

UNIVERSIDAD TÉCNICA DEL NORTE



FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERA EN SISTEMAS COMPUTACIONALES

TEMA:

**SISTEMA DE GESTIÓN Y CONTROL DE TALLERES DE VEHÍCULOS
PARA EL CENTRO INTEGRAL DE REPARACIÓN AUTOMOTRIZ
MEGA-AUTO**

AUTOR:

Jenny Alicia Ortiz Benítez

DIRECTOR:

Ing. Carpio Pineda

Ibarra – Ecuador

2012

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

Artículo Científico

1 Introducción

En la actualidad se vive en un mundo globalizado, en el que las empresas buscan cada vez cómo mantenerse en el tiempo; por tal motivo, las organizaciones deben ser entendidas como un gran sistema de información, que alcanzan el éxito a medida que quienes las dirigen entienden que: la información correcta, adecuada y oportuna conducirá a decisiones y acciones más eficientes, y las decisiones informadas serán las que le diferencien de la competencia.

Por tal motivo, las empresas requieren de una constante planificación de sus actividades, y a medida que el número de datos y procesos en una organización se incrementa, se tiene la necesidad de gestionarlos y controlarlos, estableciendo metas específicas para ayudar a alcanzar el crecimiento del negocio.

Ante esta realidad, Mega-Auto tiene como centro de atención la automatización de la información y de los procesos; que den como resultado un alto componente de lógica del negocio, y que involucren decisiones importantes en relación a cómo satisfacer los requerimientos de los clientes y tender a un óptimo uso de los recursos de la empresa.

1.1 Problema

El Centro Integral de Reparación Automotriz Mega-Auto es una empresa dedicada a la rama de reparación de vehículos, venta de repuestos y otros servicios relacionados con esta actividad, inicia sus funciones sin tener una estructura de organización que le permita el control detallado de la información y de los procesos que diariamente se realizan en su entorno de trabajo; haciendo que éste se mantenga estancado con bajos niveles de rentabilidad, con relación al monto de la inversión.

De ahí la necesidad de implantar un sistema informático con un esquema de funcionamiento, en base a estructuras organizadas de información que le permitan registrar y administrar la información de: clientes, vehículos, aseguradoras, repuestos y/o materiales; elaborar

presupuestos, cotizar repuestos, generar órdenes de trabajo, asignar a los técnicos los trabajos de mano de obra, realizar requisiciones de materiales o de repuestos, conocer el tiempo estimado para la reparación por lo tanto establecer el tiempo de entrega del vehículo, formular colores, conocer la utilidad que percibirá la empresa en la reparación del vehículo, clasificar y resumir la información de cada una de las transacciones efectuadas por la empresa; Todo esto ayudará al crecimiento y posicionamiento de la empresa en un mundo globalizado como el actual, con altos niveles de competitividad, en el que las empresas buscan cada vez como mantenerse en el tiempo.

Con la implementación del Sistema Informático se pretende mejorar los mecanismos y procesos que se han venido llevando a cabo en Mega-Auto; en caso de continuar con el trabajo que actualmente se lleva a cabo, mismo que no ha cumplido con las expectativas empresariales, se presentarán limitantes referente al control y gestión de la información y de los procesos.

1.2 Objetivos

Objetivo General:

- Optimizar la gestión y el control de la información y de los procesos del centro integral de reparación automotriz Mega-Auto; para mejorar la productividad del centro, de tal manera que le permita brindar un servicio oportuno y eficiente; todo esto con la utilización de herramientas de código abierto.

Objetivos Específicos:

- Definir los servicios que presta Mega-Auto, los que deben reflejar claramente lo que se espera del entorno en el que se desarrolla, en relación a sus funciones y cualidades de trabajo.
- Establecer la secuencia óptima de los procesos y recursos que implica el control de un taller automotriz, mediante la adquisición e identificación de los requerimientos.

- Realizar el estudio de las herramientas informáticas a utilizar para el desarrollo del sistema.
- Implementar el sistema informático que gestione y controle la información y los procesos del centro automotriz, de tal manera, que permita disponer de la información oportuna, correcta e integrada en el momento que se necesita.
- Implantar un modelo de control de taller, que permita facilitar y agilizar la capacitación, el acceso, la administración y el análisis de la información, a los usuarios del sistema mediante una interfaz amigable.

1.3 Alcance

El sistema de Gestión y Control de talleres de vehículos para el Centro Integral de reparación Automotriz Mega-Auto abarca los siguientes módulos.

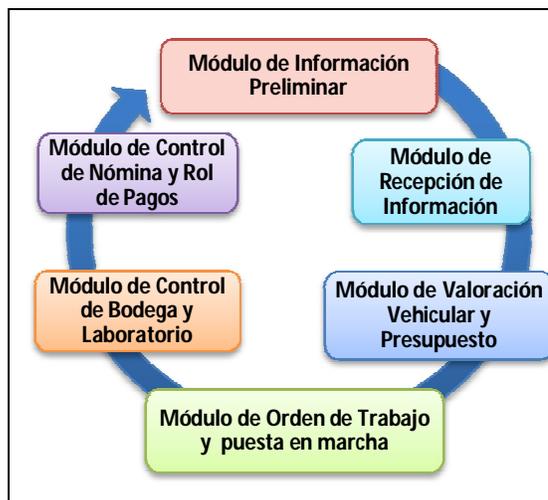


Fig.1: Esquema del Alcance del Proyecto

Administrar información preliminar:

- Registrar y administrar cuentas de usuarios.
- Asignar permisos de acceso a las cuentas de usuarios.
- Registrar y administrar la información de: Marcas y Modelos de Vehículos, Tipos de Vehículos, Tipos de Operaciones, Trabajos de Mano de Obra, Trabajos externos.

Administrar información del Cliente y Vehículo:

- Registrar y administrar datos del Cliente y del vehículo a repararse.
- Generar la relación de información entre: Cliente, Vehículo y Aseguradora (éste último si fuere necesario).

- Realizar búsquedas de información para la localización de clientes, vehículos y aseguradoras en caso de estar ya registrados.
- Imprimir Reportes de Clientes, Vehículos, Aseguradoras.

Realizar el registro de la información obtenida en base a la Valoración Vehicular:

- Definir los procesos de mano de obra que se realizarán para la reparación del vehículo.
- Definir los repuestos y materiales a utilizar para la reparación del vehículo.
- Cotizar Repuestos.
- Establecer el tiempo estimado para la realización del trabajo, por lo tanto conocer el tiempo de entrega del vehículo.
- Elaborar un presupuesto o proforma previo a la Reparación.
- Conocer el valor y el porcentaje de utilidad que genera si se repara el vehículo presupuestado.
- Imprimir proforma de reparación.

Controlar y Administrar información de las Bodegas:

- Registrar grupos, subgrupos, productos y precios.
- Registrar el Conteo Físico de productos.
- Realizar Requisiciones de materiales y/o repuestos según la orden de trabajo generada.
- Controlar las entradas y salidas de productos por Ajuste de Inventario.
- Grabar los repuestos incluidos en la Orden de Trabajo.
- Mantener al día las existencias.

Controlar y administrar información de Laboratorio:

- Formular colores.
- Realizar requisiciones de materiales para la creación de colores.

Registrar y administrar información del Personal de la Empresa:

- Registrar nómina.
- Registrar préstamos.
- Elaborar Rol de Pagos.

1.4 Justificación

El Centro Integral de reparación automotriz Mega-Auto, en sus actividades diarias genera datos como producto secundario, que son el resultado de todas las transacciones que se realizan en la empresa; la idea central es dejar de considerarlos como simples datos, para convertirlos en información que enriquezca las decisiones de los ejecutivos.

Siendo que el Mega-Auto desea maximizar el valor de su información para lograr tener una mayor ventaja competitiva; que le permita convertir los datos operacionales en verdaderas fuentes de información enfocadas a soluciones empresariales, y tener a disposición de los usuarios indicados la información en el momento que se necesita, surge la necesidad de implantar un sistema informático. De esta manera se obtendrá una buena gestión y control de la información y de los procesos del centro integral, con altas prestaciones de trabajo, que permita mejorar la productividad, la atención y el servicio a los clientes.

Conforme a las necesidades de cómo satisfacer los requerimientos de los usuarios y tender a un uso óptimo de los recursos de la empresa; se ha optado por la utilización de herramientas open source, por lo que, el sistema Informático se desarrollará como una aplicación de escritorio basado en el lenguaje de programación Java, en el entorno de desarrollo integrado Eclipse Galileo, con el motor de base de datos PostgreSQL, y como herramienta para la generación de Reportes de usuario final se utilizará iReport. Además se manejará la metodología de desarrollo RUP que ofrece un conjunto de técnicas para desarrollar y mantener software de calidad.

2 Herramientas de Desarrollo

A continuación se describen las herramientas utilizadas para el desarrollo del Aplicativo.

2.1 IDE para Java

Los IDEs son Entornos de Desarrollo Integrados que brindan un conjunto de herramientas para el programador; suelen incluir en una misma suite: un buen editor de código, administrador de proyectos y archivos, enlace transparente a compiladores y debuggers, etc.

Son varias las opciones de entornos de desarrollo integrado para Java, entre los principales y más populares se encuentran: Eclipse, BlueJ, Jbuilder, etc.

Eclipse: La plataforma Eclipse consiste en un Entorno de Desarrollo Integrado de código abierto y extensible. Cuenta con numerosas herramientas de desarrollo de software. Facilita las tareas de edición, compilación y ejecución de programas durante su fase de desarrollo.^[1]

^[1] (2012). Eclipse. Recuperado de http://es.wikipedia.org/wiki/Eclipse_%28software%29

Ventajas:

- Es un IDE universal no solo es para java, también se puede usar con otros lenguajes de programación como son C/C++, Jsp, Php, etc.
- El consumo de memoria es algo inferior a otros entornos de desarrollo.
- Existen versiones para casi cualquier sistema operativo entre ellos linux y windows.
- Tiene opción de multilinguaje.
- No consume mucho espacio en el disco duro.
- Es portable.
- Se puede trabajar varios espacios de trabajo.
- Tiene un excelente asistente de etiquetas.

2.2 Java como Lenguaje de Programación

Java es un lenguaje de desarrollo de propósito general, y como tal es válido para realizar todo tipo de aplicaciones profesionales. Actualmente es uno de los lenguajes más usados para la programación en todo el mundo.

Características de Java:^[2]

- Lenguaje Simple
- Orientado a objetos
- Distribuido
- Interpretado
- Seguro
- Independiente de la plataforma
- Robusto

¿Qué se necesita para empezar a programar en java?

- **JDK** (Java Development Kit - Kit de desarrollo de Java): Es un software que provee herramientas de desarrollo para la creación de programas en java.^[3]

A continuación se listan algunas de las utilidades que se pueden encontrar en el JDK:

- **javac:** Es el compilador de Java. Se encarga de convertir el código fuente escrito en Java a *bytecode*.
- **java:** Es el intérprete de Java. Ejecuta el *bytecode* a partir de los archivos *.class*.
- **javadoc:** Se utiliza para crear documentación en formato HTML a

^[2] Características de Java como Lenguaje de Programación. Recuperado de <http://www.infor.uva.es/~jmrr/tgp/java/JAVA.html>

^[3] Programando en Java I. Recuperado de <http://leoag.wordpress.com/2010/08/02/programando-en-java-i/>

partir del código fuente Java y los comentarios que contiene.

- javap: Es un desensamblador de Java.

Para poder ejecutar cualquier aplicación Java en cualquier sistema operativo es necesario tener instalado el JRE (Java RunTime Environment - Entorno de desarrollo de Java). El JRE se compone de herramientas necesarias como la máquina virtual de java (java.exe) y el conjunto de librerías estándar de Java. El JDK incluye a JRE.

- **JVM** (Java Virtual Machine - Máquina Virtual de Java): Es el entorno en el que se ejecutan los programas Java, su misión principal es la de garantizar la portabilidad de las aplicaciones Java. Se le llama "máquina virtual" porque, sin importar el tipo de máquina en la cual se esté ejecutando el programa, crea una máquina simulada que proporciona la plataforma correcta para ejecutar estas aplicaciones.

2.3 API de Persistencia para Java (JPA)

El Java Persistence API (JPA) es una especificación de Sun Microsystems para la persistencia de objetos Java a cualquier base de datos relacional.

La especificación JPA abarca tres áreas:

- *El API JPA*: Es un conjunto de clases e interfaces incluidas dentro del paquete javax.persistence que serán empleadas para operar con los objetos persistentes.
- *Mapeo objeto-relacional*: Es la relación entre entidades Java y tablas de la base de datos. Representa la información que permite especificar al motor de persistencia la manera en que se deben mapear los objetos con las tablas de la base de datos. El mapeo objeto/relacional se realiza mediante anotaciones en las propias clases de la entidad, las anotaciones pueden aplicarse a clases, métodos o atributos.
- *Java Persistence Query Language (JPQL)*: Es un lenguaje de manipulación de objetos, con el que se puede definir operaciones de tratamiento de objetos. Su sintaxis es similar a la del lenguaje SQL estándar, pero adaptadas al tratamiento de objetos.

Entidad:

Una entidad en JPA se refiere generalmente a una tabla de la base de datos. Las instancias de una entidad corresponden a filas dentro de la tabla.

Persistencia:

El concepto de persistencia implica el hecho de almacenar las entidades (objetos Java) en un sistema de almacenamiento, normalmente una base de datos relacional (tablas, filas, y columnas). Más allá del proceso de almacenar entidades en una base de datos, todo sistema de persistencia debe permitir recuperar, actualizar y eliminar dichas entidades.

Motor de Persistencia:

Un motor de persistencia es una capa de software que transforma transparentemente, los objetos de la aplicación en registros de una base de datos relacional.

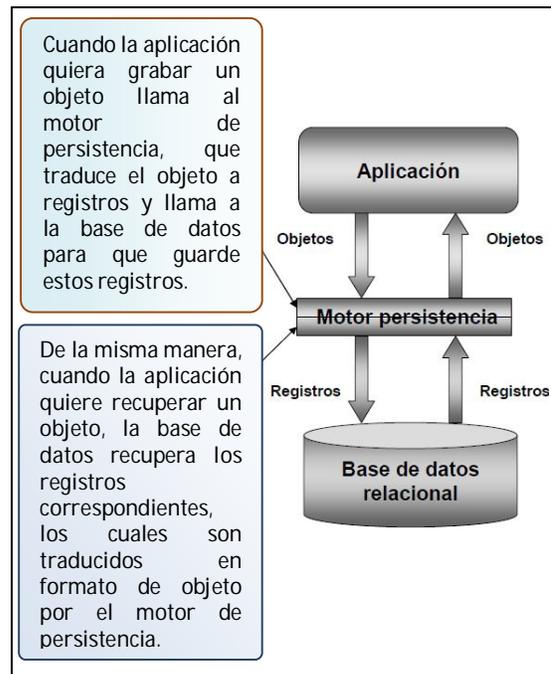


Fig.2: Motor de persistencia

Unidad de persistencia:

La agrupación de entidades en una aplicación se llama unidad de persistencia. Son el conjunto de todas las clases administradas por el EntityManager de una aplicación. Se encuentran definidas en el archivo de configuración persistence.xml dentro del directorio META-INF de la aplicación.^[4]

Una unidad de persistencia describe los siguientes datos:

- Clase con la implementación JPA del proveedor de persistencia utilizado.

^[4] (2007). Usando el API de persistencia en aplicaciones de escritorio. Recuperado de <http://www.apuntesdejava.com/2007/06/usando-el-api-de-persistencia-en.html>

- Clases de entidad que van a ser gestionadas por el proveedor de persistencia.
- Propiedades necesarias para el proveedor conectarse con la base de datos.

2.4 Servidor de Base de Datos PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos objeto-relacional, su código fuente está disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales.

Características:^[5]

- Está basado en lenguaje C.
- Interoperabilidad con otros DBMS: SQL.
- De almacenamiento confiable, consistente, robusto.
- De manipulación potente, flexible y eficiente.
- Utiliza Control de Concurrencia Multi – Versión.
- Presenta soporte completo ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad)
- Resistencia a fallas. Soporta caídas del sistema sin perder ningún dato.
- Tiene soporte para lenguajes procedurales internos, incluyendo un lenguaje nativo denominado PL/PgSQL, el cual es un lenguaje comparable a PL/SQL de Oracle.
- APIs para programar en C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC, PHP y muchos otros lenguajes, Completa documentación.
- Implementación del Estándar SQL92/SQL99.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

2.5 Metodología de desarrollo RUP

Las metodologías y estándares utilizados en el desarrollo de software proporcionan las guías para poder conocer todo el camino a recorrer desde antes de empezar la implementación, con lo cual se asegura la calidad del producto final, así como también el cumplimiento en la entrega del mismo en el tiempo establecido.

UML (Lenguaje Unificado de Modelado): Es un lenguaje estándar de modelado para desarrollo de sistemas y de software. Es usado para especificar, visualizar y documentar los

diferentes aspectos relativos a un sistema de software bajo desarrollo, así como para modelado de negocios.

Tiene una gran aplicación en la representación y modelado de la información que se utiliza en las fases de análisis y diseño. En diseño de sistemas se modela por una importante razón: gestionar la complejidad. El bloque básico fundamental de UML es un diagrama.^[6]

RUP (Proceso Unificado de Rational):

RUP es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de software. Estos procesos de ingeniería se basan en las “mejores prácticas” del desarrollo de software. Su objetivo es asegurar la producción de software de alta calidad que satisfaga la necesidad del usuario final dentro de un tiempo y presupuesto previsible.

Ciclo de vida de RUP:

RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones.^[7]

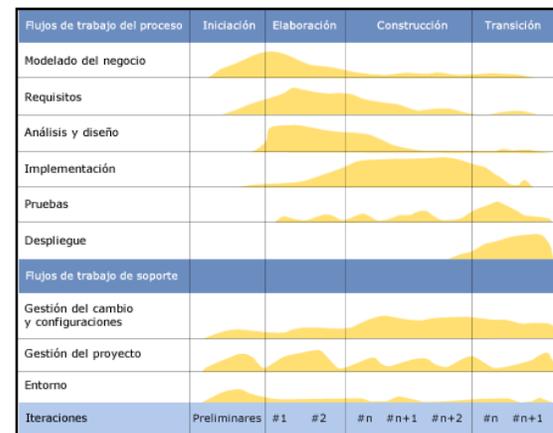


Fig. 3: Ciclo de Vida de RUP

Fases y Disciplinas del Ciclo de vida de RUP

Fase de inicio: Las iteraciones hacen mayor énfasis en actividades de Modelado del negocio (el equipo se familiarizará más al

^[5] Sabana Mendoza, M. (2006). Php con PostgreSQL8 (1ª edición). Lima Perú: Megabyte

^[6] Joyanes A., L., Zahonero Martínez, I. (2010). Programación en C, C++, JAVA y UML, México D.F.: Mc. Graw Hill

^[7] RUP Ciclo de Vida. Recuperado de http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational

funcionamiento de la empresa, sobre conocer sus procesos, entender el problema actual en la organización) y de Requisitos (se identifican los requisitos, lo que en realidad se necesita, lo que el sistema podría hacer, definir el ámbito del sistema).

Fase de Elaboración: Las iteraciones se orientan al desarrollo de la línea base de la arquitectura, abarcan más los flujos de trabajo de requisitos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación.

Fase de construcción: Se lleva a cabo la construcción del producto por medio de una serie de iteraciones. Para cada iteración se selecciona algunos Casos de Uso, se refina su análisis y diseño y se procede a su implementación y pruebas.

Fase de transición: Se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

Durante todo el Proyecto:

Gestión del proyecto: Se vigila el cumplimiento de los objetivos, gestión de riesgos y restricciones para desarrollar un producto que sea acorde a los requisitos de los clientes y los usuarios.

Configuración y control de cambios: El control de cambios permite mantener la integridad de todos los artefactos que se crean en el proceso, así como de mantener información del proceso evolutivo que han seguido.

Entorno: La finalidad de esta actividad es dar soporte al proyecto con las adecuadas herramientas, procesos y métodos. Brinda una especificación de las herramientas que se van a necesitar en cada momento, así como definir la instancia concreta del proceso que se va a seguir.

2.6 Diseñador de Reportes

Hoy en día, existen diversas librerías y herramientas dedicadas (varias de ellas, Open Source) para la rápida generación de reportes. Una de ellas es la librería JasperReports.

JasperReports:

Es una poderosa herramienta para generar reportes en java, combinada con herramientas para el diseño, facilita y agiliza la generación, la pre-visualización y la impresión de los reportes. JasperReports es una biblioteca que puede ser

embebida (incrustada) en cualquier aplicación Java.^[8]

iReport:

Es un diseñador visual de código libre para JasperReports escrito en Java. Es un programa que ayuda a los usuarios y desarrolladores que usan la librería JasperReports para diseñar reportes visualmente. A través de una interfaz rica y simple de usar, iReport provee las funciones más importantes para crear reportes amenos en poco tiempo. iReport puede ayudar a la gente que no conoce la sintaxis XML para generar reportes de JasperReports.

3 Conclusiones

- La implementación del Sistema de Gestión y Control de Talleres de vehículos para el Centro Integral de Reparación Automotriz Mega-Auto, ha permitido establecer estructuras de organización, enfocadas en las necesidades reales de los usuarios y en la administración eficiente de las actividades y recursos de la Empresa; posibilitando de esta manera, el desarrollo sustentable del negocio haciéndolo aún más rentable.
- La adaptación del sistema informático permite gestionar y controlar los procesos que diariamente se realizan en el centro automotriz, desde la recepción de datos del Cliente, proceso que implica la captura de los datos y condiciones del vehículo; luego la elaboración del presupuesto, la generación de la orden de trabajo, la puesta en marcha de la reparación; y finalmente el registro del vehículo terminado.
- Referente al aprovechamiento máximo de los recursos del taller, se ha logrado acelerar los tiempos de reparación, se ha mejorado el rendimiento e incrementado la productividad; esto ha permitido brindar un mejor servicio a los clientes y dar un trato oportuno al personal que labora en la empresa.
- Las herramientas informáticas y la metodología de desarrollo utilizadas en este aplicativo, han permitido la construcción de software de calidad, orientado a cubrir las necesidades de la empresa, como una herramienta que hará que el trabajo sea más fácil y productivo.

^[8] JasperReports. Recuperado de <http://es.wikipedia.org/wiki/JasperReports>

- El acceso a la información se ve facilitado gracias al diseño de una interfaz amigable en un ambiente automatizado, que permite disponer de información oportuna, correcta e integrada en el momento que se necesita.

4 Recomendaciones

A continuación se enumeran una serie de recomendaciones que permitirán asegurar el correcto funcionamiento del sistema de gestión y control de talleres de vehículos.

- Para dar cumplimiento a los objetivos planteados, es fundamental la participación activa de los usuarios; por lo que, la función que desempeña cada uno de ellos debe estar claramente definida, de tal manera, que se facilite la familiarización con los procesos establecidos en el aplicativo.
- Para facilitar el trabajo en el registro del Conteo Físico, Ajuste de Inventario, Requisición de Materiales o en la elaboración del presupuesto; es recomendable tener previamente cargada la información preliminar (productos, bodegas, tipos de operaciones, tipos de vehículos, trabajos de mano de obra, trabajos externos, marcas y modelos de vehículos).
- Realizada la inspección del vehículo se debe proceder a la elaboración de la proforma, en donde, se debe indicar las actividades o trabajos de mano de obra a realizarse, se deben asignar los recursos correspondientes, así también especificar los trabajos a realizarse de manera subcontratada. El resultado obtenido permitirá especificar el tiempo y costo de la reparación del vehículo.
- La orden de trabajo debe ser generada únicamente previa aceptación de la reparación del vehículo.
- En base a la orden y dependiendo de los trabajos a realizar, el jefe de taller debe encargarse de asignar los técnicos a los trabajos requeridos para la puesta en marcha de la reparación del vehículo.
- Cada trabajo de mano de obra terminado debe ser registrado en el sistema, de tal manera que se tenga control del estado del vehículo; además se debe asignar por cada trabajo terminado el nombre del técnico que lo realizó.

- Una vez terminada la reparación del vehículo, el jefe de taller debe registrar la finalización de la reparación.
- Como todos los presupuestos pendientes del vehículo son reflejados al imprimirse la Orden de Trabajo, es muy importante facturar los presupuestos que ya han sido finalizados, para que no sean incluidos en futuras órdenes de trabajo, o anular aquellos presupuestos que no sean válidos o no se vayan a ejecutar.
- El responsable de bodega y el responsable de laboratorio deben realizar el registro pertinente de las entradas y/o salidas de los productos, de esta forma se podrá tener un control detallado del inventario.
- La efectividad en la generación de roles de pagos dependerá del registro de horas trabajadas, comisiones, adelantos y toda la información pertinente al personal.
- El registro adecuado de la información permitirá tener información veraz, real y oportuna en el momento en que se necesite.
- La coordinación entre el personal administrativo y el personal del taller debe estar enfocado en solucionar o satisfacer los requerimientos de los clientes y tender a un uso óptimo de los recursos de la empresa.

5 Referencias Bibliográficas

- [1] (2012). Eclipse. Recuperado de http://es.wikipedia.org/wiki/Eclipse_%28software%29
- [2] Características de Java como Lenguaje de Programación. Recuperado de <http://www.infor.uva.es/~jmrr/tgp/java/JAVA.html>
- [3] Programando en Java I. Recuperado de <http://leoag.wordpress.com/2010/08/02/programando-en-java-i/>
- [4] (2007). Usando el API de persistencia en aplicaciones de escritorio. Recuperado de <http://www.apuntesdejava.com/2007/06/usando-el-api-de-persistencia-en.html>
- [5] Sabana Mendoza, M. (2006). Php con PostgreSQL8 (1ra edición). Lima Perú: Megabyte
- [6] Joyanes A., L., Zahonero Martínez, I. (2010). Programación en C, C++, JAVA y UML, México D.F.: Mc. Graw Hill

- [7] RUP Ciclo de Vida. Recuperado de http://es.wikipedia.org/wiki/Proceso_Unificado_de_Rational
- [8] JasperReports. Recuperado de <http://es.wikipedia.org/wiki/JasperReports>