
Figura 10 Proceso de la programación extrema	24
Figura 11 Grupos de Procesos de Dirección de Proyecto	30
Figura 12 Interacción detallada entre Grupos de Procesos de dirección de Proyecto	31
Figura 13 Áreas de Conocimiento de la Gerencia de Proyectos	34
Figura 14 Estructura del Árbol de Problemas	38
Figura 15 Estructura del Árbol de Objetivos	39
Figura 16 Estructura Analítica del Proyecto (EAP)	41
Figura 17 Estructura de la Matriz de Marco Lógico	42
Figura 18 API soportadas por cada sistema operativo móvil	44
Figura 19 Metodología para el desarrollo del sistema de información	54
Figura 20 Flujo de la metodología para el desarrollo del sistema de información	54
Figura 21 Modelo de Gestión de Proyectos de Software	58
Figura 22 Historia de usuario con ID 1	60
Figura 23 Historia de usuario con ID 5	60
Figura 24 Modelo Físico de la Base de Datos	65
Figura 25 Arquitectura Cliente - Servidor	66
Figura 26 Estructura del software con dependencias	67

Figura 27 Estructura de la aplicación móvil con dependencias	68
Figura 28 Interfaz de Inicio de Sesión	69
Figura 29 Interfaz de Logout (Cierre de Sesión)	69
Figura 30 Interfaz Mis usuarios	70
Figura 31 Interfaz de Mis proyectos (Consulta de proyectos)	71
Figura 32 Interfaz Nuevo Proyecto	71
Figura 33 Interfaz Ver más (Mostrar proyecto)	72
Figura 34 Interfaz Mostrar foto de comprobante de pago	72
Figura 35 Interfaz Programar visita	73
Figura 36 Interfaz Cancelar programación de visita	73
Figura 37 Interfaz Nueva visita (Planificar visita)	74
Figura 38 Interfaz Visitas (Consulta de visitas programadas)	74
Figura 39 Interfaz Nueva reparación (visita realizada)	75
Figura 40 Interfaz Reparaciones (Consulta de visitas realizadas)	75
Figura 41 Interfaz Mis ventas	76
Figura 42 Interfaz Visitas por proyecto	77
Figura 43 Interfaz Calificaciones por técnico	77
Figura 44 Interfaz Mis clientes	78
Figura 45 Interfaz Nuevo cliente	78
Figura 46 Interfaz Más información del cliente	79
Figura 47 Interfaz Contactos de clientes	79
Figura 48 Interfaz Mis técnicos	80
Figura 49 Interfaz Nuevo técnico	80
Figura 50 Interfaz Mis categorías de proyectos	81
Figura 51 Interfaz Nuevo tipo de proyecto	81
Figura 52 Interfaz Login de la app móvil	82
Figura 53 Interfaz Mis visitas programadas de la app móvil	83
Figura 54 Interfaz Nueva visita de la app móvil	84
Figura 55 Interfaz Mis visitas realizadas de la app móvil	85
Figura 56 Interfaz Calificación de la visita de la app móvil	86
Figura 57 Modelo Canvas	88
Figura 58 Resultados de la ejecución de las pruebas unitarias	90
Figura 59 Prueba unitaria a Contact	93
Figura 60 Resultados de la prueba a Contact	94
Figura 61 Prueba unitaria a Customer	94
Figura 62 Resultados de la prueba a Customer	95
Figura 63 Prueba unitaria a Email	95
Figura 64 Resultados de la prueba a Email	96
Figura 65 Prueba unitaria a Employee	96
Figura 66 Resultados de la prueba a Employee	97
Figura 67 Prueba unitaria a Maintenance	97
Figura 68 Resultados de la prueba a Maintenance	98
Figura 69 Prueba unitaria a Project	99
Figura 70 Resultados de la prueba a Project	99
Figura 71 Prueba unitaria a Repair	100
Figura 72 Resultados de la prueba a Repair	100
Figura 73 Prueba unitaria a Telephone	101

Figura 74 Resultados a la prueba Telephone	102
Figura 75 Prueba unitaria a TypeProject	103
Figura 76 Resultados de prueba a TypeProject	103
Figura 77 Mejora en el servicio de post-venta de INNOMECC	106
Figura 78 Mejora en el servicio de post-venta de Procesos Continuos	107
Figura 79 Diagrama de flujo del proceso de planificación de visita	109
Figura 80 Diagrama de flujo del proceso de planificación de visita con el sistema	110
Figura 81 Distribución del tiempo en el desarrollo del proyecto	111
Figura 82 Consolidación de información en INNOMECC	112
Figura 83 Consolidación de información en Procesos Continuos	113
Figura 84 Diagrama de flujo del proceso de consolidación de información	114
Figura 85 Diagrama de flujo del proceso de consolidación de info. con el sistema	115
Figura 86 Diagrama de flujo del proceso servicio técnico	116
Figura 87 Diagrama de flujo del proceso servicio técnico con el sistema	117

ÍNDICE DE TABLAS

	Página
Tabla 1 Empresas formales (2013)	4
Tabla 2 Productos más vendidos en el 2012	7
Tabla 3 Exportaciones de productos metalmecánicos en el 2012	7
Tabla 4 Exportaciones de Mipymes, según sector económico, 2013-12	8
Tabla 5 Mipymes formales, según sector económico, 2009-2013	9
Tabla 6. Ventajas y desventajas desarrollo nativo	16
Tabla 7. Ventajas y desventajas del desarrollo con HTML5	18
Tabla 8. Materiales usados en la ejecución de la tesis	50
Tabla 9. Plan de iteración	62
Tabla 10. Tarjeta CRC Project	63
Tabla 11. Lista de elementos de pruebas	91
Tabla 12. Recursos de hardware para las pruebas	92
Tabla 13. Recursos de software para las pruebas	93
Tabla 14. Clientes atendidos	105
Tabla 15. Mejora en el servicio de post-venta de INNOMECE	106
Tabla 16. Mejora en el servicio de post-venta Procesos Continuos	107
Tabla 17. Tiempo promedio de planificación de visitas	108
Tabla 18. Historial de desarrollo del proyecto	111
Tabla 19. Historiales de post-venta en INNOMECE	112
Tabla 20. Historial de post-venta en Procesos Continuos	113

RESUMEN

El sector metalmecánico es muy importante para el desarrollo del país, porque no solo genera tecnología y mejores empleos, sino que también sirve como nexo entre muchas de las actividades económicas, permitiendo su desarrollo. Cabe mencionar que las empresas metalmecánicas se encargan de realizar principalmente máquinas automáticas (como dosificadoras, etiquetadoras, envasadoras y otras) y semiautomáticas (como selladoras, soldadoras de paquetes y otras) que serán utilizadas por empresas de diferentes rubros.

El objetivo de la presente tesis es mejorar el servicio de post-venta de las Mipymes del sector metalmecánico del Perú, proporcionando un sistema de información que soporte las características de estas empresas, utilizando cloud computing (la nube), teniendo en cuenta que resultaría costoso la adquisición de equipos computacionales y de comunicación dado su limitada infraestructura tecnológica.

Por tanto, a base del modelo de negocio que se propuso en esta tesis, se adquirió una instancia de un proveedor de cloud computing y en él se desplegó el software web, brindándoles acceso al mencionado software a las Mipymes de acuerdo con el número de licencias de usuario que tengan. Además, se instaló una aplicación móvil desde la tienda virtual.

ABSTRACT

The metalworking sector is very important for the country's development, because not only creates technology and better jobs, but also serves as a link between many economic activities, allowing their development. It is necessary to mention that metalworking companies are in charge to make automatic machines (such as metering, labeling, packaging, etc) and semi-automatic (as sealers, welding packages, etc). All of these will be used by companies of different working sectors.

The objective of this thesis is to improve the service after sales of MSMEs in the metalworking sector of Peru, providing an information system that supports the features of these companies, using cloud computing, considering that it would be costly to acquisition of computer equipment and communications given their limited technological infrastructure.

Therefore, based on the business model proposed in this thesis, an instance of a cloud computing provider was acquired and in it the web software was deployed, giving access to MSMEs according to the number of user licenses they have. Also, a mobile application was installed from the virtual store.

INTRODUCCIÓN

En la presente tesis, se desarrolló un sistema de servicio de post-venta al cliente para las Mipymes del sector metalmecánico del Perú. Para ello, primero se identificó el problema, luego se estableció el objetivo general y los objetivos específicos del proyecto. Además, se definió la justificación de la realización del proyecto y se determinaron sus límites.

El primer capítulo está conformado por toda la información relevante que soporta teóricamente al proyecto, que es la información del área temática, tecnologías, metodologías y estándares en los que se basó la tesis.

El segundo abarca, específicamente, la metodología que se utilizó en el proyecto, para ello hubo una selección de metodología de desarrollo de software adaptada al producto del acotado proyecto, luego se explicó el flujo que se usó para la administración de proyecto. También, es parte del capítulo los materiales (herramientas tecnológicas) que formaron parte del desarrollo, comunicación, gestión y documentación del proyecto. Además, hay una sección referida, exclusivamente, al desarrollo del proyecto, que está dividida en el desarrollo del sistema de información y en el del modelo de negocio.

En el tercer capítulo, se realizaron las pruebas a nivel de proyecto y producto. Luego, se definieron las características y bondades del producto desarrollado y se estableció los resultados indirectos logrados.

En el cuarto y último, se interpretaron los resultados obtenidos del proyecto respecto a las empresas metalmecánicas que formaron parte del estudio (Procesos Continuos SAC e INNOMECC) proporcionando de esta manera directrices acerca del sector.

El problema, en el Perú, las micros, pequeñas y medianas empresas del sector metalmecánico utilizan un tiempo significativo para brindar servicio al cliente luego de la venta de sus productos, considerando que sus registros lo realizan manualmente y en algunos casos empleando hojas de cálculo. Así como, se debe tener en cuenta que cuando un cliente compra o adquiere un producto requiere que la empresa que lo fabricó pueda asistirlo oportunamente, en su mantenimiento.

Con relación a lo descrito en el párrafo anterior, se usó la metodología del Marco Lógico para la selección del problema, cuyo resultado fue que el problema determinado es el ineficiente servicio de post-venta de las Mipymes metalmecánicas del Perú.

Este problema se debe principalmente a tres factores o causas:

1. Existe retraso en la programación de las visitas al cliente, porque se desconoce la vigencia exacta de la garantía de los productos vendidos debiendo calcularlo a base de la fecha de fabricación del producto. Además, se desconoce la disponibilidad de los técnicos al planificar una visita.
2. Hay una escasa utilización de sistemas de información de post-venta integrados, porque las interfaces de usuario son difíciles de usar y existe una limitada infraestructura tecnológica por parte de las Mipymes contribuyendo al difícil despliegue y mantenimiento de un software.
3. Hay una limitada consolidación de información de post-venta, porque existe insuficiente información obtenida del servicio técnico, dado

que registran, de manera incompleta, el servicio realizado. Además, es difícil el acceso a la información de los proyectos vendidos.

Todo lo mencionado anteriormente trae como consecuencia que exista una difícil trazabilidad de los procesos, poca información oportuna y consistente, y un retraso en brindar el servicio de mantenimiento y reparación al cliente, originando que exista pérdida de control de procesos, disminución en la cartera de clientes e imagen negativa de la empresa, contribuyendo a que las Mipymes metalmeccánicas tengan baja rentabilidad.

Para mayor detalle sobre el problema, véase el Anexo N° 1 “Árbol de Problemas”.

Como objetivo general se planteó mejorar el servicio de post-venta de las Mipymes del sector metalmeccánico del Perú.

Los objetivos específicos son:

1. Agilizar la programación de visitas al cliente, con la cual se mejorará el tiempo de planificación de visitas, para ello se deberá conocer la vigencia exacta de la garantía de los productos vendidos y la disponibilidad de los técnicos.
2. Desarrollar un sistema de información de post-venta integral. Este sistema de información a desarrollar estará compuesto por un software web con interfaces fáciles de usar que estará comunicado y conectado con una aplicación móvil. Además, se brindará una infraestructura tecnológica utilizando el modelo de servicio de cloud computing.
3. Impulsar la consolidación de información de post-venta, que aumentará la cantidad de información consolidada de manera automatizada, incrementado la información obtenida del servicio técnico y facilitando el acceso a la información de los proyectos.

El cumplimiento del objetivo principal originará que se mejore la trazabilidad de procesos, se incremente la información oportuna y consistente, y se mejoren los tiempos en el servicio al cliente, trayendo consigo un mejor control de procesos, incremento en la cartera de clientes y mejora en la

imagen de la empresa, con la finalidad de contribuir a incrementar la rentabilidad de las Mipymes metalmecánicas que utilicen el sistema de información.

Para mayor detalle sobre los objetivos, véase el Anexo N° 2 “Árbol de Objetivos”.

Como justificación que el proyecto es importante, porque las empresas del rubro metalmecánico, especialmente las Mipymes, necesitan agilizar el servicio de mantenimiento y reparaciones de equipos y sus componentes para consolidar oportunamente la información y a base de esta se tomen mejores decisiones. Para lograr esto, es necesario tener una plataforma integral que incluya una aplicación para dispositivos móviles y software con acceso web, para la consulta de garantías, asignación de personal para mantenimientos y reparaciones, y registro de servicio técnico. De esta manera, se puede llevar un registro histórico de las incidencias y conocer las mayores fallas de los productos que venden.

Es importante mencionar que una de las novedades de este proyecto es que se usó el modelo de negocio que brinda cloud computing, que consiste en un pago mensual de un monto competitivo por la utilización del sistema, esto es mucho mejor para este tipo de empresas (Mipymes) que a comparación del clásico modelo de negocio, en el que se vendía el software a un precio alto y un solo pago único, que podía constituir un riesgo principal que el software no sea el adecuado y que las empresas hayan pagado tanto por algo que no utilizaron.

Las limitaciones que hubo fueron:

- El sistema de información está dirigido a empresas del rubro metalmecánico.
- Las empresas metalmecánicas que formaron parte de la investigación del presente proyecto fueron Procesos Continuos SAC e INNOMECE.
- Las funcionalidades del sistema de información están relacionadas al módulo de servicio de post-venta al cliente, las cuales son:

consulta de proyectos, consulta de vigencia de garantías, asignación de personal para mantenimientos y reparaciones, y registro de servicio técnico. Se puede planificar la visita (asignación de personal para mantenimientos y reparaciones) de un proyecto vendido y registrar el servicio técnico brindado ya sea que el proyecto esté dentro del tiempo de garantía o no lo esté.

- El sistema de información está compuesto por un software web, una aplicación para dispositivos móviles y una instancia de cloud computing. En la aplicación para dispositivos móviles solo se puede registrar el servicio técnico brindado al cliente y una evaluación respecto al servicio. Sin embargo, el software web ejecuta todas sus funcionalidades en cualquier dispositivo móvil con navegador web.
- El registro de las cláusulas o detalles de la garantía no forman parte de la primera versión del sistema (versión que es parte del presente trabajo de investigación).

CAPÍTULO I

MARCO TEÓRICO

1.1 Antecedentes

El sector metalmecánico comprende una gran variedad de industrias manufactureras, comenzando por la producción de pequeños productos hasta la fabricación de equipos con alto nivel tecnológico.

Este sector es importante para el desarrollo del país, debido a que, además, de generar valor agregado y desarrollo tecnológico, interacciona y apoya a otros sectores claves del país (como la minería), en otras palabras, la industria metalmecánica sirve como nexo de mejora y avance entre las diferentes industrias del país (Universidad Católica Sedes Sapientae, 2014).

La industria metalmecánica está conformada por las industrias metálicas básicas, de producción de máquinas y equipos, de fabricación de productos de acero, industria automotriz y equipos para el transporte, construcciones metálicas, electromecánicas, entre otros, por lo tanto, el sector metalmecánico es conocido como la “Industria de industrias” (Universidad Católica Sedes Sapientae, 2014).

El presente sector tiene un papel fundamental en la generación de empleo, porque necesita operarios, maquinistas, obreros, ingenieros, y otros profesionales, donde la cantidad de personal depende mucho de lo que produzca la empresa, pero siempre tiene que ser personal capacitado, tecnológicamente, por lo tanto, un personal capacitado en tecnología tiene mayores oportunidades de realizar mejor sus actividades y tareas, de mejorar la empresa en la que labora, generando mejores niveles tecnológicos para la industria y a largo plazo crecimiento para el país (Universidad Católica Sedes Sapientae, 2014).

En el sentido del avance tecnológico, el uso de herramientas informáticas se ha convertido indispensable en la cadena de valor de las empresas, porque no solo permite obtener una mayor eficiencia en las actividades que realizan a diario, sino también mejorar la relación que tienen con sus clientes.

A continuación, se muestra una ilustración con los resultados de un estudio acerca del uso de herramientas informáticas en empresas metalmeccánicas ubicadas en Lima Norte.

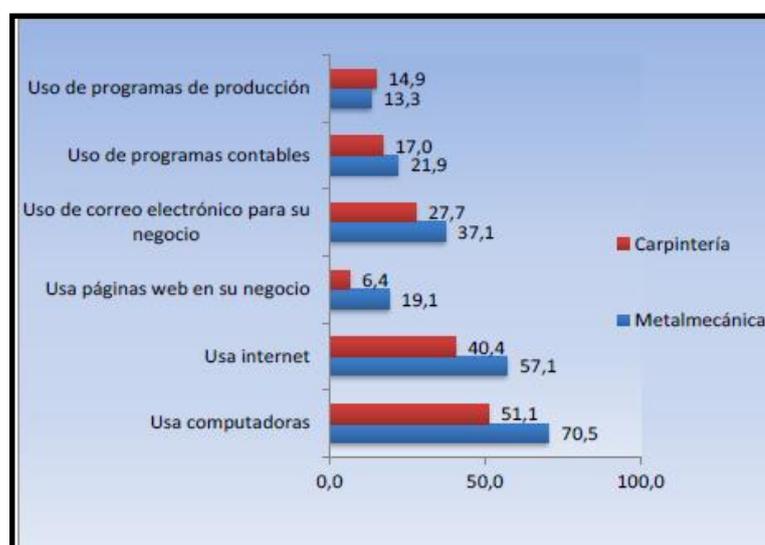


Figura 1. Uso de herramientas informáticas según actividad manufacturera

Fuente: (Universidad Católica Sedes Sapientae, 2014)

De la ilustración anterior, el 70,5% de las empresas metalmecánicas utilizan computadoras en sus actividades, lo cual es una cifra alentadora, pero la parte desfavorable es para qué lo usan.

Como podemos apreciar en el gráfico precedente, un bajo porcentaje de empresas utilizan los programas informáticos como apoyo en el desarrollo de sus actividades y/o procesos, como es el caso del uso de páginas web en el negocio y el uso de internet, que son elementos importantes para el desarrollo de la comunicación con los clientes, el mismo que contribuiría en la mejora de la rentabilidad de la empresa.

1.2 Bases teóricas

1.2.1 Mipyme

a) Definiciones

a.1. Microempresa

Las microempresas son unidades productoras que tienen ventas anuales hasta el monto máximo de 150 Unidades Impositivas Tributarias (UIT) (Ministerio de la Producción, 2015).

a.2. Pequeña empresa

Las pequeñas empresas son unidades productoras que tienen ventas anuales superiores a 150 UIT y hasta el monto máximo de 1700 Unidades Impositivas Tributarias (UIT) (Ministerio de la Producción, 2015).

a.3. Mediana empresa

Las medianas empresas son unidades productoras que tienen ventas anuales superiores a 1700 UIT y hasta el monto máximo de 2300 UIT (Ministerio de la Producción, 2015).

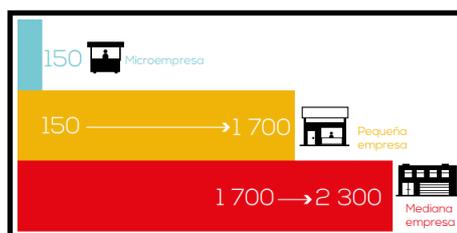


Figura 2. Clasificación de una empresa de acuerdo a sus ventas anuales (en UIT)

Fuente: (Ministerio de la Producción, 2015)

b) Situación actual

Según el último estudio realizado en el 2013, la mayoría de empresas en el Perú son microempresas, solo un 4.6% son pequeñas empresas y 0.2% son medianas empresas, por lo cual se puede mencionar que lo que mueve la economía de este país son realmente las microempresas (Ministerio de la Producción, 2015).

Lo mencionado anteriormente se puede comprobar en el siguiente cuadro que muestra la situación de las empresas formales en el 2013:

Tabla 1. Empresas formales (2013)

Estrato empresarial ¹	Nº de empresas	%
Microempresa	1 439 778	94,6
Pequeña Empresa	70 708	4,6
Mediana empresa	2 520	0,2
Total de mipymes	1 513 006	99,5
Gran Empresa	8 306	0,5
Total de empresas	1 521 312	100,0

Fuente: (Ministerio de la Producción, 2015)

Un dato importante a resaltar es el que las Mipymes han tenido un crecimiento anual promedio de 7.6%, entre 2009 y 2013. Este crecimiento está directamente relacionado con el crecimiento de la economía del país.

A continuación, se muestra un gráfico con la evolución de las Mipymes formales:

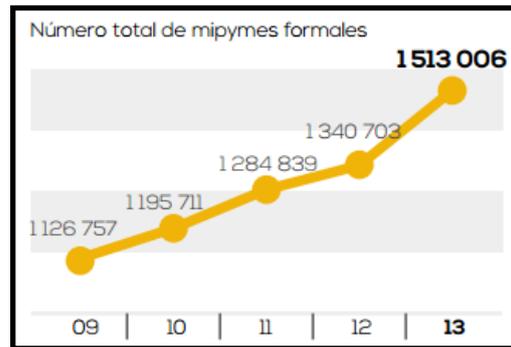


Figura 3. Evolución de las Mipymes formales, 2009-13

Fuente: (Ministerio de la Producción, 2015)

Por otro lado, según el último estudio de Ipsos-Apoyo “Uso de las TICs en Mipymes” solo uno de cada tres Pymes rezagadas (aquellas que no tienen página web, páginas en redes sociales y no usan Computación en la nube, entre otros) conocen que es cloud computing (Barrón, 2015). Además, tres de cada cinco Mipymes no usan los servicios de computación en la nube. En el año 2011, el 35% de Mipymes hacían uso de alguno de los servicios de cloud computing, mientras que el 65% no usaba ninguno, mientras que en el 2014 el 42% de Mipymes ya hacían uso de algún servicio de cloud computing, y los que no lo usan bajaron a 58% (Gestión, 2014).

1.2.2 Sector metalmeccánico

a) Concepto y generalidades

La industria metalmeccánica contiene un conjunto de actividades manufactureras, en la cual se utilizan como insumos principales los productos de la siderurgia y/o derivados, con la finalidad de procesarlos y transformarlos en algún producto funcional (como equipos y máquinas industriales) (Universidad Católica Sedes Sapientae, 2014).

La actividad metalmeccánica es la más importante dentro de las actividades productivas de un país, debido a que no solo genera tecnología y valor agregado, sino también provee una conexión en el desarrollo de otros sectores industriales.

b) Contexto nacional

Las exportaciones de la industria metalmeccánica aumentaron 12% en el año 2012 respecto al 2011, al alcanzar un valor de 545 millones de dólares (Gestión, 2013).

La recuperación de la economía mundial y el incremento del precio de los metales impulsaron a esta industria, cuyo 20% es exportado a países de América Latina. Vale decir que al cierre del 2012 hubo un incremento en la producción alrededor del 10%, con respecto a los 2,340 millones de dólares registrados el 2011 (Gestión, 2012).

Los Tratados de Libre Comercio (TLC) firmados vienen permitiendo la exportación de los productos terminados de gran demanda en la actividad minera mundial. Este cluster minero-industrial está generando más de 300 mil empleos formales. Por ejemplo, un soldador de primer nivel llega a recibir un sueldo de casi 2,200 nuevos soles semanales en promedio, debido a que realiza un trabajo muy especializado y de mucha concentración (Biznews, 2011).

Es muy importante el cluster Industria-Minería porque no solo apoya a los principales proyectos mineros del Perú y el mundo, sino que también está logrando posicionarse en la industria energética, del gas natural y hasta la petroquímica (Biznews, 2011).

La revista virtual Semana Económica (SEMANAeconómica, 2012) menciona que los productos que destacan en el sector metalmeccánico son los siguientes:

- Productos de zinc
- Conductores de cobre
- Bolas de molino
- Máquinas para la minería
- Grúas

- Palas mecánicas
- Grupos electrógenos
- Pequeñas embarcaciones
- Buses para el transporte de personal

A continuación, se presenta un cuadro con los productos vendidos que destacaron en el sector metalmecánico en el 2012:

Tabla 2. Productos más vendidos en el 2012

Producto	Monto de Venta (Millones de Dólares)
Partes de máquinas y aparatos de molinos de anillo	29
Conductores eléctricos de cobre	28
Bolas para molinos de fundición de hierro	21
Partes de máquinas como palas mecánicas, excavadoras y cargadoras	18

Fuente: (Gestión, 2013)

Entre los principales países a los cuales se exportan nuestros productos metalmecánicos son Estados Unidos y Ecuador.

Se presenta un cuadro que muestra el monto de compra de productos metalmecánicos peruanos en el año 2012 por otros países:

Tabla 3. Exportaciones de productos metalmecánicos en el 2012

País	Monto de compra (Millones de Dólares)
Estados Unidos	82
Ecuador	79
Chile	76
Venezuela	55

Fuente: (Gestión, 2013)

Cabe indicar que Chile y Estados Unidos destacan por sus actividades mineras, mientras que Venezuela y Ecuador se caracterizan por actividades petroleras

Además de estos países ya mencionados, también México, Colombia, Bolivia y Brasil compran productos metalmeccánicos peruanos (SEMANAeconómica, 2012).

Vale mencionar que Canadá es un mercado atractivo para industria metalmeccánica peruana, porque presenta comportamientos similares al de Estado Unidos (SEMANAeconómica, 2012).

c) Situación de las Mipymes metalmeccánicas

Las exportaciones del sector metalmeccánico se han mantenido en los últimos años, prueba de ello es que se muestra un cuadro que indica sus porcentajes de exportaciones en los años 2012 y 2013:

Tabla 4. Exportaciones de Mipymes, según sector económico, 2013-12

Sector económico	2012		2013	
	US\$	%	US\$	%
Textil	693	37,7	582	31,8
Agropecuario	480	26,1	486	26,6
Pesquero no tradicional	187	10,2	21102	11,1
Minero	39	2,1	173	9,5
Agrícola	64	3,5	68	3,7
Químico	102	5,5	60	3,3
Metalmecánico	51	2,8	50	2,7
Minería no metálica	31	1,7	28	1,5
Sidero-metalúrgico	13	0,7	23	1,2
Pesquero tradicional	17	0,9	16	0,9
Resto	160	8,7	140	7,7
Total	1 837	100,0	1 829	100,0

Fuente: (Ministerio de la Producción, 2015)

Del cuadro anterior, a las Mipymes del sector metalmeccánico les correspondió una cifra de 2,7 % de las exportaciones que se realizaron en el Perú en el año 2013 respecto a las micro, pequeñas y medianas empresas, por tanto, se puede deducir que las empresas pequeñas de la industria metalmeccánica exportan poco en comparación con la de otros sectores como el textil.

En el siguiente cuadro, se observa la distribución de las Mipymes por sector económico, donde las metalmecánicas se encuentran dentro del rubro manufacturero:

Tabla 5. Mipymes formales, según sector económico, 2009-2013

Sector Económico	Mipyme 2009		Mipyme 2013		VAP ¹ 2009/2013
	Nº	%	Nº	%	
Comercio	512 738	45,5	694 358	45,9	6,3
Servicios	438 964	39,0	589 357	39,0	6,1
Manufactura	115 122	10,2	144 506	9,6	4,7
Construcción	28 301	2,5	47 378	3,1	10,9
Agropecuario	22 419	2,0	24 131	1,6	1,5
Minería	5 967	0,5	9 620	0,6	10,0
Pesca	3 246	0,3	3 656	0,2	2,4
Total	1 126 757	100	1 513 006	100	6,1

Fuente: (Ministerio de la Producción, 2015)

De la tabla anterior, interesa la evolución que sufrió el sector manufacturero porque dentro están las Mipymes metalmecánicas. Se puede mencionar que se incrementó la cantidad de empresas manufactureras, pero porcentualmente ha disminuido respecto a la distribución total de empresas.

Además, en la siguiente figura, se distinguen las Mipymes manufactureras que brindan servicios metalmecánicos:

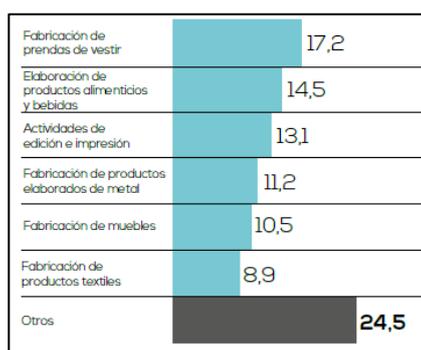


Figura 4. Mipymes formales en el sector manufacturero, 2013

Fuente: (Ministerio de la Producción, 2015)

De la figura anterior, el 11.2 % de las Mipymes manufactureras brindan servicios metalmecánicos (fabricación de productos elaborados de metal), por tanto, teniendo en cuenta la tabla 5 y la

figura anterior mínimo existen dieciséis mil ciento ochenta y cuatro (16,184) Mipymes en el Perú.

1.2.3 Servicio de post-venta al cliente

a) Generalidades

El servicio de post-venta al cliente es un factor importante para la competitividad de las empresas y se ha convertido en uno de los componentes más relevantes dentro de las organizaciones (Ortiz & Ruiz, 2015).

Este tipo de servicio se encuentra en la economía industrial (empresas manufactureras) como valor agregado al producto, otorgando una diferenciación respecto a la competencia, de esta manera, el cliente sabe que el producto que está adquiriendo se soporta y respalda en la ayuda técnica de la organización, contribuyendo a la obtención y fidelización de clientes (Ortiz & Ruiz, 2015).

En las últimas décadas, el comprador o cliente se ha vuelto parte importante de la vida de las empresas, y es este quien siempre busca las mejores oportunidades y condiciones del producto o servicio que adquirirá. En tal sentido, en un mercado tan competitivo como el actual, ya no solo el precio, costo y características propias del producto son suficientes. Ahora la ventaja competitiva está en elegir el producto que cuenta con un adecuado servicio al cliente (Ortiz & Ruiz, 2015).

La calidad del servicio está relacionada con el cumplimiento de la promesa de servicio, hacer lo que se prometió al cliente, en el tiempo que se pactó y cubrir las necesidades o especificaciones que este solicitó, esto contribuye a una mejor percepción del cliente en cuanto al producto que compra y respecto a la empresa que lo fabrica (Ortiz & Ruiz, 2015).

b) Sistemas de servicio de post-venta

Actualmente, las empresas utilizan herramientas tecnológicas para mantener contacto con sus clientes y brindar una mejor respuesta ante algún inconveniente que tenga, de esta manera, el cliente siente que está recibiendo un buen trato y que la empresa se preocupa por él (Ortiz & Ruiz, 2015).

El uso de sistemas tecnológicos se debe a la necesidad de mejorar el servicio post-venta a los clientes y generar un valor agregado importante en el momento de realizar o brindar el servicio. Se debe tener en cuenta que al desarrollar un sistema de servicio al cliente lo que se debe buscar es la facilidad y comodidad del cliente y de la organización que lo implementa (Ortiz & Ruiz, 2015).

Se muestra el porcentaje de utilización de los sistemas de servicio de post-venta al cliente:



Figura 5. Porcentaje de la utilización de los sistemas en el mercado

Fuente: (Ortiz & Ruiz, 2015)

De la figura anterior, la mayor parte de uso de sistemas de servicio al cliente corresponde a la obtención de información cuando el cliente tiene alguna duda acerca del producto; los otros dos porcentajes son importantes para realizar reclamos y transacciones, ambos esenciales cuando se trata de brindar soporte y ayuda al cliente de manera oportuna.

1.2.4 Desarrollo de interfaces de usuario

a) Generalidades

Las interfaces de usuario contribuyen a que el usuario perciba y se adapte fácilmente al contenido que se quiere mostrar, logrando facilidad de uso en el sistema de información.

En la actualidad, mínimo se debe buscar utilizar un diseño responsivo al momento de realizar un sitio web, ya que contribuirá a no tener una versión diferente por cada dispositivo, por tanto, ahorrar en los costos de desarrollo.

b) Material Design

Material Design es una corriente de diseño, que quiere decir, basado en objetos materiales, en otras palabras, las piezas están colocadas en un lugar (espacio) y con un tiempo (movimiento) determinado (El androide libre, 2014).

En esta tendencia la profundidad, las superficies, los bordes, las sombras y los colores juegan un papel importante. Utilizando Material Design los desarrolladores deben guiarse por las leyes de la física, donde las animaciones sean lógicas, los objetos se superpongan, pero no puedan atravesarse el uno al otro y demás (El androide libre, 2014).

Otro punto por resaltar en Material Design, es que debe reflejar un diseño con una tipografía clara, casillas bien ordenadas, colores e imágenes llamativos para no perder el foco y un sentido del orden y la jerarquía muy marcada (El androide libre, 2014).

Muestra de un ejemplo del diseño Material Design:

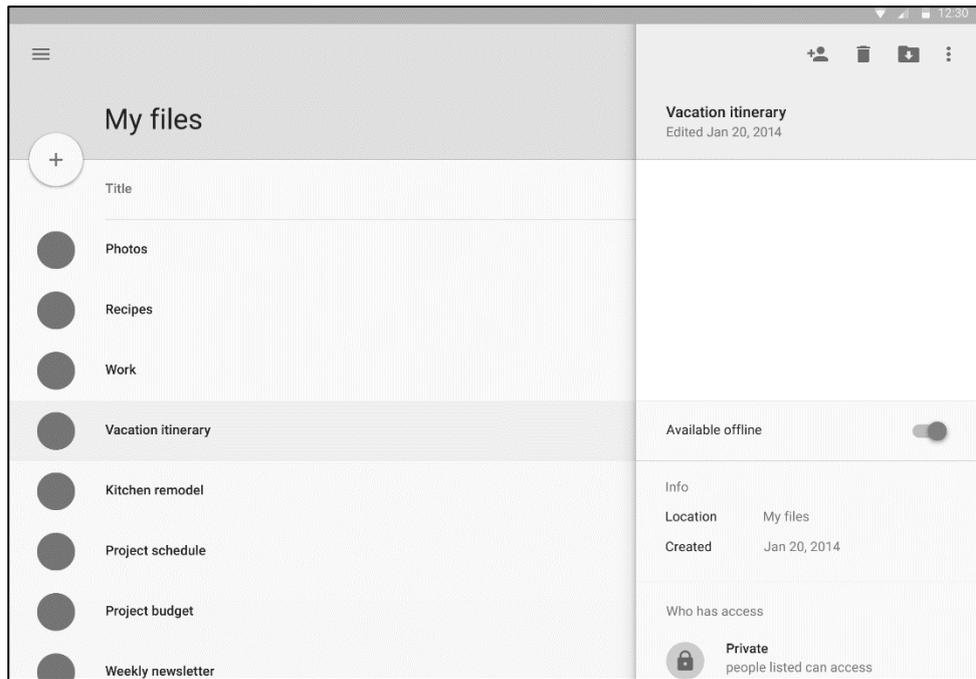


Figura 6. Ejemplo de material design

Fuente: (El androide libre, 2014)

De la figura anterior, se puede observar que el menú de la derecha, además, de tener un fondo más claro proyecta su sombra sobre el panel principal, dando la sensación de estar delante.

Es importante resaltar que el movimiento es una parte esencial dentro de Material Design, por ejemplo, un objeto que parpadea significa que está llamando la atención o un objeto que se expande significa que se acaba de abrir, por tanto, todos los movimientos siempre deben crearse en una dirección determinada y el usuario debe ser capaz de poder percibirlo (Google, 2015).

c) Responsive Design (Diseño adaptativo)

El diseño adaptativo o responsivo es una tendencia de creación de páginas web que puedan ser visualizadas, perfectamente, en todo tipo de dispositivos, desde computadoras de escritorio hasta móviles, de tal manera que no es necesario tener versiones para cada dispositivo, una sola web se adapta a todos ellos (Maestros del Web, 2013).

El modo de funcionamiento del diseño responsivo es que el sitio web detecta qué clase de dispositivo está accediendo el usuario (utilizando CSS3 media queries) y muestra la versión más correcta para ese medio, reorganizando los elementos de la web e incluso discriminando alguno de ellos, logrando tener menos imágenes, redistribuyendo las columnas en el diseño o reduciendo la cantidad de texto mostrado (Maestros del Web, 2013).

d) Parallax

Es una tendencia en el actual diseño web, que consiste en generar una visión dinámica de la página web. El efecto parallax se logra visualizar en el instante en que el usuario está haciendo scroll y de repente el fondo parece moverse a una velocidad distinta que el contenido, creando un efecto de profundidad y dejando ver partes que antes no se veía (Dueñas Villamiel, 2013).

Cuando se quiere contar una historia es recomendable utilizar Parallax, de esta manera, se logrará conseguir una experiencia gráfica y romper lo plano que pudiera estar la página (Dueñas Villamiel, 2013).

1.2.5 Desarrollo de aplicaciones móviles

a) Generalidades

El desarrollo de una aplicación móvil, dependiendo de su naturaleza y requerimientos puede ser de tres tipos: desarrollo nativo, desarrollo multiplataforma compilado a nativo y desarrollo multiplataforma basado en HTML5 (CampusMVP, 2014).

Es importante mencionar que no hay un tipo de desarrollo mejor que otro, sino que de acuerdo a las necesidades y recursos que se tengan, convendrá realizar alguno de ellos (CampusMVP, 2014).

En la siguiente figura, se muestra el tipo de desarrollo a elegir de acuerdo con las características del equipo humano:

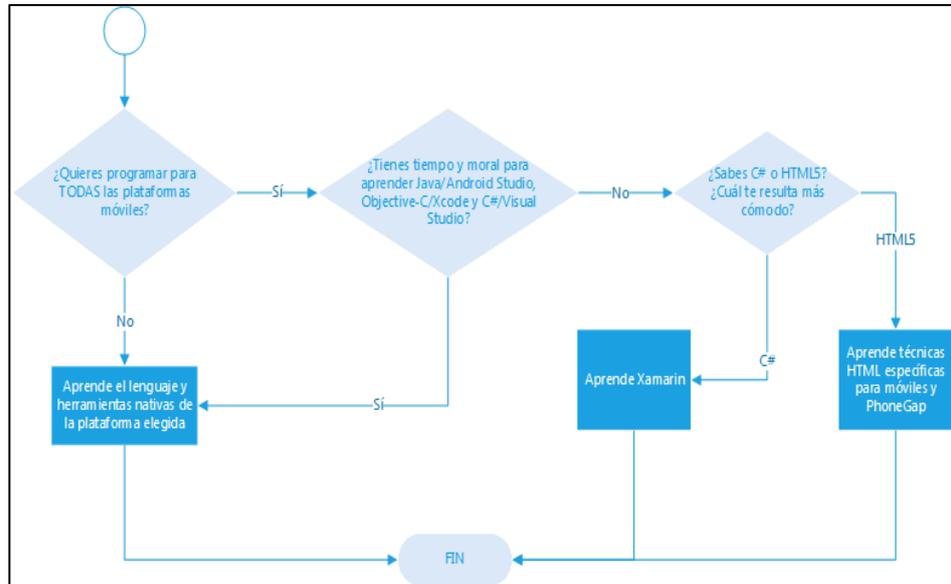


Figura 7. Tipos de desarrollo móvil

Fuente: (CampusMVP, 2014)

Según la figura anterior, convendrá elegir el desarrollo nativo si es que se tiene tiempo para aprender cada lenguaje de las diferentes plataformas móviles, si en cambio se quiere realizar una aplicación para varios sistemas operativos y se conoce un lenguaje de programación de backend en particular (como C#). Este posee herramientas para construir app multiplataforma (un mismo código reutilizarlo en varias plataformas) se ajustará mejor escoger el desarrollo multiplataforma compilado a nativo. Si sucediera que el requerimiento es desarrollar para varios sistemas operativos, no se conoce muy bien un lenguaje de programación de backend y sí se tiene conocimientos de HTML5, CSS y JavaScript, es conveniente desarrollar una aplicación móvil multiplataforma basada en HTML5.

Es importante mencionar que pueden intervenir otros factores al momento de elegir un tipo de desarrollo, como cantidad de desarrolladores que existen en el mercado que conocen acerca de una tecnología que puedan brindar mantenimiento a la aplicación, limitaciones en el presupuesto, hardware a utilizar y otros.

b) Desarrollo nativo

Una aplicación nativa es la que se desarrolla de manera específica para un determinado sistema operativo, entre los más populares están el desarrollo para Android, iOS y Windows Phone. Cada plataforma posee su propio lenguaje de programación y sus diversas API que les permiten acceder al hardware del equipo (LanceTalent, 2014).

La principal diferencia entre el desarrollo nativo y multiplataforma, es que con el primero podemos acceder fácilmente y por completo a todos los aspectos internos del hardware (como cámara, GPS, agenda, dispositivos de almacenamiento, y otros muchos), en cambio con el desarrollo multiplataforma a veces se dificulta estandarizar el acceso al hardware o de encontrar un total acceso (cuando se trata de desarrollo multiplataforma basado en HTML5) (LanceTalent, 2014).

Con respecto a sus ventajas e inconvenientes, estos se muestran en el siguiente cuadro:

Tabla 6. Ventajas y desventajas desarrollo nativo

Ventajas	Desventajas
Acceso completo al dispositivo	Diferentes habilidades, idiomas y herramientas para cada plataforma de destino
Mejor experiencia de usuario	Elevado costo de desarrollo, si se desea desarrollar en varias plataformas.
Visibilidad en APP Store de forma rápida	Código del cliente no es reutilizable entre las diferentes plataformas

Fuente: (LanceTalent, 2014)

c) Desarrollo multiplataforma compilado a nativo

Esta opción consiste en desarrollar una aplicación móvil para múltiples sistemas operativos en base a la selección de una plataforma mixta que sea independiente del lenguaje nativo de cada sistema operativo, con la finalidad de compilarlo al término del desarrollo a los sistemas operativos destino (CampusMVP, 2014).

La gran ventaja es que el código es reutilizable entre los diferentes sistemas operativos, por tanto, disminuye el costo de desarrollo, es importante mencionar que el código resultante se tiene que adaptar de acuerdo a las necesidades de acceso de las funcionalidades de cada sistema operativo del dispositivo móvil (CampusMVP, 2014).

Una de las herramientas más conocidas en el mercado es Xamarin, la cual está basada en C# de Microsoft y en la plataforma .NET. Las aplicaciones realizadas en Xamarin permiten compilar a código nativo de cada plataforma, permitiendo obtener un rendimiento igual al desarrollado de manera nativa (CampusMVP, 2014).

d) Desarrollo multiplataforma basado en HTML5

Este tipo de desarrollo consiste en crear aplicaciones web que se parecen a las aplicaciones nativas en el diseño, esto se logra porque cada dispositivo móvil trae por defecto un navegador que soporta HTML5. Es importante indicar que es cierto que se utiliza un navegador web; sin embargo, el usuario no percibe el uso del navegador, porque la aplicación se descarga de la APP Store, se instala y se oculta el navegador al momento de ejecutar la app (CampusMVP, 2014).

Para realizar una aplicación móvil basada en HTML5 es necesario conocer HTML, CSS y Javascript, una vez terminada la aplicación se procede a compilar el código a cada uno de los sistemas operativos destinos (CampusMVP, 2014).

Se debe mencionar que al igual que en el desarrollo multiplataforma compilado a nativo, también se tiene acceso al hardware del equipo,

solo que dependiendo de lo que se quiere acceder puede resultar con cierto nivel de complejidad (Márquez, 2015).

En el siguiente cuadro, se muestran las ventajas y desventajas del desarrollo móvil basado en HTML5:

Tabla 7. Ventajas y desventajas del desarrollo con HTML5

Ventajas	Desventajas
Fácil diseño	Incompleto acceso a las APIs del hardware
Un solo código que funciona en múltiples plataformas	Lenta en ocasiones, especialmente cuando se quiere trabajar con movimiento de imágenes, y procesar de videos.
Bajo costo de desarrollo	Es dependiente, porque requiere de un navegador web que soporte HTML5.
Instalación nativa y distribución a través de la APP Store	Puede ocasionar en los desarrolladores confusiones al desarrollar la app si no se tiene experiencia con HTML5, CSS y JavaScript

Fuente: (LanceTalent, 2014)

1.2.6 Cloud computing

a) Definición

Cloud computing es un modelo tecnológico que permite el acceso ubicuo, adaptado y bajo demanda en red a un conjunto compartido de recursos de computación configurables compartidos (por ejemplo, redes, servidores, equipos de almacenamiento, aplicaciones y servicios), que pueden ser rápidamente aprovisionados y liberados con un esfuerzo de gestión reducido o interacción mínima con el

proveedor del servicio (Observatorio Nacional de las Telecomunicaciones y de la SI, 2012).

b) Modelo de servicio

b.1. Infraestructura como Servicio (IaaS)

Es la forma que cloud computing provee recursos computacionales virtualizados utilizando internet (Rouse, Infrastructure as a Service (IaaS) definition, 2015).

En el modelo IaaS, un proveedor brinda hardware, software, servidores, almacenamiento y otros componentes de infraestructura para que el usuario lo utilice. También se incluye tareas para manejar mantenimientos y backups (respaldos de información) (Rouse, Base de datos relacional, 2015).

Además, en un ambiente IaaS se debe tener en cuenta la automatización de tareas administrativas, escalado dinámico (de acuerdo a los recursos demandados) y servicios de seguridad (Rouse, Infrastructure as a Service (IaaS) definition, 2015).

En la actualidad, los clientes pueden elegir el modo o plan de pago, pudiendo ser este de acuerdo con las horas, semanas o meses de consumo de los recursos tecnológicos. Es importante mencionar que dentro del plan de pago deben estar contenidos las características de la infraestructura tecnológica y sus respectivas tarifaciones (Observatorio Nacional de las Telecomunicaciones y de la SI, 2012).

Se muestra el cuadrante mágico para los proveedores de IaaS, según el estudio de Gartner:



Figura 8. Proveedores de IaaS

Fuente: (ZDNet, 2014)

De la figura anterior, se observa que los líderes indiscutibles según el estudio de Gartner son Amazon Web Service (AWS) y Microsoft Azure Cloud. La razón por la cual AWS es el líder es por su gran cartera de clientes, precios y estrategias agresivas y por contar con la mayor cantidad de casos de estudios.

b.2. Plataforma con Servicio (PaaS)

Es un modelo de cloud computing que ofrece aplicaciones a través de internet. Utilizando PaaS, el proveedor de cloud nos brinda hardware y herramientas de software generalmente para el desarrollo y pruebas (Rouse, Platform as a Service (PaaS) definition, 2015).

Dentro del paquete de PaaS incluye hardware instalado, sistemas operativos base de datos, servidores web, sistema de seguridad y aplicaciones para la gestión de los datos (Rouse, Platform as a Service (PaaS) definition, 2015).

Una de las grandes ventajas de utilizar PaaS es que permite el trabajo colaborativo, por tanto, varios usuarios pueden desarrollar el software de modo online (en línea) y ver sus actualizaciones en vivo (Observatorio Nacional de las Telecomunicaciones y de la SI, 2012).

Uno de los grandes riesgos de usar PaaS es la dependencia generada con una determinada tecnología (como lenguajes programación o frameworks de desarrollo). Por tanto, si algún día el proveedor decide no apoyar a acotada tecnología, los clientes tendrían que cambiar su plan de desarrollo o realizar un plan de migración (Rouse, Platform as a Service (PaaS) definition, 2015).

Cuadrante mágico de los proveedores de PaaS, según el estudio de Gartner:



Figura 9. Proveedores de PaaS

Fuente: (Salesforce, 2015)

b.3. Software como Servicio (SaaS)

Es un modelo de servicio de cloud computing donde el soporte lógico y los datos que maneja se alojan en servidores de una compañía de tecnologías de información y comunicación (TIC), a los que se accede vía Internet desde un cliente. La empresa proveedora TIC se ocupa del servicio de mantenimiento, de la operación diaria y del soporte del software usado por el cliente. Regularmente el software puede ser consultado en cualquier computador, se encuentre presente en la empresa o no. Se deduce que la información, el procesamiento, los insumos, y los resultados de la lógica de negocio del software, están hospedados en la compañía de TIC (redis, 2014).

c) Formas de implementación

c.1. Cloud público

Forma de implementación caracterizada por la oferta de servicios computación virtualizados (base de datos, sistemas operativos, plataformas de desarrollo, aplicaciones, etc) por parte de los proveedores para múltiples clientes (Observatorio Nacional de las Telecomunicaciones y de la SI, 2012).

c.2. Cloud privado

Forma de implementación caracterizada por el suministro por parte del proveedor, de entornos virtualizados que pueden ser implementados, usados y controlados por la misma empresa contratante del servicio. Esto indica no solo que la solución cloud puede ser administrada por la organización contratante, por el proveedor o por un tercer actor; sino que puede existir en las instalaciones propias del cliente o fuera de las mismas (Observatorio Nacional de las Telecomunicaciones y de la SI, 2012).

c.3. Cloud híbrido

Forma de implementación cuya infraestructura cloud (en la nube) se caracteriza por aunar dos o más formas de clouds (privado o público), los cuales continúan siendo entidades únicas interconectadas mediante tecnología estandarizada o propietaria, tecnología que permite la portabilidad de datos y aplicaciones (Observatorio Nacional de las Telecomunicaciones y de la SI, 2012).

1.2.7 Programación extrema

a) Proceso XP

La Programación Extrema (XP) utiliza un enfoque orientado a objetos como su paradigma de desarrollo preferido. La XP abarca un conjunto de reglas y prácticas que ocurren en el contexto de cuatro

actividades del marco de trabajo: planeación, diseño, codificación y pruebas (Pressman, 2006).

En la siguiente figura, se ilustra el proceso de la XP y se observa algunas de las ideas y tareas claves asociadas con cada actividad del marco de trabajo:

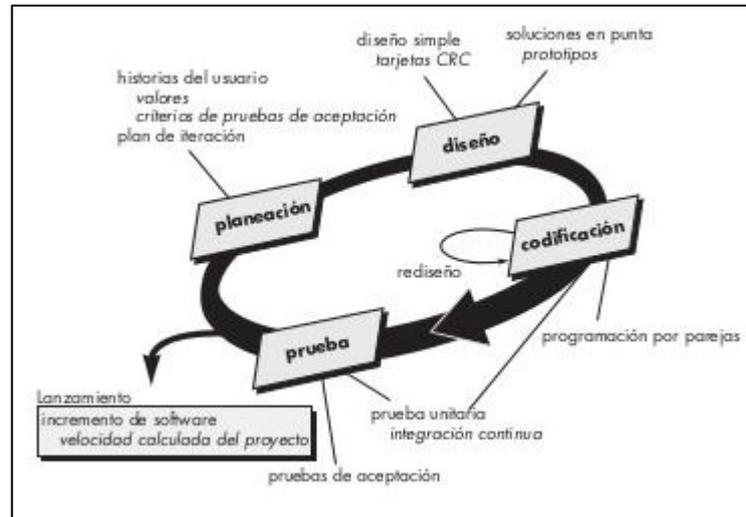


Figura 10. Proceso de la programación extrema

Fuente: (Pressman, 2006)

a.1. Planeación

La actividad de planeación comienza creando una serie de historias del usuario que describen las características y la funcionalidad requeridas para el software que se construirá.

Cada historia la escribe el cliente y se coloca en una carta índice. El cliente le asigna un valor (una prioridad) a la historia basándose en los valores generales del negocio respecto de la característica o la función (Pressman, 2006).

Los miembros del equipo de la XP evalúan entonces cada historia y le asignan un costo, el cual se mide en semanas de desarrollo. Si la historia requiere más de tres semanas de desarrollo, se le pide al cliente que la divida en historias menores, y se realiza de nuevo la asignación del valor y el costo. Es importante destacar que las nuevas historias pueden escribirse en cualquier momento (Pressman, 2006).

Los clientes y el equipo de XP trabajan juntos para decidir cómo agrupar las historias hacia el próximo lanzamiento (el siguiente incremento de software) para que el equipo de la XP las desarrolle. Una vez establecido el compromiso básico (el acuerdo de las historias que se incluirán, la fecha de entrega y otras cuestiones del proyecto) para un lanzamiento, el equipo de la XP ordena las historias que se desarrollarán de una de las siguientes tres maneras (Pressman, 2006) :

- Todas las historias serán implementadas de un modo inmediato (dentro de pocas semanas).
- Las historias con valor más alto se moverán en el programa y se implementarán al principio.
- Las historias más riesgosas se moverán dentro del programa y se implementarán al principio.

Después de que se ha entregado el primer lanzamiento del proyecto o incremento de software, el equipo de la XP calcula la velocidad de proyecto. Dicho de un modo más simple, la velocidad del proyecto es el número de historias de los clientes, implementado durante el primer lanzamiento (Pressman, 2006).

Entonces, la velocidad del proyecto puede utilizarse para:

- Ayudar a estimar fechas de entrega y el programa para lanzamiento subsecuentes.
- Determinar si se ha hecho un compromiso excesivo en algunas de las historias de todo el proyecto de desarrollo. Si se presenta un compromiso excesivo, el contenido de los lanzamientos se modifica o se cambia las fechas de las entregas finales.

Conforme avanza el trabajo de desarrollo, el cliente puede agregar historias, cambiar el valor de la historia existente, dividir historias o eliminarlas. Entonces el equipo de la XP

considera de nuevo los lanzamientos restantes y modifica sus planes de acuerdo con ello (Pressman, 2006).

a.2. Diseño

El diseño de la XP apoya el uso de tarjetas CRC como un mecanismo efecto para pensar en el software en un contexto orientado a objetos. Las tarjetas CRC (colaborador-responsabilidad-clase) identifican y organizan las clases orientadas hacia el objeto relevantes para el incremento del software actual. Las tarjetas CRC son el único producto de trabajo realizado como parte del proceso de la XP (Pressman, 2006).

Si se encuentra un problema difícil de diseño como parte del diseño de la historia, la XP recomienda la creación inmediata de un prototipo operacional de esa porción del diseño. El prototipo del diseño, llamado la solución pico, se implementa y evalúa. El propósito es reducir el riesgo cuando comience la verdadera implementación y validar las estimaciones originales en la historia que contiene el problema del diseño (Pressman, 2006).

a.3. Codificación

La XP recomienda que después de diseñar las historias el equipo no debe moverse hacia la codificación, sino que debe desarrollar una serie de pruebas de unidad que ejerciten cada una de las historias que vayan a incluirse en el lanzamiento actual (incremento de software). Una vez creada la prueba de unidad, el desarrollador es capaz de centrarse en lo que debe implementarse para pasar la prueba de unidad. No se agrega nada extraño. Una vez que el código está completo, la unidad puede probarse de inmediato, y así proporcionar una retroalimentación instantánea a los desarrolladores (Pressman, 2006).

Un concepto clave durante la actividad de codificación es la programación en pareja. La XP recomienda que dos personas trabajen juntas en una estación de trabajo de computadora para crear el código de una historia. Esto proporciona un mecanismo para la resolución de problemas en tiempo real y el aseguramiento de la calidad en las mismas condiciones. También alienta que los desarrolladores se mantengan centrados en el problema que se tiene a la mano. En la práctica, cada persona tiene un papel sutilmente diferente. Por ejemplo, una persona puede pensar en los detalles de codificación de una porción particular del diseño, mientras que la otra se asegura de que se sigan los estándares de codificación y que el código que se genera coincida con el diseño más amplio de la historia (Pressman, 2006).

Cuando los programadores completan su trabajo el código que desarrollaron se integra con el trabajo de otros. En algunos casos esto lo lleva a cabo diariamente el equipo de integración. En otros casos, la pareja de programadores es la responsable de la integración. Esta estrategia de integración continua ayuda a evitar problemas de compatibilidad e interfaz y proporciona un ambiente que ayuda a descubrir los errores desde el principio (Pressman, 2006).

a.4. Pruebas

La creación de una prueba de unidad ha sido importante antes de comenzar la codificación, esto es un elemento clave para el enfoque XP. Las pruebas de unidad que se crean deben implementarse con un marco de trabajo que permita automatizarlas (por lo tanto, pueden ejecutarse de manera fácil y repetida) (Extreme Programming, 2013).

Cuando las unidades individuales de prueba se organizan en un conjunto universal de pruebas, las pruebas de integración y validación del sistema pueden realizarse a diario. Esto

proporciona al equipo de XP una indicación continua del progreso y también puede encender luces de emergencia previas si las cosas salen mal (Pressman, 2006).

Las pruebas de aceptación de la XP, también llamadas pruebas del cliente, las especifica el cliente y se enfocan en las características generales y la funcionalidad del sistema, elementos visibles y revisables por el cliente. Las pruebas de aceptación se derivan de las historias de usuario que han implementado como parte de un lanzamiento de software (Pressman, 2006).

b) Reglas de XP

Según la documentación oficial de Programación Extrema (Extreme Programming, 2013) por cada etapa existen reglas.

b.1. Planeación

Las reglas en la planeación son:

- Las historias de usuario se escriben.
- La planificación de lanzamiento crea el calendario de lanzamientos.
- Hacer pequeños lanzamientos frecuentes.
- El proyecto se divide en iteraciones.
- La planificación de la iteración comienza en cada iteración.

b.2. Diseño

Las reglas en el diseño son:

- Simplicidad.
- Utilice tarjetas CRC en la sesión del diseño.
- Crea soluciones pico para reducir el riesgo.
- Utiliza la refabricación cuando sea posible.

b.3. Codificación

Las reglas en la codificación son:

- El cliente siempre está disponible.
- El código debe ser escrito con las normas acordadas.
- Realiza primero la prueba de unidad.
- Todo el código debe producirse en parejas.
- Integra a menudo.
- Establece un equipo de integración dedicado.

b.4. Pruebas

Las reglas en las pruebas son:

- Todo el código debe tener pruebas unitarias.
- Todo el código debe pasar todas las pruebas unitarias antes de que sea lanzado.
- Cuando se encuentra un error se crean pruebas.
- Las pruebas de aceptación se ejecutan a menudo y la puntuación se publica.

b.5. Gestión

Las reglas en la gestión son:

- Dar al equipo un espacio de trabajo abierto dedicado.
- Inicia una reunión de trabajo cada día.
- La velocidad del proyecto se mide.
- Rota a las personas a tu alrededor.
- Arregla XP, de ser necesario cambia algunas de las reglas de XP si en caso no funcionara.

1.2.8 PMBOK

a) Breve descripción

La guía del PMBOK es un estándar en la Administración de proyectos desarrollado por el Project Management Institute (PMI). Define la dirección de proyectos y otros conceptos relacionados, y describe el ciclo de vida de la dirección de proyectos y los procesos conexos. El propósito de la guía del PMBOK es la aplicación de

conocimientos, procesos, habilidades, herramientas y técnicas a las actividades de un proyecto para satisfacer los requisitos del proyecto. También proporciona y promueve un vocabulario común en el ámbito de la profesión de la dirección de proyectos (Project Management Institute, 2013).

b) Procesos de Dirección de Proyectos

Los procesos de la dirección de proyectos comunes a la mayoría de los proyectos, por lo general, están relacionados entre sí por el hecho de que se llevan a cabo para un propósito integrado. El propósito es iniciar, planificar, ejecutar, supervisar y controlar, y cerrar un proyecto (Project Management Institute, 2013).

La siguiente figura permite visualizar las interacciones entre procesos.

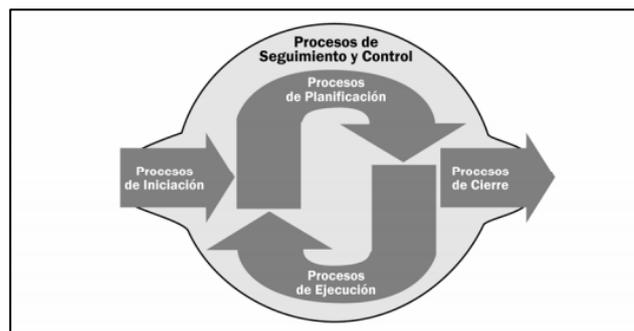


Figura 11. Grupos de Procesos de Dirección de Proyecto

Fuente: (Project Management Institute, 2013)

De la figura anterior, los cinco Grupos de Procesos son:

- Grupo de Procesos de Iniciación: Define y autoriza el proyecto o una fase del mismo (Project Management Institute, 2013).
- Grupo de Procesos de Planificación: Define y refina los objetivos, y planifica el curso de acción requerido para lograr los objetivos y el alcance pretendido del proyecto (Project Management Institute, 2013).
- Grupo de Procesos de Ejecución: Integra a personas y otros recursos para llevar a cabo el plan de gestión del proyecto.

- Grupo de Procesos de Seguimiento y Control: Mide y supervisa regularmente el avance, a fin de identificar las variaciones respecto del plan de gestión del proyecto, de tal forma que se tome medidas correctivas cuando sea para cumplir con los objetivos del proyecto (Project Management Institute, 2013).
- Grupo de Procesos de Cierre: Formaliza la aceptación del producto, servicio o resultado, y termina ordenando el proyecto o una fase del mismo (Project Management Institute, 2013).

En la siguiente figura, se muestra de una forma más detalla la interacción entre los Grupos de Procesos de Dirección de Proyecto:

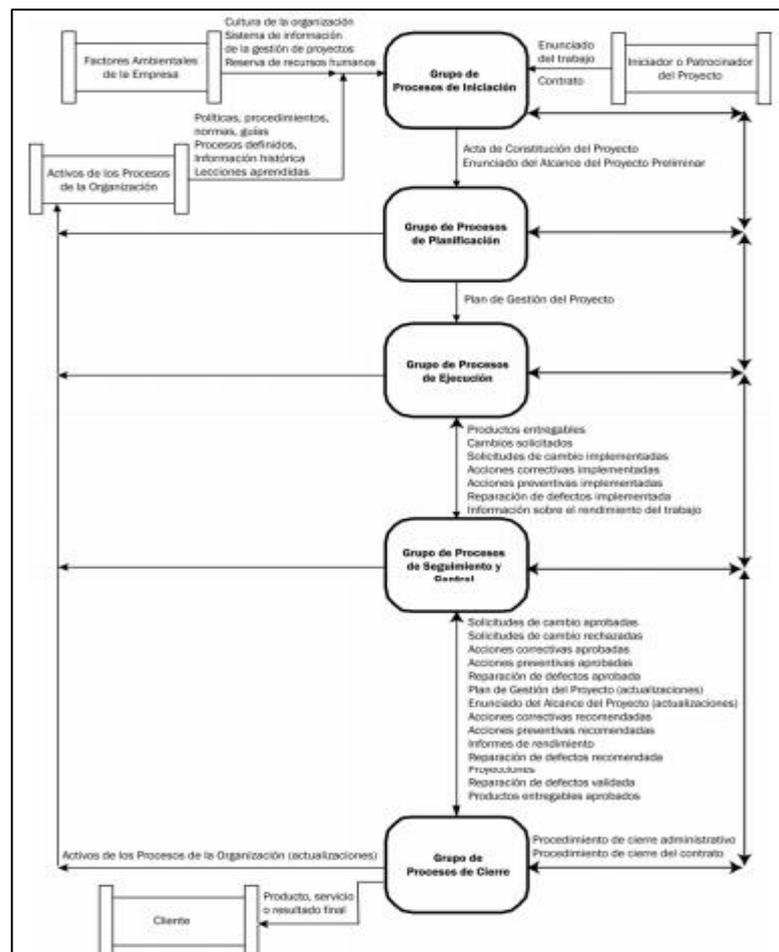


Figura 12. Interacción detallada entre Grupos de Procesos de dirección de Proyecto

Fuente: (Project Management Institute, 2013)

c) Áreas de conocimiento

Las áreas de conocimiento definidas en el PMBOK son:

- **Gestión del Alcance del Proyecto:** Describe los procesos necesarios para asegurarse de que el proyecto incluya todo el trabajo requerido para completarse satisfactoriamente. Se compone de los procesos: Planificación del Alcance, Definición del Alcance, Crear EDT, Verificación del Alcance y Control del Alcance (Morales, 2014).
- **Gestión del Tiempo del Proyecto:** Describe los procesos relativos a la puntualidad en la conclusión del proyecto. Se compone de los procesos: Definición de las Actividades, Establecimiento de la Secuencia de las Actividades, Estimación de Recursos de las Actividades, Estimación de la Duración de las Actividades, Desarrollo del Cronograma y Control del Cronograma (Morales, 2014).
- **Gestión de los Costes del Proyecto:** Describe los procesos involucrados en la planificación, estimación, presupuesto y control de costes de forma que el proyecto se complete dentro del presupuesto aprobado. Se compone de los procesos: Estimación de Costes, Preparación del Presupuesto de Costes y Control de Costes (Morales, 2014).
- **Gestión de la Calidad del Proyecto:** Describe los procesos necesarios para asegurarse de que el proyecto cumpla con los objetivos para los cuales ha sido emprendido. Se compone de los procesos: Planificación de Calidad, Realizar Aseguramiento de Calidad y Realizar Control de Calidad (Morales, 2014).
- **Gestión de los Recursos Humanos del Proyecto:** Describe los procesos para organizar y dirigir los Recursos Humanos del proyecto. Se compone de los procesos: Planificación de los Recursos Humanos, Adquirir el Equipo del Proyecto,

Desarrollar el Equipo del Proyecto y Gestionar el Equipo del Proyecto (Morales, 2014).

- **Gestión de las Comunicaciones del Proyecto:** Describe los procesos relacionados con la generación, distribución almacenamiento y destino final de la información del proyecto en tiempo y forma. Se compone de los procesos: Planificación de las Comunicaciones, Distribución de la Información, Informar el Rendimiento y Gestionar a los Interesados (Morales, 2014).
- **Gestión de la Integración del Proyecto:** Describe los procesos y actividades que forman parte de los diversos elementos de la Dirección de Proyectos, que se identifican, definen, combinan, unen y coordinan dentro de los Grupos de Procesos de Dirección de Proyectos. Se compone de los procesos: Desarrollar el Acta de Constitución del Proyecto, Desarrollar el Enunciado del Alcance del Proyecto Preliminar, Desarrollar el Plan de Gestión del Proyecto, Dirigir y Gestionar la Ejecución del Proyecto, Supervisar y Controlar el Trabajo del Proyecto, Control Integrado de Cambios y Cerrar Proyecto. **Gestión de los riesgos:** Incluye los procesos relacionados con llevar a cabo la planificación de la gestión, identificación, el análisis, la planificación de respuesta a los riesgos, así como su monitoreo y control en un proyecto (Morales, 2014).
- **Gestión de los Riesgos del Proyecto:** Describe los procesos relacionados con el desarrollo de la gestión de riesgos de un proyecto. Se compone de los procesos: Planificación de la Gestión de Riesgos, Identificación de Riesgos, Análisis Cualitativo de Riesgos, Análisis Cuantitativo de Riesgos, Planificación de la Respuesta a los Riesgos, y Seguimiento y Control de Riesgos (Morales, 2014).

- **Gestión de las Adquisiciones del Proyecto:** Describe los procesos para comprar o adquirir productos, servicios o resultados, así como para contratar procesos de dirección. Se compone de los procesos: Planificar las compras y adquisiciones, Planificar la contratación, Solicitar respuestas de vendedores, Selección de vendedores, Administración del contrato y cierre del contrato (Morales, 2014).

En la siguiente figura, se muestran los procesos que se deben hacer en cada área de conocimiento:

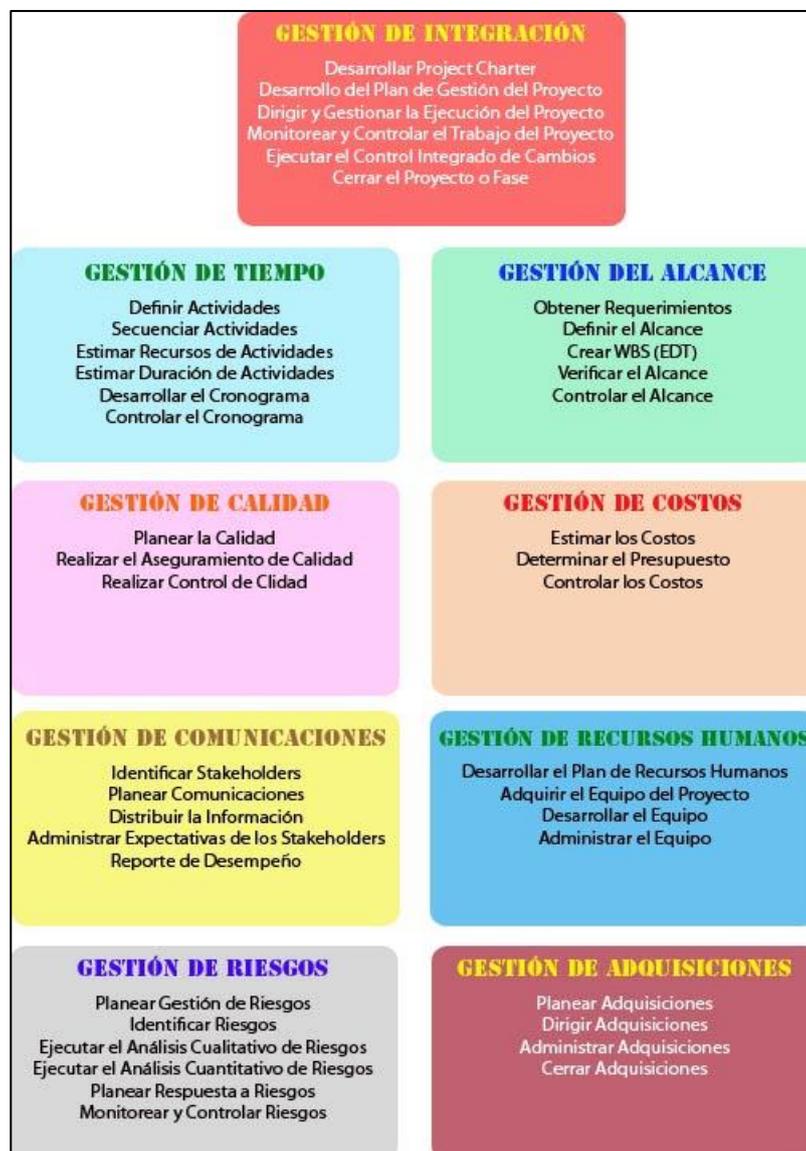


Figura 13. Áreas de Conocimiento de la Gerencia de Proyectos

Fuente: (Morales, 2014)

1.2.9 Marco lógico

a) Descripción

La metodología de Marco lógico es una herramienta para facilitar el proceso de conceptualización, diseño, ejecución y evaluación de proyectos (Ortegón, Pacheco, & Prieto, 2006).

La metodología fue elaborada, originalmente, como respuesta a tres problemas comunes a proyectos (Ortegón, Pacheco, & Prieto, 2006):

- Planificación de proyectos carentes de precisión, con objetivos múltiples que no estaban claramente relacionados con las actividades del proyecto.
- Proyectos que no se ejecutaban exitosamente, y el alcance de la responsabilidad del gerente del proyecto no estaba claramente definida.
- No existía una imagen clara de cómo luciría el proyecto si tuviese éxito, y los evaluadores no tenían una base objetiva para comparar lo que se planeaba con lo que sucedía en la realidad.

La metodología del Marco lógico encara estos problemas, y provee, además, una cantidad de ventajas sobre enfoques menos estructurados (Ortegón, Pacheco, & Prieto, 2006):

- Aporta una terminología uniforme que facilita la comunicación y que sirve para reducir ambigüedades.
- Aporta un formato para llegar a acuerdos precisos acerca de los objetivos, metas y riesgos del proyecto que comparten los diferentes actores relacionados con el proyecto.
- Suministra un temario analítico común que pueden utilizar los involucrados, los consultores y el equipo de proyecto para elaborar tanto el proyecto como el informe de proyecto, como también para la interpretación de este.

- Enfoca el trabajo técnico en los aspectos críticos y puede acortar documentos de proyecto en forma considerable.
- Suministra información para organizar y preparar en forma lógica el plan de ejecución del proyecto.
- Suministra información necesaria para la ejecución, monitoreo y evaluación del proyecto.
- Proporciona una estructura para expresar, en un solo cuadro, la información más importante sobre un proyecto.

Es importante mencionar que hay una diferencia entre la metodología de Marco lógico y la Matriz de marco lógico. La metodología contempla análisis del problema, análisis de los involucrados, jerarquía de objetos y selección de una estrategia de implementación óptima. El producto de la metodología es la Matriz del marco lógico, la cual resume lo que el proyecto pretende hacer y cómo, cuáles son los supuestos claves y cómo los insumos y productos del proyecto serán monitoreados y evaluados.

Las etapas de esta metodología son:

- Identificación del problema y alternativas de solución.
- Matriz de (planificación) Marco lógico.

b) Identificación del problema y alternativas de solución

El proceso de planificación nace con la percepción de una situación problemática y la motivación para solucionarla (Ortegón, Pacheco, & Prieto, 2006).

b.1. Análisis de involucrados

Es importante estudiar a cualquier persona o grupo, institución o empresa susceptible de tener un vínculo con un proyecto dado. El análisis de involucrados permite optimizar los beneficios sociales e institucionales del proyecto y limitar los impactos negativos. Al analizar sus intereses y expectativas se puede aprovechar y potenciar el apoyo de aquellos con

intereses coincidentes o complementarios al proyecto, disminuir la oposición de aquellos con intereses opuestos al proyecto y conseguir el apoyo de los indiferentes (Ortegón, Pacheco, & Prieto, 2006). El análisis de involucrados implica:

- Identificar todos aquellos que pudieran tener interés o que se pudieran beneficiar directa e indirectamente (pueden estar en varios niveles, por ejemplo, local, regional, nacional).
- Investigar sus roles, intereses, poder relativo y capacidad de participación.
- Identificar su posición, de cooperación o conflicto, frente al proyecto y entre ellos y diseñar estrategias con relación a dichos conflictos.
- Interpretar los resultados del análisis y definir cómo pueden ser incorporados en el diseño del proyecto.

b.2. Análisis del problema

Al preparar un proyecto, es necesario identificar el problema que se desea intervenir, así como sus causas y sus efectos (Ortegón, Pacheco, & Prieto, 2006). El procedimiento contempla los siguientes pasos:

- Analizar e identificar lo que se considere como problemas principales de la situación a aborda.
- A partir de una primera lluvia de ideas establecer el problema central que afecta a la comunidad, aplicando criterios de prioridad y selectividad.
- Definir los efectos más importantes del problema en cuestión, de esta forma se analiza y verifica su importancia.

- Anotar las causas del problema central detectado. Esto significa buscar qué elementos están o podrían estar provocando el problema.
- Una vez que tanto el problema central como las causas y los efectos están identificados, se construye el árbol de problemas. El árbol de problemas brinda una imagen completa de la situación negativa existente.
- Revisar la validez e integridad del árbol dibujado, todas las veces que sea necesarias. Esto es, asegurarse que las causas representen causas y los efectos representen efectos, que el problema central esté correctamente definido y que las relaciones (causales) estén correctamente expresadas.

En la siguiente figura, se muestra la estructura del árbol de problemas:

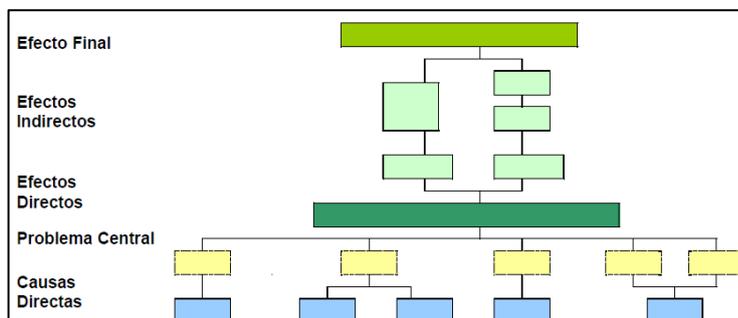


Figura 14. Estructura del Árbol de Problemas

Fuente: (Choy, Marco Lógico - Análisis de Problemas, 2014)

b.3. Análisis de objetivos

El análisis de los objetivos permite describir la situación futura a la que se desea llegar una vez que se han resuelto los problemas. Consiste en convertir los estados negativos del árbol de problemas en soluciones, expresadas en forma de estados positivos. De hecho, todos esos estados positivos son objetivos y se presentan en un diagrama de objetivos en el que se observa la jerarquía de los medios y de los fines.

Este diagrama permite tener una visión global y clara de la situación positiva que se desea. Una vez que se ha construido el árbol de objetivos es necesario examinar las relaciones de medios y fines que se han establecido para garantizar la validez e integridad del esquema de análisis. Si al revelar el árbol de causas y efectos se determinan inconsistencias es necesario volver a revisarlo para detectar las fallas que se puedan haber producido. Si se estima necesario, y siempre teniendo presente que el método debe ser todo lo flexible que sea necesario, se deben modificar las formulaciones que no se consideren correctas, se deben agregar nuevos objetivos que se consideren relevantes y que no estaban incluidos y se deben eliminar aquellos que no eran efectivos (Choy, Marco Lógico - Análisis de Objetivos, 2014).

En la siguiente figura, se muestra la estructura de un árbol de objetivos:

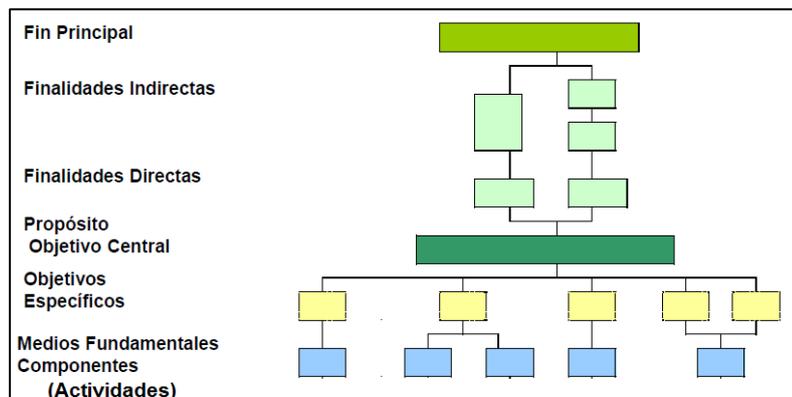


Figura 15. Estructura del Árbol de Objetivos

Fuente: (Choy, Marco Lógico - Análisis de Objetivos, 2014)

b.4. Identificación de alternativas de solución al problema

Consiste en identificar las posibles soluciones del problema, para ello se debe proponer acciones que puedan en términos operativos conseguir el medio. Es importante mencionar que si se consiguen los medios más bajos se soluciona el problema, que es lo mismo que decir que si se elimina las causas más

profundas se estará eliminado el problema. (Ortegón, Pacheco, & Prieto, 2006)

b.5. Selección de la alternativa óptima

Este análisis consiste en la selección de una alternativa que se aplicará para alcanzar los objetivos deseados (Ortegón, Pacheco, & Prieto, 2006).

Para seleccionar una alternativa se evalúan y comparan entre las identificadas como posibles soluciones al problema, para ello se realizan diferentes análisis como:

- Diagnóstico de la situación (área de estudio, área de influencia, población objetivo, demanda, oferta y déficit).
- Estudio técnico de cada alternativa (tamaño, localización, tecnología).
- Análisis de los costos de las actividades que cada alternativa demanda.
- Análisis de los beneficios.
- Se hace una comparación a través de algunos criterios e indicadores y de esta comparación se toma la que muestra los mejores resultados.

b.6. Estructura analítica del proyecto (EAP)

Teniendo ya seleccionada una alternativa, previo a la construcción de la Matriz de marco lógico es recomendable construir la EAP (Estructura Analítica del Proyecto) para establecer niveles jerárquicos, como el fin, el objetivo central del proyecto (propósito), los componentes (productos) y las actividades. Definido esto, se podrá construir la Matriz. Esto debido a la necesidad de ajustar el análisis de selección de la alternativa (estrategia) óptima y expresarla en una matriz que la resuma (Ortegón, Pacheco, & Prieto, 2006).

La EAP es la esquematización del proyecto. Dicho de otra manera, la EAP es un esquema de la alternativa de solución más viable expresada en sus rasgos más generales a la manera de un árbol de objetivos y actividades (Ortegón, Pacheco, & Prieto, 2006).

Se muestra una figura, en la cual se visualiza la EAP:

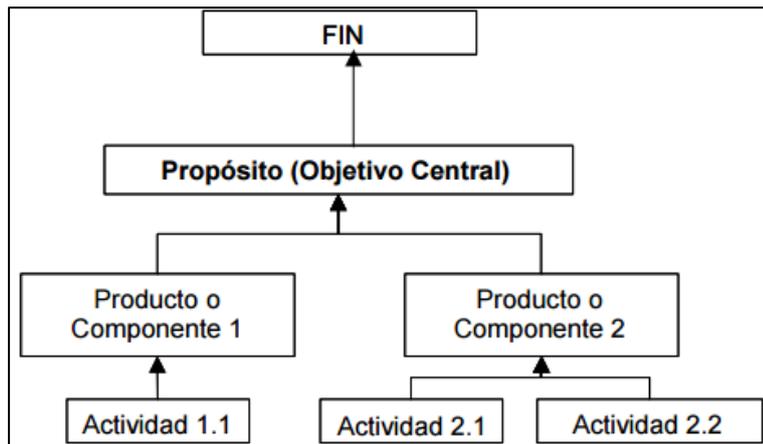


Figura 16. Estructura Analítica del Proyecto (EAP)

Fuente: (Ortegón, Pacheco, & Prieto, 2006)

c) Matriz de (planeación) marco lógico

La Matriz de marco lógico presenta, en forma resumida, los aspectos más importantes del proyecto (Ortegón, Pacheco, & Prieto, 2006). Posee cuatro columnas que suministran la siguiente información:

- Un resumen narrativo de los objetivos y las actividades.
- Indicadores (Resultados específicos por alcanzar)
- Medios de verificación (lugar donde se pueden encontrar los indicadores).
- Supuestos (factores externos que implican riesgos).

Y cuatro filas que presentan información acerca de los objetivos, indicadores, medios de verificación y supuestos en cuatro momentos diferentes en la vida del proyecto (Ortegón, Pacheco, & Prieto, 2006):

- Fin al cual el proyecto contribuye de manera significativa luego de que el proyecto ha estado en funcionamiento.
- Propósito logrado cuando el proyecto ha sido ejecutado.
- Componentes/Resultados completados en el transcurso de la ejecución del proyecto.
- Actividades requeridas para producir los Componentes/Resultados.

Estructura de la Matriz de marco lógico:

Objetivos	Indicadores verificables objetivamente	Medios de verificación	Supuestos/ riesgos importantes
FIN (OBJETIVO DE DESARROLLO)			
PROPOSITO DEL PROYECTO			
PRODUCTOS			
ACTIVIDADES	Especificaciones de insumos/costos		

Figura 17. Estructura de la Matriz de Marco Lógico

Fuente: (Choy, Marco Lógico - Análisis de Objetivos, 2014)

1.2.10 Ruby on Rails

Ruby on Rails es un framework para desarrollar aplicaciones web escrito en el lenguaje Ruby. Está diseñado para hacer más fácil la programación de las aplicaciones web partiendo de lo que necesitan los desarrolladores. Además, permite escribir menos código a comparación de muchos otros lenguajes y frameworks (Rails Guides, 2015).

La filosofía Rails incluye dos principios:

- Don't Repeat Yourself (DRY): DRY es un principio de desarrollo de software que especifica que no se debe escribir más de una vez la misma información (el mismo

código), por tanto lo que ya está hecho no tiene por qué volver a hacerse (Rails Guides, 2015).

- Convenciones sobre configuraciones: Este principio trata acerca de que Rails ya viene con ciertas configuraciones que funcionan por defecto al desarrollar y ejecutar una aplicación, permitiendo a los programadores no preocuparse por esa labor y solo concentrarse en la lógica del negocio. (Rails Guides, 2015).

Rails maneja el paradigma MVC (Modelo-Vista-Controlador), por tanto, utiliza controladores como medio de interacción entre las vistas y el modelo, realiza consultas a la base de datos con el Active Record (ORM) que forma parte del modelo, y sirve la información a través de vistas en diferentes formatos como HTML, JSON, XML, etc. (HTML5FÁCIL, 2013).

1.2.11 PhoneGap

PhoneGap es un framework para el desarrollo de aplicaciones móviles multiplataforma. Phonegap permite a los programadores desarrollar aplicaciones móviles para dispositivos móviles utilizando tecnologías como JavaScript, HTML5 y CSS3. Las aplicaciones resultantes son híbridas, es decir, no son realmente aplicaciones nativas al dispositivo (ya que el renderizado se realiza mediante vistas web y no con interfaces gráficas específicas de cada sistema operativo), pero no se tratan tampoco de aplicaciones web (teniendo en cuenta que son aplicaciones que son empaquetadas para poder ser desplegadas en el dispositivo incluso trabajando con el API del sistema nativo) (Adobe PhoneGap, 2015).

PhoneGap maneja una API que permiten tener acceso a elementos (al hardware) del dispositivo (Adobe PhoneGap, 2015). Luego se muestran las características que soporta por cada sistema operativo de los dispositivos móviles:

	iPhone / iPhone 3G	iPhone 3GS and newer	Android	BlackBerry OS 6.0+	BlackBerry 10	Windows Phone 8	Ubuntu	Firefox OS
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓
Compass	X	✓	✓	X	✓	✓	✓	✓
Contacts	✓	✓	✓	✓	✓	✓	✓	✓
File	✓	✓	✓	✓	✓	✓	✓	X
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓
Media	✓	✓	✓	X	✓	✓	✓	X
Network	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Alert)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Sound)	✓	✓	✓	✓	✓	✓	✓	✓
Notification (Vibration)	✓	✓	✓	✓	✓	✓	✓	✓
Storage	✓	✓	✓	✓	✓	✓	✓	✓

Figura 18. API soportadas por cada sistema operativo móvil

Fuente: (PhoneGap, 2015)

1.2.12 Ionic

Ionic es un poderoso framework de diseño para desarrollar aplicaciones móviles multiplataforma basadas en HTML5, que utiliza tecnologías como HTML5, CSS3 y JavaScript (Ionic, 2015).

Ionic se centra principalmente en la apariencia y en la interacción de interfaz de usuario de la aplicación, por tanto, no es un reemplazo para PhoneGap sino ambos se complementan (Ionic, 2015).

Ionic trabaja con AngularJS, este último potencia la forma de trabajar y manejar los elementos de la vista (Ionic, 2015).

1.3 Glosario técnico

API: Significa Application Programming Interfaces y es un conjunto de reglas (código) y especificaciones que las aplicaciones pueden seguir para comunicarse entre ellas: sirviendo de interfaz entre programas diferentes de la misma manera en que la interfaz de usuario facilita la interacción humano-software.

Aplicación móvil: Es un programa que se puede descargar e instalar y al que puede se puede acceder directamente desde el propio teléfono o desde algún otro aparato móvil.

Aplicación móvil híbrida: Es una aplicación escrita en el mismo lenguaje de las aplicaciones web, o websites, que está alojada en un contenedor nativo, en un dispositivo móvil. Es, entonces, la unión entre la tecnología web y la ejecución nativa., una aplicación móvil que se parece y se comporta como una aplicación móvil nativa.

Base de datos relacional: Una base de datos relacional es una colección de elementos de datos organizados en un conjunto de tablas formalmente descritas desde la que se puede acceder a los datos o volver a montarlos de muchas maneras diferentes sin tener que reorganizar las tablas de la base.

Back-end: Son todas las tecnologías que se encuentran del lado del servidor, que se encargan de interactuar con bases de datos, verificar manejos de sesiones de usuarios, montar la página en un servidor, y desde este “servir” todas las vistas que el Front-End crea.

Browser: También conocido como navegador web, es un tipo de software que permite la visualización de documentos y sitios en hipertexto, comúnmente agrupados bajo la denominación de Web o Internet.

CLI: Significa interfaz de línea de comando, la cual permite a los usuarios dar instrucciones a algún programa por medio de una línea de texto.

CSS3 media queries: Son un conjunto de expresiones que dejan que la presentación del contenido se adapte a un rango específico de dispositivos de salida sin tener que cambiar el contenido en sí.

Cloud computing: Es un modelo tecnológico que permite el acceso ubicuo, adaptado y bajo demanda en red a un conjunto compartido de recursos de computación configurables compartidos (por ejemplo: redes, servidores, equipos de almacenamiento, aplicaciones y servicios), que pueden ser rápidamente aprovisionados y liberados con un esfuerzo de gestión reducido o interacción mínima con el proveedor del servicio.

Cloud privado: Es aquel en la que solamente una organización, utilizando tecnologías como la virtualización, tiene acceso a los recursos que se utilizan para implementar la nube. Es decir, una empresa dispone de un entorno cloud en exclusiva.

Cloud público: Es un modelo de computación en la nube que se caracteriza por ofrecer recursos tecnológicos sobre infraestructuras compartidas entre múltiples clientes. A estos recursos el cliente accede a través de internet o mediante conexiones VPN. La infraestructura es proporcionada con todas las ventajas del modelo de consumo de cloud (pago por uso, aprovisionamiento ágil, elasticidad, etc.).

Cluster: Es una agrupación de empresas e instituciones relacionadas entre sí, pertenecientes a un mismo sector o segmento de mercado, que se encuentran próximas geográficamente y que colaboran para ser más competitivos

Dispositivo móvil: Es un aparato que disfruta de autonomía de movimiento y está libre de cableado, con algunas capacidades de procesamiento, con conexión permanente o intermitente a una red, con memoria limitada, que ha sido diseñado específicamente para una función, pero que puede llevar a cabo otras funciones más generales

Editor de texto: Los editores de texto son programas informáticos que crean y editan archivos digitales en un formato de texto básico o texto plano, es decir, archivos que no contengan formato de texto específico.

Facilidad de uso: Es la medida que permite evaluar la eficacia en la consulta de información de un sitio web. La facilidad de uso impacta la arquitectura del sitio, el diseño de interfaz de usuario y el desarrollo de contenidos.

Framework: Es una estructura software compuesta de componentes personalizables e intercambiables para el desarrollo de una aplicación. En otras palabras, un framework se puede considerar como una aplicación genérica incompleta y configurable a la que podemos añadirle las últimas piezas para construir una aplicación concreta.

Front-End: Son todas aquellas tecnologías que corren del lado del cliente, es decir todas aquellas tecnologías que corren del lado del navegador web, generalizándose más que nada en tres lenguajes: HTML, CSS y JavaScript.

IaaS: Las siglas significan Infraestructura como Servicio y es un tipo de cloud computing que consiste en poner a disposición del cliente el uso de la infraestructura informática (capacidad de computación, espacio de disco y bases de datos entre otros) como un servicio.

Instancia de cloud computing: Es cada servidor virtual iniciado por un período de tiempo determinado. Es decir, la instancia es un servidor (máquina virtual) que, por lo general, se utiliza estratégicamente de forma provisional, en lugar de instalarlo definitivamente.

Lenguaje de programación: es un idioma artificial diseñado para expresar computaciones que pueden ser llevadas a cabo por máquinas como las computadoras. Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina para expresar algoritmos con precisión o como modo de comunicación humana. Está formado por un conjunto de símbolos y reglas sintácticas

y semánticas que definen su estructura y el significado de sus elementos y expresiones

Mipyme(s): Siglas que significan micro, pequeña y mediana empresa.

Open-source: Significa código abierto y se aplica al Software distribuido bajo una licencia que le permita al usuario acceso al código fuente del Software, y además le permita estudiar y modificarlo con toda libertad, sin restricciones en el uso del mismo; y además le permita redistribuirlo, siempre y cuando sea de acuerdo con los términos de la licencia bajo la cual el Software original fue adquirido.

PaaS: Las siglas significan Plataforma como Servicio y es un tipo de cloud computing que consiste en la entrega como un servicio, de un conjunto de plataformas informáticas orientadas al desarrollo, testeo, despliegue, hosting y mantenimiento de los sistemas operativos y aplicaciones propias del cliente.

ORM: Significa Object Relation Mapping y consiste en una serie de objetos que permiten acceder a los datos y que contienen en su interior cierta lógica de negocio, sirviendo como interfaz para utilizar operaciones y consultas en la base de datos. Las ORM se crean a partir de la tendencia del uso de lenguajes de programación orientados a objetos, teniendo en cuenta que las bases de datos siguen una estructura relacional, en tal sentido es necesario acceder a la base de datos como si fuera orientada a objetos para simplificar el código.

PRI: Las siglas significa Periodo de retorno de inversión y es un instrumento que permite medir el plazo de tiempo que se requiere para que los flujos netos de efectivo de una inversión recuperen su costo o inversión inicial.

Renderizar: Es el proceso de generar una vista al usuario desde el modelo.

Responsivo: Es una técnica que permite que contenidos web se puedan adaptar a un dispositivo móvil, mostrando la información según sea necesario.

Servidor: Es un ordenador o máquina informática que está al “servicio” de otras máquinas, ordenadores o personas llamadas clientes y que le suministran a estos, todo tipo de información.

Servidor web: Es un programa que gestiona cualquier aplicación en el lado del servidor realizando conexiones bidireccionales y/o unidireccionales y síncronas o asíncronas con el cliente que genera una respuesta en cualquier lenguaje o aplicación en el lado del cliente.

Sistema de información: Es un conjunto de elementos que interactúan entre sí con un fin común; que permite que la información esté disponible para satisfacer las necesidades en una organización, un sistema de información no siempre requiere contar con recurso computacional, aunque la disposición del mismo facilita el manejo e interpretación de la información por los usuarios.

SGDB: Las siglas significan Sistema Gestor de Base de Datos, y es una colección de programas cuyo objetivo es servir de interfaz entre la base de datos, el usuario y las aplicaciones. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

Software web: Es un conjunto de programas, instrucciones y reglas informáticas que permite ejecutar distintas tareas en una computadora al que se accede usando internet y un navegador web.

TIR: Las siglas significan Tasa interna de retorno y es la tasa de descuento de un proyecto de inversión que permite que el VAN sea cero. La TIR es la máxima tasa de descuento que puede tener un proyecto para que sea rentable.

Usabilidad: Es la medida de la calidad de la experiencia que tiene un usuario cuando interactúa con un producto o sistema.

VAN: Las siglas significan Valor actual neto y es un indicador financiero que mide los flujos de los futuros ingresos y egresos que tendrá un proyecto para determinar si luego de descontar la inversión inicial, nos quedaría alguna ganancia.

CAPÍTULO II METODOLOGÍA

2.1 Material y métodos

2.1.1 Material

Se exponen los materiales utilizados durante el desarrollo de la tesis:

Tabla 8. Materiales usados en la ejecución de la tesis

Herramienta de Gestión de Proyectos		
Software	Versión	Descripción
MS Project	2013	Herramienta de Microsoft Office que permite controlar los tiempos y actividades del proyecto
Herramientas de Desarrollo		
Tecnología	Versión	Descripción
Ruby	2.1	Lenguaje de programación interpretado para la rápida y fácil programación orientada a objetos.
Ruby on Rails	4.1	Framework de desarrollo que lo utilizaremos para desarrollar el software web, está escrito en el lenguaje Ruby.

AngularJS	1.3.15	Framework de JavaScript para el frontend que lo utilizaremos para el desarrollo de la aplicación móvil híbrida.
Bower	1.4.1	Gestor de paquetes que lo usaremos para instalar y mantener las dependencias del software web.
Node JS	0.12.2	Plataforma escrita en JavaScript que lo utilizaremos para descargar aplicaciones necesarias (como Bower) en el desarrollo del software web y construcción de la aplicación móvil.
Apache Cordova	3.7.2	Plataforma que lo usaremos para la construcción de aplicaciones móviles usando HTML, CSS y JavaScript.
Ionic	1.3.22	Framework de desarrollo que utilizaremos para la construcción de aplicaciones móviles híbridas.
Sublime Text	3	Editor de texto que usaremos para la programación.
SQLite	3.8.7.2	Motor de Base de Datos embebido que lo utilizaremos para el ambiente de desarrollo y pruebas.
PostgreSQL	9.4.4	Base de Datos relacional open-source que utilizaremos para el ambiente de producción.
Herramientas de Documentación		
Software	Versión	Descripción
MS Word	2013	Herramienta de Microsoft Office que nos permitirá crear documentos de texto requeridos para la gestión y ejecución del proyecto y producto
MS Excel	2013	Herramienta de Microsoft Office que nos permitirá crear documentos de cálculo y diagramas requeridos para la planificación y ejecución del proyecto y producto.
MS PowerPoint	2013	Herramienta de Microsoft Office que nos permitirá crear las presentaciones (como el alcance, descripción del producto, etc)

		realizadas en toda la ejecución del proyecto.
Gliffy	Libre	Herramienta para construir diagramas y almacenarlos en la nube.
Herramienta de Comunicación		
Software	Versión	Descripción
Hotmail	Libre	Utilizaremos Hotmail para la comunicación con Correos electrónico.
Infraestructura		
Equipo	Descripción	
Computadoras personales	<p>Utilizaremos las computadoras de uso diario.</p> <p>Requerimientos mínimos:</p> <ul style="list-style-type: none"> • Procesador Intel core i5 • Memoria RAM 8 GB. • Disco duro 500 GB 	
Servidor de Desarrollo	<p>Utilizaremos como servidor, la computadora personal con mejores características en hardware. Requerimientos mínimos:</p> <ul style="list-style-type: none"> • Procesador Intel core i7 • Memoria RAM 12 GB. • Disco duro 1TB 	
Servidor de Producción (Heroku)	<p>Utilizaremos la nube para desplegar el producto, para ello utilizaremos como proveedor de cloud Computing a Heroku, el cual nos facilita el despliegue del software desarrollado en Ruby on Rails y además por basarse en el modelo PaaS nos permite utilizar sus propias herramientas de desarrollo y prueba.</p>	
Dispositivos móviles	<p>Utilizaremos los celulares de uso diario.</p> <p>Requerimientos mínimos:</p> <ul style="list-style-type: none"> • Memoria RAM DE 512 MB • Disco duro de 2 GB 	

Elaboración: Los autores

Para conocer los detalles de la selección de la base de datos en producción, véase el Anexo N° 17 “Selección de Sistema de Gestor de Base de Datos”.

En cuanto a la selección del proveedor Heroku, véase el Anexo N° 18 “Selección de proveedor de PaaS de cloud computing”.

Cabe señalar que se eligió el modelo PaaS sobre el IaaS, porque el equipo de trabajo desea concentrarse más en la parte de desarrollo que en los detalles de la infraestructura tecnológica, además desea tener herramientas integradas a la instancia para realizar el despliegue y las pruebas para acelerar dichas actividades.

2.1.2 Método

La metodología que se usó en el desarrollo del sistema de información de servicio de post-venta al cliente para empresas metalmeccánicas, está basada en la metodología Programación Extrema (XP), la cual es de tipo ágil y consta de cuatro (4) actividades: Planeación, Diseño, Codificación y Prueba. Para más detalle acerca de la selección de la metodología, véase el Anexo N° 19 “Selección de la metodología de desarrollo de software”.

Para el desarrollo del producto se decidió incluir las siguientes seis (6) fases: Exploración, Planificación del Incremento, Codificación y Pruebas, Producción, Mantenimiento y Cierre del Proyecto.

Luego se muestra una figura que gráfica la metodología utilizada con sus fases, involucrados, productos y herramientas tecnológicas necesarias:

	FASES					
	Exploración	Planificación del Incremento	Codificación y Pruebas	Producción	Mantenimiento	Cierre del Proyecto
Involucrados						
Productos	Historias de Usuario	Historias de Usuario refinadas Tarjetas CRC	Incremento funcional Pruebas Unitarias	Revisión del funcionamiento Incremento desplegado	Resultados del soporte al cliente Incremento funcionando	Producto terminado y aceptado
Herramientas tecnológicas						

Figura 19. Metodología para el desarrollo del sistema de información

Elaboración: Los autores

En la siguiente figura, se observa el flujo de la metodología, es decir, el camino que se siguió al término de una fase.

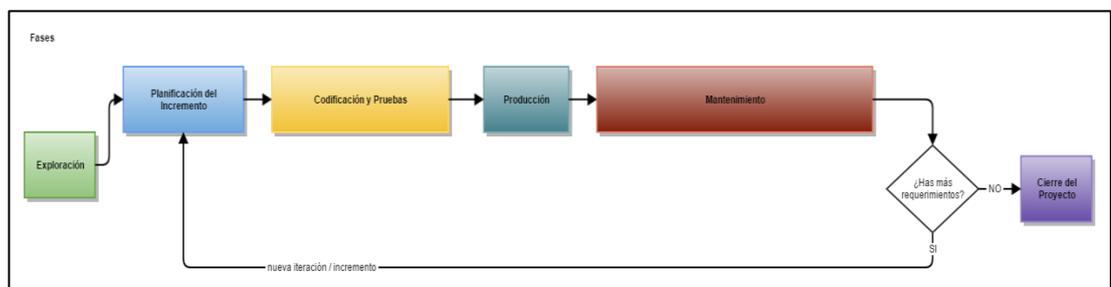


Figura 20. Flujo de la metodología para el desarrollo del sistema de información

Elaboración: Los autores

De la figura anterior, es importante mencionar que la fase de mantenimiento se ejecutó luego de poner en producción el primer incremento y no terminó hasta la entrega del producto, donde cada iteración tuvo un incremento del sistema de información que funcionó.

Como se mencionó la presente metodología está basada en XP y muchas de sus actividades se han integrado con la finalidad de conseguir que el desarrollo del sistema de información se adecue al tiempo de presentación y alcance del proyecto, por ello es que se tuvo seis (6) fases, las cuales se describirán a continuación brevemente:

- **Fase de Exploración:** En esta fase, se planteó las historias de usuario de manera general, y, a la vez, el equipo de desarrollo se capacitó en las herramientas tecnológicas a utilizar en el proyecto.

- **Fase de Planificación de la entrega:** En esta fase, se refinó las historias de usuarios resultantes de la anterior etapa, además se estableció el nivel de riesgo de desarrollo, dando como resultado un incremento planeado. La fase culminó con la realización del modelo físico de datos y de las tarjetas CRC (colaborador-responsabilidad-clase).
- **Fase de codificación y pruebas:** En esta fase, se realizó la programación del incremento y las pruebas unitarias.
- **Fase de producción:** En esta fase, se desplegó el incremento del sistema de información, y se concluyó con la revisión de su funcionamiento.
- **Fase de mantenimiento:** En esta fase, se puso el incremento del sistema a disposición del usuario. Es importante mencionar que el incremento debió de estar probado y funcionando correctamente.

Esta fase se extendió durante todo el desarrollo del sistema de información, en que el usuario pudo utilizar cada incremento y por tanto ver en vivo como avanza el desarrollo del sistema.

- **Fase de Cierre del Proyecto:** En esta, se cerró el proyecto, por tanto, no hubo más requerimientos o historias de usuario pendientes. El sistema de información tuvo que ser aceptado y entregado al cliente.

La metodología de gestión de proyecto está basada en el estándar PMBOK y en las metodologías XP y Marco Lógico.

Las fases de la metodología son:

- **Fase inicial:** En esta fase, se constituyó el proyecto, por tanto, se definió el problema a solucionar, el objetivo general del proyecto y asignó un encargado.

- **Fase de planeación:** En esta fase, se refinó el problema central y el objetivo general, además se definió los objetivos específicos y las actividades del proyecto. Luego se estimó el tiempo necesario para cumplir con cada actividad, se comprobó la viabilidad del proyecto y se realizó la evaluación de riesgos.
- **Fase de ejecución:** En esta fase, se llevó a cabo las actividades definidas en el cronograma y se elaboraron los entregables correspondientes.
- **Fase de seguimiento y control:** En esta fase, se realizó el monitoreo de las actividades del proyecto con la finalidad de poder evaluar lo planeado con lo ejecutado.
- **Fase de cierre:** En esta, se entregó al cliente para su aceptación y compromiso de término del proyecto, por tanto, el producto (resultante del proyecto) ya estuvo listo para su comercialización.

Dentro de cada fase se encuentra lo siguiente:

- **Actividades:** Definen la secuencia de pasos para llevar a cabo la fase. Describen el flujo de la fase utilizando las herramientas para convertir las entradas en entregables o productos (salidas) de la fase.
- **Entradas:** Las entradas son todos aquellos flujos de información que permitieron alimentar a las actividades. A su vez, las entradas de una actividad pudieron ser salidas de otra actividad.
- **Herramientas:** Son aquellas que se utilizaron para llevar a cabo la fase. Estas incluyen herramientas de software, frameworks de desarrollo, documentos, plantillas y guías que se usaron para ejecutar cada una de las actividades definidas en la fase.

- Involucrados: Son las personas y/o roles que forman parte de la realización de la actividad. Los involucrados pueden ser los interesados y/o trabajadores.
- Salidas: Representan los flujos de información o productos que conformaron las salidas de la actividad. Las salidas de una actividad pudieron ser las entradas de la siguiente actividad.

La siguiente figura muestra las actividades, los involucrados en las actividades, las herramientas, entradas y salidas de cada fase:

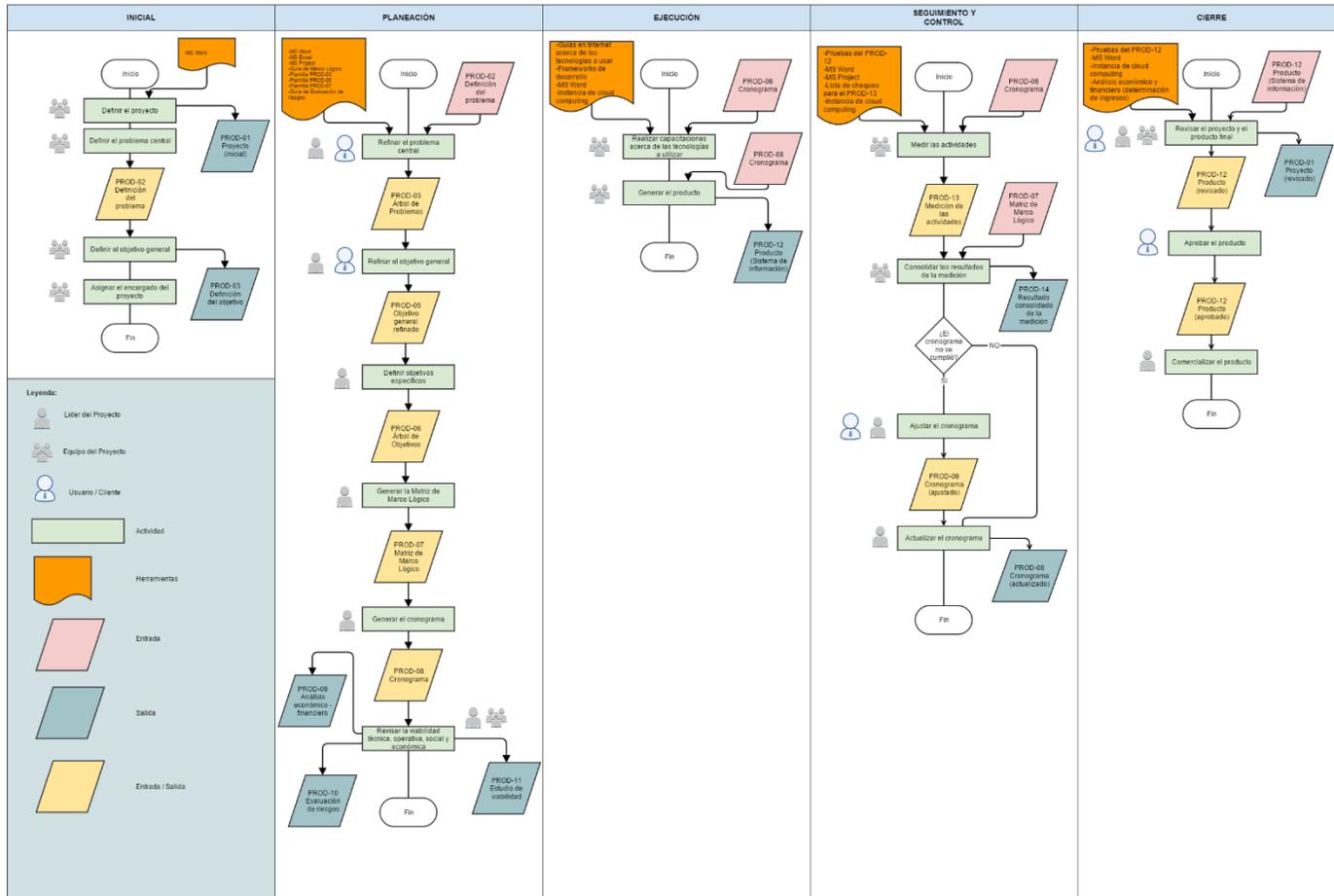


Figura 21. Modelo de Gestión de Proyectos de Software

Elaboración: Los autores

Se utilizó la metodología (framework) de programación Ruby on Rails, con el cual se realizó el software web y los servicios del backend para la aplicación móvil. Para conocer los detalles de la selección de framework, véase el Anexo N° 20 “Selección de framework de desarrollo”.

Para el desarrollo de la aplicación móvil se usó el framework de Phonegap (Apache Cordova) junto con Ionic. Para conocer los detalles de la selección de ambos frameworks (véase los Anexos N°s 21 “Selección de framework para realizar aplicaciones móviles multiplataforma basadas en HTML5” y 22 “Selección de framework de diseño para realizar aplicaciones móviles multiplataforma basadas en HTML5”).

Es importante mencionar que se escogió el desarrollo multiplataforma basadas en HTML5 para aplicaciones móviles, porque principalmente se quiere reutilizar un mismo código en varias plataformas, para ahorrar costos. Además, el equipo de trabajo tiene más experiencia utilizando HTML5, CSS y JavaScript, por todas estas características hace ideal el desarrollo de una aplicación híbrida en base a HTML5. Existe también el desarrollo multiplataforma compilado en nativo, pero para ello se debe conocer muy bien un lenguaje de programación de backend como C# o un lenguaje como JavaScript.

2.2 Desarrollo de proyecto

Durante el desarrollo del proyecto fue necesario seguir las directrices marcadas en el plan de trabajo, el cual puede visualizar en el Anexo N° 23 “Plan de trabajo”.

2.2.1 Desarrollo del sistema de información

Se utilizó como base la metodología Programación Extrema (XP) para el desarrollo del sistema de información. Este sistema está compuesto básicamente por dos componentes primordiales: el software web y la aplicación móvil.

Debido al uso de XP, el proceso de desarrollo cuenta con cuatro actividades: planeación, diseño, codificación y pruebas. Dichas actividades están incluidas dentro de las fases de la metodología de

desarrollo de la presente tesis. Estas son: exploración, planificación de la entrega, codificación y pruebas, producción, mantenimiento, y muerte del proyecto. Cabe mencionar que la descripción tanto de las actividades como de las fases están especificadas en el apartado anterior.

a) Exploración

En esta fase, se realizó las historias de usuario, las cuales se pueden apreciar en su totalidad en el Anexo N° 24 “Historias de usuario”.

Algunas de las principales historias de usuario se muestran a continuación:

Número	1	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia			Guardar información del proyecto vendido
Riesgo de desarrollo			Medio
Descripción			
Guardar la información (como código, fecha de venta, años y meses de garantía, foto del comprobante de pago, número de comprobante pago y descripción) de los proyectos vendidos con la finalidad de poder acceder a dicha información en cualquier momento.			

Figura 22. Historia de usuario con ID 1

Elaboración: Los autores

Número	5	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia			Programar visita a un proyecto determinado
Riesgo de desarrollo			Alto
Descripción			
Asignar un técnico, seleccionar una fecha y hora de visita, e ingresar las indicaciones al momento de programar la revisión de un proyecto con la finalidad de planificar y controlar las visitas de mantenimiento a los clientes.			

Figura 23. Historia de usuario con ID 5

Elaboración: Los autores

De las figuras anteriores, se observa que existe un identificador único por cada historia de usuario, un nombre, el usuario que requiere la funcionalidad, una breve descripción del requerimiento y un riesgo de desarrollo. El riesgo de desarrollo no se estableció durante la primera fase; sin embargo, es importante mencionar que sus posibles valores son: bajo, medio y alto.

Cabe mencionar que para realizar las historias de usuario se tuvo que visitar las empresas que forman parte del presente estudio y registrar sus requerimientos en cuanto al servicio de post-venta al cliente.

b) Planificación de la entrega

En esta fase, se refinaron las historias de usuario y se determinó el riesgo de desarrollo en base a las conversaciones con los clientes, también, se determinó la dificultad de codificación de la historia y su impacto en el sistema de información.

Además, se realizó el plan de iteraciones, el cual contiene los incrementos del sistema, las historias de usuario involucradas por cada incremento y el tiempo (o costo) de desarrollo. A continuación, se muestra el plan de iteraciones:

Tabla 9. Plan de iteración

Incremento	ID Historia de usuario	Historia de usuario	Tiempo
Incremento 1	1	Guardar información del proyecto vendido	2 semanas
	2	Listar los proyectos vendidos	
	3	Mostrar información de un proyecto determinado	
	4	Editar la información de un proyecto determinado	
Incremento 2	5	Programar visita a un proyecto determinado	1 semana
	6	Listar visitas programadas	
	7	Editar visita programada	
	8	Cancelar visita programada	
Incremento 3	9	Ejecutar la visita programada	1 semana
	10	Listar visitas realizadas	
	11	Editar visita ejecutada	
Incremento 4	12	Administrador del sistema	2 semanas
	13	Listar la cantidad de visitas ejecutadas por categoría de proyecto	
	14	Calificar el desempeño del técnico en una visita	
	15	Listar calificación promedio por técnico	

Elaboración: Los autores

Del cuadro anterior, se observa que se realizó cuatro incrementos en el desarrollo del sistema de información, un incremento mínimo dura una semana y máximo dos. Además, el orden de las historias de usuario que se muestra en el plan va de acuerdo a una secuencia lógica en la que para construir una funcionalidad es necesario tener realizada su predecesora.

Otro producto que forma parte de esta fase son las tarjetas CRC (colaborador-responsabilidad-clase), cuya finalidad de realizarlas es organizar las clases orientadas al objeto que son relevantes para el incremento del sistema. Una tarjeta CRC con el propósito de mostrar su interpretación:

Tabla 10. Tarjeta CRC Project

TARJETA CRC	
Clase:	Project
Descripción	
Clase encargada de los datos relevantes al proyecto (como máquinas industriales) que la empresa desarrolló y vendió	
Responsabilidad	Colaborador
Define el código, descripción, número de comprobante de pago, tiempo de garantía y estado del proyecto	
Guarda el proyecto	Customer, TypeProject, AttachynaryFiles
Muestra el proyecto	Customer, TypeProject
Edita el proyecto	Customer, TypeProject, AttachynaryFiles
Lista proyectos	Customer, TypeProject
Lista clientes para asociarlo con el proyecto	Customer
Lista tipos de proyecto para asociarlo con el project	TypeProject
Guarda la foto del comprobante de pago	AttachynaryFiles
Guarda el estado del proyecto	
Muestra la foto del comprobante de pago	AttachynaryFiles
Cancela visita programada del proyecto	

Elaboración: Los autores

La tarjeta CRC mostrada contiene tres partes principales: nombre de la clase, responsabilidades y colaboradores.

El nombre de la clase corresponde a la clase del objeto que se utilizó en el desarrollo del sistema. Las clases que se utilizaron fueron del tipo de entidad, por tanto corresponden a la parte del modelo, según el patrón MVC (modelo-vista-controlador), donde el modelo representa la sección más importante porque incorpora la lógica del negocio.

Las responsabilidades son los atributos y operaciones (métodos) que tiene la clase.

Las colaboraciones son las clases que asisten o ayudan a cumplir una determinada responsabilidad, pueden existir responsabilidades que no requieran alguna colaboración, por tanto, consiguen realizar

sus operaciones de manera autónoma. En general, una colaboración implica ya sea una solicitud de información o la solicitud de alguna acción.

Para visualizar todas las tarjetas CRC, véase el Anexo N° 25 “Tarjetas CRC”.

Además, se realizó el modelo físico, que contiene las tablas de la Base de Datos, sus atributos y la relación que existe entre tablas. Es importante mencionar que existe relación entre el modelo lógico y el físico, la cual es que este último genera un esquema físico de la base de datos teniendo en cuenta el tipo de modelo de datos seleccionado en el modelo lógico. El tipo de modelo de datos que se utiliza en el sistema es el modelo relacional, debido a la facilidad que presenta en la representación de las entidades, el buen soporte de documentación y su disponibilidad en la mayoría de Sistemas Gestores de Base de Datos (SGBD) populares en el mercado (como PostgreSQL, MySQL, MariaDB, SQLServer, Oracle, entre otros).

Vale decir, que a nivel físico el SGBD que se maneja en el sistema de información es PostgreSQL, que trabaja con el modelo relacional. Se muestra el modelo de base de datos:

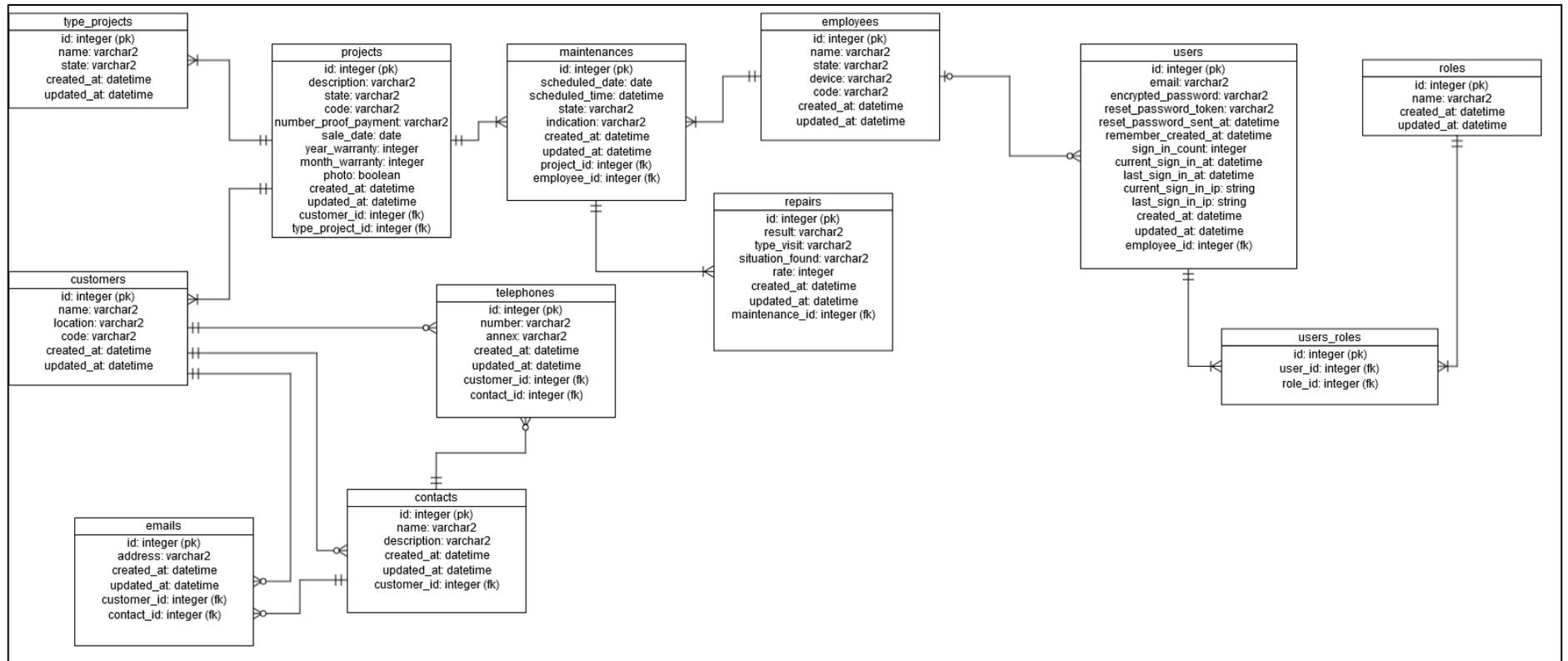


Figura 24. Modelo Físico de la Base de Datos

Elaboración: Los autores

La arquitectura del sistema desarrollado de acuerdo con el tipo de arquitectura Cliente – Servidor:

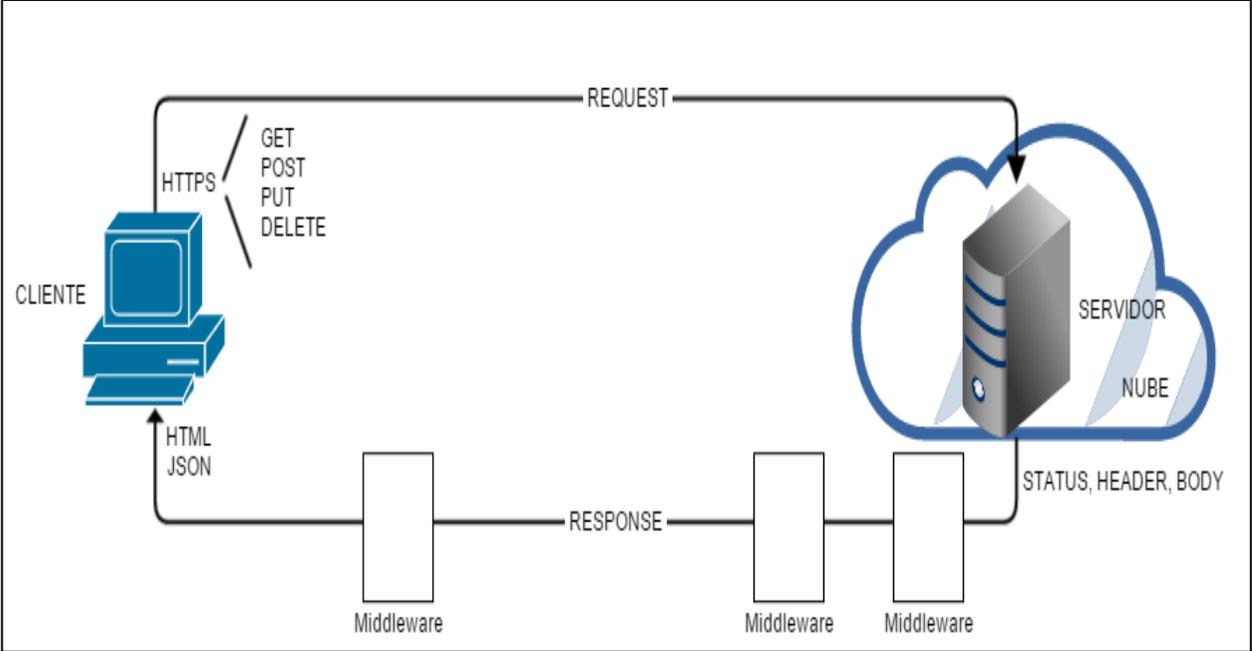


Figura 25. Arquitectura Cliente - Servidor

Elaboración: Los autores

De la figura anterior, el cliente realiza un request o solicitud al servidor utilizando el protocolo HTTPS a través de uno de los verbos (GET, POST, PUT, o DELETE). Una vez que llega la solicitud al servidor que se encuentra en una instancia del proveedor de cloud Computing es procesada por los middlewares (clases que cambian el comportamiento del request o response). Luego el servidor responde con la siguiente información: número de status, cabecera y cuerpo. La respuesta modifica su comportamiento de acuerdo con los middlewares configurados y llega al cliente en un determinado formato (como HTML o JSON).

c) Codificación y pruebas

En esta fase, se realizó la programación del sistema de información, tanto del software web como de la aplicación móvil. Además, se realizaron las pruebas unitarias automatizadas.

En la siguiente figura, se muestra la estructura del software web y las dependencias utilizadas.

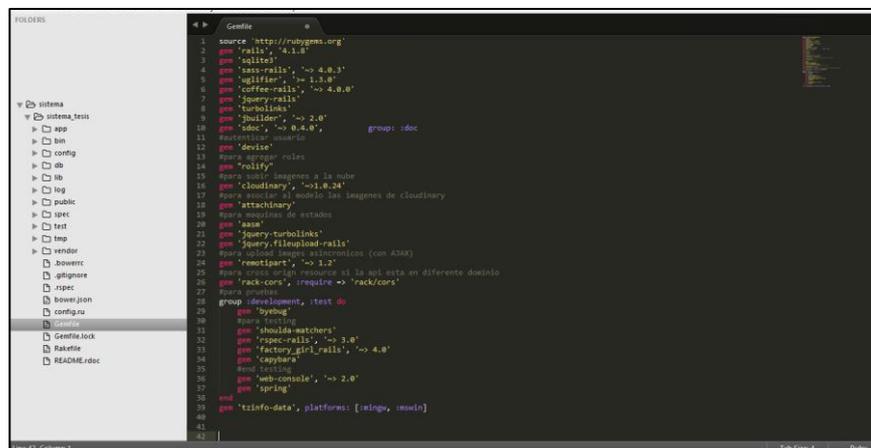


Figura 26. Estructura del software con dependencias

Elaboración: Los autores

De la figura anterior, se observa que el software web está estructurado en carpetas, con lo que se logra separar fácilmente los componentes para su posterior mantenimiento. De las carpetas anteriores las más importantes son las que tienen como nombre app,

porque dentro están los controladores, las vistas, el modelo y otros recursos (como archivos en JavaScript y CSS).

La aplicación móvil multiplataforma tiene una estructura diferente a la del software web, además, se utilizó como gestor de dependencias Bower. En la siguiente figura se muestra la estructura de la aplicación móvil:



Figura 27. Estructura de la aplicación móvil con dependencias

Elaboración: Los autores

De la figura anterior, la aplicación móvil está organizado por carpetas, dentro de estas la más importante es la denominada www, porque ahí se encontrará las interfaces de usuario, archivos en JavaScript y CSS, y las librerías utilizadas.

Otro producto importante dentro de esta fase es el script de la Base de Datos, el cual puede visualizar en el Anexo N° 26 “Script de la Base de Datos”.

En cuanto a las pruebas, estas se especifican en el capítulo de Pruebas y Resultados, pero cabe resaltar que se utilizó Rspec como framework de test llegando a automatizar el desarrollo de esta actividad.

c.1. Módulo de Seguridad

Está compuesto por el inicio y cierre de sesión, además de los callbacks para verificar la autenticidad del usuario y sus permisos por cada solicitud. También lo conforma la administración de usuarios.

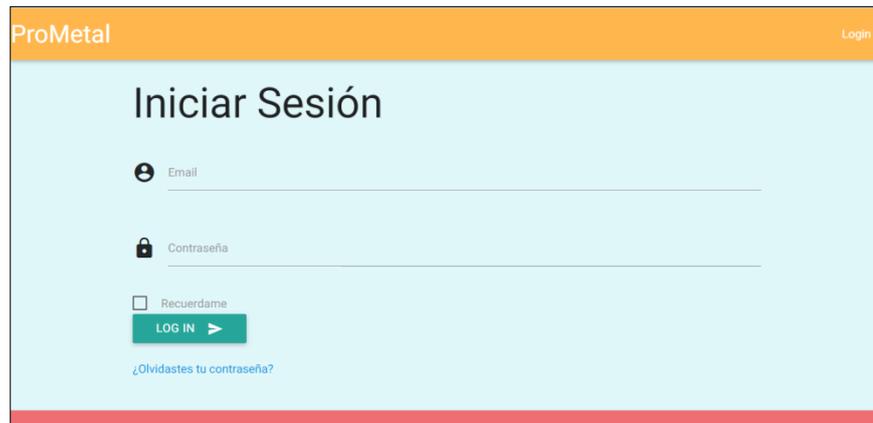


Figura 28. Interfaz de Inicio de Sesión

Elaboración: Los autores

De la figura anterior para que el usuario pueda acceder al sistema requiere como credenciales el correo electrónico y la contraseña. Además, hay un enlace para enviar la solicitud de cambio de contraseña a su correo electrónico.



Figura 29. Interfaz de Logout (Cierre de Sesión)

Elaboración: Los autores

De la figura anterior, el cierre de sesión se encuentra dentro de las opciones que muestra la barra de navegación ubicada en la sección superior de la pantalla.



Figura 30. Interfaz Mis usuarios

Elaboración: Los autores

En la figura anterior, la interfaz muestra todos los usuarios del sistema solo al administrador, además este mismo puede asignar privilegios (permisos) para que otros usuarios accedan a determinadas funcionalidades según su flujo de trabajo en la empresa. También puede revocar los privilegios si fuese necesario. Cabe mencionar que debe definir a qué técnico le corresponde su usuario, porque de esa manera, en la aplicación móvil, cuando inició sesión, el usuario técnico podrá ver las funcionalidades que se le permiten.

c.2. Módulo de Servicio al cliente

Está compuesto por las funcionalidades referidas al registro y edición del proyecto, planificación y cancelación de visitas, y registro, eliminación y edición de visitas realizadas. Además, se puede realizar consultas por proyecto, visitas programadas y visitas realizadas.



Figura 31. Interfaz de Mis proyectos (Consulta de proyectos)

Elaboración: Los autores

En la figura, se muestran todos los proyectos vendidos de la empresa con la capacidad de poder realizar una búsqueda por código, cliente, comprobante de pago o categoría de proyecto.

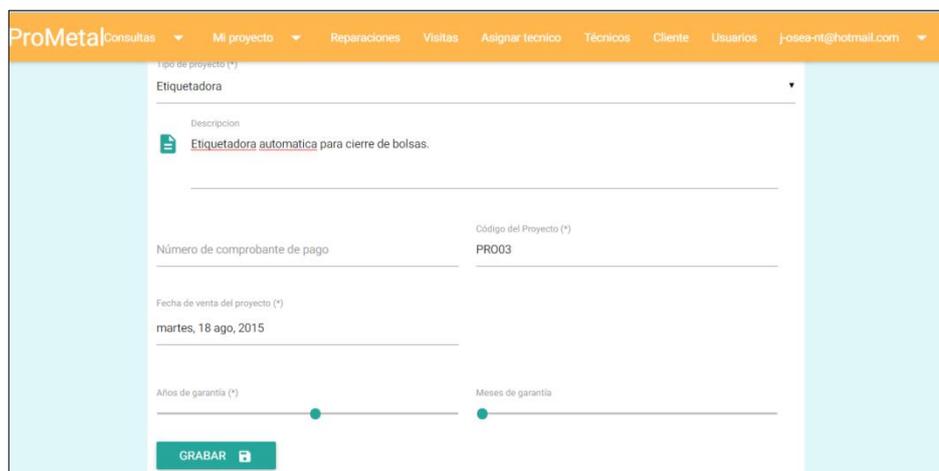


Figura 32. Interfaz Nuevo Proyecto

Elaboración: Los autores

En la figura anterior, la interfaz muestra un formulario para ingresar la foto del comprobante de pago si la hubiese, una descripción del proyecto y un código, además, se debe seleccionar el cliente, la categoría de proyecto, la fecha de venta y el tiempo de garantía. Cabe mencionar que la interfaz de nuevo proyecto es la misma para su edición, solo que los datos vienen cargados.

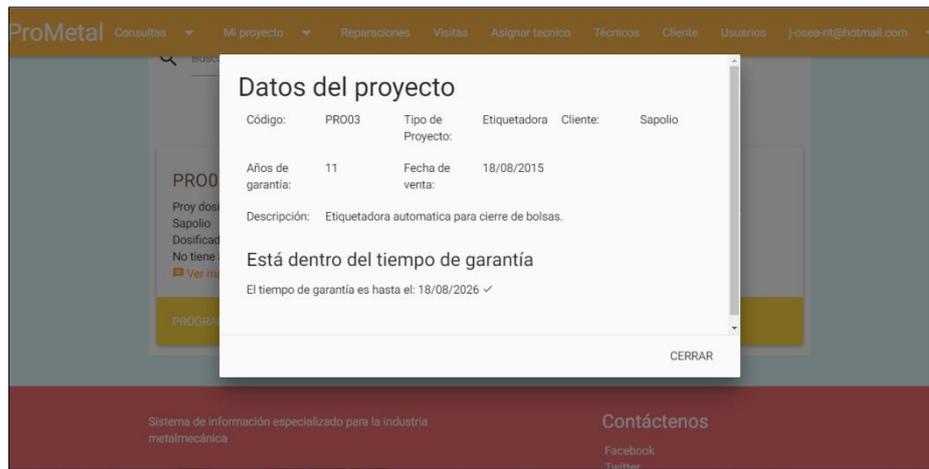


Figura 33. Interfaz Ver más (Mostrar proyecto)

Elaboración: Los autores

En la figura anterior, la interfaz muestra los datos de un determinado proyecto y si está todavía dentro del tiempo de garantía.

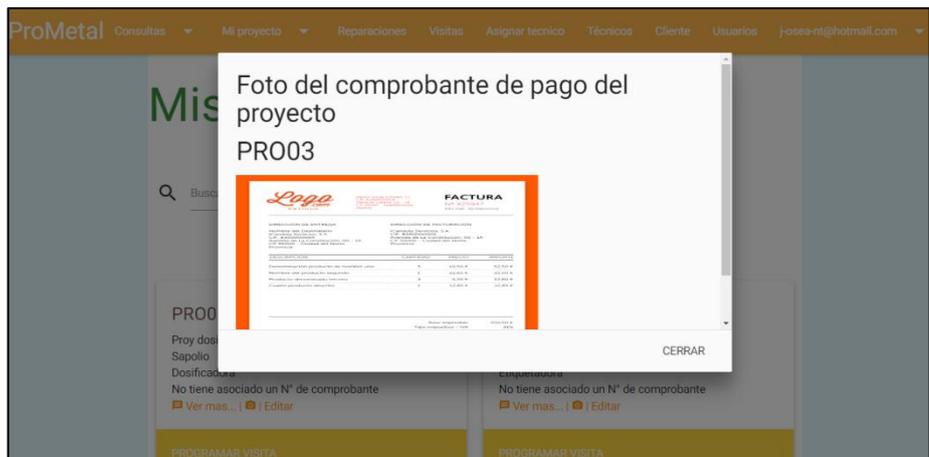


Figura 34. Interfaz Mostrar foto de comprobante de pago

Elaboración: Los autores

En la figura anterior, la interfaz muestra la foto del comprobante de pago de un determinado proyecto.

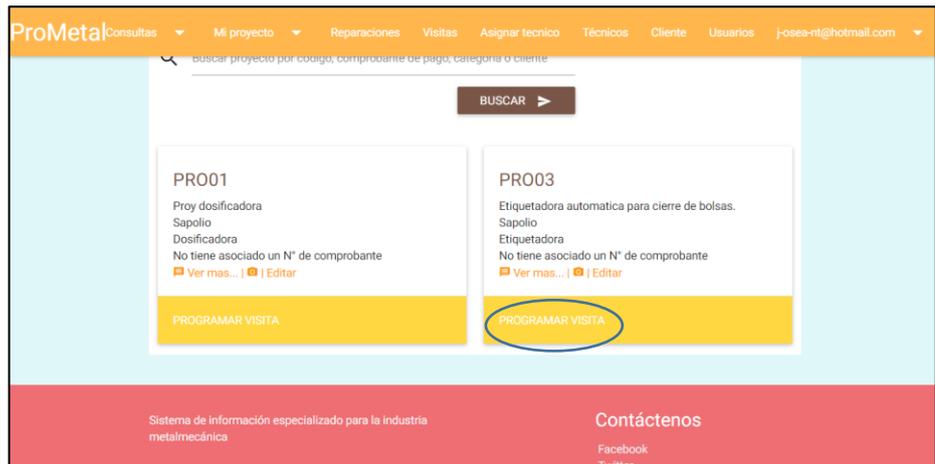


Figura 35. Interfaz Programar visita

Elaboración: Los autores

En la figura anterior, la interfaz muestra la opción de programar una visita por cada proyecto vendido que no esté con una visita planificada pendiente.

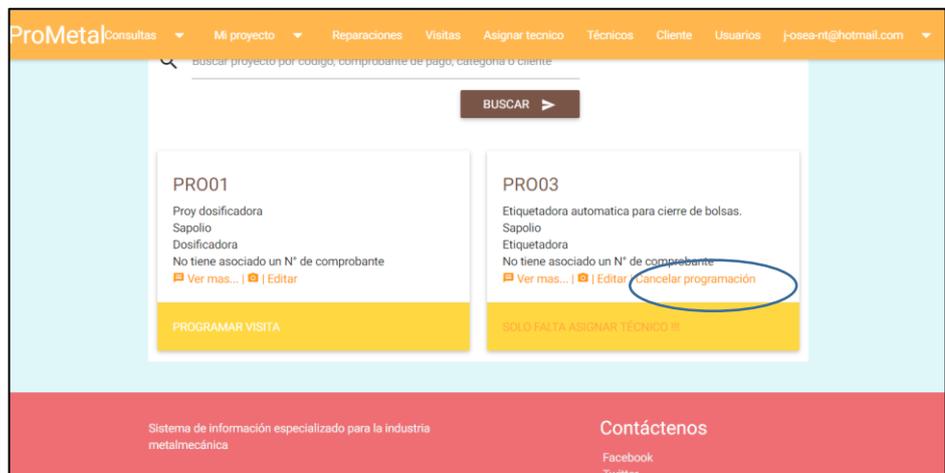


Figura 36. Interfaz Cancelar programación de visita

Elaboración: Los autores

La figura anterior muestra la interfaz con la opción para cancelar la programación de una visita por cada proyecto con visita pendiente.

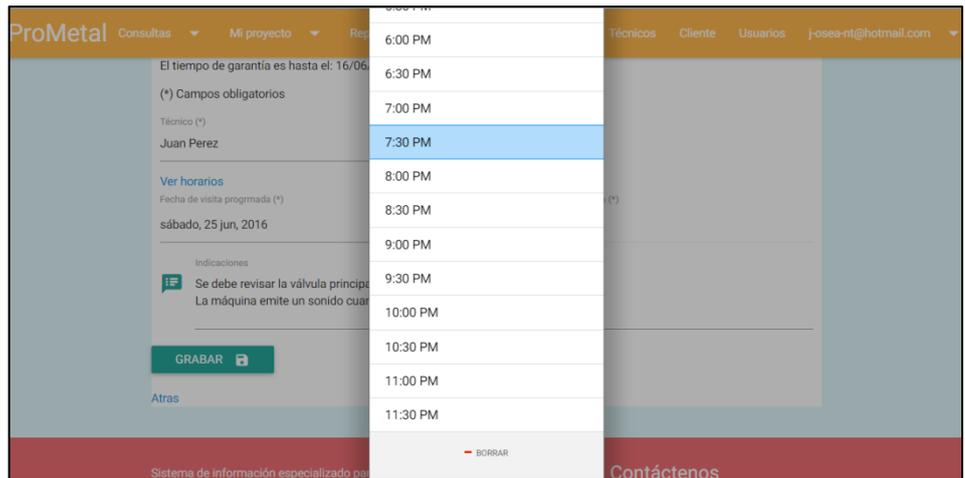


Figura 37. Interfaz Nueva visita (Planificar visita)

Elaboración: Los autores

La figura anterior, la interfaz muestra un formulario para ingresar las indicaciones de la visita, además, se debe seleccionar el técnico y la fecha de la visita. También hay un enlace para ver los horarios de las visitas planificadas por técnico y de esta manera, evitar conflictos de fecha. Cabe mencionar que la interfaz de nueva visita es la misma para su edición, solo que los datos vienen cargados.

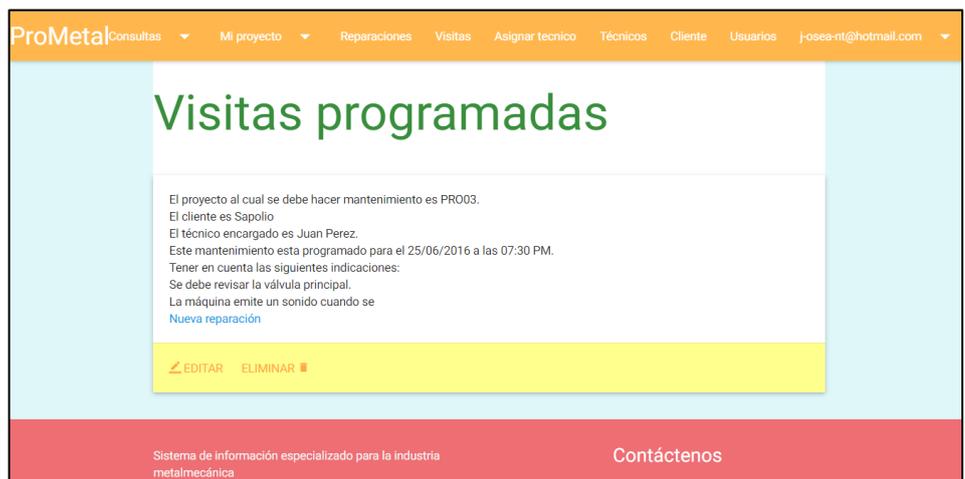


Figura 38. Interfaz Visitas (Consulta de visitas programadas)

Elaboración: Los autores

De la figura anterior, la interfaz muestra todas las visitas programadas. Además, se observa tres enlaces: el primero es

para registrar la visita realizada, el otro es para editar la visita planificada y el tercero es para eliminar la visita programada.

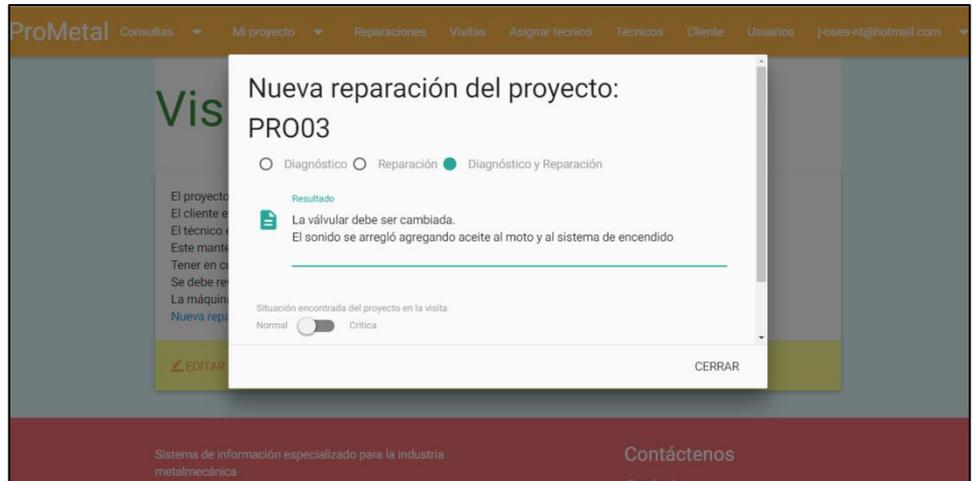


Figura 39. Interfaz Nueva reparación (visita realizada)

Elaboración: Los autores

En la figura anterior, la interfaz muestra un formulario para ingresar los resultados de la visita realizada, además, se debe seleccionar el tipo de visita y la situación encontrada. Cabe mencionar que la interfaz de nueva reparación es la misma para su edición, solo que los datos vienen cargados.



Figura 40. Interfaz Reparaciones (Consulta de visitas realizadas)

Elaboración: Los autores

La figura anterior, la interfaz muestra todas las visitas programadas, además, se observa dos enlaces por cada

elemento: el primero es para editar la visita realizada y el otro es para eliminarla. También se puede filtrar de acuerdo con el tipo de visita o por la situación encontrada en el proyecto.

c.3. Módulo de consultas

Está compuesto por las consultas de información consolidada, de esta manera, se puede conocer las ventas y visitas realizadas por categoría de producto, y resultados de la evaluación de cada técnico de acuerdo con las visitas realizadas.



Figura 41. Interfaz Mis ventas

Elaboración: Los autores

En la figura anterior, la interfaz muestra las ventas por cada categoría de proyecto, además, hay una opción para ver los proyectos que pertenecen a una categoría determinada.



Figura 42. Interfaz Visitas por proyecto

Elaboración: Los autores

En la figura anterior, la interfaz muestra las visitas realizadas por categoría de proyecto, además, hay una opción para ver los proyectos que pertenecen a una categoría determinada. Esta funcionalidad permite al usuario discernir que hay un tipo de proyecto que está presentando complicaciones porque tiene muchas visitas.



Figura 43. Interfaz Calificaciones por técnico

Elaboración: Los autores

En la figura anterior se muestra la interfaz con las calificaciones de cada técnico de acuerdo con su desempeño en las visitas.

c.4. Módulo de maestro

Está compuesto por las funcionalidades que soportan el módulo de servicio al cliente como la administración de clientes, técnicos y categoría de proyectos.

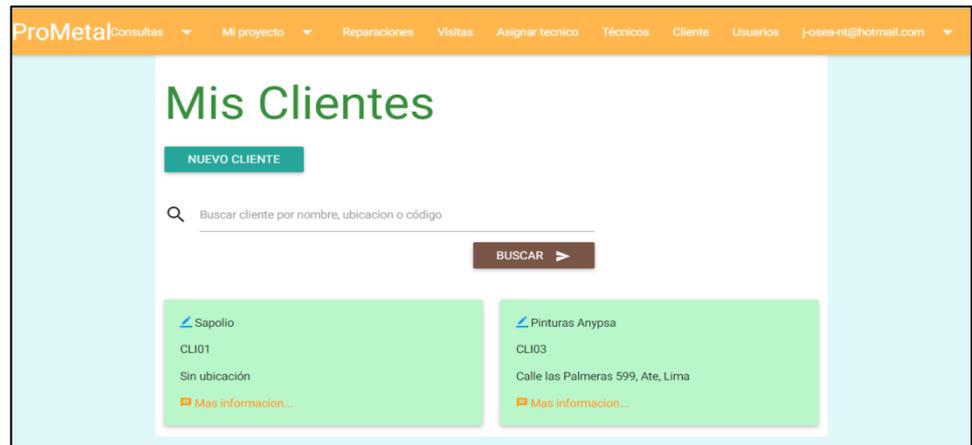


Figura 44. Interfaz Mis clientes

Elaboración: Los autores

En la figura anterior se muestra la interfaz con los datos de cada cliente con la facilidad de poder filtrarlos por nombre, ubicación o código. Además, por cada cliente se puede ver toda su información completa a través de la opción más información.

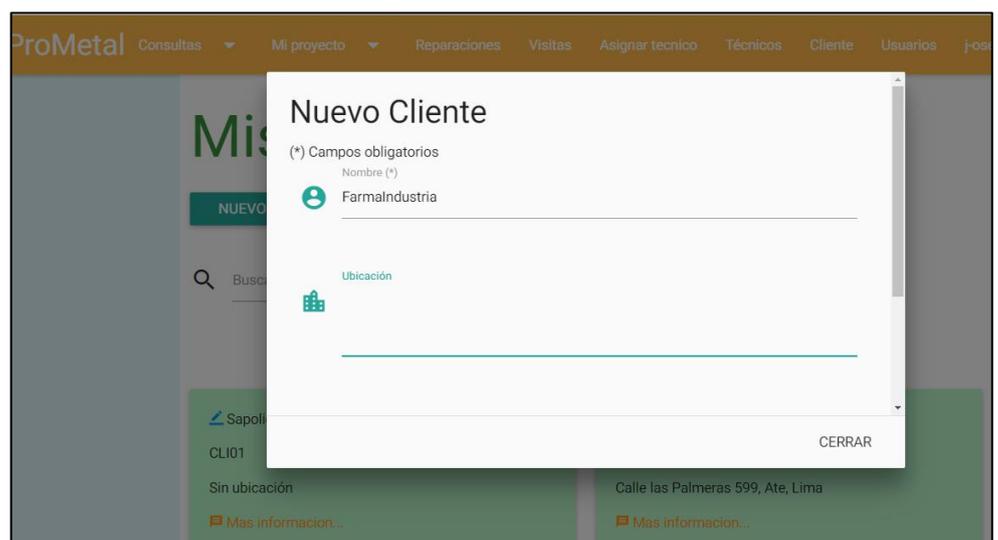


Figura 45. Interfaz Nuevo cliente

Elaboración: Los autores

La figura anterior muestra la interfaz con un formulario para ingresar el nombre, la ubicación y el código del cliente. Cabe mencionar que la interfaz de nuevo cliente es la misma para su edición, solo que los datos vienen cargados.

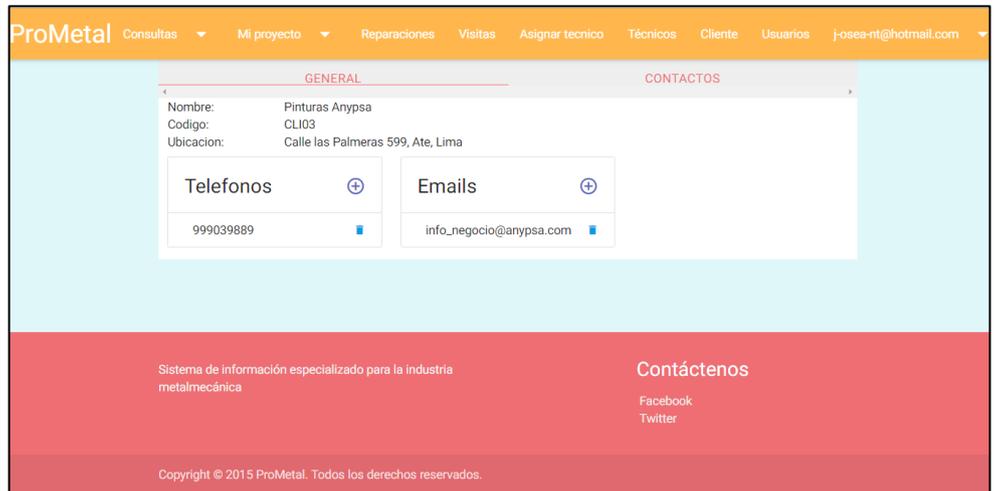


Figura 46. Interfaz Más información del cliente

Elaboración: Los autores

La figura anterior muestra la interfaz con la información de un cliente en particular y las opciones para agregar y eliminar sus teléfonos y emails.

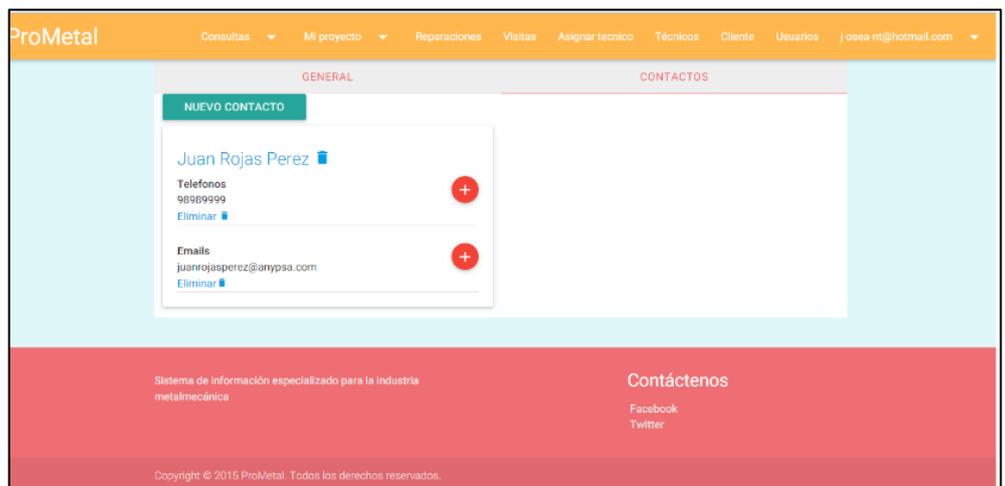


Figura 47. Interfaz Contactos de clientes

Elaboración: Los autores

La figura anterior muestra la interfaz muestra con los contactos de un determinado cliente y las opciones para agregar y

eliminar sus teléfonos y emails. Además, hay un enlace que permite registrar un nuevo contacto para el cliente.

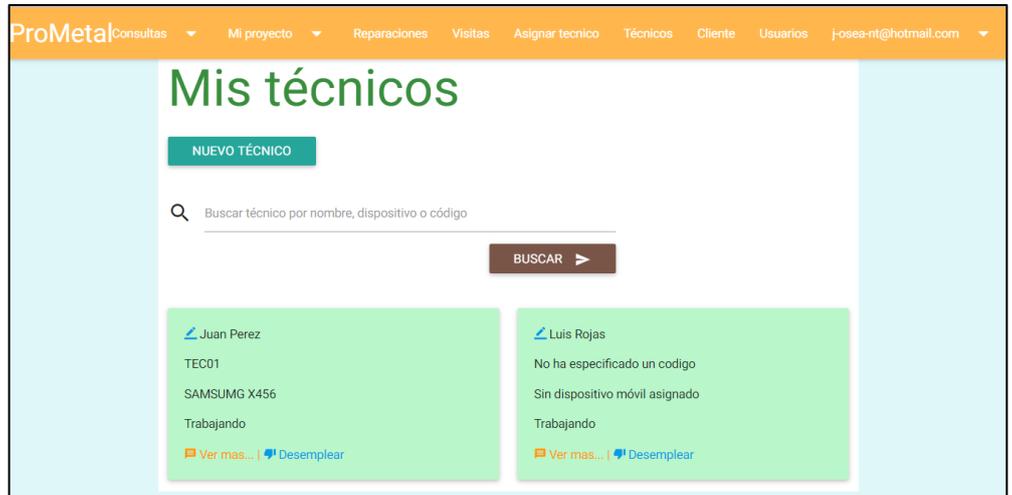


Figura 48. Interfaz Mis técnicos

Elaboración: Los autores

La figura anterior muestra la interfaz con los datos de cada técnico de la empresa con la facilidad de poder filtrarlos por nombre, dispositivo móvil asignado o código. Además, por cada técnico están las opciones para ver más información y cambiar su estado de trabajo (desempleado o empleado).

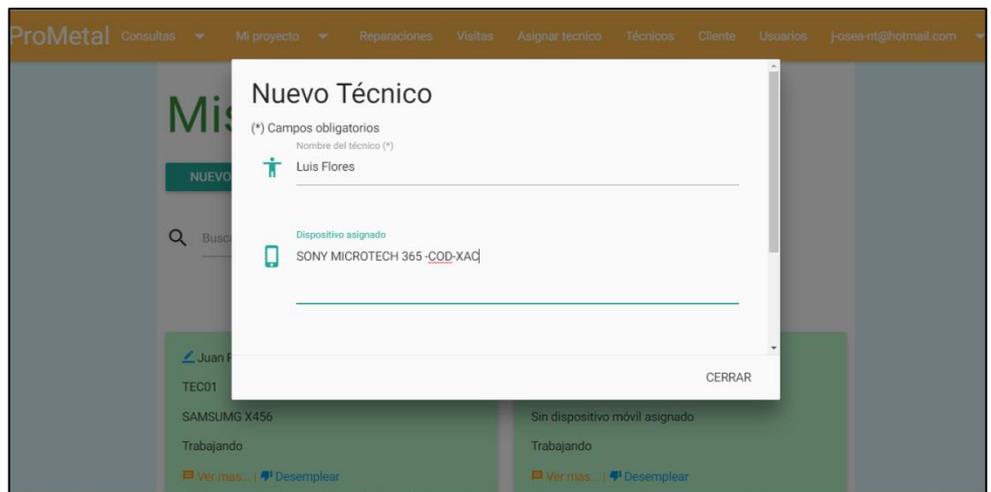


Figura 49. Interfaz Nuevo técnico

Elaboración: Los autores

La figura muestra la interfaz con un formulario para ingresar el nombre, el dispositivo móvil asignado y el código del técnico.

Cabe mencionar que la interfaz de nuevo técnico es la misma para su edición, solo que los datos vienen cargados.



Figura 50. Interfaz Mis categorías de proyectos

Elaboración: Los autores

La figura muestra la interfaz con todas las categorías de proyecto y por cada uno las opciones para editarlas o cambiarles el estado (se produce o ya no se produce).

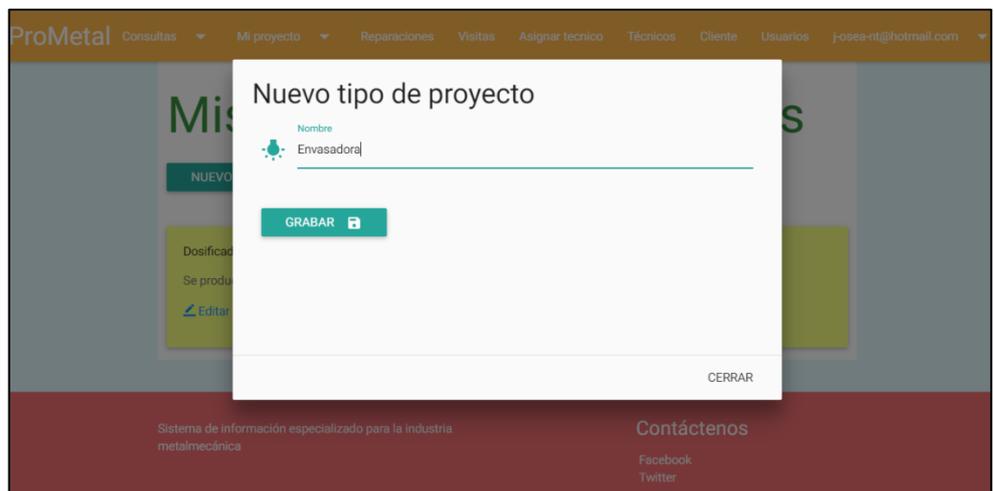


Figura 51. Interfaz Nuevo tipo de proyecto

Elaboración: Los autores

La figura anterior muestra la interfaz con un formulario para ingresar el nombre de la categoría de proyecto. Cabe mencionar que la interfaz de nueva categoría de proyecto es la misma para su edición, solo que los datos vienen cargados.

c.5. Aplicación móvil

Está compuesto por la consulta de visitas programadas de acuerdo con el técnico usuario que ingrese a la aplicación, también se puede registrar la visita y calificar el desempeño del técnico en operación.

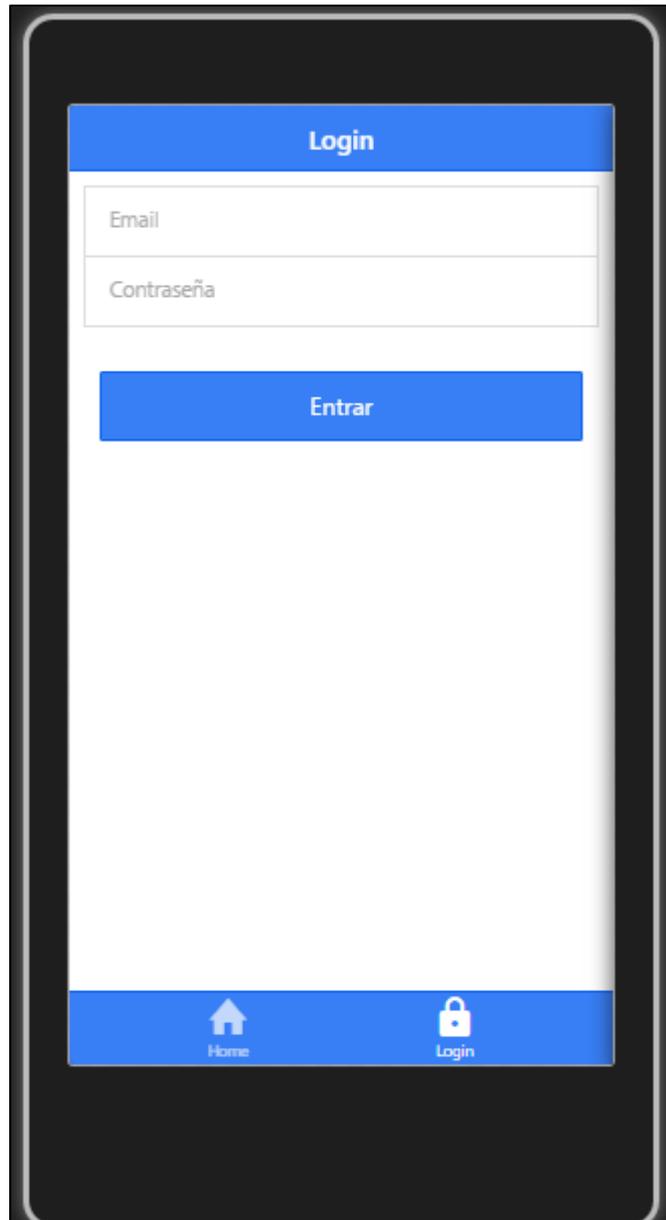


Figura 52. Interfaz Login de la app móvil

Elaboración: Los autores

La figura anterior muestra la interfaz con un formulario para ingresar las credenciales del usuario que son correo electrónico

y contraseña con la finalidad de poder acceder a la aplicación móvil.

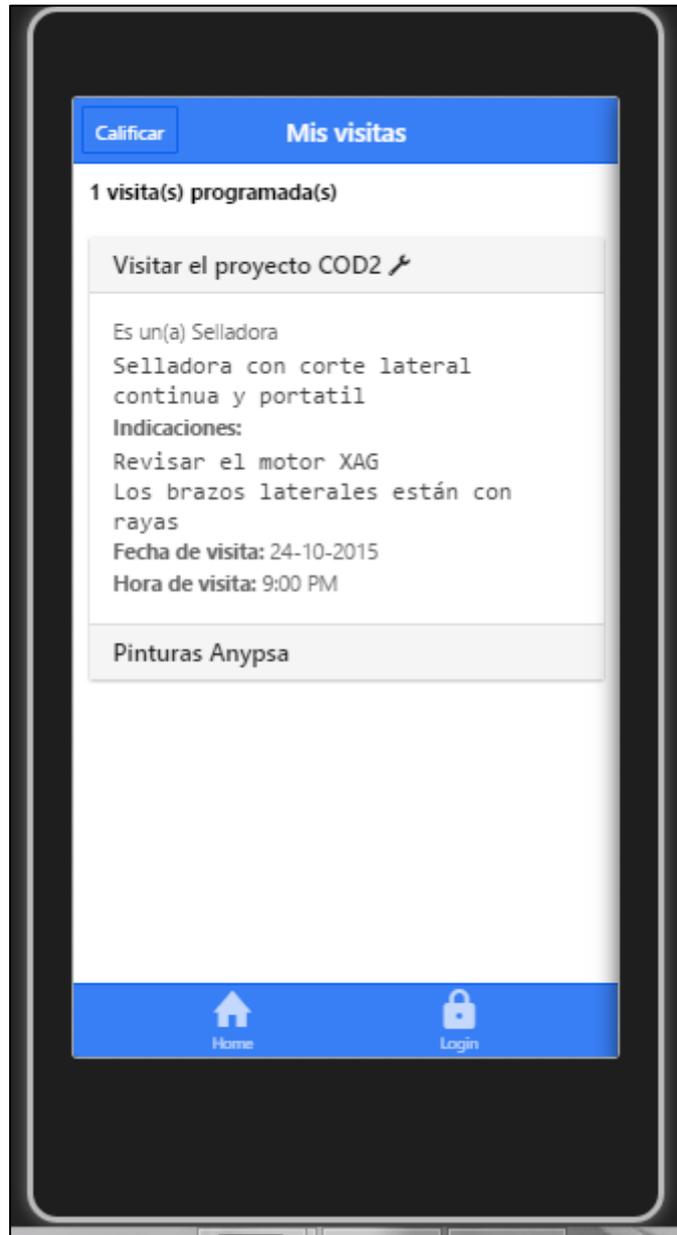


Figura 53. Interfaz Mis visitas programadas de la app móvil

Elaboración: Los autores

La figura anterior muestra la interfaz con todas las visitas programadas para un determinado técnico (que es el usuario que ingresó a la aplicación).



Figura 54. Interfaz Nueva visita de la app móvil

Elaboración: Los autores

La figura anterior muestra la interfaz con un formulario para ingresar los resultados de la visita, además se debe seleccionar el tipo de visita y la situación encontrada del proyecto.

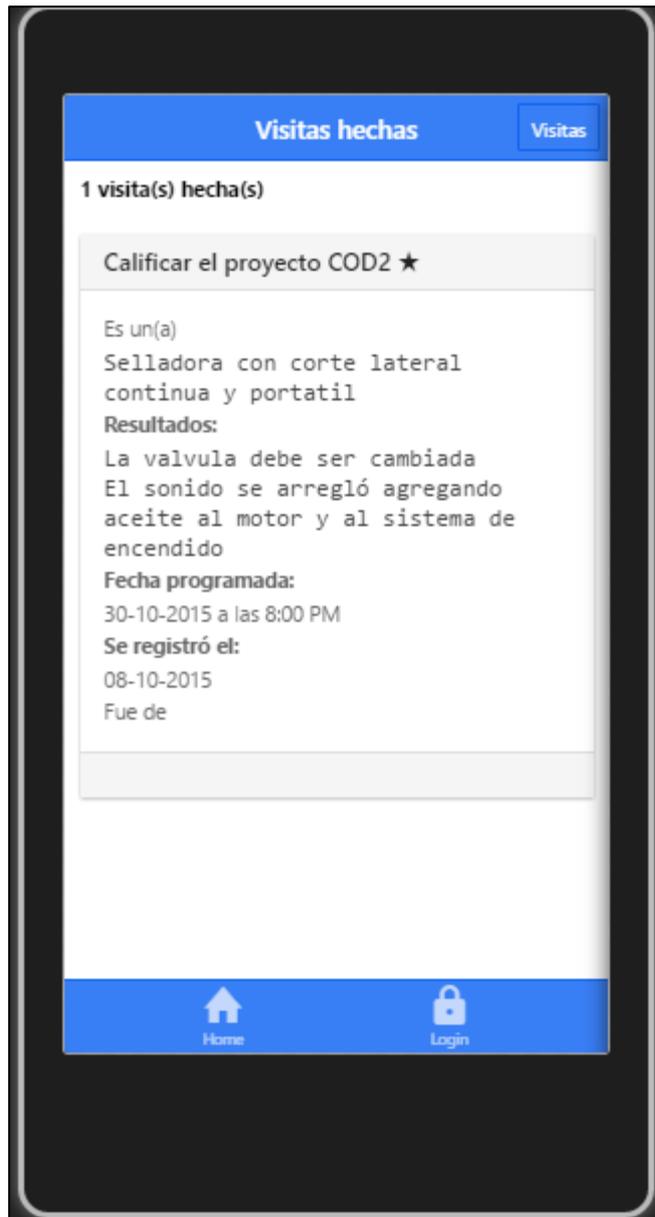


Figura 55. Interfaz Mis visitas realizadas de la app móvil

Elaboración: Los autores

La figura anterior muestra la interfaz con las visitas realizadas por el técnico usuario entre el día anterior y el actual a su registro que no han sido calificadas.

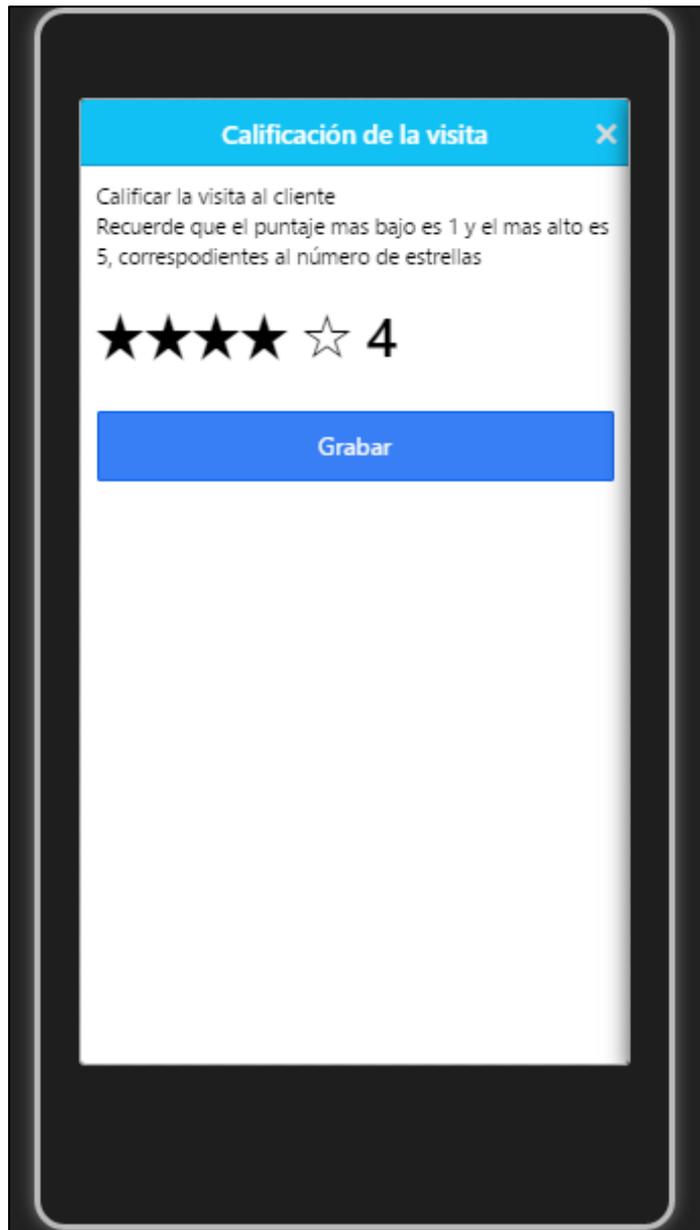


Figura 56. Interfaz Calificación de la visita de la app móvil

Elaboración: Los autores

La figura anterior muestra la interfaz con un formulario para ingresar la calificación del desempeño del técnico según el cliente.

d) Producción

En esta fase, se desplegó el sistema de información, para ello se siguió la siguiente estrategia:

- El software web se desplegó en la plataforma de Heroku
- La aplicación móvil se desplegó en la tienda virtual de Android (Play Store).

Cabe mencionar que en el capítulo de Pruebas y Resultados se verificará el despliegue del sistema de información.

e) Mantenimiento

En esta fase, se explicó a los potenciales clientes el funcionamiento del sistema de información y la forma de comercialización. Luego se probó con los posibles clientes el sistema, cuyos resultados están el capítulo de Pruebas y Resultados.

Las visitas enfocadas en la capacitación del usuario se ejecutaron de la siguiente manera:

- El 02/07/2015 se realizó una visita a INNOMECC y a Procesos Continuos SAC.
- El 22/07/2015 se realizó una visita a INNOMECC y a Procesos Continuos SAC.
- El 14/08/2015 se realizó una visita a INNOMECC y a Procesos Continuos SAC.

En todas las visitas, se brindó asistencia al usuario y explicación de los beneficios de utilizar el sistema de información como medio para mejorar su servicio de post-venta. Además, cada vez que se terminaba un incremento, se visitaba ambas empresas para que revisen lo avanzado y comiencen a interactuar con el sistema.

f) Cierre del proyecto

Una vez terminadas las visitas y teniendo la aceptación del producto por parte de los potenciales clientes, se procedió a cerrar el proyecto organizando y guardando en un repositorio físico los productos realizados (como el sistema de información).

2.2.2 Desarrollo del modelo de negocio

El modelo de negocio consiste en comercializar el producto (sistema de información) a las empresas metalmecánicas mediante el cobro mensual de ochenta y 00/100 nuevos soles (S/. 80.00) por usuario. En este modelo, está contemplado que los clientes no paguen por adquirir una infraestructura tecnológica ya que ellos no cuentan con una, sino que, a través de un pequeño pago, obtengan el servicio tecnológico y comiencen a preocuparse solo por su negocio. Además, con el avance del tiempo se incorporarán más funcionalidades y estas estarán disponibles en primera instancia para los clientes sin costo alguno por un tiempo.

Se utilizó el modelo Canvas para desarrollar el modelo de negocio, cuyo resultado es la siguiente figura:



Figura 57. Modelo Canvas

Elaboración: Los autores

De la figura anterior, se puede observar que el proveedor de cloud computing es un asociado clave para el lograr comercializar el sistema de información, porque será quien soporte la infraestructura tecnológica.

Las principales actividades que forman parte del proyecto son: desarrollar el software web, desarrollar la aplicación móvil y desplegar el sistema de información.

Durante el proyecto fue necesario tener los siguientes recursos claves: personal calificado, programas de desarrollo, hardware (computadoras) para la programación y una instancia de cloud computing (para desplegar y mantener el sistema).

La relación con los clientes se establece a través de la comunicación con ellos por medio de teléfono, correos electrónicos o forma presencial.

Los canales para comercializar el producto son dos: cobro mensual por licencia de usuario y la promoción del producto en las reuniones de la Sociedad Nacional de Industria.

El segmento de mercado al cual está dirigido el producto es claramente las Mipymes del sector metalmecánico, aunque en un tiempo se planea expandirse a otros.

Están contemplados los costos por desarrollar el sistema de información, el costo de mantener la instancia de cloud computing y los costos por probar el producto en las diferentes empresas del Perú a manera de promoción.

La fuente de ingreso del proyecto es el cobro mensual por licencia de usuario.

La propuesta de valor es un producto especializado (en servicio de post-venta al cliente) y sectorizado (industria metalmecánica) moderno y fácil de usar que permite ahorrar costos de infraestructura tecnológica a las empresas, y es que es accedido desde cualquier lugar y con cualquier dispositivo.

CAPÍTULO III

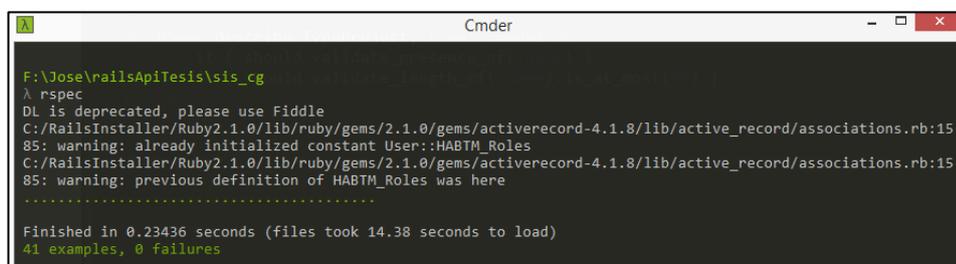
PRUEBAS Y RESULTADOS

3.1 Plan de pruebas del software

3.1.1 Resumen ejecutivo

El propósito del presente plan es verificar el funcionamiento del software a través de pruebas; dentro de ellas las de tipos unitarias realizadas por cada clase que forman parte de la lógica del negocio.

Se ejecutaron cuarenta y uno (41) pruebas, el resultado fue el que se aprecia en la siguiente figura:



```
F:\Jose\railsApiTesis\sis_cg
λ rspec
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: previous definition of HABTM_Roles was here
.....
Finished in 0.23436 seconds (files took 14.38 seconds to load)
41 examples, 0 failures
```

Figura 58. Resultados de la ejecución de las pruebas unitarias

Elaboración: Los autores

En la figura anterior, se muestra que todas las pruebas (41) pasaron con éxito, por tanto, no hubo fallas en los componentes principales del sistema.

3.1.2 Elementos de pruebas

Tabla 11. Lista de elementos de pruebas

Componente	Descripción
Contact	Clase que permite guardar y consultar la información de los contactos de un determinado cliente.
Customer	Clase que permite guardar y consultar la información de un determinado cliente.
Email	Clase que permite guardar y consultar los correos electrónicos de los contactos y/o de los clientes.
Employee	Clase que permite guardar y consultar la información de los empleados de la empresa.
Maintenance	Clase que permite guardar y consultar la información de la visita planificada o programada.
Project	Clase que permite guardar y consultar la información de los proyectos vendidos de la empresa.
Repair	Clase que permite guardar y consultar la información de la visita realizada o ejecutada.
TypeProject	Clase que permite guardar y consultar la información de las categorías de proyectos.
Telephone	Clase que permite guardar y consultar los teléfonos y anexos de los contactos y/o clientes.

Elaboración: Los autores

En la tabla anterior, los componentes listados son independientes conformando con sus métodos una unidad aislada de código.

3.1.3 Enfoque de pruebas

Se realizaron pruebas unitarias a los componentes mencionados en el apartado anterior, de esta manera, se constató el funcionamiento de un módulo del código fuente. Las pruebas unitarias

realizadas, permitieron validar el funcionamiento de manera independiente de los componentes sin interferir en otros.

Cabe mencionar, que las pruebas realizadas fueron automatizadas, por tanto, se utilizó un lenguaje de programación junto a un framework de pruebas. Teniendo en cuenta el lenguaje con el que se desarrolló el sistema, se utilizó Ruby para las pruebas y el framework RSpec.

3.1.4 Criterios de aceptación

El criterio de aceptación es completar el 100% de las pruebas unitarias sin fallas.

3.1.5 Entregables

Se realizó como parte de la ejecución del plan, las especificaciones de resultados de pruebas unitarias, en el cual se detalló cada prueba ejecutada por cada componente.

3.1.6 Requerimientos de entornos – hardware

A continuación, se listan los recursos a nivel de hardware necesarios para realizar las pruebas:

Tabla 12. Recursos de hardware para las pruebas

Recurso	Uso
2 computadoras con las siguientes características: <ul style="list-style-type: none"> • Procesador Intel core I5 • Memoria RAM 8GB • Disco duro 500 GB 	Para instalar el framework de test y ejecutar las pruebas utilizando la CLI (interfaz de línea de comando).
Base de datos de prueba (SqlLite)	Para obtener la información guardada en la base de datos y poder realizar las pruebas.
Servidor de Aplicaciones de prueba (Webrick)	Para correr la aplicación mientras se ejecutan las pruebas unitarias.

Elaboración: Los autores

3.1.7 Requerimientos de entornos – software

Los recursos a nivel de software necesarios para realizar las pruebas:

Tabla 13. Recursos de software para las pruebas

Recurso	Uso
Framework de desarrollo (Ruby on Rails)	Para desarrollar la aplicación a probar.
Framework de pruebas (RSpec)	Para realizar las pruebas unitarias por cada componente.
Librería de pruebas (Shoulda)	Para ejecutar métodos de ayuda por cada prueba unitaria dentro del framework de pruebas.

Elaboración: Los autores

3.2 Resultados de las pruebas

Se muestran las especificaciones de resultados de las pruebas unitarias por cada componente señalado en los elementos de prueba:

a) Clase Contact

En las siguientes figuras, se muestran las pruebas unitarias de la clase Contact:

```
contact_spec.rb
1  require 'rails_helper'
2
3  RSpec.describe Contact, type: :model do
4    it { should validate_presence_of(:name) }
5    it { should belong_to(:customer) }
6    it { should validate_presence_of(:customer) }
7  end
```

Figura 59. Prueba unitaria a Contact

Elaboración: Los autores

```
F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/contact_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: previous definition of HABTM_Roles was here
...
Finished in 0.04686 seconds (files took 14.35 seconds to load)
3 examples, 0 failures
```

Figura 60. Resultados de la prueba a Contact

Elaboración: Los autores

Como se puede apreciar en la figura 57, se muestran las tres pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tienen fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos a un contacto, es necesario ingresar su nombre.
2. Para registrar o modificar en la base de datos a un contacto, es necesario que el cliente esté registrado.
3. Para registrar o modificar en la base de datos a un contacto, es necesario indicar un cliente, no necesariamente debe estar registrado.

b) Clase Customer

En las siguientes figuras, se muestran las pruebas unitarias de la clase Customer:

```
customer_spec.rb
1 require 'rails_helper'
2
3 RSpec.describe Customer, type: :model do
4   it { should validate_presence_of(:name) }
5   it { should validate_length_of(:name).is_at_most(25) }
6   it { should validate_length_of(:location).is_at_most(250) }
7 end
```

Figura 61. Prueba unitaria a Customer

Elaboración: Los autores

```
F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/customer_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: previous definition of HABTM_Roles was here
...
Finished in 0.03123 seconds (files took 14.58 seconds to load)
3 examples, 0 failures
```

Figura 62. Resultados de la prueba a Customer

Elaboración: Los autores

Como se puede apreciar en la figura 59, se muestran las tres pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tienen fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos a un cliente, es necesario ingresar su nombre o razón social.
2. Para registrar o modificar en la base de datos el nombre o razón social de un cliente, este no debe poseer más de 25 caracteres.
3. Para registrar o modificar en la base de datos la ubicación de la empresa de un cliente, debe poseer no más de 250 caracteres.

c) Clase Email

En las siguientes figuras, se muestran las pruebas unitarias de la clase Email:

```
email_spec.rb
1 require 'rails_helper'
2
3 RSpec.describe Email, type: :model do
4   it { should validate_presence_of(:address) }
5   it { should validate_length_of(:address).is_at_most(50) }
6   it { should_not allow_value("probando").for(:address) }
7   it { should_not allow_value("probando@hotmail").for(:address) }
8   it { should allow_value("probando@hotmail.com").for(:address) }
9 end
```

Figura 63. Prueba unitaria a Email

Elaboración: Los autores

```

F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/email_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:
85: warning: previous definition of HABTM_Roles was here
*****
Finished in 0.03125 seconds (files took 14.22 seconds to load)
5 examples, 0 failures

```

Figura 64. Resultados de la prueba a Email

Elaboración: Los autores

Como se puede apreciar en la figura 61 se muestran las cinco pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tienen fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos a un email, es necesario ingresar la dirección de correo.
2. Para registrar o modificar en la base de datos a un email, la dirección de correo debe poseer no más de 50 caracteres.
3. Al registrar o modificar un email no debe permitirse el valor “probando” como dirección de correo.
4. Al registrar o modificar un email no debe permitirse el valor “probando@hotmail” como dirección de correo.
5. Al registrar o modificar un email debe permitirse el valor “probando@hotmail.com” como dirección de correo.

d) Clase Employee

En las siguientes figuras, se muestran las pruebas unitarias de la clase Employee:

```

employee_spec.rb
1  require 'rails_helper'
2
3  RSpec.describe Employee, type: :model do
4    it { should validate_presence_of(:name) }
5    it { should validate_length_of(:name).is_at_most(50) }
6    it { should validate_length_of(:device).is_at_most(250) }
7  end

```

Figura 65. Prueba unitaria a Employee

Elaboración: Los autores

```
F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/employee_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: previous definition of HABTM_Roles was here
...
Finished in 0.01563 seconds (files took 14.75 seconds to load)
3 examples, 0 failures
```

Figura 66. Resultados de la prueba a Employee

Elaboración: Los autores

Como se puede apreciar en la figura 63 se muestran las tres pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tienen fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos a un empleado, es necesario ingresar el nombre del trabajador.
2. Para registrar o modificar en la base de datos a un empleado, el nombre debe poseer no más de 50 caracteres.
3. Para registrar o modificar en la base de datos a un empleado, el campo dispositivo móvil del trabajador debe poseer no más de 250 caracteres.

e) Clase Maintenance

En las siguientes figuras, se muestran las pruebas unitarias de la clase Maintenance:

```
maintenance_spec.rb
1 require 'rails_helper'
2
3 RSpec.describe Maintenance, type: :model do
4   it { should belong_to(:project) }
5   it { should validate_presence_of(:project) }
6   it { should belong_to(:employee) }
7   it { should validate_presence_of(:employee) }
8   it { should validate_length_of(:indication).is_at_most(500) }
9   it { should validate_presence_of(:scheduled_date) }
10  it { should validate_presence_of(:scheduled_time) }
11 end
```

Figura 67. Prueba unitaria a Maintenance

Elaboración: Los autores

```
F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/maintenance_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: previous definition of HABTM_Roles was here
.....
Finished in 0.07812 seconds (files took 14.08 seconds to load)
7 examples, 0 failures
```

Figura 68. Resultados de la prueba a Maintenance

Elaboración: Los autores

Como se puede apreciar en la figura 65 se muestran las siete pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tienen fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos, una visita planeada, es necesario que el producto vendido esté registrado.
2. Para registrar o modificar en la base de datos, una visita planeada, es necesario indicar el producto vendido a visitar.
3. Para registrar o modificar en la base de datos, una visita planeada, es necesario que el técnico que va a realizar la visita este registrado.
4. Para registrar o modificar en la base de datos, una visita planeada, es necesario indicar el técnico que va a realizar la visita, no necesariamente debe estar registrado.
5. Para registrar o modificar en la base de datos, una visita planeada, las indicaciones del cliente deben poseer no más de 500 caracteres.
6. Para registrar o modificar en la base de datos, una visita planeada, se debe indicar la fecha a realizarse.
7. Para registrar o modificar en la base de datos, una visita planeada, se debe indicar la hora a realizarse.

f) Clase Project

En las siguientes figuras se muestran las pruebas unitarias de la clase Project:

```
project_spec.rb
1 require 'rails_helper'
2
3 RSpec.describe Project, type: :model do
4   it { should belong_to(:customer) }
5   it { should validate_presence_of(:customer) }
6   it { should belong_to(:type_project) }
7   it { should validate_presence_of(:type_project) }
8   it { should validate_length_of(:description).is_at_most(500) }
9   it { should validate_presence_of(:sale_date) }
10  it { should validate_length_of(:number_proof_payment).is_at_most(25) }
11 end
```

Figura 69. Prueba unitaria a Project

Elaboración: Los autores

```
F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/project_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: previous definition of HABTM_Roles was here
.....
Finished in 0.10936 seconds (files took 14.63 seconds to load)
7 examples, 0 failures
```

Figura 70. Resultados de la prueba a Project

Elaboración: Los autores

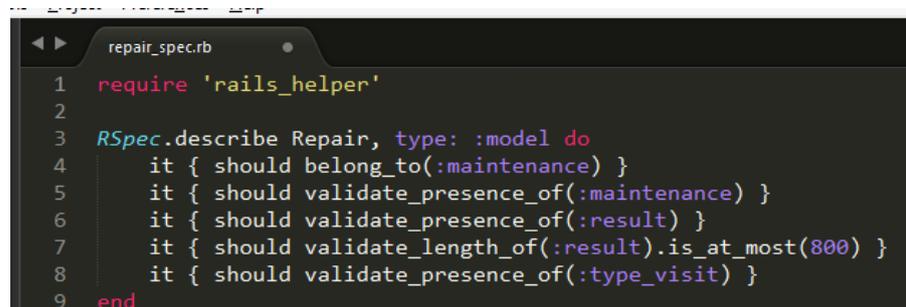
Como se puede apreciar, en la figura 67, se muestran las siete pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tienen fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos, un producto vendido, es necesario que el cliente que compró la máquina esté registrado.
2. Para registrar o modificar en la base de datos, un producto vendido, es necesario indicar el cliente que compró la máquina, no necesariamente debe estar registrado.
3. Para registrar o modificar en la base de datos, un producto vendido, es necesario que su categoría esté registrada.

4. Para registrar o modificar en la base de datos, un producto vendido, es necesario indicar su categoría, no necesariamente debe estar registrada.
5. Para registrar o modificar en la base de datos, un producto vendido, su descripción debe poseer no más de 500 caracteres.
6. Para registrar o modificar en la base de datos, un producto vendido, se debe indicar su fecha de venta.
7. Para registrar o modificar en la base de datos, un producto vendido, el número de comprobante de pago debe poseer no más de 25 caracteres.

g) Clase Repair

En las siguientes figuras, se muestran las pruebas unitarias de la clase Repair:



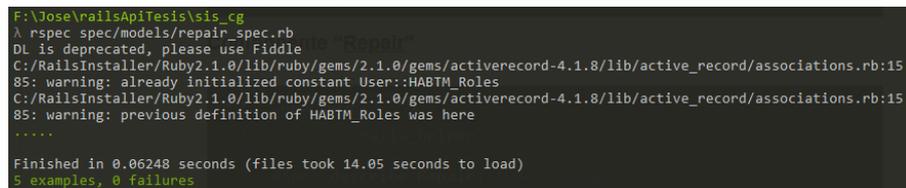
```

1 require 'rails_helper'
2
3 RSpec.describe Repair, type: :model do
4   it { should belong_to(:maintenance) }
5   it { should validate_presence_of(:maintenance) }
6   it { should validate_presence_of(:result) }
7   it { should validate_length_of(:result).is_at_most(800) }
8   it { should validate_presence_of(:type_visit) }
9 end

```

Figura 71. Prueba unitaria a Repair

Elaboración: Los autores



```

F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/repair_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: previous definition of HABTM_Roles was here
.....
Finished in 0.06248 seconds (files took 14.05 seconds to load)
5 examples, 0 failures

```

Figura 72. Resultados de la prueba a Repair

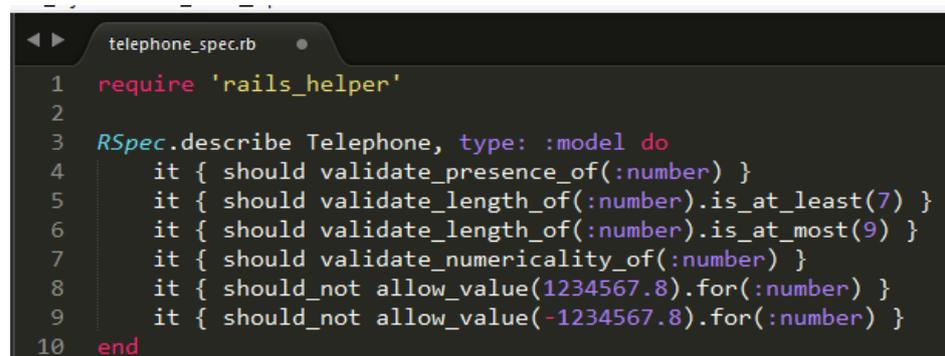
Elaboración: Los autores

Como se puede apreciar, en la figura 69, se muestran las cinco pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tienen fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos, una visita realizada, es necesario tener registrada su planeación.
2. Para registrar o modificar en la base de datos, una vista realizada, es necesario indicar el código de planeación, no necesariamente debe estar registrado.
3. Para registrar o modificar en la base de datos, una visita realizada, es necesario indicar los resultados encontrados.
4. Para registrar o modificar en la base de datos, una visita realizada, los resultados encontrados deben poseer no más de 800 caracteres.
5. Para registrar o modificar en la base de datos, una visita realizada, se debe indicar su categoría.

h) Clase Telephone

En las siguientes figuras, se muestran las pruebas unitarias de la clase Telephone:



```
1 require 'rails_helper'
2
3 RSpec.describe Telephone, type: :model do
4   it { should validate_presence_of(:number) }
5   it { should validate_length_of(:number).is_at_least(7) }
6   it { should validate_length_of(:number).is_at_most(9) }
7   it { should validate_numericality_of(:number) }
8   it { should_not allow_value(1234567.8).for(:number) }
9   it { should_not allow_value(-1234567.8).for(:number) }
10 end
```

Figura 73. Prueba unitaria a Telephone

Elaboración: Los autores

```
F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/telephone_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:15
85: warning: previous definition of HABTM_Roles was here
.....
Finished in 0.03125 seconds (files took 14.63 seconds to load)
6 examples, 0 failures
```

Figura 74. Resultados a la prueba Telephone

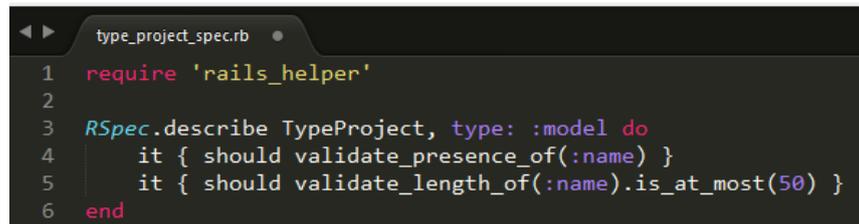
Elaboración: Los autores

Como se puede apreciar, en la figura 71, se muestran las seis pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tiene fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos a un teléfono, es necesario indicar su número.
2. Para registrar o modificar en la base de datos a un teléfono, su número debe poseer al menos 7 caracteres.
3. Para registrar o modificar en la base de datos a un teléfono, su número debe poseer no más 9 caracteres.
4. Para registrar o modificar en la base de datos a un teléfono, su número solo debe contener valores numéricos enteros.
5. Al registrar o modificar un número de teléfono no debe permitirse el valor "1234567.8".
6. Al registrar o modificar un número de teléfono no debe permitirse el valor "-1234567.8".

i) Clase TypeProject

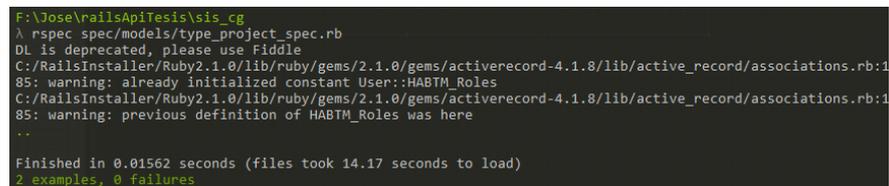
En las siguientes figuras, se muestran las pruebas unitarias de la clase TypeProject:



```
type_project_spec.rb
1 require 'rails_helper'
2
3 RSpec.describe TypeProject, type: :model do
4   it { should validate_presence_of(:name) }
5   it { should validate_length_of(:name).is_at_most(50) }
6 end
```

Figura 75. Prueba unitaria a TypeProject

Elaboración: Los autores



```
F:\Jose\railsApiTesis\sis_cg
λ rspec spec/models/type_project_spec.rb
DL is deprecated, please use Fiddle
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:1
85: warning: already initialized constant User::HABTM_Roles
C:/RailsInstaller/Ruby2.1.0/lib/ruby/gems/2.1.0/gems/activerecord-4.1.8/lib/active_record/associations.rb:1
85: warning: previous definition of HABTM_Roles was here
..
Finished in 0.01562 seconds (files took 14.17 seconds to load)
2 examples, 0 failures
```

Figura 76. Resultados de prueba a TypeProject

Elaboración: Los autores

Como se puede apreciar, en la figura 73, se muestran las dos pruebas y en la figura siguiente, el resultado de su ejecución, los mismos que indican que no tienen fallas. Las pruebas mencionadas, se detallan a continuación:

1. Para registrar o modificar en la base de datos, una categoría de producto, es necesario indicar su nombre.
2. Para registrar o modificar en la base de datos, una categoría de producto, su nombre debe poseer no más 50 caracteres.

3.3 Bondades y características del producto

El sistema de información desarrollado posee las siguientes características y bondades:

- Acceso ubicuo a la información de post-venta registrada por medio del sistema.

- Interfaces fáciles de utilizar.
- Comunicación permanente con la aplicación para dispositivos móviles (como celular), la cual contendrá el módulo para el registro del servicio técnico.
- Diseño adaptativo a cualquier resolución de pantalla.
- Suscripción fácil a través del pago mensual por licencia de usuario y fácil cancelación del servicio en cualquier momento.
- Encapsulamiento de la infraestructura tecnológica, para la Mipyme es transparente los componentes tecnológicos utilizados.
- Soportado con el modelo de servicio Plataforma como Servicio.

CAPÍTULO IV DISCUSIÓN Y APLICACIONES

4.1 Discusión

4.1.1 Objetivo general

Se logró mejorar el servicio de post-venta al cliente cuyos indicadores son: 90% de los clientes son atendidos oportunamente según las directrices establecidas y mínimo el 80% de los clientes encuestados deberán estar satisfechos con el nuevo servicio de post-venta.

Para verificar el primer indicador se realizó una evaluación en el mes de agosto del 2015, los resultados se muestran a continuación:

Tabla 14. Clientes atendidos

Ago-15		
Empresa	Clientes atendidos	Clientes atendidos oportunamente según las directrices establecidas
INNOMECC	8	8
Procesos Continuos	12	11

Elaboración: Los autores

Del cuadro anterior, se desprende o infiere que ambas empresas en estudio atendieron a más del 90% de sus clientes de manera oportuna según su flujo de trabajo definido, en caso de Procesos Continuos tuvo un inconveniente con un cliente, quien tuvo que ser atendido directamente con el gerente general.

Para verificar el segundo indicador, se realizó una encuesta a los principales clientes de ambas empresas. Los resultados se muestran a continuación:

Tabla 15. Mejora en el servicio de post-venta de INNOMECE

INNOMECE		
Grado de mejora en el servicio de post-venta	Cantidad de clientes	Porcentaje
Muy buena	8	67%
Buena	3	25%
Regular	1	8%
Poco	0	0%
Total	12	100%

Elaboración: Los autores

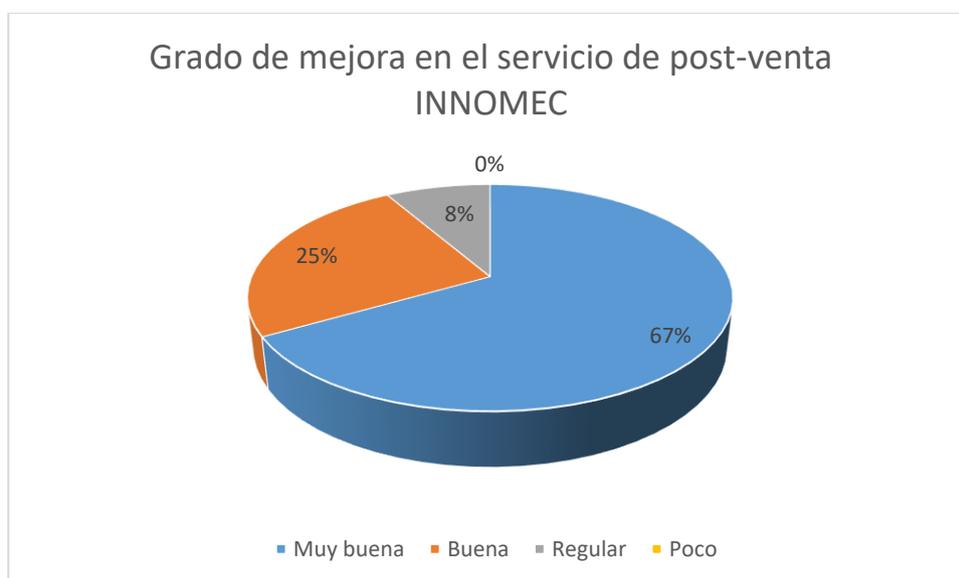


Figura 77. Mejora en el servicio de post-venta de INNOMECE

Elaboración: Los autores

Del cuadro y gráfico anterior, el 92% de los clientes encuestados opinan que hubo una mejora sustancial en el servicio de post-venta.

Asimismo, en la evaluación realizada en Procesos Continuos, se muestran los siguientes resultados:

Tabla 16. Mejora en el servicio de post-venta Procesos Continuos

Procesos Continuos		
Grado de mejora en el servicio de post-venta	Cantidad de clientes	Porcentaje
Muy buena	5	50%
Buena	3	30%
Regular	2	20%
Poco	0	0%
Total	10	100%

Elaboración: Los autores

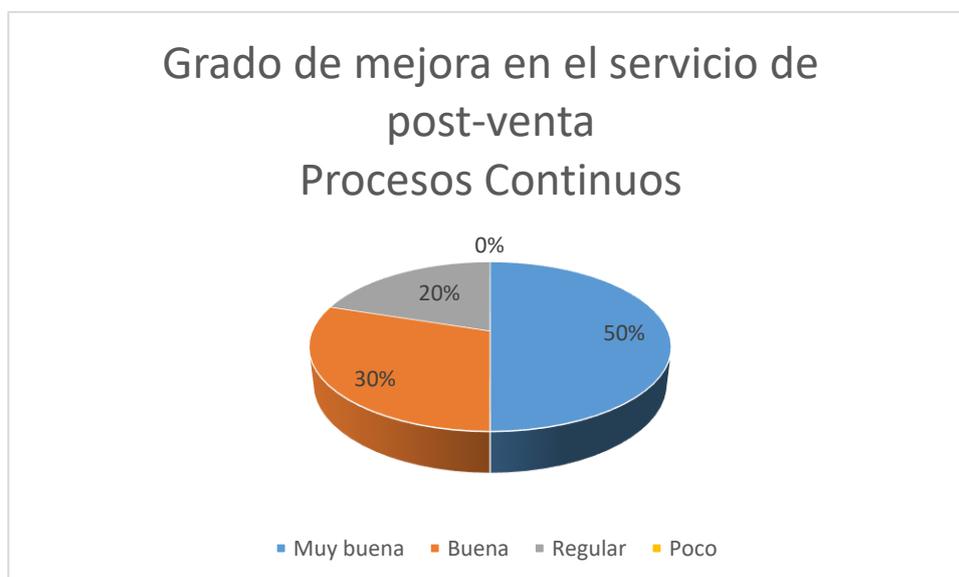


Figura 78. Mejora en el servicio de post-venta de Procesos Continuos

Elaboración: Los autores

Del cuadro anterior, el 80% de los clientes encuestados opinan que el servicio de post-venta mejoró respecto a cómo lo venían brindando.

4.1.2 Objetivos específicos

a) Objetivo específico 1:

Se logró agilizar la programación de visitas al cliente, cuyos indicadores son: mínimo una reducción del 90% en el tiempo de la planificación de las visitas a los clientes y la eliminación del 60% de las actividades burocráticas en la planificación de visitas.

Se realizó una evaluación al proceso de planificación de visitas por cada empresa que forma parte del estudio, los resultados fueron:

- Se logró reducir el tiempo para la planificación de visitas en un 200%, antes se demoraban tres o cuatro días para concretar la planificación, ahora no más de un día.
- Se eliminaron cuatro actividades burocráticas, que representan más del 90% de las actividades de este tipo. Vale decir que no se eliminó al 100%, porque por el momento solo la consulta de las especificaciones de las garantías, más no su vigencia, se hace de forma tradicional.

Se presenta un cuadro que resume la evaluación realizada:

Tabla 17. Tiempo promedio de planificación de visitas

Tiempo promedio con el sistema tradicional	Tiempo promedio con el sistema de información
3 días	1 día

Elaboración: Los autores

Se observa que utilizando el sistema tradicional, la empresa demoraba en promedio tres días para concretar una cita con el cliente; en cambio, utilizando el sistema de información no demora más 1 día.

Las actividades del proceso de planificación de visita sufrieron cambios (se eliminaron actividades burocráticas), para ello se muestra el proceso con el sistema tradicional y luego con el sistema de información:

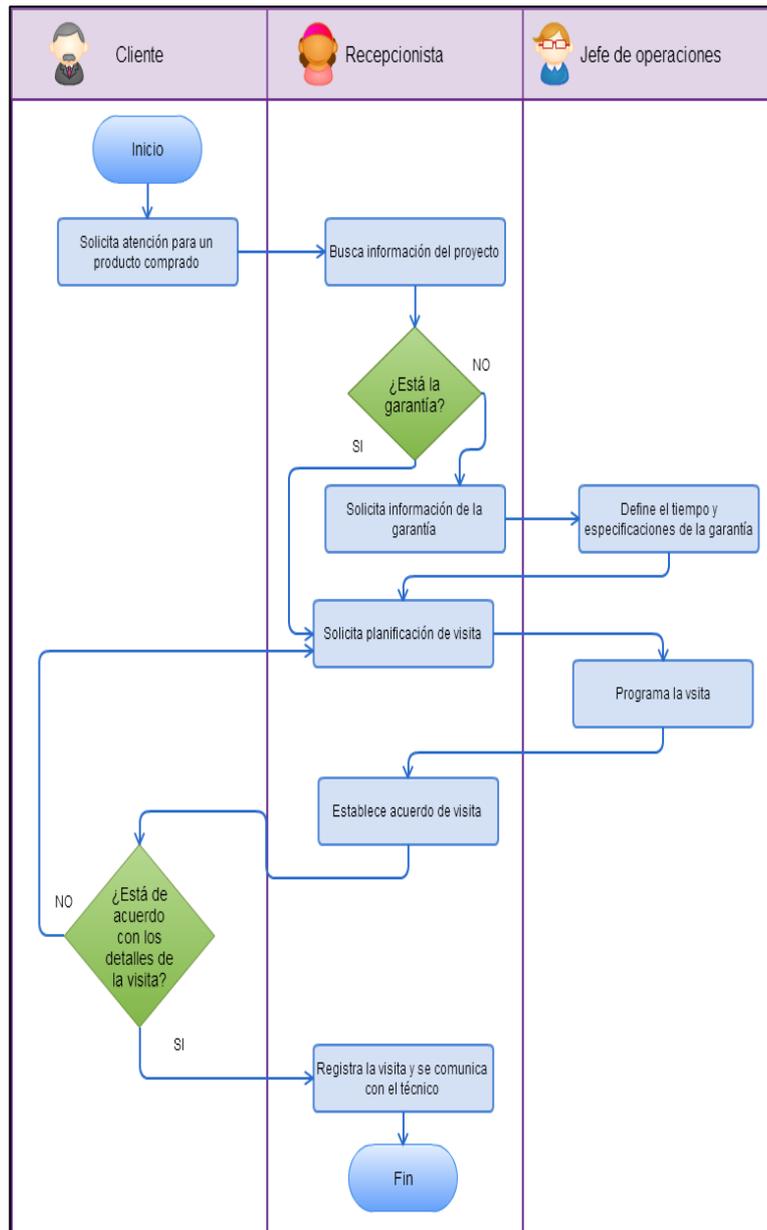


Figura 79. Diagrama de flujo del proceso de planificación de visita

Elaboración: Los autores

En la figura anterior, el proceso de planificación de visita comienza desde la atención al cliente por algún inconveniente que tenga con su producto hasta la programación de la visita. Es importante mencionar que, durante el acotado proceso, la búsqueda de información se recolecta de todos los documentos relacionados con el producto vendido. Además, el establecimiento del tiempo de garantía, por lo general es en base a la fecha de fabricación del

producto, lo cual es incorrecto, porque debería de ser a base de la fecha vendida.

Se muestra el proceso cuando se utiliza el sistema de información desarrollado:

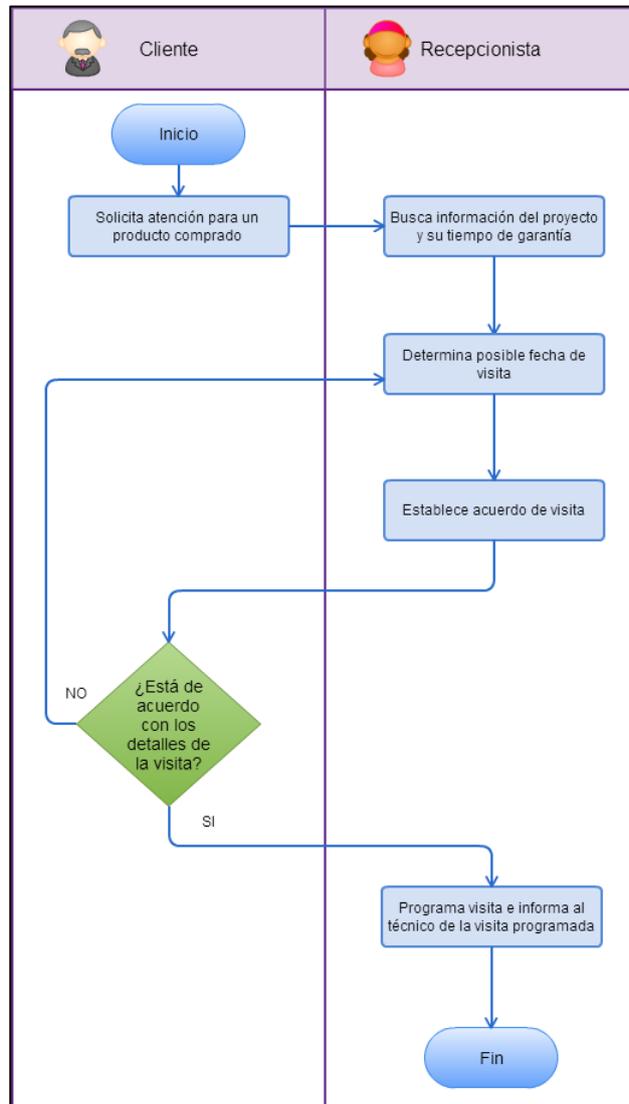


Figura 80. Diagrama de flujo del proceso de planificación de visita con el sistema

Elaboración: Los autores

En la figura anterior, se puede visualizar que el proceso inicial cambio sus actividades, ahora la consulta y programación de las visitas se realiza de forma automatizada. Además, la información de cada proyecto o producto vendido está registrado detalladamente y puede ser consultado en cualquier momento.

b) Objetivo específico 2:

Se logró desarrollar un sistema de información de post-venta integral, cuyo indicador es contar con su implementación. Las pruebas que garantizan el desarrollo del sistema de información se encuentran en el capítulo II. Metodología, en la sección desarrollo del proyecto, apartado codificación y pruebas.

El sistema de servicio de post-venta al cliente está compuesto por un software web, una aplicación móvil y soportado bajo un determinado modelo de negocio. Se muestra un cuadro que prueba la cronología del desarrollo del sistema:

Tabla 18. Historial de desarrollo del proyecto

Meses de desarrollo	Componentes funcionales desarrollados	Porcentaje de desarrollo	Descripción
Mes 1 (Abril/2015)	3	20%	Inicio del desarrollo del software web y realización del modelo
Mes 2 (Mayo/2015)	8	53%	Término del desarrollo del software web y comienzo del desarrollo de la aplicación móvil
Mes 3 (Junio/2015)	4	27%	Término del desarrollo de la aplicación móvil
Total	15	100%	Término del desarrollo del producto

Elaboración: Los autores

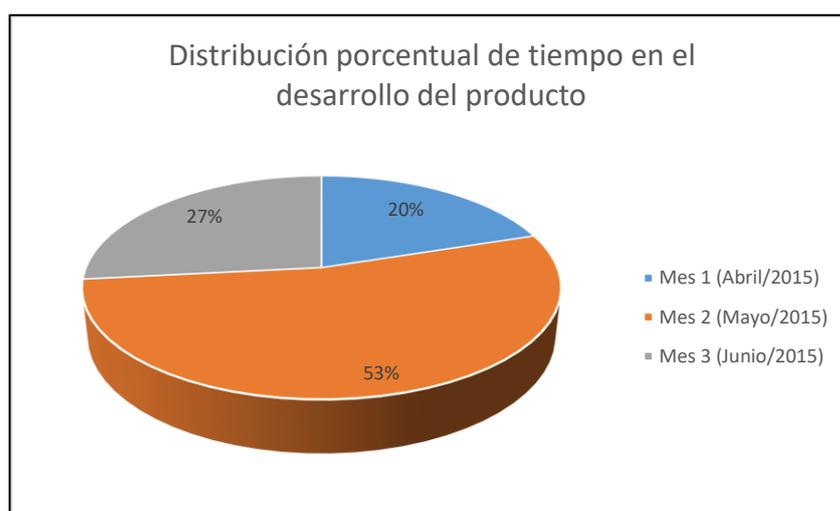


Figura 81. Distribución del tiempo en el desarrollo del proyecto

Elaboración: Los autores

En la figura anterior, se puede apreciar que en el mes de mayo del año 2015 hubo una mayor carga de trabajo, lográndose realizar la mayoría de componentes del producto final (sistema de servicio de post-venta al cliente), cabe mencionar que en ese mes se comenzó a implementar las funcionalidades principales del software web.

c) Objetivo específico 3:

Se logró impulsar la consolidación de información de post-venta, cuyo indicador es aumentar en 50% la cantidad de historiales de post-venta realizados en cada mes.

Se realizó una evaluación a ambas empresas presentes en el estudio. Los resultados fueron:

Tabla 19. Historiales de post-venta en INNOMECE

INNOMECE	
Historiales realizados mensualmente con el sistema tradicional	Historiales realizados en Julio/2015 con el sistema de información
2	5

Elaboración: Los autores

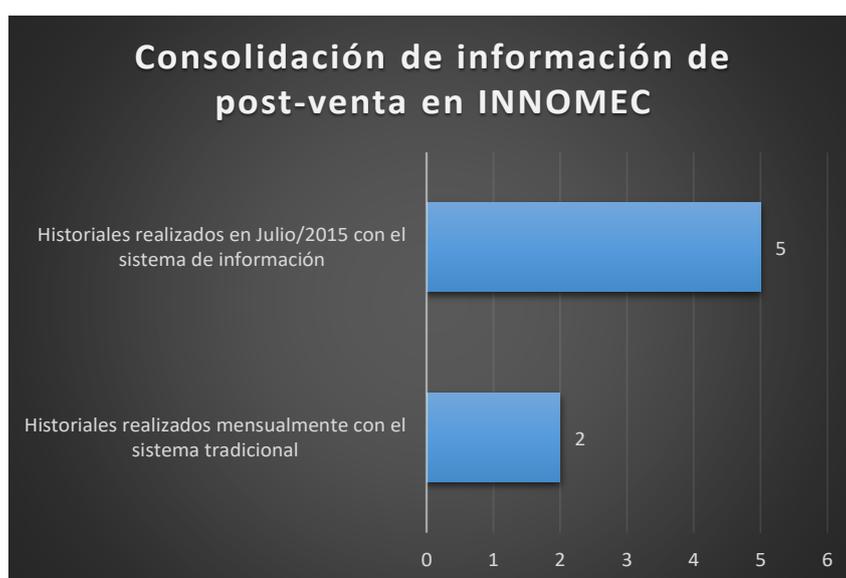


Figura 82. Consolidación de información en INNOMECE

Elaboración: Los autores

En el cuadro y figura anterior, se observa que mensualmente INNOMECE consolidaba poca información de post-venta, en cambio con el sistema de información se mejoró en 150% la cantidad de historiales que realizan. Vale decir que en cualquier momento INNOMECE puede tener a su disposición historiales de los productos más vendidos y de las máquinas con mayores problemas, además de ver el desempeño de los trabajadores luego del servicio técnico.

Con relación a la segunda empresa en estudio, los resultados se muestran a continuación:

Tabla 20. Historial de post-venta en Procesos Continuos

Procesos Continuos	
Historiales realizados mensualmente con el sistema tradicional	Historiales realizados en Julio/2015 con el sistema de información
2	4

Elaboración: Los autores

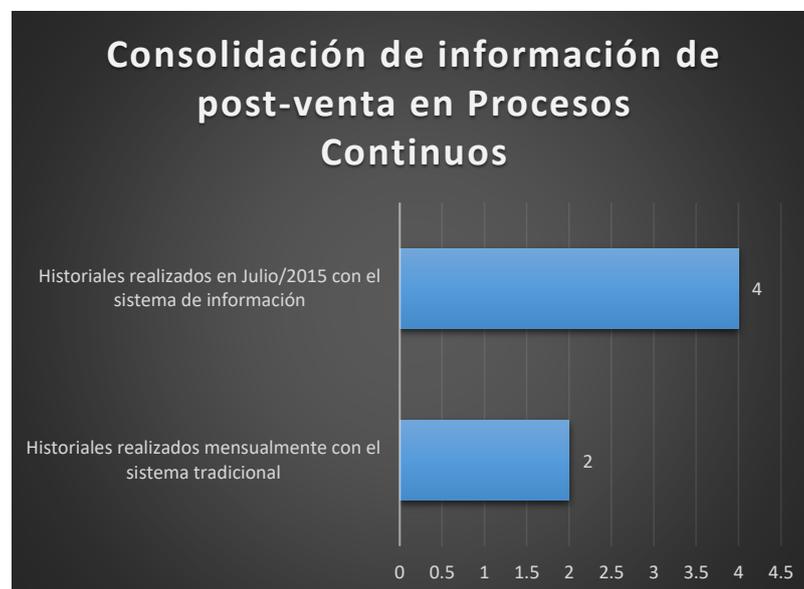


Figura 83. Consolidación de información en Procesos Continuos

Elaboración: Los autores

Del cuadro y gráfico anterior, se aprecia que hay una mejora del 100% respecto a la cantidad de historiales realizados en un mes.

4.1.3 Procesos

El proceso de consolidación de información sufrió una serie de cambios importantes, como ejemplo, se muestra el proceso utilizando el sistema tradicional:

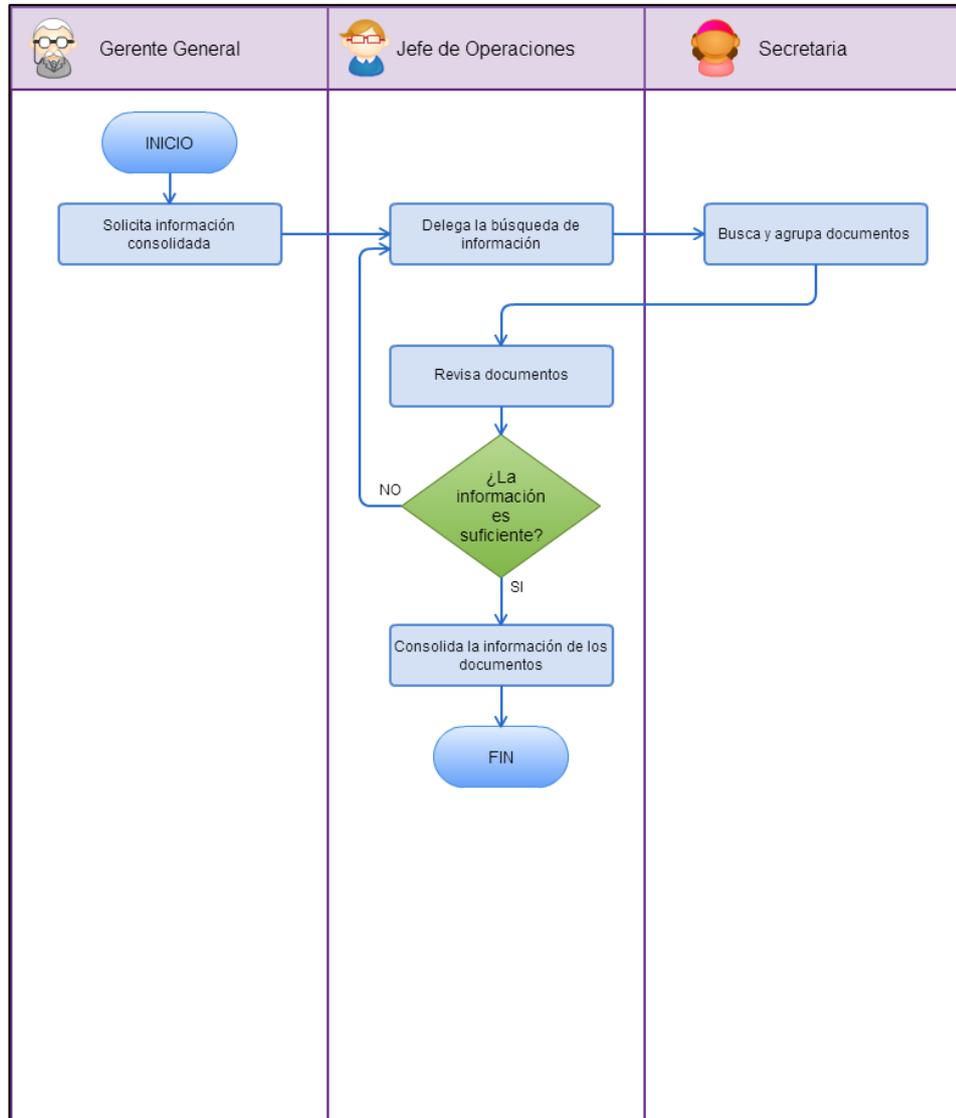


Figura 84. Diagrama de flujo del proceso de consolidación de información

Elaboración: Los autores

En la figura anterior, el proceso de consolidación de información consiste en recolectar toda la información relevante de los proyectos a base de los criterios definidos por el Gerente General para luego agruparlos y mostrarlos a la gerencia.

Usando el sistema de información, el proceso tiene el siguiente flujo de actividades:

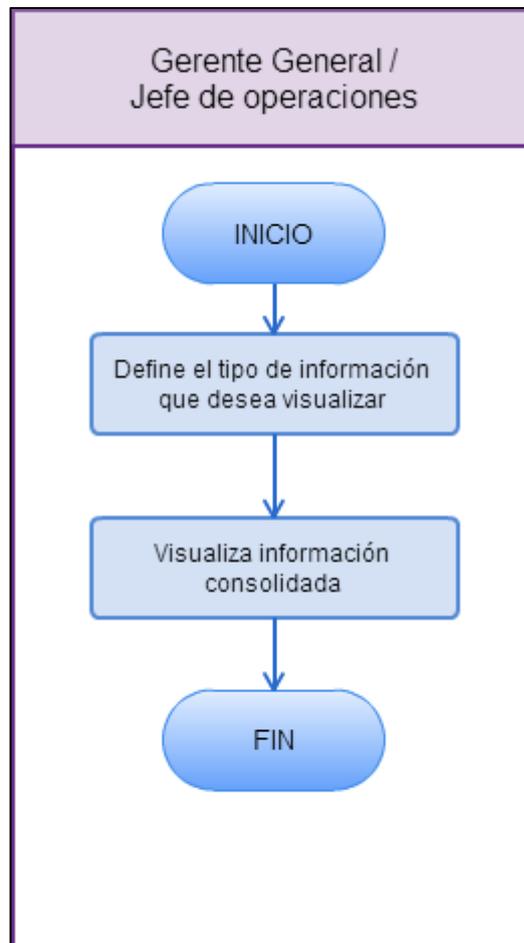


Figura 85. Diagrama de flujo del proceso de consolidación de info. con el sistema

Elaboración: Los autores

En la figura anterior, el proceso de consolidación de información, cambia en beneficio de la empresa, porque ahora el usuario solo debe definir la información que desea visualizar y esta se muestra en la pantalla, ya no es necesario recolectar y agrupar la información, porque estas actividades son parte de las funciones del sistema de información.

Asimismo, el proceso de servicio técnico también tuvo cambios sustanciales en el registro de los resultados de la visita, para ver dichos cambios primero se muestra el proceso usando el sistema tradicional:

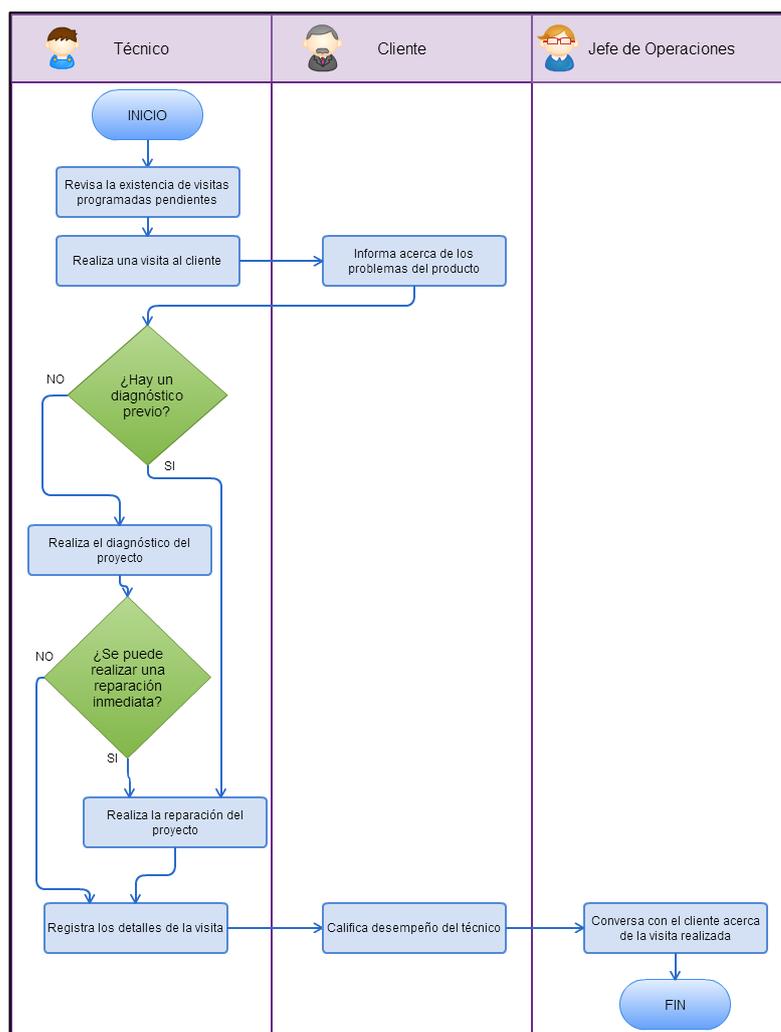


Figura 87. Diagrama de flujo del proceso servicio técnico con el sistema

Elaboración: Los autores

En la figura anterior, se puede observar que ya no es necesario que la recepcionista informe al técnico de las visitas pendientes a los clientes. Además, los detalles de la visita se registran en la aplicación móvil de tal manera que la información recabada durante la visita sirve de ayuda a la consolidación de información de post-venta.

Cabe mencionar, que la modificación del proceso de planificación de visita al cliente fue revisada en la sección anterior referida al objetivo específico 1.

4.2 Logros realizados

Se desarrolló un sistema de información que según las pruebas realizadas permitió que mejore considerablemente la calidad del servicio de post-venta, contribuyendo a generar:

- Fidelización de los clientes porque se puede solucionar con celeridad los problemas o inquietudes acerca de los proyectos o productos vendidos.
- Información oportuna y persistente, porque cuando requieran pueden tener la información actualizada acerca de los productos vendidos, sus garantías y reparaciones.
- Procesos trazables, porque pueden rastrear y relacionar el flujo de información en cualquier momento de los servicios de atención de garantías, planificación de visitas y reparación de equipos.

Asimismo, los resultados obtenidos en las pruebas aseguran:

- La eliminación de actividades burocráticas respecto a los procesos relacionados con el sistema de información desarrollado.
- Reducción de tiempo en el servicio de post-venta.
- Facilidad en la obtención de información consolidada.
- Mejora en el valor agregado del servicio de post-venta al cliente.

4.3 Campo de aplicación

El sistema de información de servicio de post-venta desarrollado es aplicable para las micros, pequeñas y medianas empresas del sector metalmeccánico del Perú; sin embargo, cualquier empresa que realice actividades manufactureras podría utilizar el sistema mencionado.

El sistema implementado abarca la consulta de la vigencia de la garantía y visualización de la factura de manera digital por cada producto vendido, consulta de la información inherente y relevante al proyecto como el tipo de proyecto, el cliente al que se le vendió y la descripción. Además, se puede conocer la disponibilidad del técnico al realizar la planificación de la visita al cliente, registrar las actividades durante la visita a través de la

aplicación para dispositivos móviles o por medio del software web. También, se puede observar un historial de visitas por cada producto e información consolidada automatizada acerca de los productos vendidos según su categoría, así como, el desempeño de los técnicos de acuerdo con el puntaje obtenido durante la visita al cliente.

CONCLUSIONES

1. El servicio de post-venta mejoró a partir de que más del 90% de los clientes son atendidos según sus directrices definidas, y el 92% de los clientes opinan que mejoró el servicio de post-venta considerablemente.
2. Se agilizó la programación de visitas al cliente logrando planificar una visita en no más de un día.
3. Se desarrolló un sistema orientado hacia el servicio de post-venta al cliente, que está conformado por tres componentes: un software web, una aplicación móvil integrada con el acotado software y un modelo de negocio con la capacidad de soportar los requerimientos de las Mipymes metalmecánicas; y con la finalidad de brindar información específica y consolidada a sus usuarios.
4. Se impulsó la consolidación de información de post-venta aumentando en un mínimo de 90% los historiales automatizados de acuerdo con los productos vendidos.

RECOMENDACIONES

1. Las Mipymes, en cada periodo, deben evaluar directa e indirectamente el servicio de post-venta brindado a través de indicadores con la finalidad de detectar las debilidades en sus procesos y mejorarlos.
2. Se recomienda extender el módulo de servicio al cliente, específicamente, la sección de garantías, de tal manera que no solo contemple su vigencia (el tiempo), sino también sus especificaciones.
3. Se debe realizar el desarrollo de APIs de las funcionalidades principales que presenta el software web en la versión próxima, para lograr comunicarse completamente con la aplicación móvil instalable, ya que por el momento solo se puede efectuar el registro del mantenimiento o reparación del producto, y la calificación del técnico.
4. Antes de comenzar a desarrollar o ampliar las interfaces de usuario se exhorta a realizar un estudio de las tendencias de diseño con la finalidad de conocer los nuevos gustos de los usuarios y brindarles satisfacción al navegar por el sistema.

5. Se recomienda incorporar la funcionalidad de consultas de disponibilidad de repuestos tanto en el software web como en la aplicación móvil en su siguiente versión.

FUENTES DE INFORMACIÓN

BIBLIOGRÁFICAS:

1. Castaño, A. d., Velthuis, M. P., & Martinez, E. M. (2000). Diseño de Base de Datos Relacionales. Mexico: Alfomega.
2. Choy, F. (2014). Marco Lógico - Análisis de Objetivos. Diplomado de Parques Industriales. Lima: Universidad San Martín de Porres.
3. Choy, F. (2014). Marco Lógico - Análisis de Problemas. Diplomado de Parques Industriales. Lima: Universidad San Martín de Porres.
4. Marinescu, D. C. (2013). Cloud Computing Theory and Practice. Elsevier.
5. Marks, E. A., & Lozano, B. R. (2010). Executive's Guide to Cloud Computing. Wiley.
6. Pressman, R. (2006). Ingeniería del Software. México: The McGraw-Hill Companies.

ELECTRÓNICAS:

1. Adobe PhoneGap. (2015). PhoneGap Documentation. Obtenido de <http://docs.phonegap.com/>
2. Alerta en Línea. (2011). Aplicaciones móviles: Qué son y cómo funcionan. Obtenido de <https://www.alertaenlinea.gov/articulos/s0018-aplicaciones-m%C3%B3viles-qu%C3%A9-son-y-c%C3%B3mo-funcionan>
3. Barrón, G. G. (2015). SEMANAeconómica. Obtenido de <http://semanaeconomica.com/tecnologiaentempresa/2015/01/30/modernizacion-de-las-pymes-peruanas-beneficios-de-las-tics/>
4. Biznews. (2011). Industria metalmeccanica peruana ya tiene estándar internacional. Obtenido de <http://biznews.pe/boletines-de-prensa/industria-metalmeccanica-peruana-ya-tiene-estandar-internacional>
5. CampusMVP. (2014). Programación móvil: Qué herramientas y lenguaje elegir. Obtenido de <http://www.campusmvp.es/recursos/post/Programacion-movil-Que-herramienta-y-lenguaje-elegir.aspx>
6. CAVSI. (2015). ¿Qué es un Sistema Gestor de Base de Datos o SGDB? Obtenido de <http://www.cavsi.com/preguntasrespuestas/que-es-un-sistema-gestor-de-bases-de-datos-o-sgdb/>
7. Centro de Lengua y Pensamiento Crítico. (2016). Manual del modelo de documentación de la Asociación de Psicología. Obtenido de <http://online.upaep.mx/LPC/online/apa/APAimp.pdf>
8. Chiyana, S. (2015). Aplicaciones móviles híbridas. Obtenido de <http://inusual.com/articulos/aplicaciones-moviles-hibridas/>
9. Choy, F. (2014). Marco Lógico - Análisis de Objetivos. Diplomado de Parques Industriales. Lima: Universidad San Martín de Porres.
10. Choy, F. (2014). Marco Lógico - Análisis de Problemas. Diplomado de Parques Industriales. Lima: Universidad San Martín de Porres.
11. CreceNegocios. (2015). El VAN y el TIR. Obtenido de <http://www.crecenegocios.com/el-van-y-el-tir/>
12. Cuello, J., & Vittone, J. (2015). Diseñando apps para móviles. Obtenido de <http://www.appdesignbook.com/es/contenidos/las-aplicaciones/>
13. Culturación. (2015). ¿Para qué se usan los editores de texto? Obtenido de <http://culturacion.com/para-que-se-usan-los-editores-de-texto/>

14. Definicion.de. (2015). Definición de Software. Obtenido de <http://definicion.de/software/>
15. Definicionabc. (2015). Definición de Navegador. Obtenido de <http://www.definicionabc.com/tecnologia/navegador.php>
16. Diaz, I. (2014). ¿Que es FrontEnd Y Backend en la programación web? Obtenido de <http://serprogramador.es/que-es-frontend-y-backend-en-la-programacion-web/>
17. Dueñas Villamiel, J. (2013). ¿Qué es el Parallax y cómo se hace? Obtenido de 40defiebre: <http://www.40defiebre.com/nueva-dimension-web-parallax/>
18. duplika. (2010). ¿Qué son los servidores web y por qué son necesarios? Obtenido de <https://www.duplika.com/blog/que-son-los-servidores-web-y-por-que-son-necesarios>
19. EcuRed. (2015). Lenguaje de Programación. Obtenido de http://www.ecured.cu/index.php/Lenguaje_de_Programaci%C3%B3n
20. EcuRed. (2015). Servidor Web. Obtenido de http://www.ecured.cu/index.php/Servidor_Web
21. El androide libre. (2014). ¿Qué es Material Design? Obtenido de <http://www.elandroidelibre.com/2014/11/que-es-material-design.html>
22. Extreme Programming. (2013). Acceptance Tests. Obtenido de <http://www.extremeprogramming.org/rules/functionaltests.html>
23. Extreme Programming. (2013). The Rules of Extreme Programming. Obtenido de <http://www.extremeprogramming.org/rules.html>
24. FOOD+i Cluster. (2015). ¿Qué son los CLUSTERS? Obtenido de <http://clusterfoodmasi.es/cluster/que-son-los-clusters/>
25. Gestión. (2012). SNI: Producción metalmecánica crecería hasta en un 10% este año. Obtenido de <http://gestion.pe/economia/sni-produccion-metalmecanica-creceria-hasta-10-este-ano-2013770>
26. Gestión. (2013). Las exportaciones del sector metalmecánico aumentaron un 12% durante el 2012. Obtenido de <http://gestion.pe/economia/exportaciones-metalmecanicas-aumentaron-12-durante-2012-2059844>
27. Gestión. (2014). Casi tres de cada cinco Mipymes en Perú no usan servicios de computación en la nube. Obtenido de <http://gestion.pe/tecnologia/casi-tres-cada-cinco-mipymes-peru-no-usan-servicios-computacion-nube-2117874>
28. Google. (2015). Material Design. Obtenido de <https://www.google.com/design/spec/material-design/introduction.html>

29. Guia Digital. (2015). ¿Qué es la Usabilidad? Obtenido de <http://www.guiadigital.gob.cl/articulo/que-es-la-usabilidad>
30. Gutiérrez, J. (2006). ¿Qué es un framework web? Obtenido de http://www.lsi.us.es/~javierj/investigacion_ficheros/Framework.pdf
31. HTML5FÁCIL. (2013). Ruby on Rails desde cero: Primeros pasos. Obtenido de <http://html5facil.com/tutoriales/ruby-on-rails-desde-cero-primeros-pasos/>
32. Ionic. (2015). Overview. Obtenido de <http://ionicframework.com/docs/overview/>
33. Isocrom Systems. (2015). ¿Qué es Open Source? Obtenido de <https://www.isocron.net/node/35>
34. LanceTalent. (2014). Los 3 tipos de aplicaciones móviles: ventajas e inconvenientes. Obtenido de <http://www.lancetalent.com/blog/tipos-de-aplicaciones-moviles-ventajas-inconvenientes/>
35. Letelier, P., & Penadés, C. (2006). Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). Obtenido de <http://www.cyta.com.ar/ta0502/v5n2a1.htm>
36. Librosweb. (2015). Mapeo de Objetos a Base de Datos (ORM). Obtenido de http://librosweb.es/libro/symfony_1_4/capitulo_1/conceptos_basicos.html
37. Maestros del Web. (2013). ¿Qué es Responsive Web Design? Obtenido de <http://www.maestrosdelweb.com/que-es-responsive-web-design/>
38. Márquez, A. (2015). Desarrollo de aplicaciones móviles multiplataforma. Obtenido de <http://asiermarques.com/2015/desarrollo-aplicaciones-multiplataforma/>
39. Microsoft Azure. (2015). Microsoft – the only vendor named a leader in Gartner Magic Quadrants for IaaS, Application PaaS, and Cloud Storage. Obtenido de <https://azure.microsoft.com/en-us/blog/microsoft-the-only-vendor-named-a-leader-in-gartner-magic-quadrants-for-iaas-application-paas-cloud-storage-and-hybrid/>
40. Ministerio de la Producción. (2015). Las MIPYMES en cifras 2013. Obtenido de <http://www.produce.gob.pe/remype/data/mype2013.pdf>
41. Morales, M. (2014). Manual de Administración de Proyectos. Obtenido de http://www.liderdeproyecto.com/manual/estructura_del_pmbok.html
42. Mozilla. (2015). CSS media queries. Obtenido de https://developer.mozilla.org/es/docs/CSS/Media_queries

43. Normas APA. (2016). Normas Apa 2016 - Edición 6. Obtenido de <http://normasapa.net/normas-apa-2016/>
44. Observatorio Nacional de las Telecomunicaciones y de la SI. (2012). Cloud Computing Retos y Oportunidades. Obtenido de http://www.ontsi.red.es/ontsi/sites/default/files/1-_estudio_cloud_computing_retos_y_oportunidades_vdef.pdf
45. Oficina de proyectos de informática. (2014). Plantilla del plan de pruebas de software. Obtenido de <https://sites.google.com/site/oficinaproyectosinformatica/home/archivo/PMOInformatica%20Plantilla%20de%20Plan%20de%20Pruebas%20de%20Software.doc?attredirects=0&d=1>
46. Ortigón, E., Pacheco, J., & Prieto, A. (2006). Metodología del marco lógico para la planificación, el seguimiento y la evaluación de proyectos y programas. Obtenido de http://www.mef.gob.pe/contenidos/pol_econ/documentos/ILPES_CEP_AL_Marco_Logico_Metodologia.pdf
47. Ortiz, D., & Ruiz, G. (2015). Sistemas de servicio al cliente implementados en las empresas: un análisis de su eficiencia en el logro de la calidad del servicio. Obtenido de <http://www.redalyc.org/articulo.oa?id=81801513>
48. OVH. (2015). Glosario Cloud. Obtenido de <https://www.ovh.es/cloud/lexico.xml>
49. PhoneGap. (2015). Supported Features. Obtenido de <http://phonegap.com/about/feature/>
50. Project Management Institute. (2013). Guía de los Fundamentos de la Dirección de Proyectos. Obtenido de <http://www.fnmt.es/documents/10179/119827/Descargar+Documentaci%C3%B3n++Gesti%C3%B3n+de+Proyectos/b34b9d76-9e62-4fcb-adbd-a0e5d675b4b4>
51. Rails Guides. (2015). Getting Started with Rails. Obtenido de http://guides.rubyonrails.org/getting_started.html
52. redis. (2014). Software como Servicio. Obtenido de <http://www.redsis.com/index.php/soluciones/servicios-cloud/saas>
53. Rouse, M. (2015). Base de datos relacional. Obtenido de <http://searchdatacenter.techtarget.com/es/definicion/Base-de-datos-relacional>
54. Rouse, M. (2015). Infrastructure as a Service (IaaS) definition. Obtenido de <http://searchcloudcomputing.techtarget.com/definicion/Infrastructure-as-a-Service-iaaS>

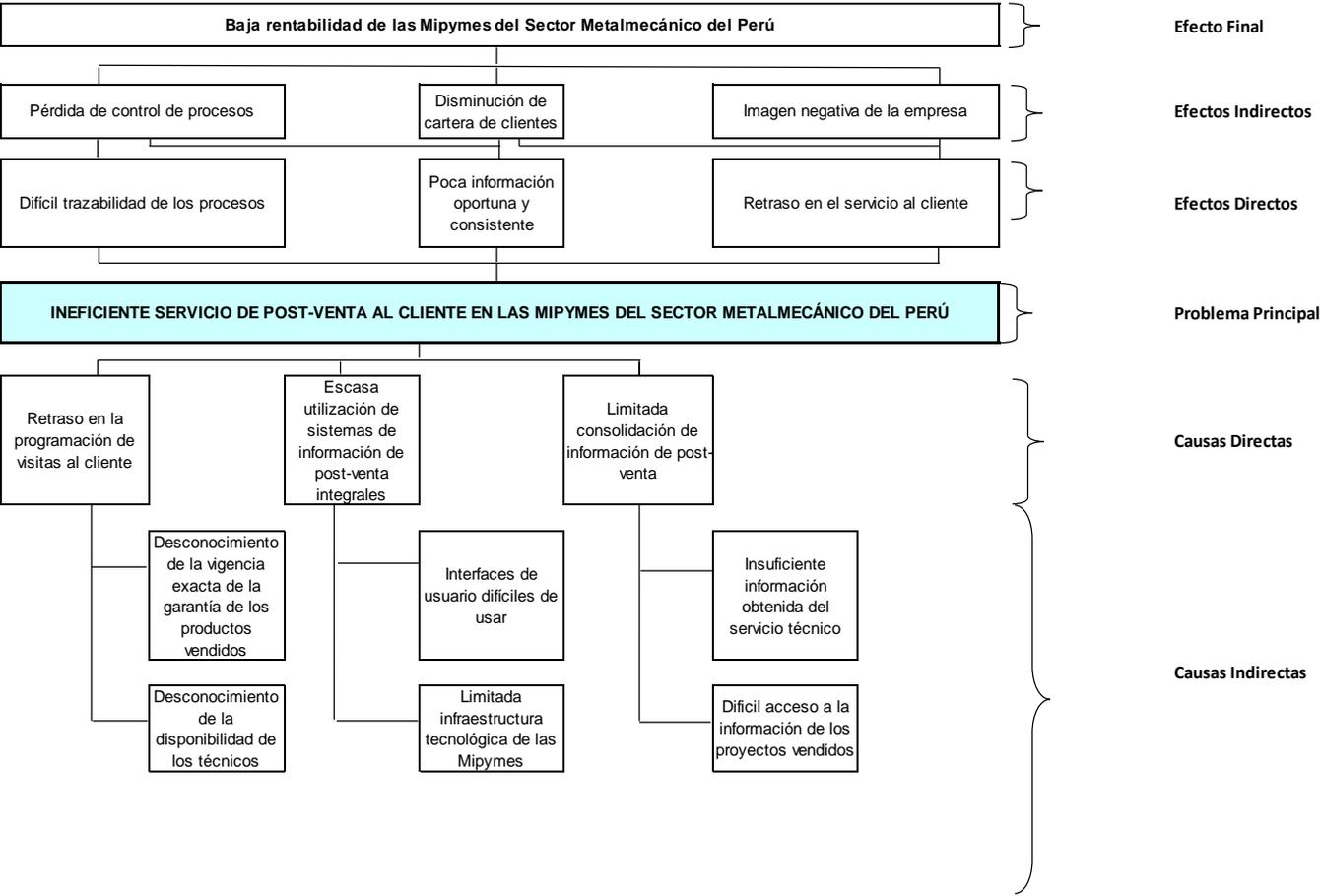
55. Rouse, M. (2015). Platform as a Service (PaaS) definition. Obtenido de <http://searchcloudcomputing.techtarget.com/definition/Platform-as-a-Service-PaaS>
56. Salesforce. (2015). Salesforce Named a Leader in Gartner's aPaaS Magic Quadrant. Obtenido de <https://www.salesforce.com/blog/2015/03/salesforce-named-a-leader-in-gartners-apaas-magic-quadrant.html>
57. SCM. (2003). Introducción a los Dispositivos Móviles. Obtenido de http://docencia.ac.upc.edu/EPSC/PSE/documentos/Trabajos/Archivo/Trabajo_PDM.pdf
58. SEMANAeconómica. (2012). SNI: Exportaciones metalmecánicas cerrarán año con crecimiento cercano a 30%. Obtenido de <http://semanaeconomica.com/article/economia/comercio-exterior/40758-sni-exportaciones-metalmecanicas-cerraran-ano-con-crecimiento-cercano-a-30-2/>
59. Sierra, M. (2015). Qué es un servidor y cuáles son los principales tipos de servidores. Obtenido de http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=542:que-es-un-servidor-y-cuales-son-los-principales-tipos-de-servidores-proxydns-webftppop3-y-smtp-dhcp&catid=57:herramientas-informaticas&Itemid=179
60. TICbeat. (2014). ¿Qué es una API y para qué sirve? Obtenido de <http://www.ticbeat.com/tecnologias/que-es-una-api-para-que-sirve/>
61. Universidad Católica Sedes Sapientiae. (2014). Características de las MYPES de metalmecánica y carpintería en Lima Norte: un análisis de Casos. Obtenido de http://www.ucss.edu.pe/osel/mercado_labor_forma/pdf/01_Estudios/2014/caracteristicas-de-las-mypes.pdf
62. Universidad de Murcia. (2006). Sistemas de Gestión de Base de datos y SIG. Obtenido de http://www.um.es/geograf/sigmur/sigpdf/temario_9.pdf
63. Universidad Nacional Autónoma de México. (2015). Recursos Web. Obtenido de <http://recursosweb.unam.mx/recursos-web/lineamientos-unam/de-uso/usabilidad/>
64. Vaquiro, J. (2015). Periodo de recuperación de la inversión - PRI. Obtenido de <http://www.pymesfuturo.com/pri.htm>
65. ZDNet. (2014). Amazon and Microsoft top Gartner's IaaS Magic Quadrant. Obtenido de <http://www.zdnet.com/article/amazon-and-microsoft-top-gartners-iaas-magic-quadrant/>

ANEXOS

	Página
1. Árbol de Problemas	130
2. Árbol de Objetivos	131
3. Matriz de Marco Lógico	132
4. Evaluación de Riesgos	133
5. Ingresos para los desarrolladores del sistema de información	135
6. Ingresos para la Mipyme que adquiere el sistema de información	136
7. Costo de desarrollo del sistema de información	137
8. Costos para la Mipyme que adquiere el sistema de información	139
9. Flujo de Caja para los desarrolladores del sistema de información	140
10. Flujo de Caja para la Mipyme que adquiere el sistema de info.	141
11. Análisis Beneficio/Costo para los desarrolladores	142
12. Indicadores VAN y TIR para los desarrolladores	143
13. Periodo de Recuperación de la Inversión para los desarrolladores	144
14. Análisis de Beneficio/Costo para la Mipyme	145
15. Indicadores VAN y TIR para la Mipyme	146
16. Periodo de Recuperación de la Inversión para la Mipyme	147
17. Selección de Sistema Gestor de Base de Datos	148
18. Selección de proveedor de PaaS de cloud computing	150
19. Selección de la metodología de desarrollo de software	152
20. Selección de framework de desarrollo	157
21. Selección de framework para realizar aplicaciones móviles multiplataforma basadas en HTML5	159
22. Selección de framework de diseño para realizar aplicaciones móviles multiplataforma basadas en HTML5	162
23. Plan de Trabajo	164
24. Historias de usuario	180
25. Tarjetas CRC	185
26. Script de la Base de Datos	191
27. Resultados de pruebas de actividades del proyecto	193

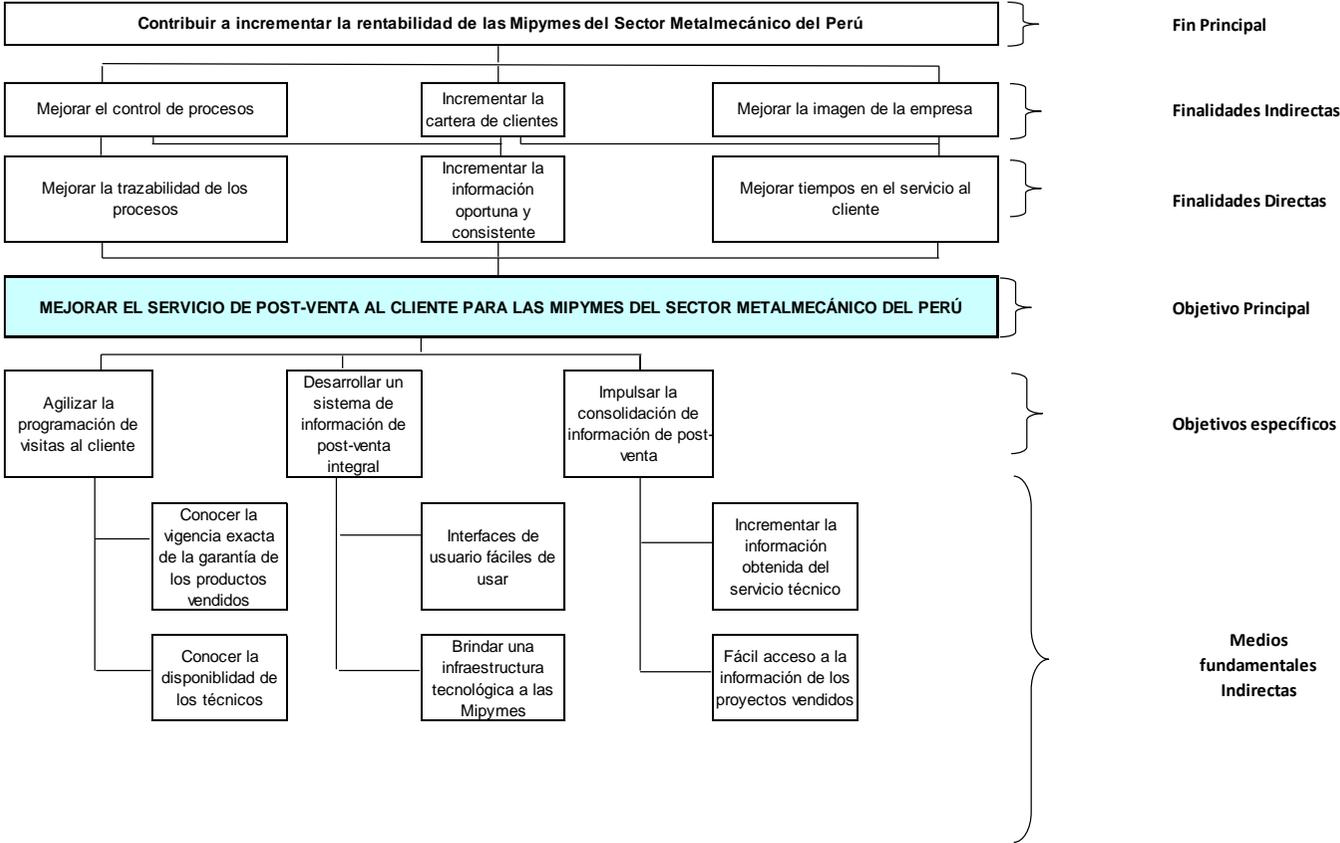
Anexo N° 1 – Árbol de Problemas

ARBOL DE PROBLEMAS



Anexo N° 2 – Árbol de Objetivos

ARBOL DE OBJETIVOS



Anexo N° 3 – Matriz de Marco Lógico

MATRIZ DE MARCO LÓGICO

		Indicadores	Medios de Verificación	Supuestos
FIN	Contribuir a incrementar la rentabilidad de las Mipymes del Sector Metalmeccánico del Perú	Mínimo un 30% de incremento en la cantidad de clientes al finalizar un periodo contable en las Mipymes que utilizarán el sistema. Mínimo el 80% de las Mipymes que utilizarán el sistema deben afirmar que los procesos de servicio al cliente han mejorado Mínimo el 70% de las Mipymes que utilizarán el sistema deben afirmar que no han tenido problemas en brindar servicio de post-venta al cliente	Encuestas telefónicas y/o presenciales	Se mantienen las condiciones macroeconómicas legales, sociales y normativas.
Objetivo General	MEJORAR EL SERVICIO DE POST-VENTA AL CLIENTE PARA LAS MIPYMES DEL SECTOR METALMECÁNICO DEL PERÚ	90% de los clientes son atendidos oportunamente según las directrices establecidas. Mínimo el 80% de los clientes encuestados deberán estar satisfechos con el servicio de post-venta	Resultados de las pruebas del proyecto	Contar con los recursos del proyecto de manera oportuna
Objetivos Específicos	Agilizar la programación de visitas al cliente	Mínimo una reducción del 90% en el tiempo de la planificación de las visitas a los clientes Eliminación del 60% de las actividades burocráticas en la planificación de visitas	Resultados de las pruebas del proyecto	Los usuarios han recibido entrenamiento para utilizar el sistema de información
	Desarrollar un sistema de información de post-venta integral	01 sistema de servicio de post-venta al cliente	Acceso al sistema Resultados de las pruebas del sistema.	El software web es desarrollado de manera oportuna. Contar con los recursos del proyecto de manera oportuna
	Impulsar la consolidación de información de post-venta	Aumentar en 50% la cantidad de historiales de post-venta realizados cada mes	Resultados de las pruebas del proyecto	Los usuarios han recibido entrenamiento para utilizar el sistema de información
Actividades				
1.1	Conocer la vigencia exacta de la garantía de los productos vendidos	100% de los productos vendidos tienen definido su vigencia de garantía	Acceso al sistema Resultados de las pruebas de actividades del proyecto	Los productos son vendidos según sus directrices definidas
1.2	Conocer la disponibilidad de los técnicos	01 calendario con la información de la disponibilidad de los técnicos	Acceso al sistema Resultados de las pruebas de actividades del proyecto	Las empresas en estudio son las que realizan su servicio técnico.
2.1	Interfases de usuario fáciles de usar	Mínimo el 60% de los trabajadores evaluados utilizaron el sistema sin complicaciones luego del primer entrenamiento.	Resultados de las pruebas de actividades del proyecto	Se mantienen las nuevas tendencias en el desarrollo de interfaces
2.2	Brindar una infraestructura tecnológica a las Mipymes	01 instancia de Cloud Computing	Acceso a la instancia de Cloud Computing	Se cumple los acuerdos del nivel de servicio por parte del proveedor de cloud computing
3.1	Incrementar la información obtenida del servicio técnico	Mínimo el 80% de las visitas tienen definido el diagnóstico y/o actividades realizadas durante el servicio técnico 100% de las visitas tienen establecido el empleado asignado al servicio técnico	Acceso al sistema Resultados de las pruebas de actividades del proyecto	Las empresas en estudio son las que realizan su servicio técnico.
3.2	Fácil acceso a la información de los proyectos vendidos	50% de las consultas son generadas de manera automatizada	Acceso al sistema Resultados de las pruebas de actividades del proyecto	La base de datos está operativa

Anexo N° 4 – Evaluación de Riesgos

Identificación de Riesgos

Lista de Riesgos

ID	Riesgos	Tipo
1	Las condiciones macroeconómicas legales, sociales y normativas cambiarán drásticamente	IN
2	Incumplimiento en los acuerdos del nivel de servicio por parte del proveedor de cloud computing	ED
3	Las tendencias de desarrollo de interfaces cambiarán drásticamente antes de finalizar el proyecto	IN
4	Falta de entrenamiento acerca de las herramientas de desarrollo	ED
5	La fecha límite de entrega de todo el proyecto estará muy próxima	IN
6	La tecnología no satisfará las expectativas de los clientes	TC
7	Las condiciones con las que fue realizado el modelo de negocio cambiarán drásticamente antes de finalizar el proyecto	IN
8	Las tecnologías de desarrollo de aplicaciones móviles cambiarán radicalmente antes de finalizar el proyecto	IN
9	El software principal no se desarrollará de manera oportuna	TP
10	El diseño web variará durante el proyecto	TC
11	El despliegue de la aplicación móvil ya no será por la tienda virtual del sistema operativo del dispositivo móvil	IN
12	Las tecnologías de despliegue a través de cloud computing (IaaS y PaaS) se convertirán en obsoletas durante el proyecto	IN
13	Abandono de algún integrante del proyecto	TEPP

Evaluación de riesgos

Riesgos priorizados

ID	Riesgos	Tipo	Probabilidad	Impacto	Orden (Prioridad)
4	Falta de entrenamiento acerca de las herramientas de desarrollo	ED	70%	3	1
5	La fecha límite de entrega de todo el proyecto estará muy próxima	IN	60%	2	
10	El diseño web variará durante el proyecto	TC	60%	2	
6	La tecnología no satisfará las expectativas de los clientes	TC	40%	2	
9	El software principal no se desarrollará de manera oportuna	TP	40%	2	
2	Incumplimiento en los acuerdos del nivel de servicio por parte del proveedor de cloud computing	ED	20%	1	2
3	Las tendencias de desarrollo de interfaces cambiarán drásticamente antes de finalizar el proyecto	IN	20%	3	
8	Las tecnologías de desarrollo de aplicaciones móviles cambiarán radicalmente antes de finalizar el proyecto	IN	15%	3	
7	Las condiciones con las que fue realizado el modelo de negocio cambiarán drásticamente antes de finalizar el proyecto	IN	10%	1	
12	Las tecnologías de despliegue a través de cloud computing (IaaS y PaaS) se convertirán en obsoletas durante el proyecto	IN	10%	1	
1	Las condiciones macroeconómicas legales, sociales y normativas cambiarán drásticamente	IN	10%	2	
11	El despliegue de la aplicación móvil ya no será por la tienda virtual del sistema operativo del dispositivo móvil	IN	10%	3	
13	Abandono de algún integrante del proyecto	TEPP	10%	3	

Anexo N° 5 – Ingresos para los desarrolladores del sistema de información

Concepto	Año 1													
	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	Mes 6	Mes 7	Mes 8	Mes 9	Mes 10	Mes 11	Mes 12		
Ingresos														
Precio del producto (S/.)	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	
Cantidad de usuarios promedio por empresa	3	3	3	3	3	3	3	3	3	3	3	3	3	
Cantidad de empresas	1	1	1	1	2	2	2	3	3	3	3	3	3	
Total (S/.)	240.00	240.00	240.00	240.00	480.00	480.00	480.00	720.00	720.00	720.00	720.00	720.00	720.00	6,000.00

Concepto	Año 2													
	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	Mes 6	Mes 7	Mes 8	Mes 9	Mes 10	Mes 11	Mes 12		
Ingresos														
Precio del producto (S/.)	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	
Cantidad de usuarios promedio por empresa	3	3	3	3	3	3	4	4	4	4	4	4	4	
Cantidad de empresas	3	4	4	8	8	8	8	8	8	11	11	11	11	
Total (S/.)	720.00	960.00	960.00	1,920.00	1,920.00	1,920.00	2,560.00	2,560.00	2,560.00	3,520.00	3,520.00	3,520.00	3,520.00	25,640.00

Concepto	Año 3													
	Mes 1	Mes 2	Mes 3	Mes 4	Mes 5	Mes 6	Mes 7	Mes 8	Mes 9	Mes 10	Mes 11	Mes 12		
Ingresos														
Precio del producto (S/.)	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	80.00	
Cantidad de usuarios promedio por empresa	4	5	5	5	5	5	5	5	5	5	5	5	5	
Cantidad de empresas	11	13	13	16	16	16	20	20	20	25	25	25	25	
Total (S/.)	3,520.00	5,200.00	5,200.00	6,400.00	6,400.00	6,400.00	8,000.00	8,000.00	8,000.00	10,000.00	10,000.00	10,000.00	10,000.00	87,120.00

De las tablas anteriores del presente anexo, los ingresos aumentarán cada año, debido a que están relacionados con la cantidad de usuarios que adquieran el sistema de información. Vale decir que las licencias de usuario se renuevan de manera mensual.

Anexo N° 6 – Ingresos para la Mipyme que adquiere el sistema de información

Concepto	Año 1 (S/.)	Año 2 (S/.)	Año 3 (S/.)
Ahorro en las comunicaciones	720.00	800.00	880.00
Ahorro en los materiales	150.00	180.00	230.00
Mejora de procesos	4,800.00	5,000.00	5,200.00

Del cuadro anterior, se observa que básicamente el mayor ingreso como concepto que percibe la Mipyme que adquirirá el sistema de información es la de mejora de procesos, la cual está relacionado al ahorro de tiempo por parte del personal en consolidar la información de los procesos de servicio al cliente.

El ahorro en las comunicaciones es el ahorro en las llamadas telefónicas necesarias para realizar las coordinaciones con los clientes así como el ahorro de tiempo que empleado necesita para comunicarse con los clientes.

El ahorro en los materiales es el ahorro en tinta y papel que de manera convencional (sin utilizar el sistema de información) necesitarían para imprimir los formularios que llenan a mano.

Anexo N° 7 – Costo de desarrollo del sistema de información

Gastos Administrativos sin servicio

Gastos administrativos por el proyecto	
Materiales	Monto (S/)
Clips, faster, grampas y otros	15.00
Engrampador	7.00
Folder	10.00
Lapicero	9.00
Papel	42.00
Perforador	6.00
Resaltador	9.00
Tinta	150.00
Total de Materiales	248.00
Total de GA sin Servicios - Mes 0	248.00

Gastos Administrativos con servicios y sin materiales

Servicios	Monto (S/)
Luz	38.00
Agua	60.00
Teléfono + Internet	30.00
Alquiler de local	480.00
Total de Servicios	608.00
Total de GA con Servicios y sin Materiales	608.00

Costos Operativos

Costos Operativos	Monto Mensual (S/.)
Mano de obra	4,800.00
Servicios	942.00
Total	5,742.00

Equipos y muebles

Equipos y muebles	Costo Unitario (S/.)	Cantidad (unidad)	Costo Total (S/.)
Escritorio	700.00	2	1,400.00
Silla	200.00	2	400.00
Estante	300.00	1	300.00
Computadora	2,500.00	2	5,000.00
Impresora	500.00	1	500.00
Total			7,600.00

Costos de desarrollo del proyecto

Egresos	Mes 0 (S/.)	Mes 1 (S/.)	Mes 2 (S/.)	Mes 3 (S/.)	Total (S/.)
Gastos Administrativos	248.00	608.00	608.00	608.00	2,072.00
Costos Operativos	0.00	5,742.00	5,742.00	5,742.00	17,226.00
Equipos y muebles	7,600.00	0.00	0.00	0.00	7,600.00
Total de egresos	7,848.00	6,350.00	6,350.00	6,350.00	26,898.00
Inversión Total del Proyecto (S/.)	26,898.00				

Anexo N° 8 – Costos para la Mipyme que adquiere el sistema de información

Inversión en Equipos

Concepto	Costo (S/.)	Cantidad	Total (S/.)
Dispositivo Movil	500.00	3	1,500.00

Licencias de usuario

Año 1				
Concepto	Costo mensual (S/.)	Costo anual (S/.)	Cantidad	Total (S/.)
Licencia de usuario	80.00	960.00	3	2,880.00
Año 2				
Licencia de usuario	80.00	960.00	4	3,840.00
Año 3				
Licencia de usuario	80.00	960.00	5	4,800.00

Servicios

Concepto	Costo mensual (S/.)	Costo anual (S/.)	Cantidad	Total (S/.)
Plan de datos	30.00	360.00	3	1,080.00

Costos de adquisición del sistema

Concepto	Monto (S/.)
3 licencias de usuario por 1 año	2,880.00
3 Planes de datos por 1 año	1,080.00
Dispositivos Móviles	1,500.00
Total	5,460.00

Anexo N° 9 – Flujo de Caja para los desarrolladores del sistema de información

Concepto / Año	Año 0 (S/.)	Año 1 (S/.)	Año 2 (S/.)	Año 3 (S/.)
Saldo Inicial	26,898.00	0.00	1,620.00	19,080.00
Ingresos				
Venta/ Prestación de Servicios	0.00	6,000.00	26,640.00	87,120.00
Total de Ingresos	0.00	6,000.00	26,640.00	87,120.00
Egresos				
Gastos Administrativos	2,072.00	1,920.00	1,920.00	1,920.00
Costos Operativos	17,226.00	2,460.00	7,260.00	9,060.00
Equipos y muebles	7,600.00	0.00	0.00	0.00
Total de egresos	26,898.00	4,380.00	9,180.00	10,980.00
Caja neta	-26,898.00	1,620.00	17,460.00	76,140.00
Caja acumulada	0.00	1,620.00	19,080.00	95,220.00

De la tabla anterior, el saldo inicial es la inversión que entrará a caja para desarrollar el proyecto, además en el año 0 no hay ingresos, porque es el momento en el que se desarrolla el proyecto, en cambio sí hay muchos egresos que afrontar, donde principalmente el de mayor cantidad son los costos operativos que son básicamente el pago al personal encargado del desarrollo del sistema de información, este es un concepto muy importante dentro del flujo de caja, ya que de este dependerá que pueda llevarse a cabo el proyecto.

Anexo N° 10 – Flujo de Caja para la Mipyme que adquiere el sistema de info.

Concepto / Año	Año 0 (S/.)	Año 1 (S/.)	Año 2 (S/.)	Año 3 (S/.)
Saldo Inicial	0.00	-1,500.00	210.00	1,270.00
Ingresos				
Ahorro en las comunicaciones	0.00	720.00	800.00	880.00
Ahorro en los materiales	0.00	150.00	180.00	230.00
Mejora de procesos	0.00	4,800.00	5,000.00	5,200.00
Total de Ingresos	0.00	5,670.00	5,980.00	6,310.00
Egresos				
Equipos	1,500.00	0.00	0.00	0.00
Licencia del sistema	0.00	2,880.00	3,840.00	4,800.00
Servicios	0.00	1,080.00	1,080.00	1,080.00
Total de egresos	1,500.00	3,960.00	4,920.00	5,880.00
Caja neta	-1,500.00	1,710.00	1,060.00	430.00
Caja acumulada	-1,500.00	210.00	1,270.00	1,700.00

Del cuadro anterior, es importante mencionar que en el año 0 no hay ingresos, esto se debe a que la Mipyme que adquiera el sistema de información no verá los resultados al momento de obtener la licencia de usuario, sino en los posteriores meses y años que valla utilizando acotado sistema.

Anexo N° 11 – Análisis Beneficio/Costo para los desarrolladores

Costos y beneficios actualizados por año

Análisis Beneficio/Costo				
Año	Costo (S/.)	Valor actualizado del costo (S/.)	Beneficio (S/.)	Valor actualizado del beneficio (S/.)
0	26,898.00	26,898.00	0.00	0.00
1	4,380.00	3,650.00	6,000.00	5,000.00
2	9,180.00	6,375.00	26,640.00	18,500.00
3	10,980.00	6,354.17	87,120.00	50,416.67
Tasa de interés	0.20			

Resultado del Indicador Beneficio/Costo

Beneficio Neto (S/.)	73,916.67
Costo (S/.)	43,277.17
B/C	1.71

Del cuadro anterior, se puede interpretar el resultado como que por cada nuevo sol (S/. 1.00) invertido en el proyecto se recuperará la inversión y se generará setenta y un céntimos de nuevo sol (S/. 0.71).

Anexo N° 12 – Indicadores VAN y TIR para los desarrolladores

Resultado del Indicador VAN

VAN	
Tasa de interés	0.20
Inversión Inicial (S/.)	26,898.00
Flujo de caja 1 (S/.)	1,350.00
Flujo de caja 2 (S/.)	12,125.00
Flujo de caja 3 (S/.)	44,062.50
VAN	30,639.50

Del cuadro anterior, el VAN es positivo por tanto el proyecto es viable a los tres años de su puesta en marcha y funcionamiento.

Resultado del Indicador TIR

TIR	59%
------------	-----

Del cuadro anterior, la máxima tasa de rendimiento que puede esperar un inversionista del proyecto es el 59%.

Anexo N° 13 – Periodo de Recuperación de la Inversión para los desarrolladores

Periodo de recuperación de la inversión (PRI)	
Año	Flujo Neto (S/)
0	-26,898.00
1	1,620.00
2	17,460.00
3	76,140.00
PRI	2.1
Año	2
Meses	1
Días	6

De la tabla anterior, el resultado se interpresa que la inversión se recuperará en 2 años, 1 mes y 6 días.

Anexo N° 14 – Análisis de Beneficio/Costo para la Mipyme

Costos y beneficios actualizados por año

Análisis Beneficio/Costo				
Año	Costo (S/.)	Valor actualizado del costo (S/.)	Beneficio (S/.)	Valor actualizado del beneficio (S/.)
0	1,500.00	1,500.00	0.00	0.00
1	3,960.00	3,700.93	5,670.00	5,299.07
2	4,920.00	4,297.32	5,980.00	5,223.16
3	5,880.00	4,799.83	6,310.00	5,150.84
Tasa de interés	0.07			

Resultado del Indicador Beneficio/Costo

Beneficio Neto (S/.)	15,673.07
Costo (S/.)	14,298.08
B/C	1.10

Del cuadro anterior, se puede interpretar el resultado como que por cada nuevo sol (S/. 1.00) invertido en el proyecto se recuperará la inversión y se generará diez céntimos de nuevo sol (S/. 0.10). Cabe mencionar, si bien es cierto que no se genera muchos beneficios económicos en primera instancia, a cambio si se produce muy buenos beneficios intangibles (como ahorro en las comunicaciones y mejora de los procesos) que a su vez permiten que la imagen de la Mipyme mejore contribuyendo a una mejora en su rentabilidad.

Anexo N° 15 – Indicadores VAN y TIR para la Mipyme

Resultado del Indicador VAN

VAN	
Tasa de interés	0.07
Inversión Inicial (S/.)	1,500.00
Flujo de caja 1 (S/.)	1,598.13
Flujo de caja 2 (S/.)	925.85
Flujo de caja 3 (S/.)	351.01
VAN	1,374.98

Del cuadro anterior, el VAN es positivo por tanto el proyecto es viable a los tres años que la Mipyme adquiere y utiliza el sistema de información.

Resultado del Indicador TIR

TIR	67%
------------	------------

Del cuadro anterior, la máxima tasa de rendimiento que puede esperar la Mipyme del proyecto es el 67%.

Anexo N° 16 – Periodo de Recuperación de la Inversión para la Mipyme

Periodo de recuperación de la inversión (PRI)	
Año	Flujo Neto (S/)
0	-1,500.00
1	1,710.00
2	1,060.00
3	430.00
PRI	0.88
Año	0
Meses	10
Días	17

De la tabla anterior, el resultado se interpreta que la inversión que realiza la Mipyme en adquirir el sistema de información se recuperará en 10 meses y 17 días, siempre y cuando utilice todos los meses mencionado sistema.

Anexo N° 17 – Selección de Sistema Gestor de Base de Datos

Para seleccionar el Sistema Gestor de Base de Datos, se tuvo que realizar una comparación cuantitativa, por ello se agregó un valor por cada criterio, el cual indica el grado de relevancia en la utilización de la base de datos, además cada tecnología recibirá un puntaje de acuerdo a sus características respecto al criterio, al final se obtendrá un valor resultante ponderado de cada tecnología y se elegirá la de mayor puntuación.

Criterio	Valor	MySQL	PostgreSQL
Rendimiento sobre operaciones simples	5	4	5
Soporte de herramientas externas	4	3	5
Integración	3	3	5
Soporte de hosting	2	5	4
Total		51	68

Se seleccionó el Sistema Gestor de Base de Datos PostgreSQL. La interpretación del cuadro anterior es la siguiente:

- El rendimiento es un factor importante en la selección de un motor de Base de Datos cuando se construye una aplicación web, debido a que esta accederá a la Base de Datos a través de consultas (queries) y realizará operaciones (como insertar, actualizar y eliminar registros). En tal sentido, PostgreSQL tiene mejor performance cuando se realiza operaciones simples a comparación de MySQL.
- PostgreSQL tiene compatibilidad con muchas herramientas externas para el diseño, administración y uso del sistema de gestión.
- La integración es la capacidad que tiene una determinada Base de Datos para poder migrar hacia otra. PostgreSQL ofrece la posibilidad de manejar este cambio de Base de Datos rápidamente, en cambio con MySQL se vuelve un poco complicado porque no posee herramientas completas que realicen esta actividad.

- El soporte de hosting es la cantidad de proveedores de host que ofrecen el servicio de administrar instancias de una determinada Base de Datos. En el mercado actual existen una mayor cantidad de soporte para instancias que tienen MySQL en vez de PostgreSQL.

Anexo N° 18 – Selección de proveedor de PaaS de cloud computing

Para seleccionar el proveedor de cloud computing, se tuvo que realizar una comparación cuantitativa, por ello se agregó un valor por cada criterio, el cual indica el grado de relevancia en la utilización de la instancia de cloud, además cada plataforma recibirá un puntaje de acuerdo a sus características respecto al criterio, al final se obtendrá un valor resultante ponderado de cada tecnología y se elegirá la de mayor puntuación.

Criterio	Valor	Heroku	Engine Yard	Elastic Beanstalk
Facilidad de instalación	5	5	3	3
Soporte	4	4	4	5
Complementos	4	5	5	3
Manejo de backups	3	5	5	5
Facturación y tarificación	3	5	3	4
Escalabilidad	1	3	5	5
Total		94	80	79

Se seleccionó Heroku como proveedor de cloud computing. La interpretación del cuadro anterior es la siguiente:

- La facilidad de instalación está relacionado con la disposición de inicio rápido por parte de la plataforma del proveedor de cloud computing. En este punto Heroku a diferencia de sus competidores ofrece una rápida disponibilidad de los recursos virtuales, sin necesidad de establecer una tarjeta de crédito predeterminada o confirmación de la misma. También ofrece una herramienta llamada heroku-toolbelt con la cual a través de líneas de comando el usuario autenticarse y utilizando su ordenador local desplegar el software solo introduciendo en un archivo del proyecto de software un par de líneas de código, situación similar sucede con Engine Yard, pero diferente es el caso de Elastic Beanstalk, en el cual la configuración inicial para el despliegue del software es un poco manual, porque no incluye la opción de introducir líneas de código de despliegue directamente al código fuente.

- El soporte de documentación que brinda Elastic Beanstalk está más especificado que el que brinda Heroku o Engine Yard.
- Los complementos son aplicaciones adicionales que están integradas a la plataforma y proveen de funcionalidades extras a las brindadas por defecto por el proveedor de cloud computing. En tal sentido Heroku y Engine Yard proveen una fácil integración con NewRelic (para analizar y monitorizar los datos de la aplicación) y SendGrid (para la gestión de correos electrónicos), en cambio con Elastic Beanstalk se vuelve complicado realizar la configuración para poder integrarse con NewRelic y SendGrid, porque previamente se tiene que ejecutar una serie de comandos y agregar permisos.
- El manejo de backups (respaldo de información) se desarrolla de una manera sencilla y de forma automática, por tanto es viable elegir en este punto cualquier de los tres proveedores que están dentro de la evaluación.
- El modelo de facturación y tarifación que proporciona Heroku es más flexible que el de los otros dos proveedores (Engine Yard y Elastic Beanstalk), debido a que cuenta con un plan más económico por similares características técnicas, por ejemplo Heroku tiene un plan estándar con un pago mensual de \$25, en Engine Yard se realiza un pago mensual de \$29 y con Elastic Beanstalk un pago de \$35 por mes. Cabe resaltar que en Heroku no hay una restricción de tiempo de uso de la plataforma cuando el software está desplegado (siempre y cuando no se ocupen más de los recursos virtuales que vienen por defecto), con Engine Yard y Elastic Beanstalk existe un periodo de prueba, seis meses en caso del primero y un año para el segundo proveedor.
- En cuanto a la escalabilidad, con cualquier de los tres proveedores de cloud se puede ampliar los recursos tecnológicos, sin embargo con Heroku se debe hacerlo de manera manual a través de los dynos, por ello es que tiene un puntaje bajo.

Anexo N° 19 – Selección de la metodología de desarrollo de software

Para seleccionar la metodología de desarrollo, se tuvo que realizar una comparación cuantitativa, por ello se agregó un valor por cada criterio, el cual indica el grado de relevancia en la utilización de la metodología, además cada una recibirá un puntaje de acuerdo a sus características respecto al criterio, al final se obtendrá un valor resultante ponderado de cada metodología y se elegirá la de mayor puntuación.

Criterio/Metodología	Valor	Scrum	XP	FDD
Escalabilidad en el equipo del proyecto	1	4	3	4
Capacidad de cambio y regulación en la iteración	2	3	4	3
Soporte	3	5	5	3
Diseño	3	2	3	4
Programación	5	2	5	3
Prueba	5	2	5	3
Seguimiento de Proyecto	1	4	2	5
Total		55	87	66

Se seleccionó la metodología XP como base para el desarrollo del software. La interpretación del cuadro anterior es la siguiente:

- La escalabilidad en la metodología FDD (Feature Driven Development) es mejor que en la de XP (Extreme Programming), porque para realizar las tareas de la iteraciones se forman grupos pequeños, por tanto si el equipo total crece no afecta en el desarrollo del producto, debido a que ya está definido los grupos que se encargarán de la iteración consiguiendo que fluya mejor la comunicación a través de pocas personas en lugar de comunicarse con todos los miembros del equipo total como si pasa en XP.

Situación muy parecida sucede con Scrum, para realizar cada iteración (o también llamado Sprint) se encarga a un grupo de trabajo multifuncional (cross-functional) determinadas tareas, los integrantes de cada grupo se pueden ayudar con sus obligaciones, de esta manera se incentiva el autoaprendizaje y la comunicación.

- XP tiene una mejor capacidad para responder al cambio de un requerimiento dentro de la iteración, porque se puede hacer cualquier cambio mientras no se haya empezado a programar el requerimiento, el proceso de cambio de funcionalidades básicamente consiste en mover un requerimiento para la siguiente iteración y poner el nuevo en la iteración presente. En cuanto al mecanismo de regulación en la iteración, si en caso no se ha completado todas las historias de usuario pactadas o la calidad no es la adecuada se pasarán para la siguiente iteración y se reducirán el número de historias de usuario que se pueden realizar automáticamente.

En cuanto a Scrum solo se puede hacer cambios en la iteración hasta antes de terminar el Planning Meeting y hasta que los compromisos para entregar los ítems del Product Backlog estén hechos, además si en caso fuese necesario cancelar un Sprint las tareas pendientes que son necesarias se pasarán para el siguiente Sprint.

FDD al igual que Scrum no permite que se hagan cambios en los requerimientos (features) una vez que el Review Meeting este completado, también si no se completó o no tuvo la calidad adecuada los features al término de la iteración existen dos mecanismos de regulación que son: la experiencia y la decisión del Chief Programmer del grupo acerca de los problemas que hubo en la iteración, y creación de otros grupos para hacerse cargo de las tareas no completadas de la iteración.

- El soporte de información en la metodología Scrum y XP es mejor que la que se brinda en FDD, ya que existe mayor documentación (paper, presentaciones en diapositivas, y cualquier otro archivo electrónico) tanto en inglés como en español.

- Tanto FDD como XP contienen dentro de su ciclo de vida una etapa o actividad clave denominada diseño. En FDD hay reuniones de revisión de diseño (Design Review Meeting), en esta etapa los desarrolladores declaran las clases principales, métodos y propiedades necesarias para implementar la función de la característica (feature) generando buena cantidad de documentación que respalde esta etapa, si fuese necesario se puede realizar diagramas de secuencia. En XP se desarrolla las tarjetas CRC (clase-responsabilidad-colaborador) y prototipos si existe un problema en el diseño de la historia de usuario. En Scrum el equipo entiende que si en caso es necesario mostrar un diagrama de secuencia, modelos, prototipos de interfaz de usuario, y otros entregables lo deberán de realizar.
- XP tiene bien definido la etapa de programación y es junto con la actividad de pruebas los pilares de esta metodología. XP promueve el refactoring para mejorar la calidad del código, también afirma que nadie es dueño del código cualquiera puede cambiarlo, todos entienden el valor del código, de esta manera si alguien sale del equipo se minimizará la pérdida ya que el equipo conocerá el código. En FDD se tiene claramente identificado el dueño del código y las consecuencias de hacer cambios en este, todos los miembros del equipo conocen el código ya que siempre se realiza el Code Review Meeting que es una reunión donde los desarrolladores opinan acerca del código de sus pares habiéndolo revisado previamente, además FDD discute mucho el tema de refactoring ya que lleva mucho tiempo realizarlo y no trae ningún valor aparente para el cliente, en lugar de eso la calidad del código se puede revisar en el Code Review Meeting. Scrum no es muy específico en el desarrollo, pero tiene claro que siempre se debe hacer un entregable o incremento funcional (del software) por cada Sprint o iteración.
- XP reconoce que una de sus principales características es realizar pruebas de manera automatizada y para ello se apoya en el TDD (Test driven development), que consiste en escribir primero las

prueba y luego verificar que falla para después desarrollar el código correcto que hará pasar la prueba. FDD también apoya las pruebas automatizadas pero no tiene alguna preferencia en que se escriban antes (como con TDD) o después de que el código haya sido realizado, pero si se deben hacer pruebas unitarias por lo menos una prueba por clase. Scrum apoya que siempre los incrementos deben pasar por pruebas pudiendo ser estas automatizadas si facilitan el trabajo al equipo, si no es el caso se deben realizar pruebas de manera manual.

- La metodología FDD tiene básicamente dos mecanismos para el seguimiento del proyecto, uno es conocer el número de features luego que el proceso de captura de las características esté terminado de esta manera se podrá comparar las features pendientes con las planeadas en la iteración, otro mecanismo es asignar a las etapas de cada iteración un porcentaje de realización, de esta manera se puede generar informes y reportes acerca de la situación real del proyecto, por tal motivo FDD es una metodología que presenta una estricta cantidad de documentos respecto al seguimiento del proyecto. Cabe mencionar que las revisiones por cada iteración son otra forma de realizar el seguimiento al proyecto. Scrum también permite el seguimiento del proyecto mediante la realización del Daily Scrum, que es un evento de 15 minutos que se realiza todos los días mientras dure la iteración, en el cual los miembros del equipo Scrum mencionan que han hecho para lograr el objetivo del Sprint, que planean hacer para lograr el objetivo y cuales han sido sus principales obstáculos para lograr el objetivo, además cuenta con la revisión del Sprint (Sprint Review) para poder examinar los elementos completados del Product Backlog y las tareas pendientes del Sprint Backlog al final de cada Sprint.

A diferencia de las dos metodologías anteriores, XP sigue el proyecto conociendo las historias de usuario que han sido hechas y cuales quedan pendientes.

Además de la comparación realizada en base a la búsqueda de información de cada metodología realizada por los autores de la presente tesis, se mostrará a continuación un cuadro comparativo realizado por el Departamento de Sistemas de Informáticos y Computación de la Universidad Politécnica de Valencia:

	ASD	Crystal	DSDM	FDD	LD	Scrum	XP
Sistema como algo cambiante	5	4	3	3	4	5	5
Colaboración	5	5	4	4	4	5	5
Características Metodología (CM)							
-Resultados	5	5	4	4	4	5	5
-Simplicidad	4	4	3	5	3	5	5
-Adaptabilidad	5	5	3	3	4	4	3
-Excelencia técnica	3	3	4	4	4	3	4
-Prácticas de colaboración	5	5	4	3	3	4	5

Del cuadro anterior, se observa que XP sobresale en repuesta al cambio, colaboración, obtención de resultados rápidos, simplicidad y prácticas de colaboración.

Anexo N° 20 – Selección de framework de desarrollo

Para seleccionar el framework de desarrollo, se tuvo que realizar una comparación cuantitativa, por ello se agregó un valor por cada criterio, el cual indica el grado de relevancia en la utilización del framework, además cada uno recibirá un puntaje de acuerdo a sus características respecto al criterio, al final se obtendrá un valor resultante ponderado de cada framework y se elegirá la de mayor puntuación.

Criterio	Valor	Ruby on Rails	Spring MVC	ASP .NET MVC
Soporte	5	5	4	4
Facilidad de sintaxis	4	5	3	4
Manejo dependencias	4	5	3	4
Manejo de la capa de datos	3	5	3	3
Total		80	53	61

Se seleccionó el framework Ruby on Rails para el desarrollo del software y los servicios backend de la aplicación móvil. La interpretación del cuadro anterior es la siguiente:

- El soporte brindado a nivel de documentación por parte de la comunidad de Ruby on Rails (RoR) es mejor que el de los otros frameworks en evaluación, porque todo está documentado con imágenes y bien detallado por cada componente. Además en la web hay muchos ejemplos. Es importante mencionar que tanto Spring MVC como ASP .NET MVC contienen dentro de su sitio web oficial documentación, pero esta no llega a ser tan explícita como lo es con RoR.
- La facilidad de sintaxis está relacionado con el lenguaje de programación asociado al framework de desarrollo y la claridad de sus palabras reservadas. En tal sentido, Ruby on Rails por utilizar un

lenguaje no tipado (no existen los tipos de datos) resulta más sencillo para el programador realizar la codificación, situación contraria sucede con Java y C#, en caso Spring MVC y ASP .NET MVC respectivamente.

Además las palabras reservadas que maneja RoR son cortas y su denominación está relacionado al nombre de un expresión de la realidad, produciendo que sea fácil de recordar dichas líneas de código.

- El manejo de dependencias permite conocer en todo momento las versiones de las extensiones, evitar conflictos entre ellas y actualizarlas rápidamente. Ruby on Rails con solo una corta línea de código permite generar una dependencia, mientras que con Maven (gestor de dependencia de Spring MVC) se debe especificar con varias líneas datos el nombre, la versión y el proveedor. ASP .NET MVC utiliza NuPack como gestor de paquetes a través del cual por medio de una interfaz gráfica se puede mantener las dependencias.
- El manejo de la capa de datos está asociado con el acceso a los datos de los registros de la Base de Datos y su manipulación. En tal sentido, Ruby on Rails utiliza ActiveRecord por defecto para manejar las operaciones en la Base de Datos de una manera sencilla y rápida, debido a que no es necesario relacionar los campos de la tabla con los de la clase relacionada, cuestión que si sucede tanto con Hibernate (para Spring MVC) y Entity Framework (para ASP .NET MVC). Cabe mencionar que la configuración de la Base de Datos utilizando ActiveRecord se realiza con unas cuantas líneas de código y las operaciones se manejan a través de métodos muy flexibles en la recepción de parámetros.

Anexo N° 21 – Selección de framework para realizar aplicaciones móviles multiplataforma basadas en HTML5

Para seleccionar la tecnología para realizar aplicaciones móviles multiplataforma basadas en HTML5, se tuvo que realizar una comparación cuantitativa, por ello se agregó un valor por cada criterio, el cual indica el grado de relevancia para el desarrollo de la aplicación móvil, además cada plataforma recibirá un puntaje de acuerdo a sus características respecto al criterio, al final se obtendrá un valor resultante ponderado de cada tecnología y se elegirá la de mayor puntuación.

Criterio	Valor	PhoneGap (Cordova)	Intel XDK
Soporte de la comunidad	5	5	3
Compatibilidad inmediata de frameworks de diseño	5	4	2
Facilidad de configuración inicial	4	4	2
Capacidad de deploy	3	4	5
Debug y pruebas	2	2	5
Total		77	58

Se seleccionó Phonegap como framework de desarrollo multiplataforma de aplicaciones móviles. La interpretación del cuadro anterior es la siguiente:

- El soporte es un punto muy importante al momento de elegir una tecnología, porque ayudará a resolver dudas y obtener un rápido aprendizaje. La comunidad de PhoneGap brinda una mayor documentación que la de Intel XDK, es importante mencionar que el proyecto PhoneGap nació como proyecto libre con el nombre de Apache Cordova, por ello es que la comunidad está siempre presente

en la atención de inquietudes acerca de dicha tecnología. Además tienen un foro en español para los desarrolladores de ese idioma, con lo cual aportan una mejor asistencia que Intel XDK.

- La compatibilidad inmediata de frameworks de diseño es un criterio significativo cuando se desarrolla aplicaciones móviles multiplataforma basadas en HTML5, ya que se deberá tener en cuenta un framework que aporte a la vista la apariencia y funcionamiento lo más parecido a una aplicación nativa. En caso de PhoneGap, principalmente puede trabajar con varios frameworks como: Ionic, Mobile Angular UI, JQuery Mobile, entre otros. En cambio con Intel XDK principalmente se usa Intel App Framework, sin embargo si podría utilizar Ionic, pero tendría que realizarse configuraciones previas, con lo cual complica a los desarrolladores nuevos en función de su aprendizaje.
- La configuración inicial hace referencia al proceso de dejar listo la aplicación para comenzar a desarrollar, por ejemplo: indicar el sistema operativo, el nombre de la aplicación, dispositivos para correr la máquina virtual, entre otros). En este punto con PhoneGap es mucho más sencillo configurar la aplicación, porque se hace directamente a través de la terminal por medio de comandos precisos, por lo cual no se presta a problemas de interpretación como puede suceder con Intel XDK que necesita de una interfaz (que lo brinda su IDE inherente) para realizar la configuración inicial.
- La capacidad de deploy es la facilidad que brinda la plataforma para desplegar el software en los diferentes sistemas operativos a través de su tienda virtual. PhoneGap permite utilizar PhoneGap Build para desplegar el paquete en la nube y transformarlo de acuerdo al sistema operativo objetivo, también si se tiene instalado de manera local los manejadores propios del sistema operativo del dispositivo móvil se puede realizar esta transformación de manera natural, es importante mencionar que PhoneGap Build solo permite convertir un número específico de aplicaciones basadas en HTML5 al paquete

nativo, luego se tendrá que pagar por este servicio; en cambio con Intel XDK esta transformación es gratis y al igual que con PhoneGap se puede realizar la transformación también en la nube, solo que no habrá un costo por ello.

- PhoneGap posee una menor capacidad al momento de realizar los procesos de debug y pruebas en comparación de Intel XDK, debido a que este último posee un IDE inherente, en el cual se puede configurar para que se pruebe de manera dinámica la aplicación en los diferentes tipos de dispositivos móviles, claro esto involucra un mayor grado de complejidad, pero resulta muy efectivo si se realiza de manera adecuada. Es importante mencionar que un proyecto realizado con PhoneGap se puede migrar rápidamente a Intel XDK.

Anexo N° 22 – Selección de framework de diseño para realizar aplicaciones móviles multiplataforma basadas en HTML5

Para seleccionar el framework de diseño para realizar aplicaciones móviles multiplataforma basadas en HTML5, se tuvo que realizar una comparación cuantitativa, por ello se agregó un valor por cada criterio, el cual indica el grado de relevancia para el desarrollo de la aplicación móvil, además cada framework recibirá un puntaje de acuerdo a sus características respecto al criterio, al final se obtendrá un valor resultante ponderado de cada tecnología y se elegirá la de mayor puntuación.

Criterio	Valor	Ionic	Mobile Angular UI	Jquery Mobile
Facilidad de aprendizaje	5	5	4	3
Soporte en la documentación	5	5	3	5
Diseño moderno	4	5	5	2
Total		70	55	48

Se seleccionó Ionic como framework de diseño para realizar aplicaciones móviles multiplataforma. La interpretación del cuadro anterior es la siguiente:

- La facilidad de aprendizaje es un punto muy importante al momento de seleccionar un framework de diseño. Ionic es sin duda el que resulta más rápido de aprender porque está basado en AngularJS, por tanto nos permite manejar los datos y las etiquetas HTML a través de componentes, situación similar ocurre con Mobile Angular UI, pero la diferencia con Ionic es que este último está más documentado. El problema con Jquery Mobile es que es un framework con demasiadas clases CSS, por tanto es más difícil aprender y recordar la real utilización de cada clase y su relación entre ellas.
- El soporte en la documentación es un criterio que está relacionado con la facilidad de aprendizaje, aquí es donde Ionic y Jquery Mobile sobresalen, en caso de Jquery Mobile porque es más antiguo que las

otras dos tecnologías y por su parte Ionic está muy bien documentado con demos (ejemplos) en línea, en cambio Mobile Angular UI se presenta como framework reciente pero con mucha proyección, por ello es que actualmente su documentación no es muy explícita.

- Contar con un diseño moderno es relevante en el desarrollo de aplicaciones móviles basadas en HTML5, porque permite simular lo más parecido a la vista y funcionamiento de una aplicación móvil nativa, en tal sentido Ionic y Mobile Angular UI cuentan con modelos de diseño en sus elementos idénticos o muy similares a la de una app móvil nativa, en cambio con JQuery Mobile da la percepción de que el diseño de los elementos se hizo primero como si fuese dirigido a las computadoras de escritorio o laptops y luego fuese adaptado a los dispositivos móviles, parte de ello se debe a que existen muchas clases CSS que no permiten una fácil configuración de la vista. Además es importante mencionar que en la parte del funcionamiento y comunicación de la data entre el la vista, el controlador y el modelo tanto Ionic como Mobile Angular UI utilizan un framework frontend que los respalda (que es AngularJS), en cambio JQuery Mobile utiliza una librería (que es JQuery).

Anexo N° 23 – Plan de Trabajo

Actividades

Para la selección de las actividades se utilizó la metodología del Marco Lógico, donde las actividades están relacionados a los objetivos específicos del proyecto. A continuación se mostrará un cuadro que mostrará la relación entre las actividades y los objetivos específicos.

Actividad General	Objetivo específico
Conocer la vigencia exacta de la garantía de los productos vendidos	Agilizar la programación de visitas al cliente
Conocer la disponibilidad de los técnicos	
Interfaces de usuario fáciles de usar	Desarrollar un sistema de información de post-venta integral
Brindar una infraestructura tecnológica a las Mipymes	
Incrementar la información obtenida del servicio técnico	Impulsar la consolidación de información de post-venta
Fácil acceso a la información de los proyectos vendidos	

Para mayor detalle acerca de las actividades generales del proyecto, véase el Anexo N°s 2 “Árbol de Objetivos” y 3 “Matriz de Marco Lógico”.

En el siguiente cuadro se visualiza una breve descripción de cada una de las actividades del proyecto:

Actividad General	Breve descripción
Conocer la vigencia exacta de la garantía de los productos vendidos	La vigencia de garantía de los productos vendidos se debe definir en base a la fecha de venta y no de acuerdo a la fecha de fabricación.
Conocer la disponibilidad de los técnicos	Se debe tener establecido cuando el técnico está disponible para atender una visita al cliente.
Interfaces de usuario fáciles de usar	Se debe construir interfaces de usuario con alta usabilidad. Los usuarios deben probar dichas interfaces y ser evaluados luego de recibir un entrenamiento.
Brindar una infraestructura tecnológica a las Mipymes	El sistema de información debe de ser desplegado totalmente, en tal sentido, el software web se publicará en una instancia proporcionada por el proveedor de cloud computing y la aplicación móvil será publicada en la tienda virtual del sistema operativo del dispositivo móvil.
Incrementar la información obtenida del servicio técnico	Se debe definir para el servicio técnico, los campos a registrar y que tendrán persistencia en la base de datos.
Impulsar la consolidación de información de post-venta	Se debe definir los criterios de consulta de información de post-venta y los historiales a visualizar.

Es importante mencionar que las actividades propuestas son actividades generales necesarias para cumplir con los objetivos específicos del proyecto, que resultan de la Matriz de Marco Lógico, por tanto cada una ellas tiene comprendida más sub-actividades que se verán en la sección del cronograma.

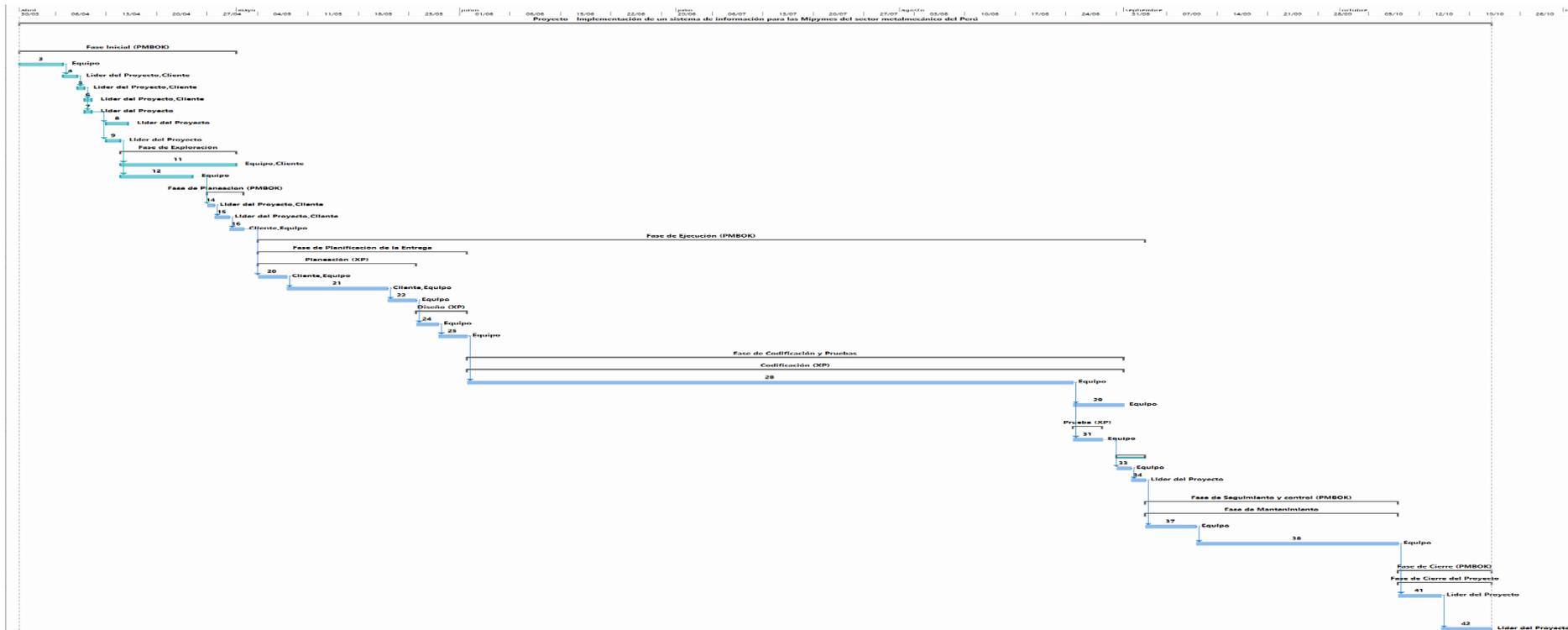
Cronograma

A continuación se mostrará el cronograma del proyecto

ID	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	Proyecto - Implementación de un sistema de información para las Mipymes del sector metalmeccánico del Perú	146 días	mié 01/04/15	mié 21/10/15		
2	Fase Inicial (PMBOK)	22 días	mié 01/04/15	jue 30/04/15		
3	Entrevista con involucrados	4 días	mié 01/04/15	lun 06/04/15		Equipo
4	Definición del problema	2 días	mar 07/04/15	mié 08/04/15	3	Lider del Proyecto, Cliente
5	Definición del objetivo general	1 día	jue 09/04/15	jue 09/04/15	4	Lider del Proyecto, Cliente
6	Definición el alcance	1 día	vie 10/04/15	vie 10/04/15	5	Lider del Proyecto, Cliente
7	Definición de las limitaciones	1 día	vie 10/04/15	vie 10/04/15	5	Lider del Proyecto
8	Estudio de viabilidad económica, operativa, técnica y social	3 días	lun 13/04/15	mié 15/04/15	7	Lider del Proyecto
9	Selección de tecnologías	2 días	lun 13/04/15	mar 14/04/15	7	Lider del Proyecto
10	Fase de Exploración	12 días	mié 15/04/15	jue 30/04/15		
11	Captura inicial de requerimientos	12 días	mié 15/04/15	jue 30/04/15	9	Equipo, Cliente
12	Capacitación en tecnologías seleccionadas	8 días	mié 15/04/15	vie 24/04/15	9	Equipo
13	Fase de Planeación (PMBOK)	5 días	lun 27/04/15	vie 01/05/15		
14	Refinación del objetivo principal	1 día	lun 27/04/15	lun 27/04/15	12	Lider del Proyecto, Cliente
15	Definición de los objetivos específicos	2 días	mar 28/04/15	mié 29/04/15	14	Lider del Proyecto, Cliente
16	Definición de indicadores	2 días	jue 30/04/15	vie 01/05/15	15	Cliente, Equipo
17	Fase de Ejecución (PMBOK)	89 días	lun 04/05/15	jue 03/09/15		
18	Fase de Planificación de la Entrega	21 días	lun 04/05/15	lun 01/06/15		
19	Planeación (XP)	16 días	lun 04/05/15	lun 25/05/15		
20	Consolidación de requerimientos	4 días	lun 04/05/15	jue 07/05/15	16	Cliente, Equipo
21	Creación de las historias de usuario	10 días	vie 08/05/15	jue 21/05/15	20	Cliente, Equipo
22	Definición del plan de iteración	2 días	vie 22/05/15	lun 25/05/15	21	Equipo
23	Diseño (XP)	5 días	mar 26/05/15	lun 01/06/15		
24	Definición de las tarjetas CRC	3 días	mar 26/05/15	jue 28/05/15	22	Equipo
25	Diseño del modelo físico de la Base de datos	2 días	vie 29/05/15	lun 01/06/15	24	Equipo
26	Fase de Codificación y Pruebas	65 días	mar 02/06/15	lun 31/08/15		
27	Codificación (XP)	65 días	mar 02/06/15	lun 31/08/15		
28	Programación del código fuente por historia de usuario	60 días	mar 02/06/15	lun 24/08/15	25	Equipo
29	Integración de código al repositorio del sistema	5 días	mar 25/08/15	lun 31/08/15	28	Equipo
30	Prueba (XP)	4 días	mar 25/08/15	vie 28/08/15		
31	Creación de las pruebas unitarias	4 días	mar 25/08/15	vie 28/08/15	28	Equipo
32	Fase de Producción	4 días	lun 31/08/15	jue 03/09/15		
33	Publicación del incremento de software	2 días	lun 31/08/15	mar 01/09/15	31	Equipo
34	Revisión de la integración del incremento de software	2 días	mié 02/09/15	jue 03/09/15	33	Lider del Proyecto
35	Fase de Seguimiento y control (PMBOK)	25 días	vie 04/09/15	jue 08/10/15		
36	Fase de Mantenimiento	25 días	vie 04/09/15	jue 08/10/15		
37	Entrenamiento a los usuarios acerca del producto	5 días	vie 04/09/15	jue 10/09/15	34	Equipo
38	Evaluación del producto de acuerdo a los indicadores definidos	20 días	vie 11/09/15	jue 08/10/15	37	Equipo
39	Fase de Cierre (PMBOK)	9 días	vie 09/10/15	mié 21/10/15		
40	Fase de Cierre del Proyecto	9 días	vie 09/10/15	mié 21/10/15		
41	Almacenamiento y verificación en el repositorio de la documentación generada durante el proyecto	4 días	vie 09/10/15	mié 14/10/15	38	Lider del Proyecto
42	Entrega y conformidad del producto al cliente	5 días	jue 15/10/15	mié 21/10/15	41	Lider del Proyecto

Del cuadro anterior, el proyecto tiene una duración de ciento cuarenta y seis (146) días, es importante mencionar que se consideró solo los días hábiles para el desarrollo del proyecto.

Para una mejor visualización de la duración de las actividades y su precedencia, se mostrará el Diagrama de Gantt del proyecto:



De la figura anterior, se ha mostrado el identificador único (ID) en la parte superior de cada actividad. Además es importante mencionar que el proyecto debe iniciar el 01 de Abril de 2015 y debe culminar el 21 de Octubre de 2015.

Cabe resaltar que las actividades o tareas listadas en el cronograma son parte de alguna actividad general señalada en la Matriz de Marco Lógico, cuyo detalle se puede apreciar en el Anexo N° 3 “Matriz de Marco Lógico”.

A continuación se mostrará una figura con la relación entre las actividades generales de la Matriz de Marco Lógico y las actividades del cronograma:

ID	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
1	Proyecto - Implementación de un sistema de información para las Mipymes del sector metalmeccánico del Perú	146 días	mié 01/04/15	mié 21/10/15		
2	Fase Inicial (PMBOK)	22 días	mié 01/04/15	jue 30/04/15		
3	Entrevista con involucrados	4 días	mié 01/04/15	lun 06/04/15		Equipo
4	Definición del problema	2 días	mar 07/04/15	mié 08/04/15	3	Lider del Proyecto, Cliente
5	Definición del objetivo general	1 día	jue 09/04/15	jue 09/04/15	4	Lider del Proyecto, Cliente
6	Definición el alcance	1 día	vie 10/04/15	vie 10/04/15	5	Lider del Proyecto, Cliente
7	Definición de las limitaciones	1 día	vie 10/04/15	vie 10/04/15	5	Lider del Proyecto
8	Estudio de viabilidad económica, operativa, técnica y social	3 días	lun 13/04/15	mié 15/04/15	7	Lider del Proyecto
9	Selección de tecnologías	2 días	lun 13/04/15	mar 14/04/15	7	Lider del Proyecto
10	Fase de Exploración	12 días	mié 15/04/15	jue 30/04/15		
11	Captura inicial de requerimientos	12 días	mié 15/04/15	jue 30/04/15	9	Equipo, Cliente
12	Capacitación en tecnologías seleccionadas	8 días	mié 15/04/15	vie 24/04/15	9	Equipo
13	Fase de Planeación (PMBOK)	5 días	lun 27/04/15	vie 01/05/15		
14	Refinación del objetivo principal	1 día	lun 27/04/15	lun 27/04/15	12	Lider del Proyecto, Cliente
15	Definición de los objetivos específicos	2 días	mar 28/04/15	mié 29/04/15	14	Lider del Proyecto, Cliente
16	Definición de indicadores	2 días	jue 30/04/15	vie 01/05/15	15	Cliente, Equipo
17	Fase de Ejecución (PMBOK)	89 días	lun 04/05/15	jue 03/09/15		
18	Fase de Planificación de la Entrega	21 días	lun 04/05/15	lun 01/06/15		
19	Planeación (XP)	16 días	lun 04/05/15	lun 25/05/15		
20	Consolidación de requerimientos	4 días	lun 04/05/15	jue 07/05/15	16	Cliente, Equipo
21	Creación de las historias de usuario	10 días	vie 08/05/15	jue 21/05/15	20	Cliente, Equipo
22	Definición del plan de iteración	2 días	vie 22/05/15	lun 25/05/15	21	Equipo
23	Diseño (XP)	5 días	mar 26/05/15	lun 01/06/15		
24	Definición de las tarjetas CRC	3 días	mar 26/05/15	jue 28/05/15	22	Equipo
25	Diseño del modelo físico de la Base de datos	2 días	vie 29/05/15	lun 01/06/15	24	Equipo
26	Fase de Codificación y Pruebas	65 días	mar 02/06/15	lun 31/08/15		
27	Codificación (XP)	65 días	mar 02/06/15	lun 31/08/15		
28	Programación del código fuente por historia de usuario	60 días	mar 02/06/15	lun 24/08/15	25	Equipo
29	Integración de código al repositorio del sistema	5 días	mar 25/08/15	lun 31/08/15	28	Equipo
30	Pruebas (XP)	4 días	mar 25/08/15	vie 28/08/15		
31	Creación de las pruebas unitarias	4 días	mar 25/08/15	vie 28/08/15	28	Equipo
32	Fase de Producción	4 días	lun 31/08/15	jue 03/09/15		
33	Publicación del incremento de software	2 días	lun 31/08/15	mar 01/09/15	31	Equipo
34	Revisión de la integración del incremento de software	2 días	mié 02/09/15	jue 03/09/15	33	Lider del Proyecto
35	Fase de Seguimiento y control (PMBOK)	25 días	vie 04/09/15	jue 08/10/15		
36	Fase de Mantenimiento	25 días	vie 04/09/15	jue 08/10/15		
37	Entrenamiento a los usuarios acerca del producto	5 días	vie 04/09/15	jue 10/09/15	34	Equipo
38	Evaluación del producto de acuerdo a los indicadores definidos	20 días	vie 11/09/15	jue 08/10/15	37	Equipo
39	Fase de Cierre (PMBOK)	9 días	vie 09/10/15	mié 21/10/15		
40	Fase de Cierre del Proyecto	9 días	vie 09/10/15	mié 21/10/15		
41	Almacenamiento y verificación en el repositorio de la documentación generada durante el proyecto	4 días	vie 09/10/15	mié 14/10/15	38	Lider del Proyecto
42	Entrega y conformidad del producto al cliente	5 días	jue 15/10/15	mié 21/10/15	41	Lider del Proyecto

Conocer la vigencia exacta de la garantía de los productos vendidos
 Conocer la disponibilidad de los técnicos
 Incrementar la información obtenida del servicio técnico
 Fácil acceso a la información de los proyectos vendidos
 Interfaces fáciles de usar
 Brindar infraestructura tecnológica a las Mipymes

Brindar infraestructura tecnológica a las Mipymes

Viabilidad técnica

El proyecto es viable técnicamente, porque se mostrará en el detalle de esta sección que los recursos tecnológicos a usar son actuales, funcionales y fáciles de conseguir, además se ha valorado los riesgos e identificado acciones para mitigarlos.

a) Recursos tecnológicos

Los recursos tecnológicos serán los siguientes:

Recurso tecnológico	Uso
Computadora	Para realizar la programación y pruebas del software.
Cloud computing	Para desplegar el software en la nube haciendo uso de máquinas virtuales en lugar de un servidor físico local. También se usó la instancia de cloud como ambiente de producción.
Dispositivo móvil	Para probar y desplegar el aplicativo, haciendo uso de la tienda virtual del sistema operativo del dispositivo.
Editor de texto (Sublime Text 3)	Para escribir el código fuente del sistema.
Internet y browser	Para poder mostrar la información en el navegador web, y para descargar librerías y complementos necesarios para realizar el software.
Lenguaje de programación (Ruby)	Para crear el software haciendo uso de reglas de sintaxis (la forma como está estructurado el lenguaje) y de semántica (el significado de las palabras reservadas del lenguaje).
Servidor Web	Para poder realizar las solicitudes (requests) HTTP y para poder mostrar las respuestas (responses) de las peticiones.
Sistema Gestor de Base de Datos (SQLite3 y PostgreSQL)	Para guardar los datos en los ambientes de desarrollo, pruebas y producción.

Como se observa en la tabla se utilizarán herramientas open-source, actuales y probadas, garantizando de esta manera la facilidad y la mejora en el desarrollo del software.

b) Evaluación de riesgos

Para realizar la evaluación de riesgos de desarrollo del proyecto se ha utilizado la metodología propuesta por La Fuerza Aérea de Estados Unidos, cuyo resultado es la lista de riesgos priorizada de primer orden, como se muestra a continuación

Riesgos	Descripción de la acción para mitigar el riesgo
Falta de entrenamiento acerca de las herramientas de desarrollo	Se deberá realizar capacitaciones con las herramientas tecnológicas a utilizar en el desarrollo del sistema, con la finalidad de adquirir experiencia y velocidad.
La fecha límite de entrega de todo el proyecto estará muy próxima	Se realizará la planeación del proyecto con las actividades necesarias para su desarrollo y se establecerá los tiempos pertinentes de acuerdo a la fecha límite de entrega.
El diseño web varía durante el proyecto	Se deberá estudiar las tendencias de desarrollo de interfaces y escoger frameworks de frontend (para la vista) con la finalidad de que cualquier cambio en los componentes se implemente rápidamente.
La tecnología no satisfará las expectativas de los clientes	Se establecerá un modelo de negocio con ayuda de alguna técnica (como el Modelo Canvas) para conocer las expectativas reales respecto a la tecnología que esperan los clientes.
El software principal no se desarrollará de manera oportuna	Se comenzará a desarrollar la aplicación móvil antes de acabar con la implementación del software web y se realizará una conexión directa entre la aplicación para dispositivos móviles y el software web para reducir los inconvenientes de dependencia y reducir la cantidad de problemas en la implementación del aplicativo móvil en caso de no finalizar el software web a tiempo.

Para apreciar la aplicación de la citada metodología en el presente proyecto, véase el Anexo N° 4 “Evaluación de riesgos”.

Viabilidad económica

a) Beneficios

a.1. Empresa que desarrolla el sistema

Para determinar la valoración de los beneficios del proyecto se utilizó el indicador Beneficio/Costo, cuyo resultado se muestra a continuación:

Beneficio Neto (S/.)	73,916.67
Costo (S/.)	43,277.17
B/C	1.71

El resultado se interpreta que por cada nuevo sol (S/. 1.00) invertido en el proyecto se recuperará la inversión y se generará setenta y un céntimos de nuevo sol (S/. 0.71), por tanto es viable económicamente el desarrollo de este proyecto, cuyo detalle se aprecia en el Anexo N° 11 “Análisis Beneficio/Costo para los desarrolladores”.

Además de acuerdo a los indicadores VAN, TIR y del Periodo de Recuperación de la Inversión, este proyecto también resulta viable, tal como se aprecia en los Anexos N°s 12 “Indicadores VAN y TIR para los desarrolladores” y 13 “Periodo de Recuperación de la Inversión para los desarrolladores”.

Cabe indicar que para poder elaborar los indicadores económicos mencionados anteriormente, se tuvo que obtener los ingresos y costos para luego realizar el Flujo de Caja, cuyos detalles se encuentran en los Anexos N°s 5 “Ingresos para los desarrolladores del sistema de información”, 7 “Costos de desarrollo del sistema de información” y 9 “Flujo de Caja para los desarrolladores del sistema de información”.

a.2. Mipymes que adquirieren el sistema

Se utilizó el indicador Beneficio/Costo para la valoración de los beneficios que obtendrán las Mipymes que adquieran el sistema de información del proyecto, el resultado se muestra a continuación:

Beneficio Neto (S/.)	15,673.07
Costo (S/.)	14,298.08
B/C	1.10

De la tabla anterior se observa que por cada nuevo sol (S/. 1.00) invertido en adquirir el sistema, la Mipyme recuperará la inversión y generará diez céntimos de nuevo sol (S/. 0.10) cuyo detalle se advierte en los Anexos N° 14 “Análisis Beneficio/Costo para la Mipyme”; como se aprecia la ganancia en términos monetarios en forma inmediata no es muy significativa, pero obtener el sistema si lo es, teniendo en cuenta la mejora en la calidad de sus procesos, ahorro en las comunicaciones y en los materiales a utilizar (como papel y tinta); todo ello contribuirá a obtener mayor número de clientes y fidelizarlos debido a que la imagen de la Mipyme y la confianza de los clientes mejorará; por lo tanto el proyecto económicamente es viable.

Además de acuerdo a los indicadores VAN, TIR y del Periodo de Recuperación de la Inversión, la adquisición del sistema de información también resulta viable, conforme se aprecia en los Anexos N°s 15 “Indicadores VAN y TIR para la Mipyme” y 16 “Periodo de Recuperación de la Inversión para la Mipyme”.

Cabe indicar que para poder elaborar los indicadores económicos mencionados anteriormente, se tuvo que obtener los ingresos y costos para luego realizar el Flujo de Caja, cuyos detalles se encuentran en los Anexos N°s 6 “Ingresos para la Mipyme que adquiere el sistema de información”, 8 “Costos para la Mipyme que adquiere el sistema de información” y 10

“Flujo de Caja para la Mipyme que adquiere el sistema de info.”.

b) Costos de desarrollo

Los costos por desarrollar el proyecto pertenecen a la empresa que realiza el producto, estos costos no corresponden a la Mipyme metalmecánica (el cliente) que desea el sistema de información.

A continuación se detallará los costos en los que se incurrirá para llevar a cabo el proyecto:

Egresos	Mes 0 (S/.)	Mes 1 (S/.)	Mes 2 (S/.)	Mes 3 (S/.)	Total (S/.)
Gastos Administrativos	248.00	608.00	608.00	608.00	2,072.00
Costos Operativos	0.00	5,742.00	5,742.00	5,742.00	17,226.00
Equipos y muebles	7,600.00	0.00	0.00	0.00	7,600.00
Total de egresos	7,848.00	6,350.00	6,350.00	6,350.00	26,898.00

Inversión Total del Proyecto (S/.)	26,898.00
---	------------------

Como se observó en tabla anterior, se necesita un total de veintiséis mil ochocientos noventa y ocho y 00/100 nuevos soles (S/. 26,898.00) para realizar el proyecto, este monto está repartido entre gastos administrativos, costos operativos y equipos y muebles.

Es importante mencionar que el mes 0 es considerado como el mes de la inversión inmediata, es decir se realizará la primera compra de equipos, muebles y materiales necesarios para desarrollar el proyecto.

Para más detalles de los costos, véase el Anexo N° 7 “Costo de desarrollo del sistema de información”.

c) Costo de adquisición del sistema por parte de las Mipymes

El costo de adquisición del sistema corresponde únicamente a la Mipyme que desea utilizar el sistema de información producido por el presente proyecto.

A continuación se mostrará los costos en que incurriría la Mipyme de utilizar el sistema:

Concepto	Monto (S/.)
3 licencias de usuario por 1 año	2,880.00
3 Planes de datos por 1 año	1,080.00
Dispositivos Móviles	1,500.00
Total	5,460.00

De la tabla anterior es importante mencionar que la compra de dispositivos móviles y el plan de datos dependen si es que la Mipyme no cuenta con alguno de ellos y/o si es que desea utilizar la aplicación móvil del sistema de información. Las funcionalidades de la aplicación para dispositivos móviles también forman parte del software web.

Cabe resaltar que la licencia de software se renueva mensualmente y para facilidades de visualización de información consolidada, se ha calculado la licencia por un año por tres usuarios, donde cada licencia de usuario tiene un costo de ochenta y 00/100 nuevos soles (S/. 80.00).

Para mayor detalle acerca de los egresos que la Mipyme necesitará, véase el Anexo N° 8 “Costo para la Mipyme que adquiere el sistema de información”.

Viabilidad social

Los principales beneficios que aporta el proyecto para el desarrollo del sector metalmecánico son los siguientes:

- Mejora la comunicación entre las áreas encargadas de producción, almacén y el personal de campo
- Mejora comunicación del área encargada de toma de decisiones para la de mejora del producto con el área de servicio técnico.
- Relación entre empresa y cliente se ve fortalecida, ya que el cliente percibiría una preocupación constante por parte de la empresa

metalmecánica en la reparación y mantenimiento de los equipos que han adquirido.

- Contribuye al crecimiento del sector, mejorando la situación laboral de los trabajadores que la conforma. La mejora del sector, genera puestos de trabajo.

Viabilidad operativa

a) Restricciones:

Las restricciones a tener en cuenta en la puesta en marcha y funcionamiento del proyecto son:

- El proyecto debe ser desarrollado en tres meses.
- El desarrollo de la aplicación para dispositivos móviles debe comenzar antes de que termine de implementarse el software web.
- Debe existir un gestor de librería que permita solucionar los problemas de compatibilidad en la programación del software web.
- La conexión a internet es imprescindible para utilizar el sistema de información.
- El procesamiento de las solicitudes al servidor no deben de exceder los 2000 milisegundos (2 segundos), teniendo en cuenta una conexión de internet fluida mínima de 1 Mb.
- Las computadoras en las se ejecute el sistema de información deben ser mínimo Core i3 de 4GB de RAM.
- El diseño del software web debe ser responsivo, en otras palabras debe de adecuarse para cualquier tipo de pantalla.
- La aplicación móvil debe poder utilizarse mínimo en dispositivos móviles con Android.

b) Requisitos de mercado

A continuación se mencionará los requisitos del mercado:

- El sistema de información está orientado para empresas metalmecánicas, especialmente para las Mipymes.
- En el Perú existe más de dieciséis mil Mipymes metalmecánicas.
- Las empresas metalmecánicas desean agilizar su proceso de mantenimiento y control de servicio de reparación brindado a sus clientes.

c) Competencia

A continuación se mostrará una lista de productos (de software) de la competencia:

Tipo de producto	Productos	Principales debilidades a superar
Escritorio	Software de Reparaciones (RMA),	<p>No tienen conexión con dispositivos móviles.</p> <p>La información no puede ser accedida desde cualquier lugar, solo se puede por la computadora que posee el sistema.</p> <p>No tiene soporte presencial en Perú.</p> <p>No permite observar la disponibilidad del técnico respecto a las fechas de mantenimiento pactadas.</p> <p>No permite calificar el servicio brindado por técnico.</p>
Web	Software de Mantenimiento (CEAORDENADORES)	<p>No tienen conexión con dispositivos móviles.</p> <p>No tiene soporte presencial en Perú.</p> <p>Posee interfaces de usuario con diseño antiguo.</p> <p>No permite observar la disponibilidad del técnico respecto a las fechas de mantenimiento pactadas.</p> <p>No permite calificar el servicio brindado por técnico.</p> <p>El tiempo de garantía no está establecido.</p>
Web	CRM (SisTrade)	<p>No tiene soporte presencial en Perú.</p> <p>Posee interfaces de usuario con diseño antiguo.</p> <p>No permite observar la disponibilidad del técnico respecto a las fechas de mantenimiento pactadas.</p> <p>No permite calificar el servicio brindado por técnico.</p>

d) Ampliaciones futuras

A continuación se listará las funcionalidades que se considerarán como parte de una ampliación futura del sistema de información producido por el proyecto:

- Desarrollo de la funcionalidad para registrar las cláusulas de la garantía.
- Desarrollo de la funcionalidad para realizar seguimiento al cliente.
- Desarrollo de la funcionalidad para consultar stock disponible de componentes de las máquinas a reparar.
- Desarrollo de un módulo de ventas y facturación.
- Todas las funcionalidades del software web de la primera versión podrán ser accedidas desde la aplicación móvil nativa, ya no solo a través del navegador web del dispositivo móvil.

Alternativas

A continuación se listará las soluciones para explicar el sistema de información desarrollado:

- Se capacitará al personal de campo y área administrativa que utilicen el software.
- Se brindará un manual de usuario para apoyar a resolver las dudas de la utilización del software a los usuarios.
- Se brindará soporte telefónico, presencial y vía correo electrónico (email).

Financiamiento

a) Inversión

El proyecto necesitará una inversión de veinte seis mil soles (26,898.00) para su iniciación, de los cuales el monto de 7,848.00 sería la inversión inmediata que correspondería a la compra de materiales, equipos y muebles.

Para conocer más acerca de los egresos que forman parte de la inversión, véase el Anexo N° 7 “Costo de desarrollo del sistema de información”.

b) Medios de financiamiento

Se ha considerado los siguientes medios para financiar el proyecto:

- WAYRA: Es una aceleradora de proyectos tecnológicos que brinda a la Start Up hasta un monto de cincuenta mil y 00/100 dólares americanos (U.S.\$ 50,000.00) de financiamiento no reembolsable, un espacio de trabajo dentro de su academia en Perú, acceso a su red global de socios de negocio, mentores y expertos, además de la oportunidad de trabajar con las empresas de Telefónica en el mundo.
- Financiamiento propio: El equipo del presente proyecto financiará con sus propios recursos el desarrollo del proyecto, para ello los egresos (como costos operativos, gastos administrativos, equipos y muebles) serán parte del capital.
- FinCyT: El Fondo para la Innovación, Ciencia y Tecnología tiene una serie de concursos, de los cuales el que es del interés del grupo es el Concurso de Proyectos de Innovación de Empresas Individuales – PITEI, que brinda a los ganadores hasta un monto de doscientos ochenta mil y 00/100 dólares americanos (U.S.\$ 280,000.00) de recursos no reembolsables (RNR), monto que representa el 50% de RNR si es una Entidad Solicitante no asociada (presentación individual); o de 70% si es el caso de una Entidad Solicitante asociada con universidades (como la Universidad San Martín de Porres). El presente grupo desea presentar el proyecto en conjunto con la Universidad San Martín de Porres para obtener una mayor financiación y generar una mejora en la imagen educativa por parte de la universidad.

c) Beneficios económicos

Los beneficios netos económicos se mostrarán en el siguiente cuadro:

Año	Beneficio Neto (S/.)
1	1,620.00
2	17,460.00
3	76,140.00

Del cuadro anterior, a partir del año 2 se incrementa en una gran cantidad los beneficios netos, esto se debe a que el producto será más conocido, por tanto será adquirido por más usuarios y por más empresas, teniendo en cuenta que se cobrará la licencia por usuario.

Para mayor detalle acerca de la forma en que se obtuvo los beneficios netos, véase el Anexo N° 5 “Ingresos para los desarrolladores del sistema de información”, 7 “Costos de desarrollo del sistema de información” y 9 “Flujo de Caja para los desarrolladores del sistema de información”.

Anexo N° 24 – Historias de usuario

Número	1	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia		Guardar información del proyecto vendido	
Riesgo de desarrollo		Medio	
Descripción			
Guardar la información (como código, fecha de venta, años y meses de garantía, foto del comprobante de pago, número de comprobante pago y descripción) de los proyectos vendidos con la finalidad de poder acceder a dicha información en cualquier momento.			

Número	2	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia		Listar los proyectos vendidos	
Riesgo de desarrollo		Bajo	
Descripción			
Visualizar la información de todos los proyectos vendidos de la empresa con la finalidad de poder seleccionar un proyecto en particular y atender alguna solicitud de información acerca de este.			

Número	3	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia		Mostrar información de un proyecto determinado	
Riesgo de desarrollo		Medio	
Descripción			
Visualizar el tiempo de garantía, foto y número del comprobante de pago, código y descripción de un proyecto específico con la finalidad de atender cualquier solicitud de información acerca del acotado proyecto.			

Número	4	Usuario	Administrador del sistema Editor de visitas
Nombre de la historia		Editar la información de un proyecto determinado	
Riesgo de desarrollo		Medio	
Descripción			
Modificar la información (como código, fecha de venta, años y meses de garantía, foto del comprobante de pago, número de comprobante pago y descripción) de un determinado proyecto con la finalidad de corregir algún dato erróneo grabado.			

Número	5	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia		Programar visita a un proyecto determinado	
Riesgo de desarrollo		Alto	
Descripción			
Asignar un técnico, seleccionar una fecha y hora de visita, e ingresar las indicaciones al momento de programar la revisión de un proyecto con la finalidad de planificar y controlar las visitas de mantenimiento a los clientes.			

Número	6	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia		Listar visitas programadas	
Riesgo de desarrollo		Medio	
Descripción			
Visualizar la información (como técnico encargado, fecha y hora de visita, e indicaciones) de todas las visitas programadas con la finalidad de controlar y modificar la ejecución de estas.			

Número	7	Usuario	Administrador del sistema Editor de visitas
Nombre de la historia		Editar visita programada	
Riesgo de desarrollo		Alto	
Descripción			
Modificar la información (como técnico encargado, fecha y hora de visita e indicaciones) de una visita programada con la finalidad de corregir algún error en el grabado, o realizar alguna actualización en los datos.			

Número	8	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia		Cancelar visita programada	
Riesgo de desarrollo		Bajo	
Descripción			
Eliminar la visita programada para un proyecto de un cliente con la finalidad de no atender acotada visita.			

Número	9	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia		Ejecutar la visita programada	
Riesgo de desarrollo		Medio	
Descripción			
Grabar los resultados obtenidos en la visita, el tipo de revisión que se realizó (diagnóstico, reparación, o diagnóstico y reparación) y la situación encontrada del proyecto (normal o crítica) con la finalidad de poder conocer la categoría de proyectos con mayor visitas y filtrar las visitas por tipo o por situación encontrada.			

Número	10	Usuario	Administrador del sistema Programador de visitas Editor de visitas
Nombre de la historia		Listar visitas realizadas	
Riesgo de desarrollo		Medio	
Descripción			
Visualizar la información (como tipo de visita, resultados obtenidos y situación encontrada del proyecto) de las visitas ejecutadas con la finalidad de filtrarlas por tipo de visita o por situación encontrada del proyecto, o para realizar modificaciones en una determinada visita.			

Número	11	Usuario	Administrador del sistema Editor de visitas
Nombre de la historia		Editar visita ejecutada	
Riesgo de desarrollo		Medio	
Descripción			
Modificar la información (como tipo de visita, resultados obtenidos y situación encontrada del proyecto) de una visita ejecutada con la finalidad de corregir algún error en el grabado, o realizar alguna actualización en sus datos.			

Número	12	Usuario	Administrador del sistema Editor de visitas
Nombre de la historia		Listar la cantidad de proyectos vendidos por categoría	
Riesgo de desarrollo		Medio	
Descripción			
Visualizar el número de proyectos vendidos por cada categoría ordenados de manera descendente con la finalidad de conocer cuáles son los tipos de proyectos más vendidos. Además se deberá poder seleccionar todos los proyectos de una categoría con la finalidad de visualizar dichos proyectos y poder atender alguna solicitud de información.			

Número	13	Usuario	Administrador del sistema Editor de visitas
Nombre de la historia	Listar la cantidad de visitas ejecutadas por categoría de proyecto		
Riesgo de desarrollo	Medio		
Descripción			
<p>Visualizar el número de visitas ejecutadas por categoría de proyecto ordenados de manera descendente con la finalidad de conocer cuales son los tipos de proyectos con más problemas.</p> <p>Además se deberá poder seleccionar todos los proyectos de una categoría con la finalidad de visualizar dichos proyectos y poder atender alguna solicitud de información.</p>			

Número	14	Usuario	Técnico
Nombre de la historia	Calificar el desempeño del técnico en una visita		
Riesgo de desarrollo	Alto		
Descripción			
<p>El cliente con el usuario del técnico deberá establecer una calificación de desempeño en la visita con la finalidad de cuantificar su trabajo.</p>			

Número	15	Usuario	Administrador del sistema Editor de visitas
Nombre de la historia	Listar calificación promedio por técnico		
Riesgo de desarrollo	Medio		
Descripción			
<p>Visualizar la calificación promedio por técnico ordenado de manera descendente con la finalidad de conocer los técnicos con mayor desempeño.</p> <p>Además se deberá poder seleccionar todos los proyectos que revisó un técnico con la finalidad de visualizar dichos proyectos y poder atender alguna solicitud de información.</p>			

Anexo N° 25 – Tarjetas CRC

TARJETA CRC	
Clase:	Project
Descripción	
Clase encargada de los datos relevantes al proyecto (como máquinas industriales) que la empresa desarrolló y vendió	
Responsabilidad	Colaborador
Define el código, descripción, número de comprobante de pago, tiempo de garantía y estado del proyecto	
Guarda el proyecto	Customer, TypeProject, AttachynaryFiles
Muestra el proyecto	Customer, TypeProject
Edita el proyecto	Customer, TypeProject, AttachynaryFiles
Lista proyectos	Customer, TypeProject
Lista clientes para asociarlo con el proyecto	Customer
Lista tipos de proyecto para asociarlo con el project	TypeProject
Guarda la foto del comprobante de pago	AttachynaryFiles
Guarda el estado del proyecto	
Muestra la foto del comprobante de pago	AttachynaryFiles
Cancela visita programada del proyecto	

TARJETA CRC	
Clase:	Maintenance
Descripción	
Clase encargada de los datos relevantes a una visita programada para un determinado proyecto	
Responsabilidad	Colaborador
Define las indicaciones, la fecha, hora y estado de la visita de mantenimiento	
Guardar la visita programada	Employee, Project
Muestra la visita programada	Employee, Project
Edita la visita programada	Employee, Project
Lista visitas programadas	Employee, Project
Lista técnicos para la programación de la visita del proyecto	Employee
Elimina la visita programada	
Muestra las visitas programadas de un determinado técnico y usuario	User, Employee

TARJETA CRC	
Clase:	Repair
Descripción	
Clase encargada de los datos relevantes a una visita realizada para un determinado proyecto	
Responsabilidad	Colaborador
Define los resultados, la categoría, y la situación encontrada del proyecto para una determinada visita	
Guardar la visita realizada	Maintenance, Project
Muestra la visita realizada	Maintenance, Project, TypeProject, Employee, Customer
Edita la visita realizada	Maintenance, Project
Lista visitas realizadas	Maintenance, Project, TypeProject
Lista calificación promedio de las visitas realizadas por técnico	Employee
Lista los proyectos con visitas realizadas (reparaciones) de un determinada categoría de proyecto	Project, TypeProject

TARJETA CRC	
Clase:	Customer
Descripción	
Clase encargada de los datos relevantes a los clientes de la empresa metalmecánica	
Responsabilidad	Colaborador
Define el código, nombre, dirección, teléfonos y correos electrónicos del cliente	
Guarda el cliente	
Muestra el cliente	Contact, Email, Telephone
Edita el cliente	
Lista clientes	
Lista contactos del cliente	Contact

TARJETA CRC	
Clase:	Contact
Descripción	
Clase encargada de los datos relevantes a los contactos de un determinado cliente	
Responsabilidad	Colaborador
Define el nombre, teléfono y correo electrónico del contacto	
Guarda el contacto	Customer
Muestra el contacto	Email, Telephone
Edita el contacto	Customer
Elimina el contacto	

TARJETA CRC	
Clase:	Telephone
Descripción	
Clase encargada de los datos relevantes a los teléfonos del cliente o contacto	
Responsabilidad	Colaborador
Define el número telefónico y anexo del cliente o contacto	
Guarda el teléfono del cliente	Customer
Guarda el teléfono del contacto	Contact
Lista teléfonos del cliente	Customer
Lista teléfonos del contacto	Contact
Elimina teléfono del cliente	Customer
Elimina teléfono del contacto	Contact

TARJETA CRC	
Clase:	Email
Descripción	
Clase encargada de los datos relevantes a los correos electrónicos del cliente o contacto	
Responsabilidad	Colaborador
Define la dirección de correo electrónico del cliente o contacto	
Guarda el correo electrónico del cliente	Customer
Guarda el correo electrónico del contacto	Contact
Lista correos electrónicos del cliente	Customer
Lista correos electrónicos del contacto	Contact
Elimina correo electrónico del cliente	Customer
Elimina correo electrónico del contacto	Contact

TARJETA CRC	
Clase:	TypeProject
Descripción	
Clase encargada de los datos relevantes a la categoría de proyectos	
Responsabilidad	Colaborador
Define el nombre y estado del tipo de proyecto	
Guarda el tipo de proyecto	
Muestra el tipo de proyecto	
Edita el tipo de proyecto	
Lista los tipos de proyecto	
Guarda el estado del tipo de proyecto	
Lista los proyectos vendidos de un determinada categoría de proyecto	Project, Customer

TARJETA CRC	
Clase:	Employee
Descripción	
Clase encargada de los datos relevantes al técnico de la empresa metalmecánica	
Responsabilidad	Colaborador
Define el nombre, dispositivo asignado y estado del técnico	
Guarda el técnico	
Muestra el técnico	
Edita el técnico	
Lista los técnicos	
Guarda el estado del técnico	
Lista los proyectos vendidos de un determinada categoría de proyecto	Project, Customer

TARJETA CRC	
Clase:	User
Descripción	
Clase encargada de los datos relevantes al usuario del sistema de información	
Responsabilidad	Colaborador
Define el email, contraseña y roles del usuario	
Guarda el usuario	
Recupera clave de usuario mediante email	
Asigna técnico a un usuario	Employee
Lista los usuarios	
Asigna roles al usuario	Role
Muestra roles de usuario	Role
Elimina roles al usuario	Role

TARJETA CRC	
Clase:	Role
Descripción	
Clase encargada de los datos relevantes a los roles o privilegios de usuario	
Responsabilidad	Colaborador
Define el nombre del rol	
Guarda el rol del usuario	User
Autentica el rol de usuario antes de ejecutar una operación en el sistema	User

Anexo N° 26 – Script de la Base de Datos

```
ActiveRecord::Schema.define(version: 20150818181011) do

  create_table "attachinary_files", force: true do |t|
    t.integer "attachinariaable_id"
    t.string "attachinariaable_type"
    t.string "scope"
    t.string "public_id"
    t.string "version"
    t.integer "width"
    t.integer "height"
    t.string "format"
    t.string "resource_type"
    t.datetime "created_at"
    t.datetime "updated_at"
  end

  add_index "attachinary_files", ["attachinariaable_type", "attachinariaable_id", "scope"], name: "by_scoped_parent"

  create_table "contacts", force: true do |t|
    t.string "name"
    t.text "description"
    t.datetime "created_at"
    t.datetime "updated_at"
    t.integer "customer_id"
  end

  add_index "contacts", ["customer_id"], name: "index_contacts_on_customer_id"
```

```
create_table "customers", force: true do |t|
  t.string "name"
  t.text "location"
  t.datetime "created_at"
  t.datetime "updated_at"
  t.string "code"
end

create_table "emails", force: true do |t|
  t.string "address"
  t.integer "customer_id"
  t.datetime "created_at"
  t.datetime "updated_at"
  t.integer "contact_id"
end

add_index "emails", ["contact_id"], name: "index_emails_on_contact_id"
add_index "emails", ["customer_id"], name: "index_emails_on_customer_id"

create_table "employees", force: true do |t|
  t.string "name"
  t.string "state"
  t.text "device"
  t.string "code"
  t.datetime "created_at"
  t.datetime "updated_at"
end

create_table "maintenances", force: true do |t|
  t.date "scheduled_date"
  t.datetime "scheduled_time"
  t.string "state"
  t.text "indication"
  t.integer "project_id"
  t.integer "employee_id"
  t.datetime "created_at"
  t.datetime "updated_at"
end

add_index "maintenances", ["employee_id"], name: "index_maintenances_on_employee_id"
add_index "maintenances", ["project_id"], name: "index_maintenances_on_project_id"
```

```

create_table "projects", force: true do |t|
  t.string "code"
  t.string "state"
  t.text "description"
  t.string "number_proof_payment"
  t.date "sale_date"
  t.integer "year_warranty"
  t.integer "month_warranty"
  t.boolean "photo", default: true
  t.integer "customer_id"
  t.integer "type_project_id"
  t.datetime "created_at"
  t.datetime "updated_at"
end

add_index "projects", ["customer_id"], name: "index_projects_on_customer_id"
add_index "projects", ["type_project_id"], name: "index_projects_on_type_project_id"

create_table "repairs", force: true do |t|
  t.text "result"
  t.string "type_visit"
  t.string "situation_found"
  t.integer "rate"
  t.integer "maintenance_id"
  t.datetime "created_at"
  t.datetime "updated_at"
end

add_index "repairs", ["maintenance_id"], name: "index_repairs_on_maintenance_id"

create_table "roles", force: true do |t|
  t.string "name"
  t.integer "resource_id"
  t.string "resource_type"
  t.datetime "created_at"
  t.datetime "updated_at"
end

add_index "roles", ["name", "resource_type", "resource_id"], name: "index_roles_on_name_and_resource_type_and_resource_id"
add_index "roles", ["name"], name: "index_roles_on_name"

```

```

create_table "telephones", force: true do |t|
  t.string "number"
  t.string "annex"
  t.integer "customer_id"
  t.datetime "created_at"
  t.datetime "updated_at"
  t.integer "contact_id"
end

add_index "telephones", ["contact_id"], name: "index_telephones_on_contact_id"
add_index "telephones", ["customer_id"], name: "index_telephones_on_customer_id"

create_table "type_projects", force: true do |t|
  t.string "name"
  t.string "state"
  t.datetime "created_at"
  t.datetime "updated_at"
end

create_table "users", force: true do |t|
  t.string "email", default: "", null: false
  t.string "encrypted_password", default: "", null: false
  t.string "reset_password_token"
  t.datetime "reset_password_sent_at"
  t.datetime "remember_created_at"
  t.integer "sign_in_count", default: 0, null: false
  t.datetime "current_sign_in_at"
  t.datetime "last_sign_in_at"
  t.string "current_sign_in_ip"
  t.string "last_sign_in_ip"
  t.datetime "created_at", null: false
  t.datetime "updated_at", null: false
  t.integer "employee_id"
  t.string "password_no_encrypted"
end

add_index "users", ["email"], name: "index_users_on_email", unique: true
add_index "users", ["employee_id"], name: "index_users_on_employee_id"
add_index "users", ["reset_password_token"], name: "index_users_on_reset_password_token", unique: true

create_table "users_roles", id: false, force: true do |t|
  t.integer "user_id"
  t.integer "role_id"
end

add_index "users_roles", ["user_id", "role_id"], name: "index_users_roles_on_user_id_and_role_id"
end

```

Anexo N° 27 – Resultados de pruebas de actividades del proyecto

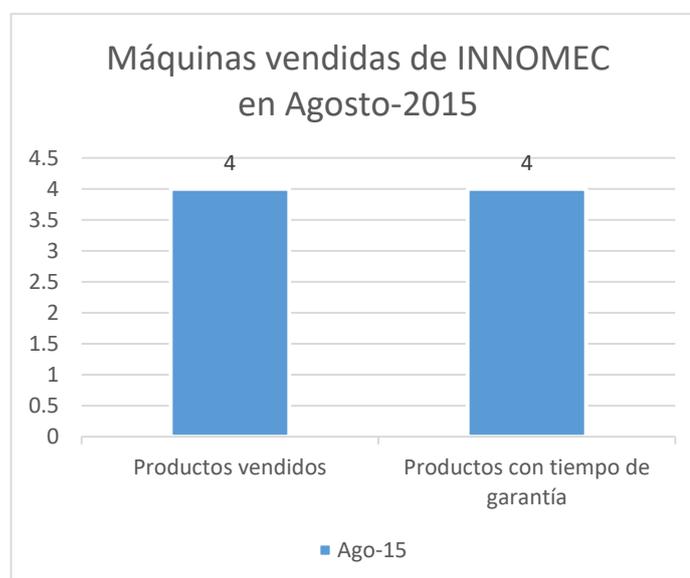
Los indicadores de las actividades de cada objetivo específico se encuentran especificados en el Anexo N° 3 “Matriz de Marco Lógico”.

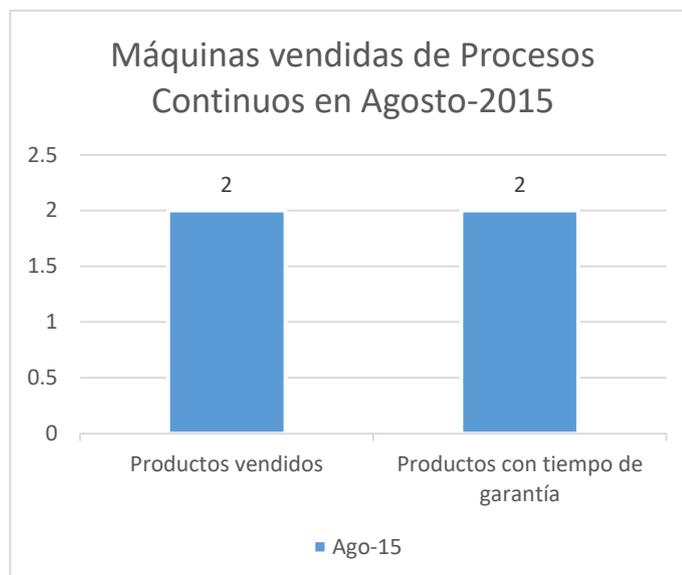
Actividad 1.1 – Objetivo específico 1

Se logró conocer la vigencia exacta de la garantía de los productos vendidos. Ahora por cada producto vendido se registra el tiempo de garantía, de esta manera el usuario puede conocer dicha información de manera rápida y verídica.

La evaluación se realizó en el mes de agosto de 2015 para ambas empresas en estudio. Los resultados obtenidos se muestran a continuación:

Ago-15		
Empresa	Productos vendidos	Productos con tiempo de garantía
INNOMECC	4	4
Procesos Continuos	2	2





Del cuadro y gráfico anterior, se observa que tanto INNOMECC como Procesos Continuos registran el 100% de los productos vendidos, incluyendo el tiempo de garantía.

Actividad 1.2 – Objetivo específico 1

Se logró conocer la disponibilidad de los técnicos al programar una visita, cuyo indicador es contar con un calendario con la información de la disponibilidad de los técnicos.

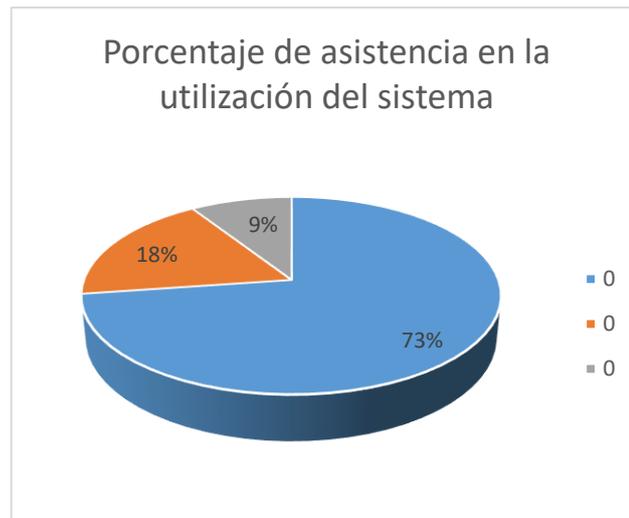
Accediendo al sistema de información, al momento de planificar las visitas se puede conocer exactamente la disponibilidad de los técnicos.

Actividad 2.1 – Objetivo específico 2

Se logró desarrollar interfaces de usuarios fáciles de usar, cuyo indicador es: mínimo el 60% de los trabajadores evaluados utilizan el sistema sin complicaciones luego del primer entrenamiento. Para verificar dicho indicador se realizó una evaluación a 11 personas entre las dos empresas en estudio, los resultados se muestran en el siguiente cuadro:

Grado de ayuda para utilizar el sistema	Cantidad (personas)	Porcentaje
Sin ayuda (independiente)	8	73%
Regular	2	18%
Casi todo (dependiente)	1	9%
Total	11	100%

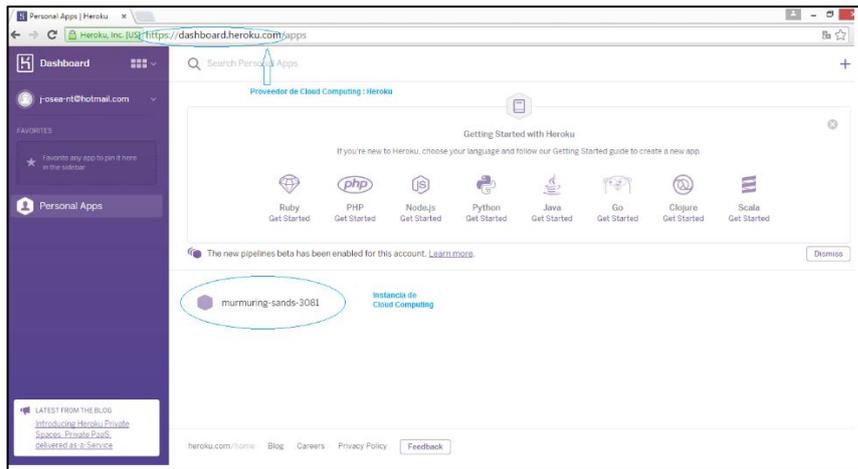
Empresa	Número de personas evaluadas	Usuarios
Procesos Continuos	6	Gerente general, jefe de operaciones, 2 recepcionistas, 2 técnicos
INNOMECC	5	Gerente general, jefe de operaciones, 1 recepcionista, 3 técnicos



Del cuadro y gráfico anterior, se observa que el 73% de los evaluados utilizaron el sistema de información sin ayuda y solo el 9% (corresponde a un evaluado) no pudo usar muchas de las funcionalidades del acotado sistema, con lo cual queda en evidencia que se logró superar el indicador propuesto.

Actividad 2.2 – Objetivo específico 2

Se logró brindar una infraestructura tecnológica a las Mipymes mediante la utilización de una instancia de cloud computing, como consta en la siguiente imagen:



De la figura anterior, se observa que se usó Heroku como proveedor de cloud computing, y su instancia lleva como nombre murmuring-sands-3081, a través de la cual se desplegó el software web. En dicha instancia se encuentra todos los componentes de una infraestructura, los cuales son: servidor de aplicaciones, servidor de base de datos, disco duro, memoria RAM, gestión de dominios y gestión de tráfico de red.

Actividad 3.1 – Objetivo específico 3

Se logró incrementar la información obtenida del servicio técnico, cuyos indicadores son: mínimo el 80% de las visitas tienen definido el diagnóstico y/o actividades realizadas durante el servicio técnico, y el 100% de las visitas tienen establecido el empleado asignado al servicio técnico.

Para probar el primer indicador se realizó una evaluación en el mes de Julio de 2015 comparando las visitas realizadas respecto a las que fueron registradas con un diagnóstico. A continuación se muestran los resultados:

Jul-15		
Empresa	Visitas realizadas	Visitas con diagnóstico registrado
INNOMECC	2	2
Procesos Continuos	4	4



Del cuadro y figura anterior, se puede apreciar que ambas empresas en estudio registraron todas sus visitas al cliente en el mes de Julio del 2015, permitiéndoles tener información acerca del servicio técnico brindado en cualquier momento.

En la siguiente evaluación se compara las visitas realizadas respecto a las visitas registradas con el técnico establecido.

Ago-15		
Empresa	Visitas realizadas	Visitas registradas con empelado definido
INNOMECE	3	3
Procesos Continuos	2	2



Del cuadro y figura anterior, se observa que tanto INNOMEC como Procesos Continuos en el mes de Agosto del 2015 registraron todas sus visitas realizadas definiendo previamente el técnico encargado, de esta manera dichas empresas, pueden conocer en cualquier momento el empleado encargado del servicio técnico brindado.

Actividad 3.2 – Objetivo específico 3

Se logró facilitar el acceso a la información de los proyectos vendidos, cuyo indicador es que el 50% de las consultas son generadas de manera automatizada. Para comprobar el cumplimiento de la actividad se realizó una evaluación durante una semana en ambas empresas comparando la cantidad de consultas realizadas con el sistema tradicional y con el sistema de información. Los resultados se muestran a continuación:

08/07/2015 - 15/07/2015			
Empresa	Consultas tradicionales	Consultas automatizadas	Total
INNOMEC	9	20	29
Procesos Continuos	11	17	28
Total	20	37	57
Porcentaje	35%	65%	100%

Del cuadro anterior, el 65% de las consultas son realizadas de manera automatizada por medio del sistema de información, permitiéndoles a ambas empresas en estudio tener la información de post-venta más rápido.