

# UNIVERSIDAD TECNICA DE AMBATO

FACULTAD DE INGENIERIA EN SISTEMAS

CARRERA DE INGENIERIA EN SISTEMAS  
COMPUTACIONALES E INFORMÁTICOS

TEMA:

---

SOFTWARE DE CONTROL DE VEHÍCULOS DE LA EMPRESA  
MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO DE  
AMBATO “EMAPA”

---

Trabajo de graduación modalidad pasantía presentado como requisito previo a la  
obtención del Título de Ingeniero en Sistemas Computacionales e Informáticos.

Autor:

Ricardo Fabián Vásquez Mena

Tutor:

Ing. Janio Jadán

Ambato - Ecuador

2007

## **APROBACIÓN DEL TUTOR**

En calidad de Tutor del trabajo de Investigación del tema “SOFTWARE DE CONTROL DE VEHÍCULOS DE LA EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO DE AMBATO EMAPA”, realizado por el señor RICARDO FABIAN VASQUEZ MENA estudiante décimo semestre de la carrera de Sistemas Computacionales e Informáticos, de la Facultad de Ingeniería en Sistemas de la Universidad Técnica de Ambato, considero que dicho informe investigativo reúne los requisitos y méritos suficientes para ser sometidos a la evaluación del tribunal de Grado, que el Honorable Concejo Directivo de la facultad designe para su correspondiente estudio y calificación.

Atentamente:

---

Ing. Janio Jadán

TUTOR

# **AUTORÍA**

El presente trabajo de Investigación con el tema “SOFTWARE DE CONTROL DE VEHÍCULOS DE LA EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO DE AMBATO EMAPA” está enfocado a mejorar el área vehicular de la Empresa, todo cuanto consta en esta investigación es responsabilidad del autor.

---

Sr. Ricardo Fabián Vásquez Mena

C.I 1803735990

## **DEDICATORIA**

El presente trabajo está dedicado a todas las personas que me han apoyado en mi carrera Universitaria, a mis padres y hermanas que siempre estuvieron para apoyarme durante mis estudios, a todos mis compañeros y amigos con los que he aprendido y vivido muchos momentos agradables, a todos mis maestros que a mas de el conocimientos científicos inculcaron en mi valores y enseñanzas que me servirán para llegar a ser un gran profesional.

## **AGRADECIMIENTO**

Agradezco a Dios que día a día guía mis actos, a mi familia que han demostrado apoyo incondicional y gracias a sus consejos y sacrificio me permitieron lograr este tan anhelado título, y a mis amigos y maestros que me han acompañado durante todos estos años.

## **RESUMEN EJECUTIVO**

La Empresa Municipal de Agua Potable y Alcantarillado de la ciudad de Ambato ha visto la necesidad de optimizar el proceso del manejo y control de suministros de los vehículos pertenecientes a la empresa, considerando que esta área realiza sus actividades en forma manual, esto sin duda conlleva a una serie de desventajas de recursos humanos, materiales, información errónea lo que conlleva a la pérdida de recursos económicos.

Por esto se vio la necesidad de automatizar los procesos realizados dentro del área vehicular para que, con ello, se produzcan ahorros económicos y principalmente mejorar el servicio a la comunidad ambateña, los cuales mejorarán sustancialmente con la creación del sistema de control vehicular.

## **INTRODUCCIÓN**

En este mundo globalizado y cambiante cada día crecen las necesidades de las empresas por mejorar y optimizar sus procesos en todas las áreas, para esto disponemos también de una tecnología que evoluciona rápidamente y que nos permite automatizar las operaciones de la empresa, lo cual nos permite mejorar considerablemente el desarrollo de los trabajos y administración de todas las áreas, esto a su vez produce ganancias y ahorros económicos dentro de la organización.

Tal es el caso de la Empresa Municipal de Agua Potable y Alcantarillado de la ciudad de Ambato que ya cuenta con un sistema integrado que en un futuro no tan lejano abarcará todas las áreas y departamentos de la Empresa, por esto este sistema Vehicular se unirá con el sistema integrado para así brindar un mejor servicio a la empresa y a la comunidad.

## INDICE GENERAL

### CAPITULO I

#### GENERALIDADES

1.1 TEMA DE INVESTIGACIÓN.....	1
1.2 ANTECEDENTES DE LA INSTITUCIÓN.....	1
1.2.1 NOMBRE.....	1
1.2.2 MISIÓN.....	1
1.2.3 VISIÓN.....	1
1.3 PLANTEAMIENTO DEL PROBLEMA.....	2
1.4 JUSTIFICACIÓN.....	4
1.5 OBJETIVOS.....	5
1.5.1 OBJETIVO GENERAL.....	5
1.5.2 OBJETIVOS ESPECÍFICOS.....	6

### CAPITULO II

#### MARCO TEÓRICO

2.1 ANTECEDENTES INVESTIGATIVOS.....	7
2.2 FUNDAMENTACIÓN LEGAL.....	7
2.3 CATEGORÍAS FUNDAMENTALES.....	9
• La Automatización.....	9
- Principios de los Sistemas Automatizados.....	9
- Medición.....	9
- Evaluación.....	10
• Sistemas.....	10
- Características de los sistemas.....	11
- Tipos de Sistemas.....	12

• Sistemas de control.....	12
- Componentes Básicos de un sistema de control.....	13
• Control.....	14
- Fundamentos de control.....	14
- Definición de control.....	15
- Factores de control.....	15
- Importancias del control.....	16
- Tipos de control.....	16
• Recursos.....	17
- Definición de Recursos.....	17
• Vehículos.....	17
- Área vehicular EMAPA.....	18
• SQL Server.....	18
- Administración de SQL Server.....	19
- Los Usuarios de SQL Server.....	20
- El administrador de SQL Server.....	20
• ASP.NET.....	21
- Ventajas de ASP.NET en relación al ASP Clásico.....	22
- Mejoras de ASP.NET.....	23
• Arquitectura Cliente-Servidor.....	23
- Elementos de la arquitectura Cliente-Servidor.....	24
- ¿Qué es un cliente? .....	24
- ¿Qué es un Servidor? .....	24
- Tipos de cliente.....	25
- Tipos de servidor.....	25
- Componentes básicos de la arquitectura Cliente-Servidor.....	26
- Características del modelo Cliente-Servidor.....	26
- Ventajas de la arquitectura Cliente-Servidor.....	27
• Intranet.....	27
- ¿Cómo funciona una Intranet?.....	28
- Seguridad de una Intranet.....	29
2.4 HIPÓTESIS.....	29

2.5 DETERMINACIÓN DE VARIABLES.....	30
2.4.1 VARIABLE INDEPENDIENTE.....	30
2.4.2 VARIABLE DEPENDIENTE.....	30

### **CAPITULO III**

#### METODOLOGÍA

3.1 MODALIDAD DE INVESTIGACIÓN.....	31
3.2 TIPO DE INVESTIGACIÓN.....	31
3.3 POBLACIÓN Y MUESTRA.....	32
3.4 TÉCNICAS E INSTRUMENTOS DE INVESTIGACIÓN.....	32
3.5 RECOLECCIÓN DE INFORMACIÓN.....	32
3.6 PROCESAMIENTO DE LA INFORMACIÓN.....	33

### **CAPITULO IV**

#### ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

• Análisis Previo.....	34
• Análisis de la documentación.....	35

### **CAPITULO V**

#### COLCLUSIONES Y RECOMENDACIONES

5.1 COLCLUSIONES.....	38
5.2 RECOMENDACIONES.....	38

## CAPITULO VI

### ANÁLISIS DEL SISTEMA

6.1 DATOS INFORMATIVOS.....	40
6.2 ANTECEDENTES DE LA PROPUESTA.....	41
6.3 ANÁLISIS DE RIESGOS.....	41
6.4 ANÁLISIS DE FACTIBILIDADES.....	45
6.4.1 FACTIBILIDAD OPERACIONAL.....	45
6.4.2 FACTIBILIDAD TÉCNICA.....	45
6.4.3 FACTIBILIDAD ECONÓMICA.....	47
6.5 INGENIERÍA DE SOFTWARE.....	48
6.5.1 FUNCIONES PRINCIPALES DEL SISTEMA.....	48
6.5.2 ASPECTOS DE FUNCIONAMIENTO.....	49
6.5.3 RESTRICCIONES TÉCNICAS Y DE FUNCIONAMIENTO.....	50
6.5.4 ENTRADAS DE USUARIO.....	50
6.5.5 SALIDAS DE USUARIO.....	51
6.6 ANÁLISIS ESTRUCTURADO.....	52
6.6.1 DIAGRAMAS DE FLUJO DE DATOS.....	52
6.6.2 DICCIONARIO DE DATOS.....	57
• Nivel 0.....	57
• Nivel 1.....	71
• Nivel 2 (Edición de datos).....	75
• Nivel2 (Mantenimiento).....	80
• Nivel3 (Combustible).....	84

## CAPITULO VII

### DISEÑO DEL SISTEMA

7.1 DISEÑO DE LA BASE DE DATOS.....	90
7.1.1 DISEÑO FÍSICO.....	90

7.1.2 DISEÑO LÓGICO.....	91
7.2 DISEÑO DE ENTRADAS Y SALIDAS.....	92
7.2.1 DISEÑO DE ENTRADAS.....	92
7.2.2 DISEÑO DE SALIDAS.....	93
7.3 MAPA DEL SITIO Web DE EMAPA.....	94
7.4 MAPA CONCEPTUAL DEL SITIO Web DE EMAPA.....	95
7.5 DIAGRAMA FÍSICO DE LA RED.....	96

## **CAPITULO VIII**

### **CODIFICACIÓN Y PRUEBAS**

8.1 CODIGO SQL DE LA BASE DE DATOS.....	97
8.1.1 CREACIÓN DE TABLAS.....	97
8.1.2 CREACIÓN DE VISTAS .....	101
8.1.4 CREACIÓN DE STORE PROCEDURES.....	102
8.2 CODIGO DE LA APLICACIÓN.....	106
8.2.1 CODIGO DE CONFIGURACIÓN.....	106
8.2.2 CODIGO DE PÁGINAS.....	107
8.3 PRUEBAS.....	142
8.3.1 FUNCIONES JAVA SCRIPT PARA EL CONTROL.....	142

### **BIBLIOGRAFIA**

### **ANEXOS**

### **MANUALES**

## INDICE DE TABLAS

Tabla 1 (Identificación de riegos).....	42
Tabla 2 (Posibles impactos) .....	42
Tabla 3 (Posibilidad de ocurrencia) .....	43
Tabla 4 (Evaluación de riesgos) .....	44
Tabla 5 (Comparación de Hardware) .....	46
Tabla 6 (Comparación de Software) .....	46
Tabla 7 (Posible inversión en el sistema) .....	48
Tabla 8 (Diccionario de datos – Entidades).....	57
Tabla 9 (Diccionario de datos – Procesos Nivel 0).....	58
Tabla 10 (Diccionario de datos – Estructuras Nivel 0).....	58
Tabla 11 (Diccionario de datos – Elementos Dato Nivel 0).....	63
Tabla 12 (Diccionario de datos – Procesos Nivel 1).....	71
Tabla 13 (Diccionario de datos – Estructuras Nivel 1).....	73
Tabla 14 (Diccionario de datos – Elementos dato Nivel 1).....	73
Tabla 15 (Diccionario de datos – Procesos Nivel 2 Edición de datos).....	75
Tabla 16 (Diccionario de datos – Almacenes Nivel 2 Edición de datos)....	76
Tabla 17 (Diccionario de datos – Estructuras Nivel 2 Edición de datos)....	77
Tabla 18 (Diccionario de datos-Elementos Dato Nivel 2 Edición de datos)	80
Tabla 19 (Diccionario de datos – Procesos Nivel 2 Combustible).....	80
Tabla 20 (Diccionario de datos – Almacenes Nivel 2 Combustible).....	81
Tabla 21 (Diccionario de datos – Estructuras Nivel 2 Combustible).....	82
Tabla 22 (Diccionario de datos – Procesos Nivel 2 Mantenimiento).....	84
Tabla 23 (Diccionario de datos – Almacenes Nivel 2 Mantenimiento).....	85
Tabla 24 (Diccionario de datos – Estructuras Nivel 2 Mantenimiento).....	87

## **CAPITULO I**

### **GENERALIDADES**

#### **1.1 TEMA DE INVESTIGACIÓN**

“SOFTWARE DE CONTROL DE VEHÍCULOS DE LA EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO DE AMBATO EMAPA”

#### **1.2 ANTECEDENTES DE LA INSTITUCIÓN**

##### **1.2.1 Nombre**

Empresa Municipal de Agua Potable y Alcantarillado de Ambato

##### **1.2.2 Misión**

La Empresa Municipal de Agua Potable y Alcantarillado de Ambato, EMAPA, tiene como finalidad la dotación, prestación, mantenimiento, comercialización, control, regulación y desarrollo de los servicios de agua potable y alcantarillado en la ciudad de Ambato y sus parroquias rurales mirando en interés social y sin ánimo de lucro.

##### **1.2.3 Visión**

Durante los próximos tres años la EMAPA será la entidad de entrega de servicios de agua potable y alcantarillado más importante y sólida de la provincia, de reconocido prestigio, confianza y credibilidad como empresa del Ilustre Municipio de Ambato, dando resultado a sus crecientes niveles

de productividad, administración transparente, alto nivel de tecnificación, así como, por sus activos totales y patrimonio.

### **1.3 PLANTEAMIENTO DEL PROBLEMA**

#### **Contextualización**

A Medida que el mundo y el ser humano evolucionan, crecen las necesidades de manejar con mayor y mejor precisión los procesos integrados dentro de una empresa, tomando en consideración el aspecto económico de la organización, es por esto que la automatización de procesos ha copado la mayoría de los aspectos dentro de las empresas e instituciones ya que facilita el trabajo diario.

Es conocido por todos que en este mundo globalizado, el que no está en la era tecnológica poco a poco irá perdiendo competencia frente a las grandes corporaciones y empresas del mundo.

En nuestro país se han hecho muchos esfuerzos por contar sistemas que automaticen todos los procesos especialmente en las empresas privadas pero lamentablemente estos esfuerzos no son suficientes y se están perdiendo muchos recursos por este problema, más aún en las empresas públicas en las cuales cuenta mucho el factor económico que en el caso del Ecuador es poco optimista, muchas empresas han quedado marginadas total o parcialmente del mundo, de la tecnología y de la información.

El ámbito provincial no cambia mucho, ya que el sistema en el que trabaja el gobierno Ecuatoriano es centralizado a las provincias grandes y de mayor población, este no es el caso de la provincia de Tungurahua la misma que carece de suficientes recursos para el perfecto control de los procesos en las empresas.

Se puede decir que en las empresas investigadas no se cuenta con algún sistema que automatice todo lo que corresponde a los automotores de una organización, en la mayoría de instituciones este control se lo lleva a mano o controlado con archivos planos u hojas electrónicas, esto da lugar a múltiples pérdidas económicas y de tiempo dentro de esta área.

En el caso de la Empresa municipal de agua potable y alcantarillado de Ambato “EMAPA” no se cuenta con ningún sistema automatizado completo e íntegro que controle el área vehicular, al momento este aspecto es controlado mediante hojas electrónicas realizadas en EXCEL del paquete de Microsoft Office 2000 y en archivos manuales, esto tiene grandes inconvenientes de preescisión, manejo y comprensión, además no proporciona datos automáticos y exactos sobre los suministros y servicios que corresponden a los automotores, es decir, que la empresa no cuenta en el momento con ningún sistema que automatice esta área de la organización.

Esta falta de un sistema completo e integral para vehículos y maquinaria es debido a la inexistencia de procesos automatizados y la poca de atención que se le presta a esta área, lo cual ocasiona un mal funcionamiento del servicio vehicular de la empresa lo que se refleja en grandes pérdidas económicas, retrasos y daños en los vehículos los cuales también provocan el deteniimiento de los automotores.

De continuar este problema se llegará a que se pierdan en suministros grandes cantidades de dinero, se darían de baja gran cantidad de automotores que bien podrían continuar en servicio y a su vez sin servicio de transporte se llegará a la total insatisfacción de la ciudadanía lo que provocaría gran cantidad de denuncias en contra de la institución.

## **Formulación del Problema**

¿Qué impacto tiene la realización de un software de control en el área de de administración de vehículos para el mejoramiento del servicio?

## **Delimitación**

El presente trabajo de investigación se realizará en la Empresa Municipal de Agua Potable y Alcantarillado de la Ciudad de Ambato dentro del Departamento Administrativo en el área de Vehículos en el periodo comprendido entre Abril del 2006 y Septiembre del 2006. Se trabajará bajo Windows XP y Windows 2000 Server utilizando las herramientas de desarrollo C# de la misma que se trabajará bajo ASP.NET, así mismo se trabajará bajo la Base de Datos SQL Server 2000.

## **1.4 JUSTIFICACIÓN**

La empresa Municipal de Agua Potable y Alcantarillado de Ambato “EMAPA” buscando siempre el mejoramiento en la calidad de servicios para sus usuarios desea implementar un sistema de control, lo cual es de suma importancia porque permitirá reducir los costos en el área, con la finalidad de lograr satisfacer los requerimientos de sus usuarios.

En el Software que se desarrollará se tomarán en cuenta nuevas técnicas y procesos que servirán para controlar de manera eficiente y eficaz los vehículos de la empresa. Se crearán módulos inteligentes que ayuden al departamento a determinar el tiempo y la cantidad justa de distribución de los suministros a los vehículos y control de los servicios que necesitarán los automotores para su mantenimiento, poniendo de esta manera en práctica los conocimientos adquiridos durante la carrera.

De este Proyecto se beneficiarán, en primer lugar el Departamento administrativo de la “EMAPA” que tendrá mayor control de sus procesos, así como también el área vehicular a la cual se le acorta el tiempo de realización de su trabajo, en sí la empresa y el estado ya que ahorrarán muchos recursos los cuales se pueden invertir en nuevas y mejores actividades para la propia organización.

La ciudadanía también se verá beneficiada con este proyecto ya que tendrá un mejor servicio y verá cumplidas sus expectativas porque con vehículos y maquinaria en buenas condiciones se pueden realizar obras en menor tiempo.

La realización de la presente investigación se justifica porque se cuenta con facilidades de acceso a la Institución para obtener la información que permitirá establecer el problema a investigarse, además se cuenta con suficientes recursos económicos, materiales y humanos ya que se tendrá el apoyo tanto de la Gerencia, el Departamento Administrativo y el Departamento de Sistemas de la Empresa Municipal de Agua Potable y Alcantarillado de Ambato “EMAPA”, además se cuentan con los recursos necesarios para incursionar con éxito en las actividades previstas porque se tendrá la posibilidad de obtener la suficiente bibliografía que permita explicar científicamente el tema y problema de investigación .

Por lo tanto el reto consiste en cumplir con éxito los objetivos planteados así como cubrir todas y cada una de las necesidades tanto de la empresa como de la comunidad.

## **1.5 OBJETIVOS**

### **1.5.1 Objetivo General**

Optimizar el área de control y administración de vehículos a través de un sistema para el mejoramiento del servicio vehicular de la EMAPA.

### **1.5.2 Objetivos específicos**

- Conocer, Analizar y Establecer los recursos y parámetros necesarios para realizar el software de control vehicular dentro de la empresa.
- Optimizar y Automatizar el control de consumo de combustibles y lubricantes necesarios para los vehículos de la empresa.
- Optimizar y Automatizar el control de consumo de partes y piezas necesarios para los vehículos de la empresa.
- Controlar el aumento, rendimiento y dada de baja de los automotores pertenecientes a la empresa.
- Diseñar e implementar el sistema de control de vehículos y maquinaria en la Empresa Municipal de Agua Potable y Alcantarillado de Ambato “EMAPA”.

## **CAPITULO II**

### **MARCO TEÓRICO**

#### **2.1 ANTECEDENTES INVESTIGATIVOS**

Revisados los archivos de la Facultad de Ingeniería en Sistemas no se encontró tesis alguna con un tema similar o relacionado al presente trabajo.

#### **2.2 FUNDAMENTACIÓN LEGAL**

##### **Institución de la Empresa**

Constituyese con domicilio en la ciudad de Ambato y por tiempo indefinido la Empresa Municipal de Agua Potable y Alcantarillado de Ambato, EMAPA AMBATO con personería Jurídica, patrimonio, recursos propios y capacidad para ejercer derechos y contraer obligaciones

##### **Tipo de empresa**

La Empresa Municipal de Agua Potable y Alcantarillado de Ambato EMAPA es una empresa pública regida por la Ley de Régimen Municipal, por la presente Ordenanza; por sus propias reglamentaciones generales y otras Ordenanzas del Ilustre Municipio de Ambato que tenga relación con el servicio que presta entre las que se pueden enumerar:

- Acta de constitución de la Empresa, fines y objetivos.
- Ley de régimen municipal.
- Ley de servicio civil y carrera administrativa

- Ley orgánica de administración financiera y control. (El sistema de presupuestos y contabilidad gubernamental)
- Ley de bienes del sector público
- Ley del control del gasto público.
- Código de trabajo

Los reglamentos internos sobre los que se basa la Empresa son:

- Reglamentos de representación y residencia.
- Reglamentos de procedimientos de proyectos hidráulicos sanitarios.
- Reglamento del funcionamiento del directorio.
- Reglamento de remate y baja de bienes.
- Reglamento de interconexiones.
- Reglamento de adquisiciones.
- Reglamento de fondo de caja chica

### **Políticas**

Respetar, cumplir y hacer cumplir las leyes, ordenanzas, reglamentos y disposiciones de la Institución.

## **2.3 CATEGORÍAS FUNDAMENTALES**

### **LA AUTOMATIZACIÓN**

En un mundo globalizado y digitalizado que día tras día genera nueva tecnología cabe hablar sobre la Automatización que no es sino una amplia variedad de sistemas y procesos que operan con mínima o sin intervención del ser humano. En los más modernos sistemas de automatización, el control de las máquinas es realizado por ellas mismas gracias a sensores de control que le permiten percibir cambios en sus alrededores de ciertas condiciones tales como temperatura, volumen y fluidez de la corriente eléctrica y otros, sensores los cuales le permiten a la máquina realizar los ajustes necesarios para poder compensar estos cambios. Y una gran mayoría de las operaciones industriales de hoy son realizadas por enormes máquinas de este tipo.

#### **Los Principios de los Sistemas Automatizados.**

Un sistema automatizado ajusta sus operaciones en respuesta a cambios en las condiciones externas en tres etapas: medición, evaluación y control

#### **Medición**

Para que un sistema automatizado reaccione ante los cambios en su alrededor debe estar apto para medir aquellos cambios físicos. Por ejemplo, si la fluidez de la corriente eléctrica de una máquina cambia, una medición debe ser llevada a cabo para determinar cuál ha sido este cambio. Estas medidas realizadas suministran al sistema de ingreso de corriente eléctrica de la máquina la información necesaria para poder realizar un control.

## **Evaluación**

La información obtenida gracias a la medición es evaluada para así poder determinar si una acción debe ser llevada a cabo o no. Por ejemplo, si un vehículo empieza a tener varios problemas mecánicos, se deberá evaluar la magnitud de los daños y desperfectos mecánicos para así poder tomar la mejor decisión en cuanto a este vehículo, si se lo repara o si se lo da de baja.

## **SISTEMAS**

Luego de haber hablado de Automatización es necesario hablar sobre los sistemas.

Generalmente los sistemas en sí tienen varios conceptos, entre los cuales podemos anotar los siguientes:

- Un sistema es una combinación de componentes que actúan juntos y realizan un objetivo determinado. Un sistema no necesariamente es físico. El concepto de sistema se aplica a fenómenos abstractos y dinámicos, tales como los que se encuentran en la economía. Por tanto, la palabra sistema debe interpretarse como una implicación de sistemas físicos, biológicos, económicos y similares.
- Un sistema es un todo organizado. Este concepto tiene dos usos muy diferenciados, que se refieren respectivamente a los sistemas de conceptos y a los objetos reales más o menos complejos y dotados de organización.
- Un conjunto de elementos dinámicamente relacionados que forman una actividad para alcanzar un objetivo operando sobre datos/energía/materia para proveer información/energía/materia. Según LEWIS Paul, en su libro *Sistemas de control en ingeniería*.

## Características de los sistemas

Entre las características principales de los sistemas podemos citar las siguientes:

- Sistema es un todo organizado y complejo.
- Es un conjunto de objetos unidos por alguna forma de interacción o interdependencia.
- Según Bertalanffy, sistema es un conjunto de unidades recíprocamente relacionadas. De ahí se deducen dos características: propósito (u objetivo) y globalismo (o totalidad).
- Propósito u objetivo: todo sistema tiene uno o algunos propósitos los cuales tratan siempre de alcanzar un objetivo.
- Globalismo o totalidad: un cambio en una de las unidades del sistema producirá cambios en las otras. Entonces hay una relación de causa/efecto. De estos cambios, se derivan dos fenómenos: entropía y homeostasia.
- Entropía: es la tendencia de los sistemas a desgastarse, a desintegrarse. La entropía aumenta con el correr del tiempo.
- Homeostasia: es el equilibrio dinámico entre las partes del sistema. Los sistemas tienen una tendencia a adaptarse con el fin de alcanzar un equilibrio interno frente a los cambios externos del entorno.

## **Tipos de sistemas**

En cuanto a su constitución, pueden ser físicos o abstractos:

- **Sistemas físicos o concretos:** compuestos por equipos, maquinaria, objetos y cosas reales. Es lo que se conoce como Hardware.
- **Sistemas abstractos:** compuestos por conceptos, planes, hipótesis e ideas. Muchas veces solo existen en el pensamiento de las personas. Es lo que se conoce como Software.

En cuanto a su naturaleza, pueden ser cerrados o abiertos:

- **Sistemas cerrados:** no presentan intercambio con el medio ambiente que los rodea, son herméticos a cualquier influencia ambiental. No reciben ningún recurso externo.
- **Sistemas abiertos:** presentan intercambio con el ambiente, a través de entradas y salidas. Intercambian energía y materia con el ambiente. (<http://es.wikipedia.org/wiki/Sistema>)

## **SISTEMAS DE CONTROL**

Luego de hablar y haber entendido a los sistemas cabe hablar sobre los sistemas de control, ya que estos han desempeñado una función vital en el avance de la ingeniería y la ciencia. Además se ha vuelto una parte importante e integral de los procesos modernos e industriales de manufactura. Por ejemplo el control automático es esencial en las industrias de manufactura, la industria aeroespacial, la industria automotriz, entre otras.

La esencia de los sistemas de control es un esfuerzo para avanzar de forma continua nuestra comprensión de las metodologías que proporcionan la capacidad de controlar sistemas.

Teniendo en cuenta esto podemos decir que sistemas de control se define como la forma de aplicar las mejores estrategias para buscar y alcanzar los objetivos planteados con éxito.

Es por esto que todos los ingenieros y científicos deben tener conocimientos bien fundados en el campo de los sistemas de control ya que estos permiten obtener un desempeño óptimo de los sistemas dinámicos, Además mejora la productividad al aligerar las cargas de trabajo.

### **Componentes básicos de un sistema de control.**

Los componentes básicos de un sistema de control se pueden describir mediante:

- Objetivos de control
- Componentes del sistema de control.
- Resultados o salidas.

La relación básica entre estos tres componentes se ilustra en la figura 1. En términos más técnicos, los objetivos se pueden identificar como entradas y los resultados también se pueden llamar salidas. En general, el objetivo de un sistema de control es controlar las salidas en alguna forma preescrita mediante las entradas a través de los elementos del sistema de control. Según LEWIS Paul, en su libro Sistemas de control en ingeniería.



Figura 1

Además debemos mencionar que podemos clasificar a los sistemas de control en realimentados y no realimentados:

## **CONTROL**

### **Fundamentos de control**

Para empezar a hablar de control tenemos que empezar desde el término controlar.

Con el avance de la tecnología las empresas se vuelven más complejas y más modernas, el problema del control aumenta. Por tanto se debe tener un conocimiento bien fundado sobre los puntos del control.

Con esto en mente podemos decir que controlar es permitir que las cosas y eventos ocurran tal como se planificaron, esto nos lleva a una unión significativa de planeación y control.

La ley de Murphy establece que los gerentes deben controlar continuamente, es decir, verificar que las actividades y problemas organizacionales se están llevando a cabo de la forma planteada. También establece que lo que está susceptible de fallar fallará. Esta ley recuerda a los gerentes que deben estar alerta a posibles problemas, puesto que, aunque un sistema parezca funcionar bien, podría estar erosionando bajo la superficie.

## **Definición de control**

Control es el proceso que los gerentes siguen para controlar. Según Robert Mockler, control es:

“Un esfuerzo sistemático de la administración de negocios de comparar el desempeño con los estándares, planes u objetivos predeterminados, con el fin de identificar si el rendimiento cumple esos estándares y aplicar la acción correctiva que se requiera para garantizar que los recursos humanos y otros recursos corporativos se utilicen de la forma más efectiva y eficaz para alcanzar los objetivos corporativos”

También se puede decir que control es el proceso para asegurar que las actividades reales se ajusten a las actividades planificadas. Permite mantener a la organización o sistema en buen camino.

## **Factores Del Control**

Existen cuatro factores que deben ser considerados al aplicar el proceso de control.

- Cantidad
- Tiempo
- Costo
- Calidad

Los tres primeros son de carácter cuantitativo y el último es eminentemente cualitativo. El factor cantidad se aplica a actividades en la que el volumen es importante. A través del factor tiempo se controlan las fechas programadas. El costo es utilizado como un indicador de la eficiencia administrativa, ya que por medio de él se determinan las erogaciones de ciertas actividades. La calidad se refiere a las especificaciones que deben reunir un cierto producto o ciertas funciones de la empresa.

## **Importancias Del Control**

- Establece medidas para corregir las actividades, de tal forma que se alcancen los planes exitosamente.
- Se aplica a todo: a las cosas, a las personas y a los actos.
- Determina y analiza rápidamente las causas que pueden originar desviaciones para que no vuelvan a presentarse en el futuro.
- Localiza los sectores responsables de la administración, desde el momento en que se establecen medidas correctivas.
- Reduce costos y ahorra tiempo al evitar errores.
- Su aplicación incide directamente en la racionalización de la administración y consecuentemente, en el logro de la productividad de todos los recursos de la empresa.

## **Tipos de control**

Existen tres tipos de control.

- Control previsorio
- Control simultáneo
- Control de retroalimentación.

## RECURSOS

### DEFINICIÓN DE RECURSOS:

La organización, para lograr sus objetivos requiere de una serie de recursos, estos son elementos que, administrados correctamente, le permitirán o le facilitarán alcanzar sus objetivos. Existen tres tipos de recursos:

- **RECURSOS MATERIALES:** Aquí quedan comprendidos el dinero, las instalaciones físicas, la maquinaria, los muebles, las materias primas, etc.
- **RECURSOS TÉCNICOS:** Bajo este rubro se listan los sistemas, procedimientos, organigramas, instructivos, etc.
- **TALENTO HUMANO:** No solo el esfuerzo o la actividad humana quedan comprendidos en este grupo, sino también otros factores que dan diversas modalidades a esa actividad: conocimientos, experiencias, motivación, intereses vocacionales, aptitudes, actitudes, habilidades, potencialidades, salud, etc.

## VEHÍCULOS

Existen varias definiciones para los vehículos entre las cuales tenemos:

- Un **vehículo** es un medio que permite el traslado de un lugar a otro. Respecto de personas u objetos se hace referencia a vehículos de transporte como por ejemplo tren, automóvil, camión, carro, barco, avión, bicicleta y motocicleta, entre otros. (<http://es.wikipedia.org/wiki>).

- Vehículo a motor que sirve, normalmente, para el transporte de personas o cosas o de ambas a la vez, o para la tracción de otros vehículos con aquel fin. Se excluyen de esta definición los vehículos especiales. (<http://elephas.worldonline.es/valen-ar/glosario.html>)

### **Área vehicular EMAPA**

Esta sección se encuentra bajo el área administrativa de la empresa, se encarga del control y administración de todos los vehículos que son propiedad de la organización, tomando en cuenta que dentro del área vehicular se toma en cuenta como vehículos a automóviles, camionetas, camiones pequeños y una mini escavadora, además controla los recursos y la utilización de los automotores. Es así que también esta área, aunque no se encarga de la contratación y disposición del personal, se encarga de la asignación de personal “conductores” a cada vehículo; además de la asignación de los vehículos a cada Departamento de la EMAPA.

### **SQL SERVER**

SQL Server es una herramienta de Base de Datos que cuenta con múltiples funciones para brindar al usuario entre estas tenemos la posibilidad de manejar:

- Store Procedures, o procedimientos con líneas de código almacenados en el servidor,
- Triggers, de diferente tipo como son los denominados “antes de”, “después de”, y los “combinados”, a más de esto se puede combinar los tipos con las operaciones sobre las tablas como son: Update (Actualizar) y Delete (Borrar), etc.

- Vistas, en las cuales nos permite crear vistas ya sea por código o eligiendo directamente los campos y sus restricciones y condiciones, para esto cuenta con un visor de creación de vistas que permite realizar estas operaciones.
- Diagramas: SQL Server cuenta con un visor para la creación de diagramas de la base de datos con lo cual podemos observar un modelo físico y lógico de la base de datos escogida o parte de ella.
- Usuarios: SQL Server permite el manejo de usuarios para el acceso a la base de datos con lo cual mejora la seguridad para la información.

A más de esto cuenta con muchas funciones más como la de roles, permisos, etc.

### **Administración de SQL Server**

Para la administración de SQL Server es necesario tener presente todo momento la seguridad del servidor, por esto es muy importante la creación y control de los usuarios.

Para ello es necesario crear primeramente una cuenta para el dominio de SQL Server la cual deberá tener los siguientes atributos:

- Miembro del grupo local administradores
- Activado el atributo “La contraseña no caduca nunca”
- Permitido el inicio de sesión a todas las horas
- Permitido iniciar sesión como servicio.

## **Los Usuarios de SQL Server**

Los usuarios de SQL Server pueden ser de dos tipos desde el punto de vista de seguridad:

- Los Usuarios de Windows: son creados en el dominio
- Usuarios propios de SQL Server: para los cuales es necesario configurar el Servidor SQL en modo autenticación mixta.

Se aconseja utilizar usuarios de Windows ya que es la única manera de asignar directivas de seguridad a las cuentas de usuario como son; caducidades de contraseñas, monitorear quien trató ingresar ilícitamente a nuestra base de datos, etc.

La facilidad que presenta la creación de este tipo de usuarios es que se los puede administrar desde el propio SQL Server.

## **El administrador de SQL Server**

El administrador es necesario para SQL Server para que el pueda manejar la seguridad de la Base de datos.

SQL Server permite fácilmente definir uno o varios administradores de la Base de Datos para que ayuden al buen funcionamiento del servidor, pero por razones de seguridad debería haber un usuario administrador que tenga ciertas características administrativas o a su vez distribuir las características a cada usuario administrador. Estas características son:

- Poder crear y modificar cuentas de usuarios

- Tener acceso a las claves del registro del servidor donde este instalado el SQL Server,
- Tener acceso total al sistema de archivos del equipo donde está instalado el SQL Server
- Tener privilegios para poder iniciar y detener los servicios SQL Server, SQL Server Agent, Servicio Microsoft Search
- Poder acceder a los registros de equipos clientes (si se van a utilizar controladores ODBC)
- Poder leer y buscar a través del visor de sucesos del servidor
- Poder utilizar el monitor del sistema y crear registros de log de cualquier tipo de contador
- Instalar y reparar el software del SQL Server
- Realizar copias de seguridad en dispositivos externos

## **ASP.NET**

ASP.NET Se lo puede definir como un conjunto de herramientas y tecnologías de desarrollo de aplicaciones Web, forma parte de la plataforma .NET de Microsoft Visual Studio.

Este conjunto de tecnologías son las sucesoras de la tecnología ASP (Active Server Pages).

Como sabemos la programación Web es una mezcla de varios lenguajes de etiquetas, pequeños programas script y plataformas de servidor, así apareció

ASP.NET con el objetivo de resolver problemas y limitaciones presentadas por ASP clásico, es una plataforma unificada de desarrollo WEB con el objetivo de brindar los servicios necesarios para realizar aplicaciones WEB a nivel empresarial, es por esto que se debe contar con esta herramienta ya que este tipo de aplicaciones cuenta con varios puntos de vista nuevos como son:

- Aplicaciones cliente/servidor: Que es la arquitectura con la que tendrán que trabajar las nuevas aplicaciones
- Aplicaciones que utilizan el navegador: cuentan con una interfaz Web rica y muy útil, puede estar integrada por varias tecnologías tales como HTML, XHTML, scripting, etc. Solo limitado por la versión del navegador que este utilizando.

### **Ventajas de ASP.NET en Relación a ASP Clásico**

- La primera ventaja que podemos mencionar es que ASP solo se podía escribir con el lenguaje VBScript mientras que ASP.NET puede ser escrito en cualquier lenguaje soportado por el .net Framework, es decir VB.NET, C# y Jscript.NET
- ASP.NET es un lenguaje totalmente orientado a objetos
- ASP necesita muchas líneas de código para resolver problemas pequeños mientras que ASP.NET disminuye sustancialmente esta cantidad de código.
- La organización: ASP se lo puede considerar como desorganizado ya que no hay distinción formal entre el contenido de una página y su comportamiento, ASP.NET impone un cierto orden sobre el modelo de programación estándar ASP

## **Mejoras De Asp.Net**

Sin duda. ASP.NET es mucho mejor que ASP clásico y que cuenta con diversas mejoras entre las que se destacan:

- **Rendimiento:** Se compila desde el código nativo lo que permite un mejor rendimiento y permite el almacenamiento del caché en el servidor.
- **Rapidez en la programación:** Se puede decir que con pocas líneas de código se puede realizar muchas acciones, lo cual nos permite ahorrar mucho tiempo.
- **Servicios Web:** Contiene herramientas para compartir información entre distintos sitios.
- **Seguridad:** Tiene diversas herramientas que mejoran la seguridad de nuestras aplicaciones.

## **ARQUITECTURA CLIENTE SERVIDOR**

La arquitectura cliente servidor constituye una forma de dividir las responsabilidades de un sistema de información separando la interfaz de usuario (cliente) de la gestión de información (Servidor).

En esta arquitectura una de los componentes pide servicios a otro los cuales pueden colaborar entre sí para dar a los usuarios toda la información requerida.

La arquitectura cliente Servidor debe estar conformada por un mínimo de dos computadoras las cuales deberán tener un acceso transparente a las

aplicaciones, datos, servicios de cómputo o cualquier otro recurso perteneciente al grupo de trabajo, con esto se puede llegar a tener un ambiente “distribuido” con lo cual un usuario puede acceder a la información de varios servidores que cooperan entre sí.

En este tipo de arquitectura la capacidad de proceso se reparte entre los clientes y los servidores y presenta una centralización de la gestión de la información y la separación de responsabilidades lo que facilita el diseño del sistema.

La arquitectura cliente servidor sustituye la arquitectura monolítica en la que no hay distribución tanto a nivel físico como a nivel lógico.

### **Elementos de la arquitectura Cliente Servidor**

Los elementos principales de la arquitectura cliente servidor son justamente el elemento CLIENTE y el elemento SERVIDOR

#### **¿Qué es un cliente?**

Es el que solicita requerimientos o servicios los cuales pueden ser uno o varios, estos se pueden transportar por redes ya sea LAN, MAN o WAN. La ubicación de la información es totalmente transparente para el cliente.

#### **¿Qué es un Servidor?**

Es el elemento dedicado a responder los requerimientos del cliente tales como impresión acceso a base de datos entre otros. Los servidores pueden estar conectados a los clientes a través de redes LAN, MAN o WAN.

## **Tipos de cliente**

- Cliente “Flaco”: la cual presenta un servidor rápidamente saturado y una gran circulación de datos de interfase en la red.
- Cliente “Gordo”: en la cual casi todo el trabajo se realiza en el cliente y una gran circulación de datos inútiles en a red

## **Tipos de Servidor**

Entre los principales tipos de servidor tenemos:

- Servidor Web: Se utilizan para la comunicación vía Internet o Intranet, permite alojar las páginas Web o sitios Web en un servidor, también permite transacciones por medio de un browser específico.
- Servidores de Base de Datos: Donde se almacenan las bases de datos, tablas, vistas, etc. Constituye uno de los servidores que más carga tiene.
- Servidor de transacciones: Procesa todas las transacciones las cuales primero las valida y luego realiza la petición al servidor.
- Servidor de archivos o aplicaciones: donde se almacenan todos los archivos o aplicaciones como por ejemplo procesadores de texto u hojas electrónicas.

## **Componentes Básicos de la Arquitectura Cliente Servidor**

Toda aplicación de un sistema de información está caracterizada por tres componentes básicos:

- Presentación de la información
- Procesos
- Almacenamiento de la información

## **Características del modelo Cliente Servidor**

El modelo Cliente Servidor tiene las siguientes características:

- Sus componentes pueden actuar como una sola entidad o también pueden actuar como entidades separadas realizando actividades independientes.
- Las funciones de Cliente Servidor pueden estar en plataformas separadas o en la misma plataforma.
- El servidor puede dar múltiples servicios a múltiples clientes de forma concurrente.
- Los cambios realizados en las plataformas tanto de cliente y de servidor se realizan de una manera transparente para el usuario final.
- Un sistema de servidores realiza múltiples funciones al mismo tiempo y presenta una imagen de un solo sistema a las estaciones Clientes.

En conclusión la arquitectura Cliente Servidor puede incluir múltiples plataformas, bases de datos, sistemas operativos y redes, a su vez estos pueden ser de distintos proveedores y funcionando al mismo tiempo.

### **Ventajas de la arquitectura Cliente Servidor**

Entre las principales ventajas tenemos:

- Centralización del control: control de acceso, recurso e integridad de datos.
- Escalabilidad: Se puede aumentar la capacidad tanto de clientes y servidores sin depender uno de otro.
- Reducción de tráfico: permite reducir considerablemente el tráfico en la red.

### **INTRANET**

Es una red de ordenadores de una Red de Área Local LAN privada, empresarial o educativa que proporciona herramientas de Internet. La Intranet fue creada para brindar mayor seguridad en la comparación de archivos carpetas y recursos.

Es una red privada que la tecnología de Internet usó como arquitectura elemental, se pueden ejecutar en diferentes plataformas de hardware, pero el hardware no es lo que constituye principalmente una Intranet, lo importante son los protocolos de interconexión tales como el TCP/IP para su transporte básico, IPX, etc.

Una de las principales ventajas es la capacitación que ya ha sido dada a los usuarios es decir, ya casi todos los usuarios están familiarizadas con el Internet lo cual constituye en su interfaz lo mismo.

Una Intranet está constituida generalmente por varias redes dentro de una empresa que se comunica con otra a través de TCP/IP. Estas redes se conocen generalmente como sub redes.

El software utilizado en una intranet se lo puede denominar como estándar ya que se utiliza browsers tales como Internet Explorer, Fire Fox, etc. Además de Navigator, Netscape, etc.

### **¿Cómo funciona una Intranet?**

El centro de una Intranet es la WWW (Word Wide Web), la razón principal por la que se creó las intranets es la Web la cual facilita la publicación de información por toda la compañía, además los hipertextos permiten saltar entre Webs, lo que quiere decir que podemos saltar de una Intranet al Internet.

Las intranets están basadas en la arquitectura cliente servidor por lo cual los clientes que previamente tienen instalado el browser y el protocolo TCP/IP pueden acceder a las páginas situadas en un Servidor.

Cada dirección de una página constituye una URL (Localizador uniforme de recursos) para lo cual también se necesita tener DNS (Domain Name Service) el cual permite transformar IPs en texto fácilmente reconocible, al momento de teclear y direccionar la URL los navegadores envían la petición de esa página Web a la Intranet. Luego un servidor recibe la petición usando http y la devuelve al navegador del Cliente Web, luego de realizar esta acción la conexión http se cierra para hacer un uso más eficaz de los recursos de la red.

## **Seguridad de una Intranet**

Aunque una Intranet constituya una red privada en la que se tengan grupos bien definidos esta no se encuentra exenta de ataques que pudiesen poner en riesgo la información de la organización ya que la mayoría de estos son realizados por sus mismos usuarios, y la mayoría de veces sin conocimiento de el ataque.

Para el control de la seguridad se aconseja tener un modelo de solución la cual debería contener:

- Políticas de seguridad: Documentos que describen principalmente la forma adecuada de uso de los recursos de un sistema de cómputo, responsabilidades y derecho tanto de clientes como de administradores.
- Control de acceso: utilizando firewall que es una combinación de hardware y software que controla el tipo de servicios permitidos desde y hacia la Intranet.
- Transacciones seguras.
- Anti Virus: Para protegernos de ataque de gusanos, troyanos, etc.

### **2.4 Hipótesis:**

El desarrollo de un sistema de control de Vehículos dentro de la Empresa Municipal de Agua Potable y Alcantarillado de Ambato EMAPA mejorará el funcionamiento tanto en el departamento Administrativo como en el área vehicular, además ayudará a brindar un mejor y mayor servicio a la ciudadanía.

## **2.5 Determinación de variables:**

### **2.5.1 Variable independiente**

Software de Control

### **2.5.2 Variable dependiente**

Mejorar el Servicio de vehículos en la Empresa Municipal de Agua potable y alcantarillado de Ambato “EMAPA”.

## **CAPITULO III**

### **METODOLOGÍA**

#### **3.1 MODALIDAD DE INVESTIGACIÓN**

La presente investigación se contextualiza en la modalidad de investigación de campo y bibliográfica, debido a que los hechos serán estudiados en primera instancia en base a normas legales que se encuentran tipificadas en diversos códigos, leyes, reglamentos, etc.

Además se realizará el trabajo en la Empresa municipal de Agua Potable y Alcantarillado de Ambato EMAPA, lo cual será de gran ayuda para obtener elementos de juicio necesarios para la configuración de esta investigación.

#### **3.2 TIPO DE INVESTIGACIÓN**

La investigación abarcará el nivel exploratorio pues, a través de ella se reconocerá la variables que nos competen, el nivel descriptivo permitirá caracterizar la realidad investigada, el nivel correlacional dilucidará el grado de relación entre las variables en estudio y finalmente el nivel explicativo detectará las causas de determinados comportamientos y canalizará la estructuración de propuestas de solución a la problemática que se analizará.

Por el enfoque será una investigación cualitativa pues se obtendrá información directa de los investigados, en virtud de los cuales será factible desarrollar un análisis crítico de los resultados y proponer alternativas de solución.

### **3.3 POBLACIÓN Y MUESTRA**

La presente investigación se la realizará en todo el departamento administrativo de la EMAPA, especialmente con los profesionales encargados del área vehicular.

### **3.4 TÉCNICAS E INSTRUMENTOS DE INVESTIGACIÓN**

Las técnicas que se emplearán en la presente investigación serán: la entrevista y la observación.

La entrevista será empleada para obtener datos significativos referentes a la manera en la cual se está controlando el área vehicular dentro de la Empresa Municipal de Agua Potable y Alcantarillado de Ambato EMAPA.

La técnica de la observación será de gran valor en la apreciación directa y sin filtros de la realidad, circunstancias que permitirán confrontar los hechos con palabras, elementos medulares para imprimir un sello de transparencia e imparcialidad de la investigación.

Los instrumentos que se utilizarán para obtener la información serán: el cuestionario de entrevista y complementariamente se utilizarán registros de información.

### **3.5 RECOLECCIÓN DE INFORMACIÓN**

Para la recolección eficaz de la información de campo, se recurrirá a las siguientes estrategias:

- Diseño y elaboración de los instrumentos de recolección de información a partir de registros de observación y entrevistas.

### **3.6 PROCESAMIENTO DE LA INFORMACIÓN**

Una vez aplicados los instrumentos se analizará la validez y se procederá a la tabulación de datos cualitativos los cuales se presentarán gráficamente en los términos de porcentaje a fin de facilitar la interpretación y realizar el trabajo práctico.

Finalmente, como parte medular de la investigación crítica propositiva, se estructurará una propuesta que será el informe técnico del trabajo realizado en EMAPA.

## CAPITULO IV

### ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

#### **Análisis Previo**

Los resultados se obtuvieron luego de realizar entrevistas a los encargados del área vehicular de EMAPA, y todas las personas que intervienen en el proceso de control vehicular, dichas entrevistas se realizaron con el formato que consta como **Anexo 1** en la documentación y de las cuales se pudo obtener la siguiente información.

- El área vehicular está encargada de diferentes procesos tales como:
  - control de mantenimiento de las unidades vehiculares
  - suministro de combustible tanto de las unidades vehiculares como de máquinas de las diferentes áreas.
  - Control de costos y gastos producidos por las unidades vehiculares.
  - Control de Salvoconductos del área vehicular.
  
- El área vehicular la conforman tres personas las cuales pueden realizar cualquier trámite sin tener restricciones de acceso.
  
- Los documentos legales para realizar las diferentes transacciones son:
  - Solicitud de mantenimiento.
  - Despacho de Combustible.
  - Solicitud de Combustible
  - Orden de movilización.
  
- El despacho de combustible (documento) se lo realiza diariamente una vez por día.

- En caso de no estar dentro del despacho de combustible se realizará una solicitud de combustible, también se lo hará en casos de viajes.
- La solicitud de mantenimiento deberá ser previamente aprobada por el departamento financiero para luego enviarla a los proveedores de servicios.
- La orden de movilización se lo utiliza para realizar salvoconductos y tendrá que ser firmada y aprobada por el jefe del departamento Administrativo.
- Existen diferentes informes que se enviarán al encargado del departamento Administrativo. Estos informes son:
  - Informe de Salvoconductos.
  - Informe de costos por partidas general.
  - Informe de costos por partidas solo de combustible.
  - Informe de costos por partidas solo de mantenimiento.
  - Informe de las unidades vehiculares con su detalle respectivo.

### **Análisis de la documentación**

La documentación manejada dentro del área vehicular se la puede dividir o clasificar en documentación de mantenimiento, documentación de combustible y documentación de movilización.

Entre la documentación de combustible tenemos las siguientes:

- Despacho de combustible

Este documento constituye el procedimiento para el control diario de parámetros como el kilometraje, costo de gasolina, gastos de combustible clasificado por vehículo y conductor designado a la fecha.

Lamentablemente este documento no se podrá automatizar al 100% ya que se lo realiza en el propio despachador de combustible, es decir se necesitaría computadores tipo pocketPC o equivalente, dispositivos con lo que si cuenta la empresa pero no están designadas para el uso del área vehicular, esta observación se hizo conocer al departamento de sistemas para su posterior desarrollo, la cual fue tomada con agrado.

De este documento se eliminará el parámetro de “hora”, ya que tras haber realizado entrevistas (ANEXO 1), se llegó a la conclusión de que no era necesario ya que siempre el despacho se realizaba por la mañana a la misma hora. Este documento lo podemos consultar en el ANEXO 2

- Solicitud de combustible

Este documento se lo realiza para vehículos que no han ingresado en el despacho de combustible diario, su uso mas común se produce cuando un vehículo sale de la ciudad, es decir este documento nos sirve para tanqueos extras a lo normal. Este documento se lo podrá automatizar en su totalidad con pequeños cambios como por ejemplo la agrupación del tipo de combustible. Este documento se lo puede consultar en el ANEXO 3

Entre la documentación de mantenimiento tenemos:

- Solicitud de materiales

Este documento se lo realiza para pedir uno o varios mantenimientos a vehículos, el proceso de este documento es el siguiente: primero se lo envía a financiero para su previa autorización, luego es enviada al proveedor para su ejecución. Este documento será automatizado en su totalidad. Se lo puede consultar en el ANEXO 4

Entre la documentación de movilización tenemos:

- Orden de movilización

Se lo realiza para realizar los Salvoconductos que son permisos de movilización de un vehículo a otra ciudad.

Aquí se detalla el tiempo de duración del salvoconducto junto con los datos del vehículo. Se lo puede consultar en el ANEXO 5

## **CAPITULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

#### **5.1 CONCLUSIONES**

- El sistema de control vehicular implementado en la EMAPA reduce considerablemente el tiempo de realización de procesos, es por esto que se agiliza la realización de trámites y se obtiene mayor eficiencia y eficacia del área.
- Este sistema reduce en gran medida los desperdicios de materiales y la posibilidad de equivocaciones al momento de realizar el papeleo correspondiente a cada operación.
- La implementación del sistema de control vehicular en la EMAPA cubre las necesidades de la empresa ya que se mejorará el servicio prestado a la comunidad lo cual constituye un objetivo fundamental de la EMAPA, que es la satisfacción del cliente.
- Por todas estas razones el sistema implementado reducirá los costos dentro de la empresa, ayudará a un mejor control del área vehicular e integrará a todos los departamentos ya que este software forma parte de un sistema integrado que abarca a toda la empresa.

#### **5.2 RECOMENDACIONES**

- La recomendación principal es que se continúe con la utilización de este y todos los sistemas dentro de la empresa, y se los permita evolucionar junto con el tiempo.

- Revisar periódicamente las contraseñas y claves de acceso para resguardar correctamente los datos de la empresa.
- La integración entre el departamento de sistemas y el área vehicular para la realización de controles de datos e información.
- Se recomienda realizar el mantenimiento del software y de la información en un rango de tiempo moderado.

## **CAPITULO VI**

### **ANÁLISIS DEL SISTEMA**

SOFTWARE DE CONTROL DE VEHÍCULOS DE LA EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO DE AMBATO “EMAPA”

#### **6.1 DATOS INFORMATIVOS**

**Nombre:**

Departamento Administrativo – Área Vehicular de la Empresa Municipal de Agua Potable y Alcantarillado de la Ciudad de Ambato

**Ubicación:**

Ambato, Bolívar 1960 y 5 de Junio

**Tipo de Empresa:**

Estatal

#### **6.2 ANTECEDENTES DE LA PROPUESTA**

Para el control de los vehículos del Área vehicular de la EMAPA se contaba con el programa Excel del paquete de Office 2003, esto ayudaba a la realización de reportes que eran presentados al jefe departamental.

Además el control de procesos y todo el papeleo de solicitudes de mantenimiento, solicitudes de combustible y salvoconductos respectivo se lo realizaba manualmente, esto no satisfacía las necesidades del departamento y

producía la pérdida de tiempo dentro del área, ya que se debía analizar todos los documentos para realizar un informe.

No utilizaba ninguna Base de datos y ningún programa automático que permita realizar de forma confiable los procesos, con lo cual no se estaba dando un tratamiento adecuado a la información ya que los datos no estaban siendo almacenados en una base de datos en su totalidad.

A más de esto la empresa está empezando a trabajar con herramientas de ASP.NET y base de datos SQL Server las dos herramientas de la empresa Microsoft con lo cual se agiliza el manejo de la información ya que como se sabe la utilización de programas de la misma empresa se acoplan más rápidamente permitiendo la fluidez de la información.

Por estas razones se ha planteado la alternativa de desarrollar un sistema para el control de toda el área vehicular que eliminen todos estos problemas, lo cual se detalla en el Análisis del Sistema.

### **6.3 Análisis de Riesgos**

Para todo proyecto se debe realizar un análisis previo de los posibles riesgos que se pueden suscitar durante el desarrollo del mismo, para esto nos valemos de diferentes herramientas que nos permitan visualizar de una forma cuantitativa estos riesgos, para esto podemos observar las tablas de la 1 a la 4 para su mayor comprensión.

#### **Identificación de Riesgos**

<b>RIESGOS</b>	<b>CONSECUENCIA</b>
<b>Proyecto:</b>	
Cronograma	Retraso y modificación del cronograma de actividades.

Recursos económicos	Aumento de gastos no planeados.
Experiencia del Personal	Consumo de tiempo en capacitar correctamente al personal.
Requerimiento	Cambio en los requerimientos del sistema por parte del usuario.
<b>Técnicos:</b>	
Mantenimiento	Inexperiencia en el manejo de la herramienta.
Interfaz	Incumplimiento de expectativas en el manejo del manejo de la interfaz.
<b>Varios:</b>	
Tecnología de punta	Consumo de tiempo en la capacitación a los usuarios.

Tabla 1 (Identificación de riesgos)

### Tabla de posibles impactos

RIESGOS	IMPACTO
<b>Proyecto:</b>	
Cronograma	3
Recursos económicos	2
Experiencia del Personal	4
Requerimiento	2
<b>Técnicos:</b>	
Mantenimiento	3
Interfaz	3
<b>Varios:</b>	
Tecnología de punta	2

Tabla 2 (Posibles impactos)

Donde:

1 = Catastrófico

2 = Crítico

3 = Marginal

4 = Despreciable

### Probabilidad de ocurrencia

<b>RIESGOS</b>	<b>PROBABILIDAD DE OCURRENCIA</b>
<b>Proyecto:</b>	
Cronograma	40%
Recursos económicos	20%
Experiencia del Personal	30%
Requerimiento	20%
<b>Técnicos:</b>	
Mantenimiento	10%
Interfaz	20%
<b>Varios:</b>	
Tecnología de punta	30%

Tabla 3 (Posibilidad de ocurrencia)

### Evaluación de Riesgos

<b>RIESGOS</b>	<b>NIVEL DE REFERENCIA</b>
<b>Proyecto:</b>	
Cronograma	Retrasos en entrega de informes y retraso en la entrega del proyecto final.
Recursos económicos	Aumento de costos, Faltante de

	recursos para la realización del proyecto.
Experiencia del Personal	Errores al momento de realizar trámites, Consumo de tiempo, lo cual puede retrasar la finalización del proyecto.
Requerimiento	Retrasos de entrega o incumplimiento de expectativas del usuario.
<b>Técnicos:</b>	
Mantenimiento	Aumento de costos de capacitación.
Interfaz	Retrasos en entrega de informes y retraso en la entrega del proyecto final.
<b>Varios:</b>	
Tecnología de punta	Aumento de costos para la empresa.

Tabla 4 (Evaluación de riesgos)

Luego de este análisis podemos sacar como conclusiones que no existen riesgos de nivel catastrófico que impidan la realización del proyecto, pero si existen riesgos que podrían aumentar el tiempo de realización del proyecto o que puedan aumentar los costos para la realización del sistema.

Se tendrá que prestar mucha atención en la experiencia que presenta el personal del área vehicular para buscar la mejor manera de capacitarlos sin perder recursos.

## **6.4 Análisis De Factibilidades**

### **6.4.1 Factibilidad Operacional:**

El proyecto planeado a realizarse en el área vehicular sí cumple con las factibilidades operacionales ya que cuenta con el visto bueno de las autoridades de la empresa quienes tienen una visión globalizada por lo cual ven pertinente automatizar los procesos dentro de la EMAPA.

Además se cuenta con el total apoyo del personal del área vehicular quienes demuestran su entusiasmo para la realización de este proyecto, muestran total apertura para la toma de información y están concientes que este sistema traerá muchos beneficios para toda el área y los ayudará sustancialmente a la realización de sus actividades.

A más del personal del área vehicular también se cuenta con el apoyo y visto bueno del departamento Administrativo el cual maneja el área de vehículos.

Y finalmente se cuenta con el apoyo del área de Sistemas de la empresa quienes trabajan como tutores del proyecto pero principalmente como apoyando y controlando este sistema.

### **6.4.2 Factibilidad Técnica:**

La factibilidad técnica está cubierta gracias a que la empresa con todas las herramientas necesarias para dar funcionamiento al proyecto vehicular.

#### **Hardware:**

Este aspecto fundamental para la realización del proyecto está controlado como se lo podrá ver en la siguiente tabla comparativa.

<b>REQUERIDO</b>	<b>DISPONIBLE</b>
<p><b>Computadora Servidor</b></p> <ul style="list-style-type: none"> <li>- Procesador P4 o equivalente.</li> <li>- Memoria RAM 2 Gb</li> <li>- Disco Duro 10 Gb</li> </ul> <p><b>Computadora cliente</b></p> <ul style="list-style-type: none"> <li>- Procesador P4 o equivalente</li> <li>- Memoria RAM 2 Gb</li> <li>- Tarjeta de red 100/10 Mbps</li> </ul> <p><b>Red</b></p> <ul style="list-style-type: none"> <li>- Hub o Switch 8 canales.</li> </ul>	<p><b>Computadora Servidor</b></p> <ul style="list-style-type: none"> <li>- Servidor Proliant Hp D1380 G4.</li> <li>- Doble Procesador XEON 3.6 GHZ</li> <li>- 1MB Caché</li> <li>- 2GB de Memoria RAM</li> <li>- Disco duro de 120 Gb</li> </ul> <p><b>Computadora cliente</b></p> <ul style="list-style-type: none"> <li>- Procesador P4 o equivalente.</li> <li>- Memoria RAM 2 Gb</li> <li>- Tarjeta de red 100/10 Mbps</li> </ul> <p><b>Red</b></p> <ul style="list-style-type: none"> <li>- Switch 16 canales (vehículos)</li> </ul>

Tabla 5 (Comparación de Hardware)

**Software**

<b>REQUERIDO</b>	<b>DISPONIBLE</b>
<p><b>Computadora Servidor</b></p> <ul style="list-style-type: none"> <li>- Microsoft Visual Studio. NET 2003 o posterior.</li> <li>- Microsoft SQL Server 2000 o posterior.</li> </ul> <p><b>Computadora cliente</b></p> <ul style="list-style-type: none"> <li>- Windows XP SP2 con IIS incorporado o Windows 2000 Server SP4</li> <li>- Internet Explorer Versión 5.0 o posterior</li> </ul>	<p><b>Computadora Servidor</b></p> <ul style="list-style-type: none"> <li>- Microsoft Visual Studio. NET 2003 o posterior.</li> <li>- Microsoft SQL Server 2003 o posterior.</li> </ul> <p><b>Computadora cliente</b></p> <ul style="list-style-type: none"> <li>- Windows XP SP2 con IIS incorporado</li> <li>- Internet Explorer Version 6.0</li> </ul>

Tabla 6 (Comparación de Software)

El aspecto del Software esta plenamente abastecido ya que también se cuenta con el software certificado como se puede ver en la tabla.

### **6.4.3 Factibilidad Económica:**

A continuación se analizarán los diferentes aspecto que influyen en la factibilidad económica del proyecto.

Primero se puede decir que en relación a licencias y compra de software el costo será nulo, ya que ya se cuenta en la empresa con todo el software con sus licencias necesarios para la realización e implementación del sistema de control vehicular. También el gasto de hosting no aumentará en nada los gastos ya que la Intranet se ya cuenta con el servidor Web.

En cuanto al Hardware, tampoco se deberá realizar gasto alguno ya que también como ya revisamos en las factibilidades operacionales, se cuenta con todo el equipo necesario, tanto de computadoras clientes, computadora servidor, y dispositivos de interconexión.

Además el costo de sueldo o pago a personal encargado también será nulo ya que se lo realizará por medio de pasantía en la cuál se deberá desarrollar y cumplir todos los objetivos planteados sin remuneración alguna, además los empleados del área vehicular colaborarán con toda la realización del software sin remuneración extra, ya que este trabajo ya está contemplado en sus funciones normales.

La inversión se verá aumentada en aspectos de oficina y gastos como el Internet, costo de impresiones, costos de materiales de oficina, costo de transporte, y costos varios.

Pero todos estos gastos, que en realidad resultan mínimos, se ven recuperados fácilmente, ya que encontramos beneficios en recursos

materiales, recursos humanos, recursos de tiempo, etc. En síntesis se podrán realizar mayor número de procesos en mucho menos tiempo.

Es por esto que la inversión en la realización del proyecto resulta muy buena para la empresa.

A continuación se describen los gastos que se tendrán durante el desarrollo del proyecto.

<b>DESCRIPCION</b>	<b>COSTO USD.</b>
Licencia de SQL Server	0.00 USD.
Licencia de Microsoft Visual Studio . NET	0.00 USD
Dispositivos de Red	0.00 USD
Computadoras	0.00 USD
Sueldo	0.00 USD
Internet	300.00 USD
Implementos de oficina	50.00 USD
Impresiones	50.00 USD
Costos indirectos	200.00 USD
<b>TOTAL</b>	<b>600.00 USD</b>

Tabla 7 (Posible inversión en el sistema)

## **6.5 Ingeniería De Software**

### **6.5.1 Funciones principales del sistema**

Las funciones que tendrá el sistema son las siguientes:

- Facilitar el ingreso, edición y eliminación de la información del área vehicular.

- Organizar la información de los vehículos, detallando su información individual.
- Facilitar el acceso a la información de los vehículos, impresión de reporte de vehículos actualizado.
- Organizar la información de solicitudes de mantenimiento, despachos de combustible y salvoconductos.
- Facilitar el acceso a la información de todos los procesos del área vehicular.
- Permitir la consulta rápida de la información de gastos de los vehículos.
- Realizar Informes tales como informe de costos general, informe de costos de mantenimiento, informe de costos de combustible e informa de salvoconductos.

### **6.5.2 Aspectos de funcionamiento:**

El sistema de control vehicular funcionará bajo la plataforma Microsoft Visual Studio. NET, de la cual se utilizará ASP.NET junto con C#.

Para el almacenamiento de la información utilizaremos la Base de Datos Microsoft SQL Server, para esto se utilizará el servidor principal de la empresa.

Para el almacenamiento de las páginas Web se utilizará el mismo servidor ya que funciona como servidor de Base de Datos y Servidor Web.

### **6.5.3 Restricciones técnicas y de funcionamiento:**

Como hemos visto antes no tenemos restricciones técnicas ni de funcionamiento ya que en la empresa si se cuenta con un departamento de Sistemas bien estructurado y formado, el departamento tiene bien distribuido las tareas tanto de redes como del software, es por esto que contamos con el apoyo necesario dentro de la empresa.

### **6.5.4 Entradas del Usuario:**

- Ingreso de nuevo vehículo
- Edición de un vehículo
- Datos de vehículo
- Creación de nueva solicitud de mantenimiento
- Datos del nuevo mantenimiento
- Datos de la factura de los proveedores
- Creación de nuevo despacho de combustible
- Datos del nuevo despacho de combustible
- Creación de nuevo Salvoconducto
- Datos del nuevo Salvoconducto

- Ingreso y edición de proveedores de combustible.
- Datos del nuevo proveedor de combustible
- Edición de datos de impuestos (IVA)
- Datos del nuevo valor del impuesto

#### **6.5.5 Salidas del Usuario**

- Consultas de Salvoconductos
- Consultas de solicitudes de mantenimiento
- Consultas de despachos de Combustible
- Reportes de vehículos
- Reportes de costos general
- Reportes de costos de mantenimiento
- Reportes de costos de Combustible.
- Reportes de Salvoconductos.

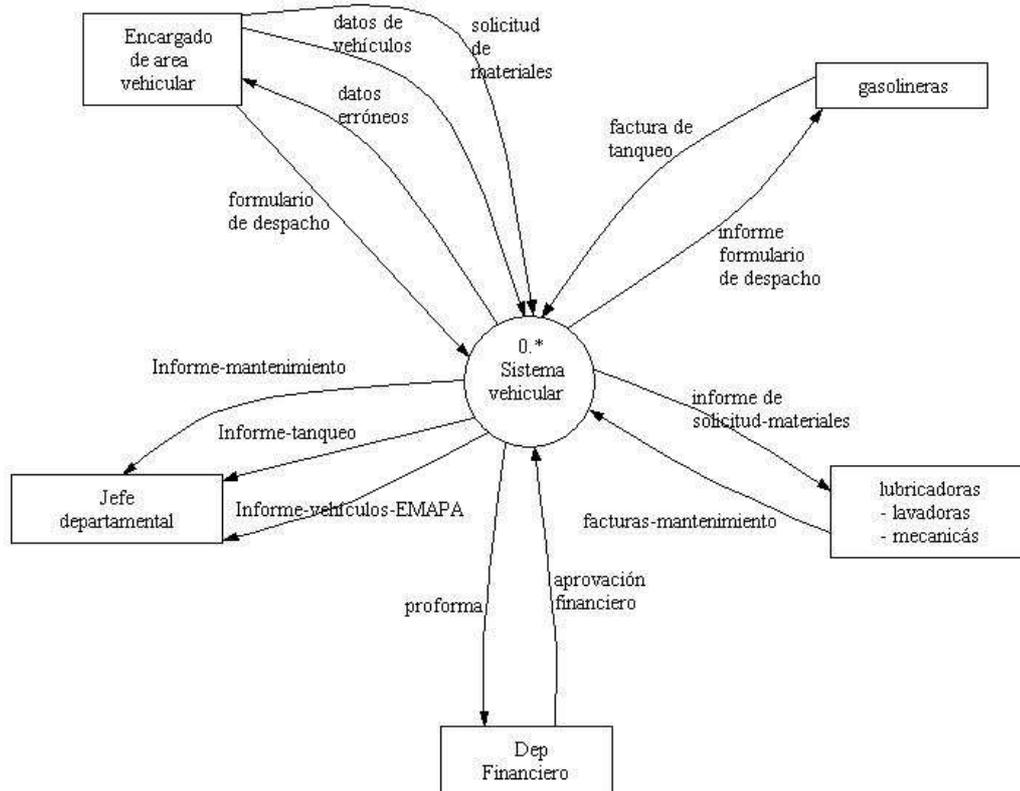
## 6.6 Análisis estructurado

### 6.6.1 Diagrama de flujo de Datos (DFD)

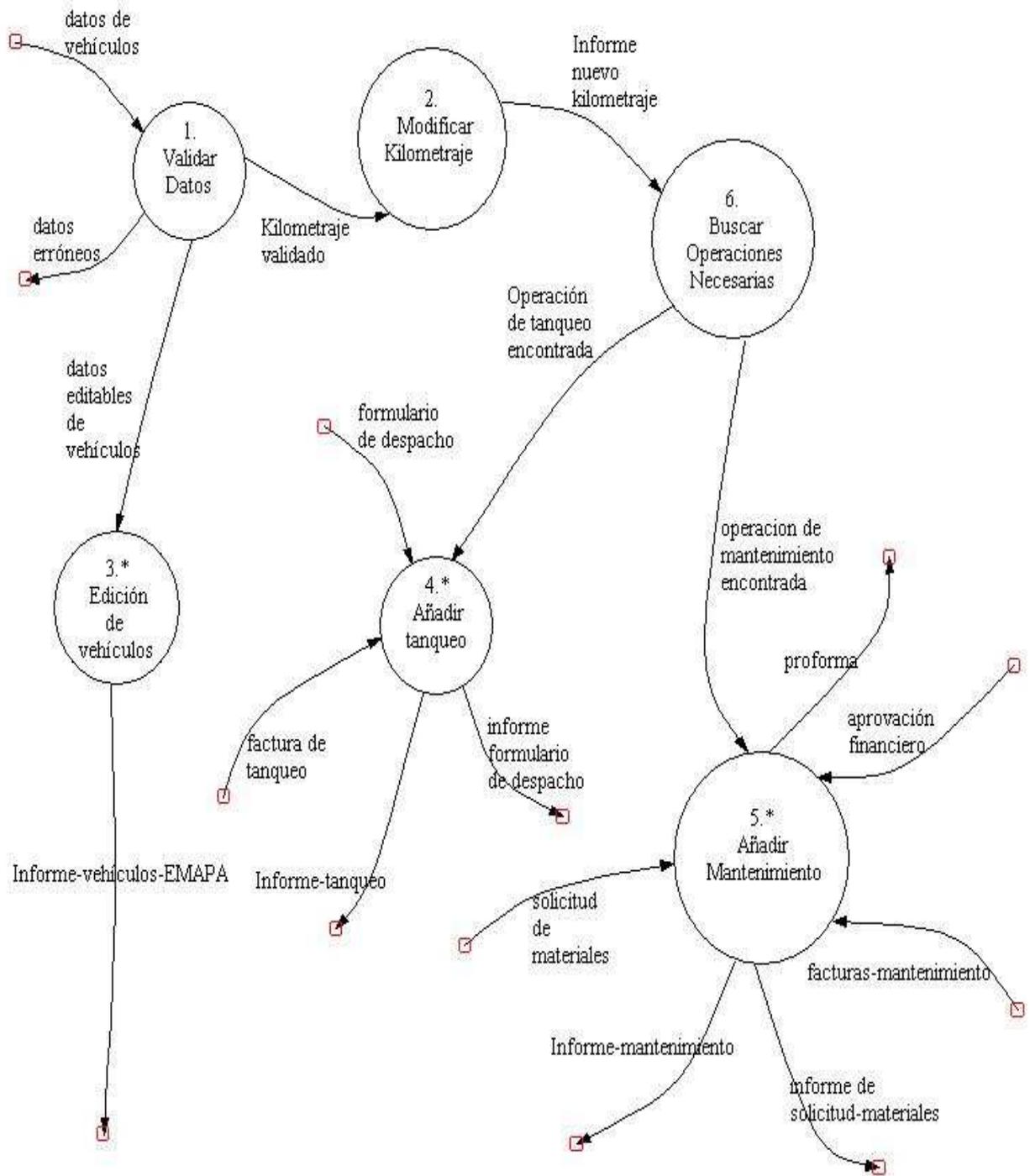
SOFTWARE DE CONTROL DE VEHÍCULOS DE LA EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO DE AMBATO “EMAPA”

#### SISTEMA VEHICULAR

##### NIVEL 0

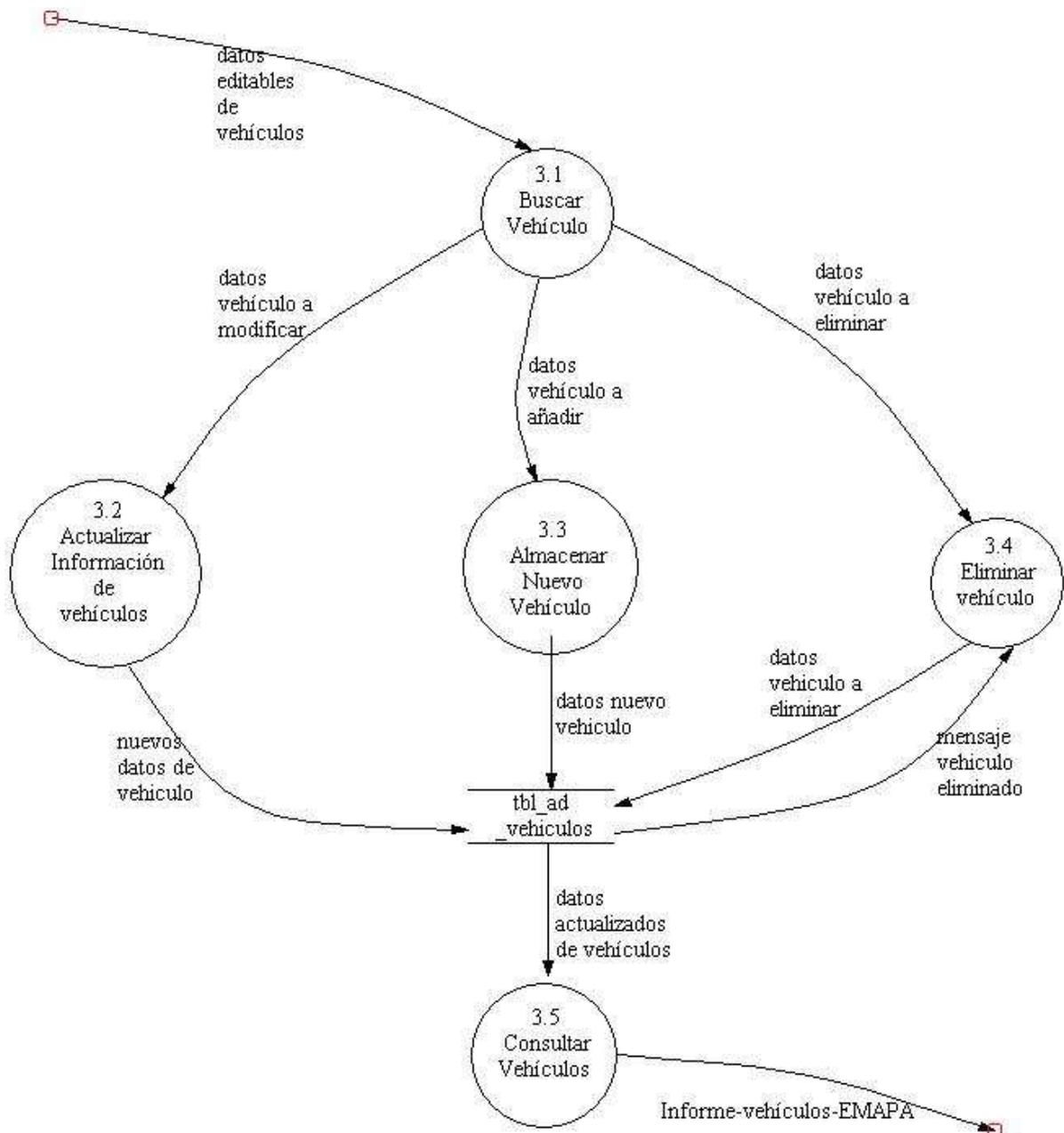


## NIVEL 1



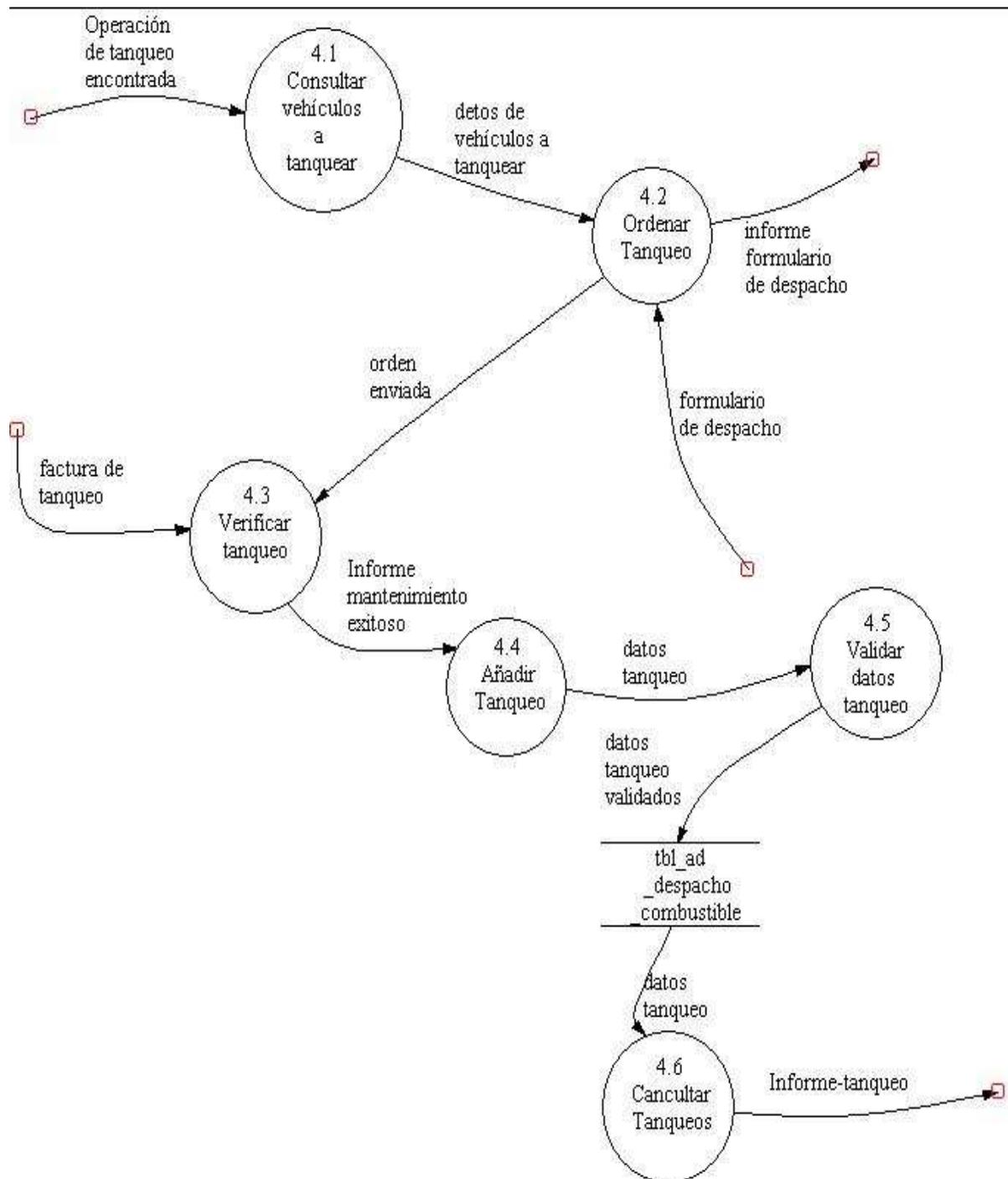
## NIVEL 2

### EDICIÓN DE VEHÍCULOS (3)



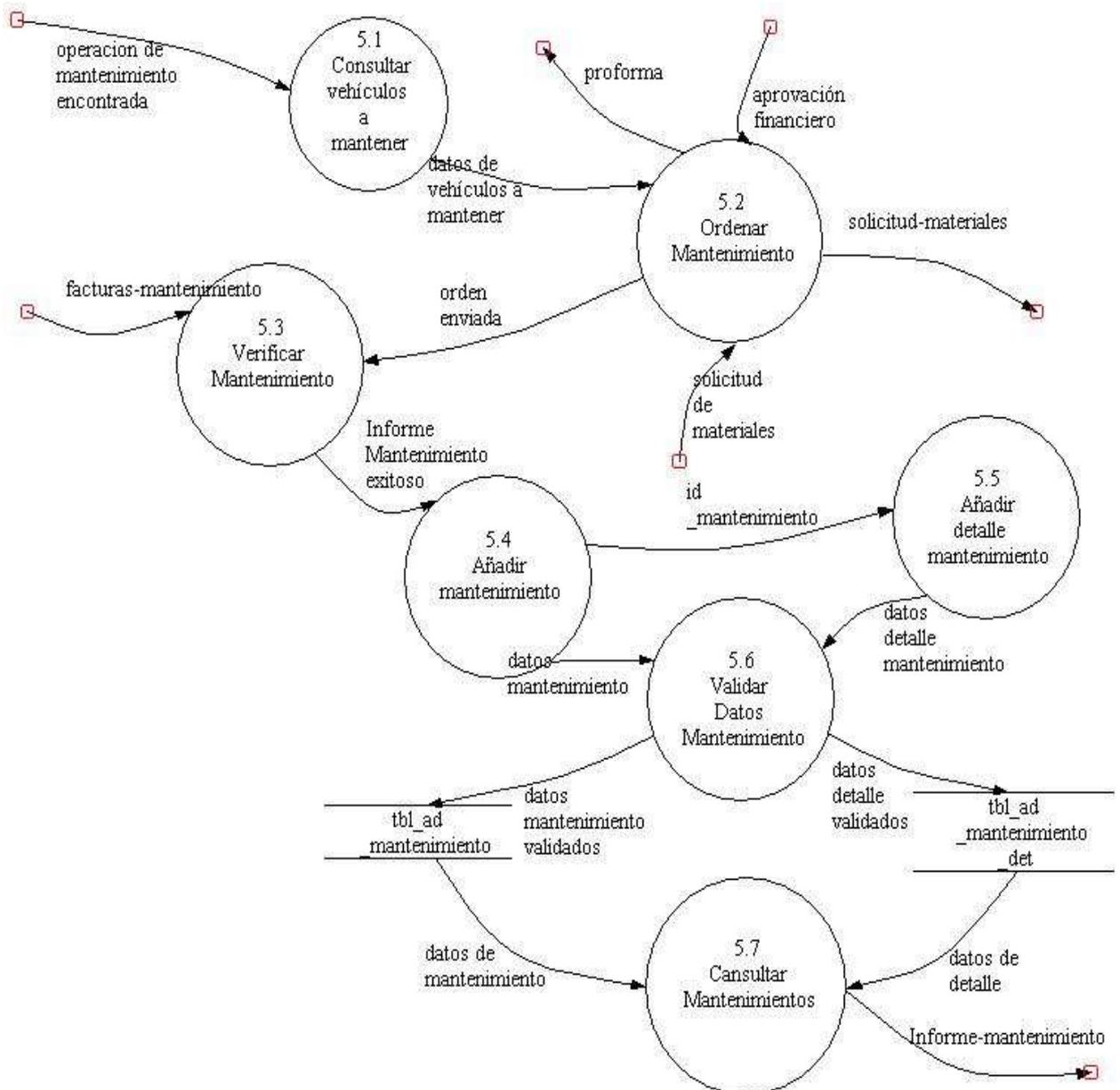
## NIVEL 2

### AÑADIR TANQUEO (4)



## NIVEL 2

### AÑADIR MANTENIMIENTO (5)

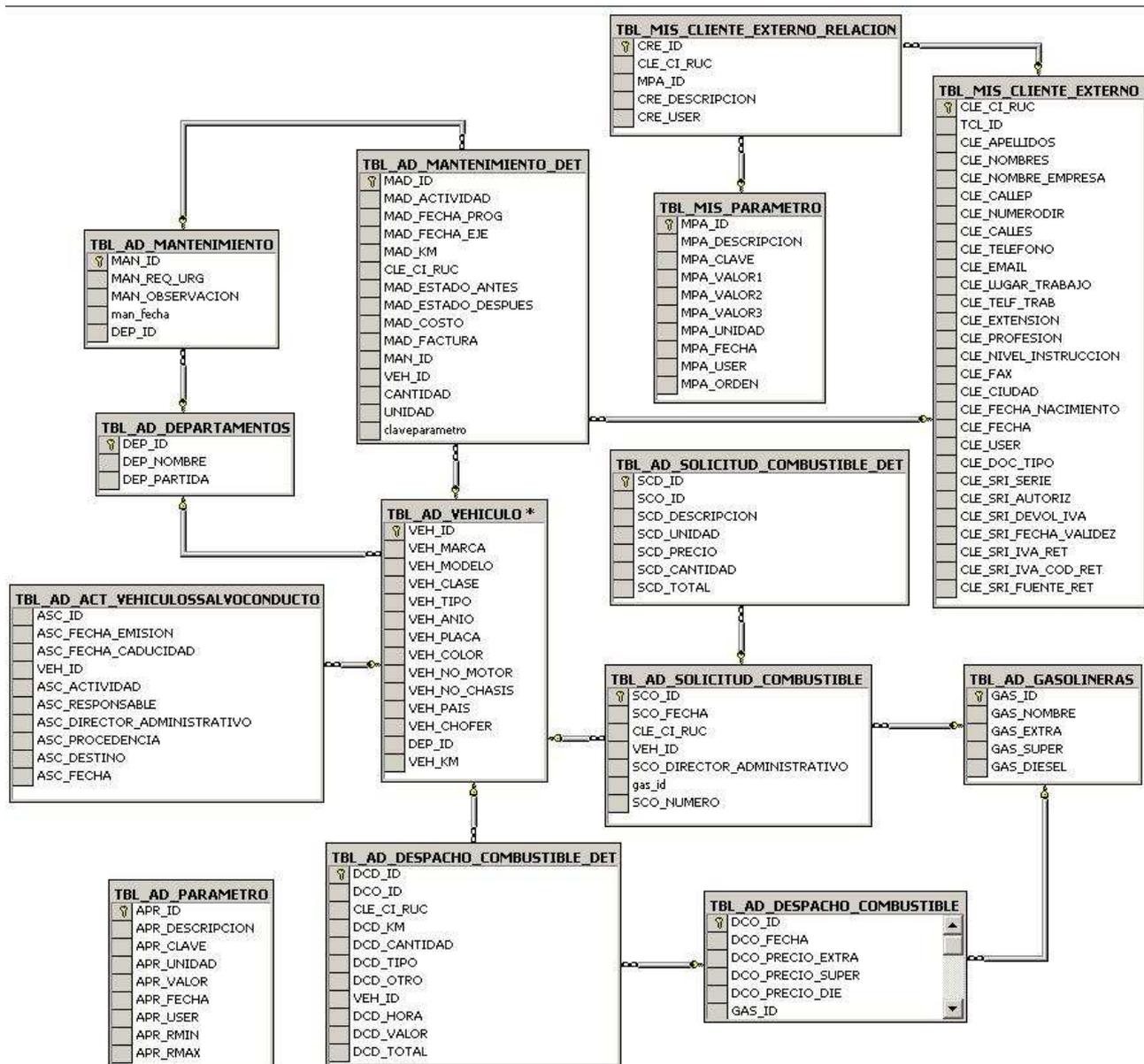


## CAPÍTULO VII

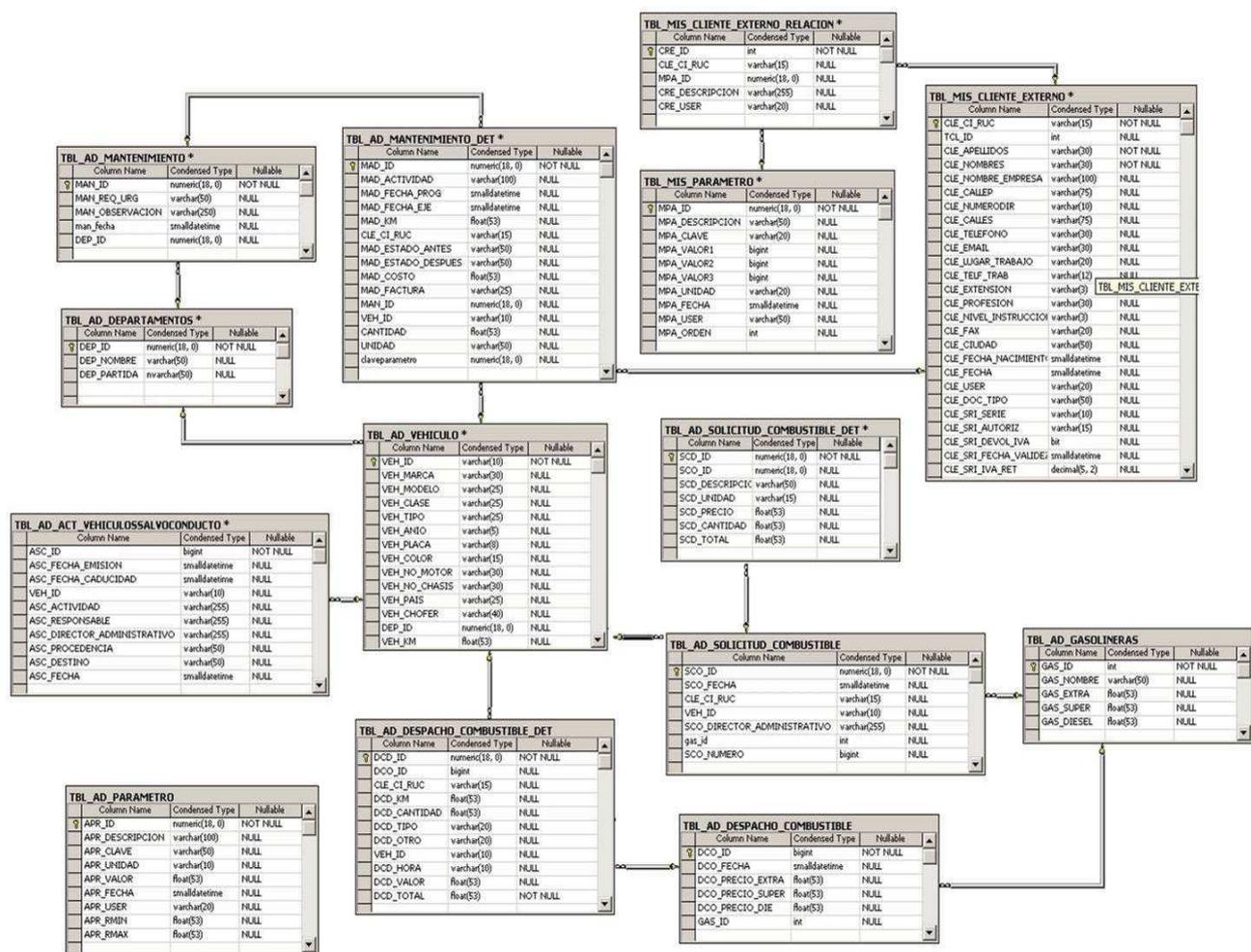
### DISEÑO DEL SISTEMA

#### 7.1 Diseño de la Base de Datos

##### 7.1.1 Diseño Físico



## 7.1.2 Diseño Lógico



## 7.2 DISEÑO DE ENTRADAS Y SALIDAS

### 7.2.1 DISEÑO DE ENTRADAS

Pantalla de autenticación, ingreso de nombre de usuario y contraseña.

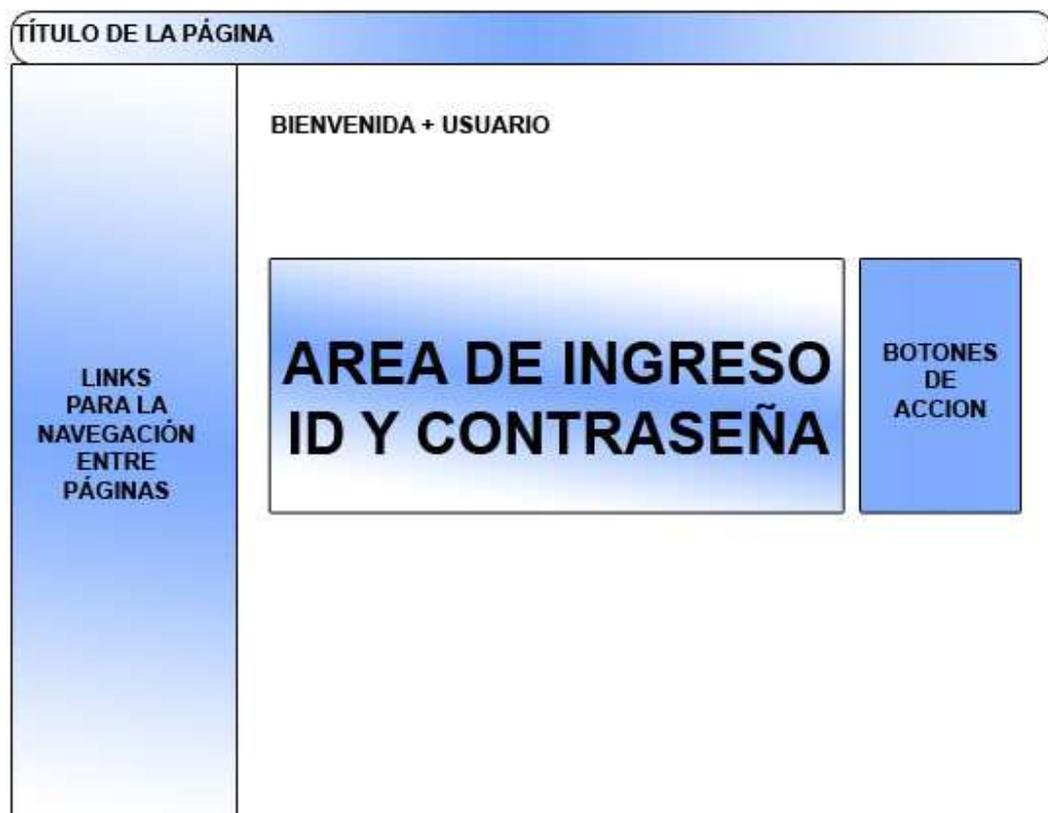


Figura 2

Pantalla de ingresos de los diferentes procedimientos



Figura 3

### 7.2.2 DISEÑO DE SALIDAS

Pantalla de búsqueda de información previa a la obtención del reporte



Figura 4

Pantalla de visualización de reportes



Figura 5







## CAPITULO VIII

### CODIFICACIÓN Y PRUEBAS

#### 8.1 CÓDIGO SQL DE LA BASE DE DATOS

##### 8.1.1 CREACIÓN DE TABLAS

➤ **TBL\_AD\_ACT\_VEHICULOSSALVOCONDUCTO**

```
CREATE TABLE [dbo].[TBL_AD_ACT_VEHICULOSSALVOCONDUCTO] (  
    [ASC_ID] [bigint] IDENTITY (1, 1) NOT NULL ,  
    [ASC_FECHA_EMISION] [smalldatetime] NULL ,  
    [ASC_FECHA_CADUCIDAD] [smalldatetime] NULL ,  
    [VEH_ID] [varchar] (10) COLLATE Modern_Spanish_CI_AS NULL ,  
    [ASC_ACTIVIDAD] [varchar] (255) COLLATE Modern_Spanish_CI_AS NULL ,  
    [ASC_RESPONSABLE] [varchar] (255) COLLATE Modern_Spanish_CI_AS NULL ,  
    [ASC_DIRECTOR_ADMINISTRATIVO] [varchar] (255) COLLATE  
Modern_Spanish_CI_AS NULL ,  
    [ASC_PROCEDENCIA] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [ASC_DESTINO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [ASC_FECHA] [smalldatetime] NULL  
) ON [PRIMARY]  
GO
```

➤ **TBL\_AD\_DEPARTAMENTOS**

```
CREATE TABLE [dbo].[TBL_AD_DEPARTAMENTOS] (  
    [DEP_ID] [numeric](18, 0) IDENTITY (1, 1) NOT NULL ,  
    [DEP_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [DEP_PARTIDA] [nvarchar] (50) COLLATE Modern_Spanish_CI_AS NULL  
) ON [PRIMARY]  
GO
```

➤ **TBL\_AD\_DESPACHO\_COMBUSTIBLE**

```
CREATE TABLE [dbo].[TBL_AD_DESPACHO_COMBUSTIBLE] (  
    [DCO_ID] [bigint] NOT NULL ,  
    [DCO_FECHA] [smalldatetime] NULL ,  
    [DCO_PRECIO_EXTRA] [float] NULL ,  
    [DCO_PRECIO_SUPER] [float] NULL ,  
    [DCO_PRECIO_DIE] [float] NULL ,  
    [GAS_ID] [int] NULL  
) ON [PRIMARY]  
GO
```

➤ **TBL\_AD\_DESPACHO\_COMBUSTIBLE\_DET**

```

CREATE TABLE [dbo].[TBL_AD_DESPACHO_COMBUSTIBLE_DET] (
    [DCD_ID] [numeric](18, 0) IDENTITY (1, 1) NOT NULL ,
    [DCO_ID] [bigint] NULL ,
    [CLE_CI_RUC] [varchar] (15) COLLATE Modern_Spanish_CI_AS NULL ,
    [DCD_KM] [float] NULL ,
    [DCD_CANTIDAD] [float] NULL ,
    [DCD_TIPO] [varchar] (20) COLLATE Modern_Spanish_CI_AS NULL ,
    [DCD_OTRO] [varchar] (20) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_ID] [varchar] (10) COLLATE Modern_Spanish_CI_AS NULL ,
    [DCD_HORA] [varchar] (10) COLLATE Modern_Spanish_CI_AS NULL ,
    [DCD_VALOR] [float] NULL ,
    [DCD_TOTAL] [float] NOT NULL
) ON [PRIMARY]
GO

```

### ➤ **TBL\_AD\_GASOLINERAS**

```

CREATE TABLE [dbo].[TBL_AD_GASOLINERAS] (
    [GAS_ID] [int] IDENTITY (1, 1) NOT NULL ,
    [GAS_NOMBRE] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [GAS_EXTRA] [float] NULL ,
    [GAS_SUPER] [float] NULL ,
    [GAS_DIESEL] [float] NULL
) ON [PRIMARY]
GO

```

### ➤ **TBL\_AD\_MANTENIMIENTO**

```

CREATE TABLE [dbo].[TBL_AD_MANTENIMIENTO] (
    [MAN_ID] [numeric](18, 0) IDENTITY (1, 1) NOT NULL ,
    [MAN_REQ_URG] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [MAN_OBSERVACION] [varchar] (250) COLLATE Modern_Spanish_CI_AS NULL ,
    [man_fecha] [smalldatetime] NULL ,
    [DEP_ID] [numeric](18, 0) NULL
) ON [PRIMARY]
GO

```

### ➤ **TBL\_AD\_MANTENIMIENTO\_DET**

```

CREATE TABLE [dbo].[TBL_AD_MANTENIMIENTO_DET] (
    [MAD_ID] [numeric](18, 0) IDENTITY (1, 1) NOT NULL ,
    [MAD_ACTIVIDAD] [varchar] (100) COLLATE Modern_Spanish_CI_AS NULL ,
    [MAD_FECHA_PROG] [smalldatetime] NULL ,
    [MAD_FECHA_EJE] [smalldatetime] NULL ,
    [MAD_KM] [float] NULL ,
    [CLE_CI_RUC] [varchar] (15) COLLATE Modern_Spanish_CI_AS NULL ,
    [MAD_ESTADO_ANTES] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [MAD_ESTADO_DESPUES] [varchar] (50) COLLATE Modern_Spanish_CI_AS
NULL ,
    [MAD_COSTO] [float] NULL ,
    [MAD_FACTURA] [varchar] (25) COLLATE Modern_Spanish_CI_AS NULL ,
    [MAN_ID] [numeric](18, 0) NULL ,
    [VEH_ID] [varchar] (10) COLLATE Modern_Spanish_CI_AS NULL ,
    [CANTIDAD] [float] NULL ,
    [UNIDAD] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [claveparametro] [numeric](18, 0) NULL

```

```
) ON [PRIMARY]
GO
```

➤ **TBL\_AD\_SOLICITUD\_COMBUSTIBLE**

```
CREATE TABLE [dbo].[TBL_AD_SOLICITUD_COMBUSTIBLE] (
    [SCO_ID] [numeric](18, 0) IDENTITY (1, 1) NOT NULL ,
    [SCO_FECHA] [smalldatetime] NULL ,
    [CLE_CI_RUC] [varchar] (15) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_ID] [varchar] (10) COLLATE Modern_Spanish_CI_AS NULL ,
    [SCO_DIRECTOR_ADMINISTRATIVO] [varchar] (255) COLLATE
Modern_Spanish_CI_AS NULL ,
    [gas_id] [int] NULL ,
    [SCO_NUMERO] [bigint] NULL
) ON [PRIMARY]
GO
```

➤ **TBL\_AD\_SOLICITUD\_COMBUSTIBLE\_DET**

```
CREATE TABLE [dbo].[TBL_AD_SOLICITUD_COMBUSTIBLE_DET] (
    [SCD_ID] [numeric](18, 0) IDENTITY (1, 1) NOT NULL ,
    [SCO_ID] [numeric](18, 0) NULL ,
    [SCD_DESCRIPCION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
    [SCD_UNIDAD] [varchar] (15) COLLATE Modern_Spanish_CI_AS NULL ,
    [SCD_PRECIO] [float] NULL ,
    [SCD_CANTIDAD] [float] NULL ,
    [SCD_TOTAL] [float] NULL
) ON [PRIMARY]
GO
```

➤ **TBL\_AD\_VEHICULO**

```
CREATE TABLE [dbo].[TBL_AD_VEHICULO] (
    [VEH_ID] [varchar] (10) COLLATE Modern_Spanish_CI_AS NOT NULL ,
    [VEH_MARCA] [varchar] (30) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_MODELO] [varchar] (25) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_CLASE] [varchar] (25) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_TIPO] [varchar] (25) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_ANIO] [varchar] (5) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_PLACA] [varchar] (8) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_COLOR] [varchar] (15) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_NO_MOTOR] [varchar] (30) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_NO_CHASIS] [varchar] (30) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_PAIS] [varchar] (25) COLLATE Modern_Spanish_CI_AS NULL ,
    [VEH_CHOFER] [varchar] (40) COLLATE Modern_Spanish_CI_AS NULL ,
    [DEP_ID] [numeric](18, 0) NULL ,
    [VEH_KM] [float] NULL
) ON [PRIMARY]
GO
```

➤ **TBL\_MIS\_PARAMETRO**

```
CREATE TABLE [dbo].[TBL_MIS_PARAMETRO] (
    [MPA_ID] [numeric](18, 0) IDENTITY (1, 1) NOT NULL ,
```

```

[MPA_DESCRIPCION] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[MPA_CLAVE] [varchar] (20) COLLATE Modern_Spanish_CI_AS NULL ,
[MPA_VALOR1] [bigint] NULL ,
[MPA_VALOR2] [bigint] NULL ,
[MPA_VALOR3] [bigint] NULL ,
[MPA_UNIDAD] [varchar] (20) COLLATE Modern_Spanish_CI_AS NULL ,
[MPA_FECHA] [smalldatetime] NULL ,
[MPA_USER] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[MPA_ORDEN] [int] NULL
) ON [PRIMARY]
GO

```

### ➤ TBL\_MIS\_CLIENTE\_EXTERNO

```

CREATE TABLE [dbo].[TBL_MIS_CLIENTE_EXTERNO] (
[CLE_CI_RUC] [varchar] (15) COLLATE Modern_Spanish_CI_AS NOT NULL ,
[TCL_ID] [int] NULL ,
[CLE_APELLIDOS] [varchar] (30) COLLATE Modern_Spanish_CI_AS NOT NULL ,
[CLE_NOMBRES] [varchar] (30) COLLATE Modern_Spanish_CI_AS NOT NULL ,
[CLE_NOMBRE_EMPRESA] [varchar] (100) COLLATE Modern_Spanish_CI_AS
NULL ,
[CLE_CALLEP] [varchar] (75) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_NUMERODIR] [varchar] (10) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_CALLES] [varchar] (75) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_TELEFONO] [varchar] (30) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_EMAIL] [varchar] (30) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_LUGAR_TRABAJO] [varchar] (20) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_TELF_TRAB] [varchar] (12) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_EXTENSION] [varchar] (3) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_PROFESION] [varchar] (30) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_NIVEL_INSTRUCCION] [varchar] (3) COLLATE Modern_Spanish_CI_AS
NULL ,
[CLE_FAX] [varchar] (20) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_CIUDAD] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_FECHA_NACIMIENTO] [smalldatetime] NULL ,
[CLE_FECHA] [smalldatetime] NULL ,
[CLE_USER] [varchar] (20) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_DOC_TIPO] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_SRI_SERIE] [varchar] (10) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_SRI_AUTORIZ] [varchar] (15) COLLATE Modern_Spanish_CI_AS NULL ,
[CLE_SRI_DEVOL_IVA] [bit] NULL ,
[CLE_SRI_FECHA_VALIDEZ] [smalldatetime] NULL ,
[CLE_SRI_IVA_RET] [decimal](5, 2) NULL ,
[CLE_SRI_IVA_COD_RET] [tinyint] NULL ,
[CLE_SRI_FUENTE_RET] [decimal](5, 2) NULL
) ON [PRIMARY]
GO

```

### ➤ TBL\_MIS\_CLIENTE\_EXTERNO\_RELACION

```

CREATE TABLE [dbo].[TBL_MIS_CLIENTE_EXTERNO_RELACION] (
[CRE_ID] [int] IDENTITY (1, 1) NOT NULL ,
[CLE_CI_RUC] [varchar] (15) COLLATE Modern_Spanish_CI_AS NULL ,
[MPA_ID] [numeric](18, 0) NULL ,
[CRE_DESCRIPCION] [varchar] (255) COLLATE Modern_Spanish_CI_AS NULL ,
[CRE_USER] [varchar] (20) COLLATE Modern_Spanish_CI_AS NULL
) ON [PRIMARY]
GO

```

## 8.1.2 CREACIÓN DE VISTAS

### ➤ VW\_AD\_DESPACHO\_DETALLE

```
CREATE VIEW dbo.VW_AD_DESPACHO_DETALLE
AS
SELECT  VEH_ID AS UNIDAD, DCD_HORA AS HORA, CLE_CI_RUC AS CHOFER,
DCD_KM AS KM, DCD_TIPO AS TIPO, DCD_CANTIDAD AS GALONES,
        DCD_VALOR AS USD, DCD_OTRO AS OTROS, DCD_TOTAL AS TOTAL,
DCO_ID AS NUM
FROM    dbo.TBL_AD_DESPACHO_COMBUSTIBLE_DET
```

### ➤ VW\_AD\_MANTENIMIENTO\_SOLICITUD

```
CREATE VIEW dbo.VW_AD_MANTENIMIENTO_SOLICITUD
AS
SELECT  VEH_ID AS VEHICULO, CANTIDAD AS CANTIDAD, UNIDAD AS UNIDAD,
MAD_ACTIVIDAD AS DESCRIPCION, MAN_ID AS #SOLICITUD
FROM    dbo.TBL_AD_MANTENIMIENTO_DET
```

### ➤ VW\_AD\_MANTENIMIENTOSSOLICITADOS

```
CREATE VIEW dbo.vw_ad_MantenimientosSolicitados
AS
SELECT  dbo.TBL_AD_DEPARTAMENTOS.DEP_NOMBRE AS Departamento,
dbo.TBL_AD_MANTENIMIENTO.man_fecha AS Fecha,
        dbo.TBL_AD_MANTENIMIENTO.MAN_OBSERVACION AS Observaciones,
dbo.TBL_AD_MANTENIMIENTO.MAN_ID AS Numero_Solicitud
FROM    dbo.TBL_AD_MANTENIMIENTO INNER JOIN
        dbo.TBL_AD_MANTENIMIENTO_DET ON
        dbo.TBL_AD_MANTENIMIENTO.MAN_ID =
        dbo.TBL_AD_MANTENIMIENTO_DET.MAN_ID INNER JOIN
        dbo.TBL_AD_DEPARTAMENTOS ON
        dbo.TBL_AD_MANTENIMIENTO.DEP_ID = dbo.TBL_AD_DEPARTAMENTOS.DEP_ID
WHERE   (dbo.TBL_AD_MANTENIMIENTO_DET.MAD_FACTURA IS NULL)
GROUP BY dbo.TBL_AD_MANTENIMIENTO.MAN_ID,
        dbo.TBL_AD_MANTENIMIENTO_DET.MAD_FACTURA,
        dbo.TBL_AD_DEPARTAMENTOS.DEP_ID,
        dbo.TBL_AD_DEPARTAMENTOS.DEP_NOMBRE,
        dbo.TBL_AD_MANTENIMIENTO.man_fecha,
        dbo.TBL_AD_MANTENIMIENTO.MAN_OBSERVACION,
        dbo.TBL_AD_MANTENIMIENTO.DEP_ID,
        dbo.TBL_AD_MANTENIMIENTO.MAN_ID
```

### ➤ VW\_AD\_VEHICULOS

```
CREATE VIEW dbo.vw_ad_VEHICULOS
```

```

AS
SELECT  dbo.TBL_AD_VEHICULO.VEH_ID AS UNIDAD,
dbo.TBL_AD_VEHICULO.VEH_MARCA AS MARCA,
dbo.TBL_AD_VEHICULO.VEH_MODELO AS MODELO,
        dbo.TBL_AD_VEHICULO.VEH_CLASE AS CLASE,
dbo.TBL_AD_VEHICULO.VEH_TIPO AS TIPO, dbo.TBL_AD_VEHICULO.VEH_ANIO AS
AÑO,
        dbo.TBL_AD_VEHICULO.VEH_PLACA AS PLACA,
dbo.TBL_AD_VEHICULO.VEH_COLOR AS COLOR,
        dbo.TBL_AD_VEHICULO.VEH_NO_MOTOR AS MOTOR,
dbo.TBL_AD_VEHICULO.VEH_NO_CHASIS AS CHASIS,
        dbo.TBL_AD_VEHICULO.VEH_PAIS AS PAIS,
dbo.TBL_AD_VEHICULO.VEH_CHOFER AS CHOFER,
dbo.TBL_AD_VEHICULO.VEH_KM AS KILOM,
        dbo.TBL_AD_DEPARTAMENTOS.DEP_NOMBRE AS DEPART
FROM      dbo.TBL_AD_VEHICULO INNER JOIN
        dbo.TBL_AD_DEPARTAMENTOS ON dbo.TBL_AD_VEHICULO.DEP_ID =
dbo.TBL_AD_DEPARTAMENTOS.DEP_ID
WHERE     (dbo.TBL_AD_VEHICULO.VEH_MARCA IS NULL) OR
        (dbo.TBL_AD_VEHICULO.VEH_MARCA <> 'maq00')

```

### ➤ **VW\_AD\_SALVOCONDUCTOS**

```

CREATE VIEW dbo.vw_ad_salvoconductos
AS
SELECT  dbo.TBL_AD_ACT_VEHICULOSSALVOCONDUCTO.ASC_FECHA_EMISION
AS [fecha emision],

dbo.TBL_AD_ACT_VEHICULOSSALVOCONDUCTO.ASC_FECHA_CADUCIDAD AS
[fecha caducidad],
        dbo.TBL_AD_ACT_VEHICULOSSALVOCONDUCTO.ASC_ACTIVIDAD AS
actividad,
        dbo.TBL_AD_ACT_VEHICULOSSALVOCONDUCTO.ASC_RESPONSABLE
AS reponsable,

dbo.TBL_AD_ACT_VEHICULOSSALVOCONDUCTO.ASC_DIRECTOR_ADMINISTRATIV
O AS [director administrativo],
        dbo.TBL_AD_VEHICULO.VEH_MARCA AS Marca,
dbo.TBL_AD_VEHICULO.VEH_MODELO AS Modelo,
dbo.TBL_AD_VEHICULO.VEH_PLACA AS Placa,
        dbo.TBL_AD_VEHICULO.VEH_CHOFER AS Chofer
FROM      dbo.TBL_AD_ACT_VEHICULOSSALVOCONDUCTO INNER JOIN
        dbo.TBL_AD_VEHICULO ON
dbo.TBL_AD_ACT_VEHICULOSSALVOCONDUCTO.VEH_ID =
dbo.TBL_AD_VEHICULO.VEH_ID

```

### **8.1.3 CREACIÓN DE STORE PROCEDURES**

```

CREATE PROCEDURE SP_AD_DESPACHO_OTROS
(
@MAN_REQ_URG VARCHAR(250),
@MAN_FECHA SMALLDATETIME,
@DEP_ID NUMERIC(18,0),

```

```

@MAD_ACTIVIDAD VARCHAR(100),
@CLE_CI_RUC VARCHAR(15),
@MAD_COSTO FLOAT,
@MAD_ID NUMERIC(18,0),
@VEH_ID VARCHAR(10),
@CANTIDAD INT
)
AS
BEGIN TRANSACTION
INSERT INTO TBL_AD_MANTENIMIENTO(MAN_REQ_URG,MAN_FECHA,DEP_ID)
VALUES(@MAN_REQ_URG,@MAN_FECHA,@DEP_ID)
INSERT INTO
TBL_AD_MANTENIMIWENTO_DET(MAD_ACTIVIDAD,CLE_CI_RUC,MAD_COSTO,MA
D_ID,VEH_ID,CANTIDAD)
VALUES(@MAD_ACTIVIDAD,@CLE_CI_RUC,@MAD_COSTO,@MAD_ID,@VEH_ID,@
CANTIDAD)
COMMIT

```

```

CREATE PROCEDURE SP_AD_INSERTARDESPACHO

```

```

(
@DCO_ID BIGINT,
@DCO_FECHA SMALLDATETIME,
@DCO_GASOLINERA VARCHAR(50),
@DCO_PRECIO_EXTRA FLOAT,
@DCO_PRECIO_SUPER FLOAT,
@DCO_PRECIO_DIE FLOAT
)
AS
BEGIN TRANSACTION
INSERT INTO TBL_AD_DESPACHO_COMBUSTIBLE
VALUES(@DCO_ID,@DCO_FECHA,@DCO_GASOLINERA,@DCO_PRECIO_EXTRA,
@DCO_PRECIO_SUPER,@DCO_PRECIO_DIE)
COMMIT

```

```

CREATE PROCEDURE SP_AD_INSERTARDESPACHO_DETALLE

```

```

(
@DCO_ID BIGINT,
@CLE_CI_RUC VARCHAR(15),
@DCD_KM FLOAT,
@DCD_CANTIDAD FLOAT,
@DCD_TIPO VARCHAR(20),
@DCO_OTRO VARCHAR(20),
@VEH_ID VARCHAR(10),
@DCD_HORA VARCHAR(10),
@DCD_VALOR FLOAT,
@DCD_TOTAL FLOAT
)
AS
BEGIN TRANSACTION
INSERT INTO TBL_AD_DESPACHO_COMBUSTIBLE_DET
VALUES(@DCO_ID,@CLE_CI_RUC,@DCD_KM,@DCD_CANTIDAD,@DCD_TIPO,
@DCO_OTRO,@VEH_ID,@DCD_HORA,@DCD_VALOR,@DCD_TOTAL)
COMMIT

```

```

CREATE PROCEDURE SP_AD_INSERTARSOLICITUD
(
@SCO_FECHA SMALLDATETIME,
@CLE_CI_RUC VARCHAR(15),
@VEH_ID VARCHAR(10),
@SCO_DIRECTOR_ADMINISTRATIVO VARCHAR(255),
@GAS_ID INT,
@SCO_NUMERO BIGINT
)
AS
BEGIN TRANSACTION
INSERT INTO TBL_AD_SOLICITUD_COMBUSTIBLE
VALUES(@SCO_FECHA,@CLE_CI_RUC,@VEH_ID,@SCO_DIRECTOR_ADMINISTRATI
VO,@GAS_ID,@SCO_NUMERO)
COMMIT

```

```

CREATE PROCEDURE SP_AD_INSERTARSOLICITUDETALLE
(
@SCO_ID NUMERIC(18,0),
@SCD_DESCRIPCION VARCHAR(50),
@SCD_UNIDAD VARCHAR(15),
@SCD_PRECIO FLOAT,
@SCD_CANTIDAD FLOAT,
@SCD_TOTAL FLOAT
)
AS
BEGIN TRANSACTION
INSERT INTO TBL_AD_SOLICITUD_COMBUSTIBLE_DET
VALUES(@SCO_ID,@SCD_DESCRIPCION,@SCD_UNIDAD,@SCD_PRECIO,@SCD_CAN
TIDAD,@SCD_TOTAL)
COMMIT

```

```

CREATE PROCEDURE SP_AD_INSERTARVEHICULO
(
@VEH_ID VARCHAR(10),
@VEH_MARCA VARCHAR(30),
@VEH_MODELO VARCHAR(25),
@VEH_CLASE VARCHAR(25),
@VEH_TIPO VARCHAR(25),
@VEH_ANIO VARCHAR(5),
@VEH_PLACA VARCHAR(8),
@VEH_COLOR VARCHAR(15),
@VEH_NO_MOTOR VARCHAR(30),
@VEH_NO_CHASIS VARCHAR(30),
@VEH_PAIS VARCHAR(25),
@VEH_CHOFER VARCHAR(40),
@DEP_ID NUMERIC(18,0),
@VEH_KM float)
AS
BEGIN TRANSACTION
INSERT INTO TBL_AD_VEHICULO
VALUES(@VEH_ID,@VEH_MARCA,@VEH_MODELO,@VEH_CLASE,@VEH_TIPO,@VE
H_ANIO,
@VEH_PLACA,@VEH_COLOR,@VEH_NO_MOTOR,@VEH_NO_CHASIS,@VEH_PAIS,@
VEH_CHOFER,@DEP_ID,@VEH_KM)
COMMIT

```

```

CREATE PROCEDURE [dbo].[SP_AD_MANTENIMIENTO]
(
    @MAN_REQ_URG VARCHAR(250),
    @MAN_OBSERVACION VARCHAR(250),
    @MAN_FECHA SMALLDATETIME,
    @DEP_ID NUMERIC(18,0)
)
AS
BEGIN TRANSACTION
INSERT INTO TBL_AD_MANTENIMIENTO
VALUES(@MAN_REQ_URG,@MAN_OBSERVACION,@MAN_FECHA,@DEP_ID)
COMMIT

CREATE PROCEDURE SP_AD_MANTENIMIENTO_DET_SOLICITUD
(
    @MAD_ACTIVIDAD VARCHAR(100),
    @MAD_FECHA_PROG SMALLDATETIME,
    @MAD_KM FLOAT,
    @MAN_ID BIGINT,
    @VEH_ID VARCHAR(10),
    @CANTIDAD float,
    @UNIDAD VARCHAR(50),
    @CLAVEPARAMETRO NUMERIC(18,0),
    @MAD_ESTADO_ANTES VARCHAR(50)
)
AS
BEGIN TRANSACTION
INSERT INTO
TBL_AD_MANTENIMIENTO_DET(MAD_ACTIVIDAD,MAD_FECHA_PROG,MAD_KM,M
AN_ID,VEH_ID,CANTIDAD,UNIDAD,CLAVEPARAMETRO,MAD_ESTADO_ANTES)
VALUES(@MAD_ACTIVIDAD,@MAD_FECHA_PROG,@MAD_KM,@MAN_ID,@VEH_ID
,@CANTIDAD,@UNIDAD,@CLAVEPARAMETRO,@MAD_ESTADO_ANTES)
COMMIT

```

## 8.2 CÓDIGO DE LA APLICACIÓN

### 8.2.1 CÓDIGO DE CONFIGURACIÓN

Para iniciar la aplicación hay que tener en cuenta cierto parámetros de configuración, estos parámetros son de conexión a la Base de datos y de autenticación de usuario.

Las siguientes líneas de código se las debe agregar en el archivo Web.config de la aplicación.

```
<add key="Cadena" value="Server=favas-computer\emapa;  
database=emapaa;User Id =fabian;Password=fabian111"/>  
  
<add key="frm_ad_consultarvehiculos.aspx"  
value="pg_ad_vehiculos"/>
```

A más de esto debemos agregar la clase de inicio con el siguiente código:

```
using System;  
using System.Data;  
using System.Data.SqlClient;  
  
namespace emapaSistema  
{  
    public class clase_inicio  
    {  
        public string conexion =  
            System.Configuration.ConfigurationSettings.  
                AppSettings["Cadena"];  
  
        public clase_inicio()  
        {  
        }  
    }  
}
```

A más de esto es cada página debemos aumentar el siguiente código en el evento

Load:

```
string pr_user="";
pr_user=this.Request.QueryString.Get("pr_user");
string grupoPagina=System.Configuration.ConfigurationSettings
.AppSettings["frm_ad_consultarvehiculos.aspx"];
this.permisos.s_comprobar_permiso(pr_user,grupoPagina,con
.conexion);
int aparecioSeñal=0;
for (int i=0;i < respuestaPermisos.Length;i++)
{
    if(respuestaPermisos.Substring(i,1) == "+")
    {
        aparecioSeñal=1;
    }
    if(respuestaPermisos.Substring(i,1)!="+" && aparecioSeñal == 0)
    {
        lectura += respuestaPermisos.Substring(i,1);
    }
    else
    {
        if(respuestaPermisos.Substring(i,1)!="+"&&aparecioSeñal== 1)
        {
            edicion += respuestaPermisos.Substring(i,1);
        }
    }
}
if(lectura=="255" || edicion=="255")
{
    //Código de la página
}
else
{
    Response.Redirect("../aspnet/pgerror/accesodenegado.htm");
}
```

## 8.2.2 CODIGO DE LAS PÁGINAS

Para empezar cada página debemos añadir los using que son referencias a librerías.

```
using System;
using System.Collections;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Web;
using System.Web.SessionState;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Web.UI.HtmlControls;
using System.Data.SqlClient;
using System.Collections.Specialized;
```

- **frm\_ad\_consultarvehiculos**

### **Variables globales de la página:**

```
BiblioPermisos.Class1 permisos=new BiblioPermisos.Class1();
clase_inicio con=new clase_inicio();
string respuestaPermisos;
string lectura="";
string edicion="";
```

### **Funciones de usuario**

Esta función se la utiliza para ordenar los vehículos por varias categorías

```
private void ordenarvehiculos(string parametro)
{
    SqlConnection conectar=new SqlConnection(this.con.conexion);
    SqlDataAdapter daVehiculosVista=new SqlDataAdapter("Select *
from vw_ad_vehiculos order by "+parametro,conectar);
    DataSet dsVehiculos=new DataSet();
    daVehiculosVista.Fill(dsVehiculos,"vw_ad_VEHICULOS");
    if(Session["edicion"].ToString()=="true")
    {
        this.dgVistaVehiculos.DataSource=dsVehiculos;
        this.dgVistaVehiculos.DataMember="vw_ad_VEHICULOS";
        this.dgVistaVehiculos.DataBind();
    }
    else
    {
        this.dgVistaVehiculos2.DataSource=dsVehiculos;
        this.dgVistaVehiculos2.DataMember="vw_ad_VEHICULOS";
        this.dgVistaVehiculos2.DataBind();
    }
}
```

### **Funciones de la página**

```
private void Page_Load(object sender, System.EventArgs e)
{
    string pr_user="";
    pr_user=this.Request.QueryString.Get("pr_user");
    string grupoPagina= System.Configuration.
ConfigurationSettings.AppSettings
["frm_ad_consultarvehiculos.aspx"];
    respuestaPermisos="255+255";
    int aparecioSeñal=0;
    for (int i=0;i < respuestaPermisos.Length;i++)
    {
        if(respuestaPermisos.Substring(i,1) == "+")
        {
            aparecioSeñal=1;
        }
    }
    if(respuestaPermisos.Substring(i,1)!="+"&& aparecioSeñal==0)
    {
```

```

        lectura += respuestaPermisos.Substring(i,1);
    }
    else
    {
        if(respuestaPermisos.Substring(i,1)!="&&aparecioSeñal==
        1)
        {
            edicion += respuestaPermisos.Substring(i,1);
        }
    }
}
if(lectura=="255")
{
    if(!IsPostBack)
    {
        this.txtBuscarNum.Attributes.Add("onkeydown",
        "atrapaenter("+this.Button1.ClientID+",event);");
        this.txtBuscar.Attributes.Add("onkeydown",
        "atrapaenter("+this.Button1.ClientID+",event);");
        Session.Clear();
        SqlConnection conectar=new
        SqlConnection(this.con.conexion);
        SqlDataAdapter daVehiculosVista=new
        SqlDataAdapter("Select * from vw_ad_VEHICULOS where marca
        is null or marca!='maq00'",conectar);
        DataSet dsVehiculos=new DataSet();
        daVehiculosVista.Fill(dsVehiculos,"vw_ad_VEHICULOS");
        if(edicion=="255")
        {
            Session.Add("edicion","true");
            this.dgVistaVehiculos.Visible=true;
            this.dgVistaVehiculos2.Visible=false;
            this.dgVistaVehiculos.DataSource=dsVehiculos;
            this.dgVistaVehiculos.DataMember="vw_ad_VEHICULOS";
            this.dgVistaVehiculos.DataBind();
        }
        else
        {
            Session.Add("edicion","false");
            this.dgVistaVehiculos.Visible=false;
            this.dgVistaVehiculos2.Visible=true;
            this.dgVistaVehiculos2.DataSource=dsVehiculos;
            this.dgVistaVehiculos2.DataMember="vw_ad_VEHICULOS";
            this.dgVistaVehiculos2.DataBind();
            this.ImageButton2.Visible=false;
        }
    }
}
else
{
    Response.Redirect("../aspnet/pgerror/accesodenegado.htm");
}
}
}

```

### Función para buscar o filtrar vehículos

```

private void btnBuscar_Click(object sender, System.EventArgs e)
{
    SqlConnection conectar=new SqlConnection(this.con.conexion);

```

```

SqlDataAdapter daVehiculosVista=new SqlDataAdapter("Select *
from vw_ad_VEHICULOS where (unidad='"+this.txtBuscarNum.Text+"'
or placa ='"+this.txtBuscar.Text+"')",conectar);
DataSet dsVehiculos=new DataSet();
daVehiculosVista.Fill(dsVehiculos,"vw_ad_VEHICULOS");
if(Session["edicion"].ToString()=="true")
{
    this.dgVistaVehiculos.DataSource=dsVehiculos;
    this.dgVistaVehiculos.DataMember="vw_ad_VEHICULOS";
    this.dgVistaVehiculos.DataBind();
}
else
{
    this.dgVistaVehiculos2.DataSource=dsVehiculos;
    this.dgVistaVehiculos2.DataMember="vw_ad_VEHICULOS";
    this.dgVistaVehiculos2.DataBind();
}
}
}

```

- **frm\_ad\_mdificarveh**

### Variables globales de la página

```

clase_inicio con=new clase_inicio();
private string claveDepartamento="";

```

### Funciones de Usuario

```

private void boton_nuevomod()
{
    this.Label1.Text="NUEVO";
    //blanquear
    this.txtUnidad.Text="";
    this.txtTipo.Text="";
    this.txtPlaca.Text="";
    this.txtPais.Text="";
    this.txtMotor.Text="";
    this.txtModelo.Text="";
    this.txtMarca.Text="";
    this.txtKilometraje.Text="0";
    this.txtColor.Text="";
    this.txtClase.Text="";
    this.txtChasis.Text="";
    this.ddlChofer.Items.Clear();
    this.ddlDepartamento.Items.Clear();
    //habilitar
    this.txtUnidad.ReadOnly=false;
    this.txtTipo.ReadOnly=false;
    this.txtPlaca.ReadOnly=false;
    this.txtPais.ReadOnly=false;
    this.txtMotor.ReadOnly=false;
    this.txtModelo.ReadOnly=false;
    this.txtMarca.ReadOnly=false;
    this.txtKilometraje.ReadOnly=false;
    this.txtColor.ReadOnly=false;
    this.txtClase.ReadOnly=false;
    this.txtChasis.ReadOnly=false;
}

```

```

this.ddlChofer.Enabled=true;
this.ddlDepartamento.Enabled=true;
this.ddlAño.Enabled=true;
//realizar coneccion
SqlConnection conectar=new SqlConnection(this.con.conexion);
SqlDataReader drlector;
conectar.Open();
//modificar llenar ddldepto
SqlCommand cmdPrincipal;
if(Session["operacion"].ToString()=="Edit")
{
    this.txtUnidad.Enabled=false;
    if(Session["año"].ToString()!="&nbsp;")
    {
        this.ddlAño.Items.Add(Session["año"].ToString());
    }
    if(Session["chofer"].ToString()!="&nbsp;")
    {
        this.ddlChofer.Items.Add(Session["chofer"].ToString());
    }
    cmdPrincipal=new SqlCommand("select * from
tbl_ad_departamentos where
dep_nombre='"+
Session["departamento"].ToString().TrimStart()
+"'",conectar);
drlector=cmdPrincipal.ExecuteReader();
if(drlector.Read())
{
    this.ddlDepartamento.Items.Add
(drlector["DEP_ID"].ToString()+
" "+Session["departamento"].ToString());
}
drlector.Close();
}
//seleccionar cliente externo (chofer)
cmdPrincipal=new SqlCommand("SELECT
TBL_MIS_CLIENTE_EXTERNO.CLE_NOMBRES,TBL_MIS_CLIENTE_EXTERNO.
CLE_APELLIDOS FROM TBL_MIS_CLIENTE_EXTERNO_RELACION INNER
JOIN TBL_MIS_CLIENTE_EXTERNO ON
TBL_MIS_CLIENTE_EXTERNO_RELACION.CLE_CI_RUC =
TBL_MIS_CLIENTE_EXTERNO.CLE_CI_RUC WHERE
(TBL_MIS_CLIENTE_EXTERNO_RELACION.MPA_ID = 9) order
by(cle_apellidos)",conectar);
drlector=cmdPrincipal.ExecuteReader();
while(drlector.Read())
{
    this.ddlChofer.Items.Add(drlector
["CLE_APELLIDOS"].ToString()+"
"+drlector["CLE_NOMBRES"].ToString());
}
this.ddlChofer.DataBind();
drlector.Close();
//seleccionar departamentos
cmdPrincipal=new SqlCommand("select * from
tbl_ad_departamentos",conectar);
drlector=cmdPrincipal.ExecuteReader();
int c=0;
while(drlector.Read())
{

```

```

        if(c>0)
        {
            this.claveDepartamento=drlector["DEP_ID"].ToString();
            c++;
        }
        this.ddlDepartamento.Items.Add(drlector["DEP_ID"]
        .ToString()+" "+drlector["DEP_NOMBRE"].ToString());
    }
    this.ddlDepartamento.DataBind();
    drlector.Close();
    conectar.Close();
}

```

## Funciones de la página

Función para modificar los datos del vehículo escogido a la Base de Datos

```

private void ibtnAceptar_Click(object sender,
System.Web.UI.ImageClickEventArgs e)
{
    if(this.txtKilometraje.Text=="")
    {
        this.txtKilometraje.Text="0";
    }
    SqlConnection conectar=new SqlConnection(this.con.conexion);
    conectar.Open();
    //sacar la id del departamento
    int i=0;
    string semiClave="";
    string letra="";
    semiClave=this.ddlDepartamento.SelectedValue.ToString();
    while(letra!=" ")
    {
        letra=semiClave.Substring(i,1);
        this.claveDepartamento=this.claveDepartamento+letra;
        i++;
        this.claveDepartamento=this.claveDepartamento.TrimEnd();
    }
    //INSERTAR EN TABLA VEHICULOS
    //modificar vehiculo
    SqlCommand cmdPrincipal = new
    SqlCommand("sp_ad_modificarvehiculo", conectar);
    cmdPrincipal.CommandType =
    System.Data.CommandType.StoredProcedure;
    SqlParameter p0 = new
    SqlParameter("@veh_id",System.Data.SqlDbType.VarChar);
    p0.Value = this.txtUnidad.Text;
    p0.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p0);
    SqlParameter p1 = new
    SqlParameter("@veh_marca",System.Data.SqlDbType.VarChar);
    p1.Value = this.txtMarca.Text;
    p1.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p1);
    SqlParameter p2 = new
    SqlParameter("@veh_modelo",System.Data.SqlDbType.VarChar);
    p2.Value = this.txtModelo.Text;
    p2.Direction = ParameterDirection.Input;
}

```

```

cmdPrincipal.Parameters.Add(p2);
SqlParameter p3 = new
SqlParameter("@veh_clase", System.Data.SqlDbType.VarChar);
p3.Value = this.txtClase.Text;
p3.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p3);
SqlParameter p4 = new
SqlParameter("@veh_tipo", System.Data.SqlDbType.VarChar);
p4.Value = this.txtTipo.Text;
p4.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p4);
SqlParameter p5 = new
SqlParameter("@veh_anio", System.Data.SqlDbType.VarChar);
p5.Value = this.ddlAño.SelectedValue;
p5.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p5);
SqlParameter p6 = new
SqlParameter("@veh_placa", System.Data.SqlDbType.VarChar);
p6.Value = this.txtPlaca.Text;
p6.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p6);
SqlParameter p7 = new
SqlParameter("@veh_color", System.Data.SqlDbType.VarChar);
p7.Value = this.txtColor.Text;
p7.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p7);
SqlParameter p8 = new
SqlParameter("@veh_no_motor", System.Data.SqlDbType.VarChar);
p8.Value = this.txtMotor.Text;
p8.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p8);
SqlParameter p9 = new
SqlParameter("@veh_no_chasis", System.Data.SqlDbType.VarChar);
p9.Value = this.txtChasis.Text;
p9.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p9);
SqlParameter p10 = new
SqlParameter("@veh_pais", System.Data.SqlDbType.VarChar);
p10.Value = this.txtPais.Text;
p10.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p10);
SqlParameter p11 = new
SqlParameter("@veh_chofer", System.Data.SqlDbType.VarChar);
p11.Value = this.ddlChofer.SelectedValue;
p11.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p11);
SqlParameter p12 = new
SqlParameter("@dep_id", System.Data.SqlDbType.VarChar);
p12.Value = this.claveDepartamento;
p12.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p12);
SqlParameter p13 = new
SqlParameter("@veh_km", System.Data.SqlDbType.Float);
p13.Value = float.Parse(this.txtKilometraje.Text);
p13.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p13);
cmdPrincipal.ExecuteNonQuery();
conectar.Close();
Server.Transfer("frm_ad_consultarvehiculos.aspx", true);

```

```
}
```

## Función para almacenar los datos del nuevo vehículo en la Base de Datos

```
private void ibtnAceptarNuevo_Click(object sender,
System.Web.UI.ImageClickEventArgs e)
{
    if(this.txtKilometraje.Text=="")
    {
        this.txtKilometraje.Text="0";
    }
    //realizar coneccion
    SqlConnection conectar=new SqlConnection(this.con.conexion);
    SqlDataReader drlector;
    conectar.Open();
    //seleccionar cliente externo (chofer)
    SqlCommand cmdPrincipal=new SqlCommand("select * from
tbl_mis_cliente_externo order by(cle_apellidos)",conectar);
    drlector=cmdPrincipal.ExecuteReader();
    while(drlector.Read())
    {
        this.ddlChofer.Items.Add(drlector["CLE_APELLIDOS"].ToString
        ()+" "+drlector["CLE_NOMBRES"].ToString());
    }
    this.ddlChofer.DataBind();
    drlector.Close();
    //seleccionar departamentos
    cmdPrincipal=new SqlCommand("select * from
tbl_ad_departamentos",conectar);
    drlector=cmdPrincipal.ExecuteReader();
    int c=0;
    while(drlector.Read())
    {
        if(c>0)
        {
            this.claveDepartamento=drlector["DEP_ID"].ToString();
            c++;
        }
        this.ddlDepartamento.Items.Add(drlector["DEP_ID"].
        ToString()+" "+drlector["DEP_NOMBRE"].ToString());
    }
    this.ddlDepartamento.DataBind();
    drlector.Close();
    conectar.Close();
    //insertar
    if(this.txtPlaca.Text!=""&&int.Parse(this.ddlAño.SelectedValue
    )>0 && this.txtUnidad.Text!="")
    {
        this.lblErrorNumero.Visible=false;
        this.lblErrorPlaca.Visible=false;
        conectar.Open();
        //verificar uinidad repetida
        cmdPrincipal=new SqlCommand("select * from
tbl_ad_vehiculo",conectar);
        drlector=cmdPrincipal.ExecuteReader();
        bool repetido=false;
        while(drlector.Read())
        {
```

```

        if(drlector["veh_id"].ToString()==this.txtUnidad.Text)
        {
            repetido=true;
        }
    }
drlector.Close();
if(repetido!=true)
{
    this.lblErrorRep.Visible=false;
    //sacar la id del departamento
    int i=0;
    string semiClave="";
    string letra="";
    semiClave=this.ddlDepartamento.Items[0].Value.ToString();
    while(letra!=" ")
    {
        letra=semiClave.Substring(i,1);
        this.claveDepartamento=this.claveDepartamento+letra;
        i++;
        this.claveDepartamento=this.claveDepartamento.TrimEnd();
    }
    //INSERTAR EN TABLA VEHICULOS
    cmdPrincipal = new SqlCommand("sp_ad_insertarvehiculo",
    conectar);
    cmdPrincipal.CommandType =
    System.Data.CommandType.StoredProcedure;
    SqlParameter p0 = new
    SqlParameter("@veh_id",System.Data.SqlDbType.VarChar);
    p0.Value = this.txtUnidad.Text;
    p0.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p0);
    SqlParameter p1 = new
    SqlParameter("@veh_marca",System.Data.SqlDbType.VarChar);
    p1.Value = this.txtMarca.Text;
    p1.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p1);
    SqlParameter p2 = new
    SqlParameter("@veh_modelo",System.Data.SqlDbType.VarChar);
    p2.Value = this.txtModelo.Text;
    p2.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p2);
    SqlParameter p3 = new
    SqlParameter("@veh_clase",System.Data.SqlDbType.VarChar);
    p3.Value = this.txtClase.Text;
    p3.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p3);
    SqlParameter p4 = new
    SqlParameter("@veh_tipo",System.Data.SqlDbType.VarChar);
    p4.Value = this.txtTipo.Text;
    p4.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p4);
    SqlParameter p5 = new
    SqlParameter("@veh_anio",System.Data.SqlDbType.VarChar);
    p5.Value = this.ddlAño.SelectedValue;
    p5.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p5);
    SqlParameter p6 = new
    SqlParameter("@veh_placa",System.Data.SqlDbType.VarChar);
    p6.Value = this.txtPlaca.Text;
}

```

```

        p6.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p6);
        SqlParameter p7 = new
        SqlParameter("@veh_color", System.Data.SqlDbType.VarChar);
        p7.Value = this.txtColor.Text;
        p7.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p7);
        SqlParameter p8 = new
        SqlParameter("@veh_no_motor", System.Data.SqlDbType.VarChar);
        p8.Value = this.txtMotor.Text;
        p8.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p8);
        SqlParameter p9 = new
        SqlParameter("@veh_no_chasis", System.Data.SqlDbType.VarChar);
        p9.Value = this.txtChasis.Text;
        p9.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p9);
        SqlParameter p10 = new
        SqlParameter("@veh_pais", System.Data.SqlDbType.VarChar);
        p10.Value = this.txtPais.Text;
        p10.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p10);
        SqlParameter p11 = new
        SqlParameter("@veh_chofer", System.Data.SqlDbType.VarChar);
        p11.Value = this.ddlChofer.SelectedValue;
        p11.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p11);
        SqlParameter p12 = new
        SqlParameter("@dep_id", System.Data.SqlDbType.VarChar);
        p12.Value = this.claveDepartamento;
        p12.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p12);
        SqlParameter p13 = new
        SqlParameter("@veh_km", System.Data.SqlDbType.Float);
        p13.Value = float.Parse(this.txtKilometraje.Text);
        p13.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p13);
        cmdPrincipal.ExecuteNonQuery();
        conectar.Close();

        Server.Transfer("frm_ad_consultarvehiculos.aspx", true);
    }
    else
    {
        this.lblErrorRep.Visible=true;
    }
}
else
{
    if(this.txtUnidad.Text=="")
    {
        this.lblErrorNumero.Visible=true;
    }
    else
    {
        this.lblErrorNumero.Visible=false;
    }
    if(this.txtPlaca.Text=="")
    {

```

```

        this.lblErrorPlaca.Visible=true;
    }
    else
    {
        this.lblErrorPlaca.Visible=false;
    }
}
}

```

- **frm\_ad\_mantenimientovehiculo**

### Variables Globales de la página

```

clase_inicio con=new clase_inicio();
BiblioPermisos.Class1 permisos=new BiblioPermisos.Class1();

```

### Funciones de usuario

Función para inicializar la página

```

private void inicializar_pagina()
{
    this.txtNumero.Enabled=true;
    this.txtObservaciones.Enabled=true;
    this.ddlDepartamento.Enabled=true;
    this.txtCantidad.Text="";
    this.txtDescripcion.Text="";
    this.txtKm.Text="";
    this.txtNumero.Text="";
    this.txtObservaciones.Text="";
    this.txtUnidad.Text="";
    this.ddlDescripcion.Items.Clear();
    this.ddlDepartamento.Items.Clear();
    this.ddlUnidad.Items.Clear();
    SqlConnection conectar=new SqlConnection(this.con.conexion);
    SqlCommand cmdPrincipal;
    SqlDataReader drlector;
    conectar.Open();
    DateTime f=DateTime.Parse(Session["fecha"].ToString());
    this.lblFecha.Text="Ambato "+f.Day+" de "+mes(f.Month)+" del
    "+f.Year;
    //llenar datos de departamentos
    cmdPrincipal=new SqlCommand("select * from
    tbl_ad_departamentos",conectar);
    drlector=cmdPrincipal.ExecuteReader();
    int c=0;
    while(drlector.Read())
    {
        if(c==0)
        {
            Session.Add("claveDep",drlector["DEP_ID"].ToString());
            c=1;
        }
        this.ddlDepartamento.Items.Add(drlector["DEP_ID"].ToString()
        +" "+drlector["DEP_NOMBRE"].ToString());
    }
}

```

```

this.ddlDepartamento.DataBind();
drlector.Close();
//llenar tipos de parametros
cmdPrincipal=new SqlCommand("select * from tbl_ad_parametro
where apr_clave='paraTipoSolicitud'",conectar);
drlector=cmdPrincipal.ExecuteReader();
while(drlector.Read())
{
    this.ddlTipo.Items.Add(drlector["apr_descripcion"].
    ToString());
}
this.ddlTipo.DataBind();
drlector.Close();
//llenar vehiculos segun departamento
cmdPrincipal=new SqlCommand("select * from tbl_ad_vehiculo
where veh_marca<>'maq00' and
dep_id="+Session["claveDep"].ToString(),conectar);
drlector=cmdPrincipal.ExecuteReader();
while(drlector.Read())
{
    this.ddlUnidad.Items.Add("Unidad
    "+drlector["veh_id"].ToString());
    this.txtKm.Text=drlector["veh_km"].ToString();
}
this.ddlDepartamento.DataBind();
drlector.Close();
//llenar datos de detalles de mantenimiento
cmdPrincipal=new SqlCommand("select * from tbl_ad_parametro
where apr_clave='paraManDet'",conectar);
drlector=cmdPrincipal.ExecuteReader();
c=0;
while(drlector.Read())
{
    if(c==0)
    {
        Session.Add("claveParametro",drlector["apr_id"].
        ToString());
        this.txtDescripcion.Text=drlector["apr_descripcion"].
        ToString();
        c=1;
    }
    this.ddlDescripcion.Items.Add(drlector["apr_descripcion"]
    .ToString());
}
this.ddlDepartamento.DataBind();
drlector.Close();
conectar.Close();
}

```

Función para cambiar el formato de fecha reducido al formato largo

```

private string mes(int varMes)
{
    string mesCompleto="";
    if(varMes==1)
    {
        mesCompleto="Enero";
    }
}

```

```

    }
    if (varMes==2)
    {
        mesCompleto="Febrero";
    }
    if (varMes==3)
    {
        mesCompleto="Marzo";
    }
    if (varMes==4)
    {
        mesCompleto="Abril";
    }
    if (varMes==5)
    {
        mesCompleto="Mayo";
    }
    if (varMes==6)
    {
        mesCompleto="Junio";
    }
    if (varMes==7)
    {
        mesCompleto="Julio";
    }
    if (varMes==8)
    {
        mesCompleto="Agosto";
    }
    if (varMes==9)
    {
        mesCompleto="Septiembre";
    }
    if (varMes==10)
    {
        mesCompleto="Octubre";
    }
    if (varMes==11)
    {
        mesCompleto="Noviembre";
    }
    if (varMes==12)
    {
        mesCompleto="Diciembre";
    }
    return mesCompleto;
}

```

## Funciones de la página

Función para añadir la solicitud de combustible a la Base de Datos

```

private void IBtnAñadir_Click(object sender,
System.Web.UI.ImageClickEventArgs e)
{
    if (this.txtCantidad.Text!=" " && this.txtCantidad.Text!="0")
    {
        this.lblErrorCantidad.Visible=false;
    }
}

```

```

this.txtObservaciones.Enabled=false;
this.ddlDepartamento.Enabled=false;
SqlConnection conectar=new SqlConnection(this.con.conexion);
SqlCommand cmdPrincipal;
string fecha;
fecha=Session["fecha"].ToString();
conectar.Open();
if(Session["insertarMan"].ToString()=="true")
{
    Session["insertarMan"]="false";
    //insertar mantenimiento
    cmdPrincipal = new SqlCommand("SP_AD_MANTENIMIENTO",
conectar);
    cmdPrincipal.CommandType =
System.Data.CommandType.StoredProcedure;
    SqlParameter p0 = new
SqlParameter("@MAN_REQ_URG",System.Data.SqlDbType.VarChar);
    p0.Value = "";
    p0.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p0);
    SqlParameter p1 = new
SqlParameter("@MAN_OBSERVACION",System.Data.SqlDbType
.VarChar);
    p1.Value = this.txtObservaciones.Text;
    p1.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p1);
    SqlParameter p2 = new
SqlParameter("@MAN_FECHA",System.Data.SqlDbType.
SmallDateTime);
    p2.Value = fecha;
    p2.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p2);
    SqlParameter p3 = new
SqlParameter("@DEP_ID",System.Data.SqlDbType.Int);
    p3.Value = Session["claveDep"].ToString();
    p3.Direction = ParameterDirection.Input;
    cmdPrincipal.Parameters.Add(p3);
    cmdPrincipal.ExecuteNonQuery();
}
//mostrar Numero
cmdPrincipal=new SqlCommand("SELECT max(man_id) as maximo
from tbl_ad_mantenimiento",conectar);
SqlDataReader drlector=cmdPrincipal.ExecuteReader();
if(drlector.Read())
{
    this.txtNumero.Text=drlector["maximo"].ToString();
}
drlector.Close();
//insertar mantenimiento detalle
cmdPrincipal = new
SqlCommand("SP_AD_MANTENIMIENTO_DET_SOLICITUD", conectar);
cmdPrincipal.CommandType =
System.Data.CommandType.StoredProcedure;
SqlParameter p00 = new
SqlParameter("@MAD_ACTIVIDAD",System.Data.SqlDbType.VarChar);
p00.Value = this.txtDescripcion.Text;
p00.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p00);

```

```

SqlParameter p01 = new
SqlParameter("@MAD_FECHA_PROG", System.Data.SqlDbType.SmallDa
teTime);
p01.Value = fecha;
p01.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p01);
SqlParameter p02 = new
SqlParameter("@MAD_KM", System.Data.SqlDbType.Float);
p02.Value = this.txtKm.Text;
p02.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p02);
SqlParameter p03 = new
SqlParameter("@MAN_ID", System.Data.SqlDbType.BigInt);
p03.Value = this.txtNumero.Text;
p03.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p03);
SqlParameter p04 = new
SqlParameter("@VEH_ID", System.Data.SqlDbType.VarChar);
p04.Value = this.ddlUnidad.SelectedValue.ToString()
.Substring(7, this.ddlUnidad.SelectedValue.ToString()
.Length-7).Trim();
p04.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p04);
SqlParameter p05 = new
SqlParameter("@CANTIDAD", System.Data.SqlDbType.Float);
p05.Value = this.txtCantidad.Text;
p05.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p05);
SqlParameter p06 = new
SqlParameter("@UNIDAD", System.Data.SqlDbType.VarChar);
p06.Value = this.txtUnidad.Text;
p06.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p06);
SqlParameter p07 = new
SqlParameter("@CLAVEPARAMETRO", System.Data.SqlDbType.Int);
p07.Value = Session["claveParametro"].ToString();
p07.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p07);
SqlParameter p08 = new
SqlParameter("@MAD_ESTADO_ANTES", System.Data.SqlDbType
.VarChar);
p08.Value = this.ddlTipo.SelectedValue.ToString();
p08.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p08);
cmdPrincipal.ExecuteNonQuery();
this.dgDetalle.Visible=true;
//llenar grid de detalles de la presente solicitud
SqlDataAdapter daManVista=new SqlDataAdapter(
"Select vehiculo,cantidad,unidad,descripcion from
VW_AD_MANTENIMIENTO_SOLICITUD where
#SOLICITUD='"+this.txtNumero.Text+"'", conectar);
DataSet dsVehiculos=new DataSet();
daManVista.Fill(dsVehiculos, "VW_AD_MANTENIMIENTO_SOLICITUD");
this.dgDetalle.DataSource=dsVehiculos;
this.dgDetalle.DataMember="VW_AD_MANTENIMIENTO_SOLICITUD";
this.dgDetalle.DataBind();
conectar.Close();
this.txtObservaciones.Text="";
this.txtCantidad.Text="";

```

```

        this.txtUnidad.Text="";
    }
    else
    {
        if(this.txtCantidad.Text=="")
        {
            this.lblErrorCantidad.Visible=true;
        }
        else
        {
            if(this.txtCantidad.Text=="0")
            {
                this.lblErrorCantidad.Visible=true;
            }
            else
            {
                this.lblErrorCantidad.Visible=false;
            }
        }
    }
}
}
}

```

- **frm\_ad\_mantenimiento\_factura**

### Variables Globales de la página

```

clase_inicio con=new clase_inicio();
BiblioPermisos.Class1 permisos=new BiblioPermisos.Class1();

```

### Funciones de la página

```

private void ibtnAceptarFactura_Click(object sender,
System.Web.UI.ImageClickEventArgs e)
{
    if(this.txtNumero.Text!=" " && this.txtCosto.Text!=" " &&
this.txtCosto.Text!="0")
    {
        this.lblErrorCosto.Visible=false;
        this.lblErrorNumero.Visible=false;
        DateTime fecha=DateTime.Parse(Session["fecha"].ToString());
        SqlConnection conectar=new SqlConnection(this.con.conexion);
        SqlCommand cmdPrincipal;
        conectar.Open();
        cmdPrincipal = new
        SqlCommand("SP_AD_ACTUALIZAR_MANTENIMIENTO_DET", conectar);
        cmdPrincipal.CommandType =
        System.Data.CommandType.StoredProcedure;
        SqlParameter p0 = new
        SqlParameter("@mad_fecha_eje",System.Data.SqlDbType.SmallDat
eTime);
        p0.Value = fecha.ToShortDateString();
        p0.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p0);
        SqlParameter p1 = new
        SqlParameter("@cle_ci_ruc",System.Data.SqlDbType.VarChar);
        p1.Value = (this.lbxProveedores.Items[this.ddlProveedor

```

```

.SelectedIndex]).Text;
p1.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p1);
SqlParameter p2 = new
SqlParameter("@mad_costo",System.Data.SqlDbType.Float);
p2.Value = this.txtCosto.Text;
p2.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p2);
SqlParameter p3 = new
SqlParameter("@mad_factura",System.Data.SqlDbType.Int);
p3.Value = this.txtNumero.Text;
p3.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p3);
SqlParameter p4 = new SqlParameter("@mad_id
",System.Data.SqlDbType.Int);
p4.Value =
this.dgVistaDetalleMantenimiento.SelectedItem.Cells[5].Text;
p4.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p4);
cmdPrincipal.ExecuteNonQuery();
//verificar factura vacia
bool retornar=true;
cmdPrincipal = new SqlCommand("SELECT
TBL_AD_MANTENIMIENTO_DET.VEH_ID AS
Vehiculo,TBL_AD_MANTENIMIENTO_DET.CANTIDAD AS
Cant,TBL_AD_MANTENIMIENTO_DET.UNIDAD AS
Unidad,TBL_AD_MANTENIMIENTO_DET.MAD_ACTIVIDAD AS
Actividad,TBL_AD_MANTENIMIENTO_DET.MAD_ID AS id FROM
TBL_AD_MANTENIMIENTO INNER JOIN TBL_AD_MANTENIMIENTO_DET ON
TBL_AD_MANTENIMIENTO.MAN_ID =
TBL_AD_MANTENIMIENTO_DET.MAN_ID WHERE
(TBL_AD_MANTENIMIENTO.MAN_ID =
'+Session["idMan"].ToString()+'' ) AND
(dbo.TBL_AD_MANTENIMIENTO_DET.MAD_FACTURA IS NULL)",
conectar);
cmdPrincipal.CommandType = System.Data.CommandType.Text;
SqlDataReader drlector=cmdPrincipal.ExecuteReader();
if(drlector.Read())
{
    retornar=false;
}
drlector.Close();
SqlDataAdapter daVehiculosVista=new SqlDataAdapter("
SELECT TBL_AD_MANTENIMIENTO_DET.VEH_ID AS
Vehiculo,TBL_AD_MANTENIMIENTO_DET.CANTIDAD AS
Cant,TBL_AD_MANTENIMIENTO_DET.UNIDAD AS
Unidad,TBL_AD_MANTENIMIENTO_DET.MAD_ACTIVIDAD AS
Actividad,TBL_AD_MANTENIMIENTO_DET.MAD_ID AS id FROM
TBL_AD_MANTENIMIENTO INNER JOIN TBL_AD_MANTENIMIENTO_DET ON
TBL_AD_MANTENIMIENTO.MAN_ID =
TBL_AD_MANTENIMIENTO_DET.MAN_ID WHERE
(TBL_AD_MANTENIMIENTO.MAN_ID =
'+Session["idMan"].ToString()+'' ) AND
(dbo.TBL_AD_MANTENIMIENTO_DET.MAD_FACTURA IS
NULL)",conectar);
DataSet dsVehiculos=new DataSet();
daVehiculosVista.Fill(dsVehiculos,"vw_ad_VEICULOS");
this.dgVistaDetalleMantenimiento.DataSource=dsVehiculos;
this.dgVistaDetalleMantenimiento.DataBind();

```

```

this.dgVistaDetalleMantenimiento.SelectedIndex=0;
conectar.Close();
if(retornar==true)
{
    editarVisible(false);
    daVehiculosVista=new SqlDataAdapter("Select
Numero_Solicitud,Fecha,Departamento,Observaciones from
vw_ad_MantenimientosSolicitados",conectar);
    dsVehiculos=new DataSet();
    daVehiculosVista.Fill(dsVehiculos,"vw_ad_VEHICULOS");
    this.dgVistaVehiculos.DataSource=dsVehiculos;
    this.dgVistaVehiculos.DataMember="vw_ad_VEHICULOS";
    this.dgVistaVehiculos.DataBind();
    this.dgVistaVehiculos.Visible=true;
    this.txtCosto.Text="";
    this.txtNumero.Text="";
}
this.txtCosto.Text="";
this.txtNumero.Text="";
}
else
{
    if(this.txtCosto.Text==" || this.txtCosto.Text=="0")
    {
        this.lblErrorCosto.Visible=true;
    }
    else
    {
        this.lblErrorCosto.Visible=false;
    }
    if(this.txtNumero.Text=="")
    {
        this.lblErrorNumero.Visible=true;
    }
    else
    {
        this.lblErrorNumero.Visible=false;
    }
}
}
}

```

- **frmad\_tanqueovehiculo**

### **Variables Globales de la página**

```

clase_inicio con=new clase_inicio();
BiblioPermisos.Class1 permisos=new BiblioPermisos.Class1();

```

### **Funciones de la Página**

Función para agregar el despacho a la Base de datos

```

private void IBtnAñadir_Click(object sender,
System.Web.UI.ImageClickEventArgs e)
{
    DateTime f=DateTime.Parse(Session["fecha"].ToString());
}

```

```

if(this.txtNumero.Text!="")
{
    this.lblErrorNumero.Visible=false;
    if(this.TextBox9.Text=="")
    {
        this.txtValor.Text="0";
    }
    if(this.ddlTipo.SelectedValue=="Super")
    {
        this.txtValor.Text=(float.Parse(Session["precio_super"]
        .ToString())*float.Parse(this.TextBox9.Text)).ToString();
    }
    else
    {
        if(this.ddlTipo.SelectedValue=="Extra")
        {
            this.txtValor.Text=(float.Parse(Session["precio_extra"]
            .ToString())*float.Parse(this.TextBox9.Text)).ToString();
        }
        else
        {
            this.txtValor.Text=(float.Parse(Session["precio_diesel"]
            .ToString())*float.Parse(this.TextBox9.Text)).ToString();
        }
    }
}
if(this.txtTotal.Text!="0")
{
    this.lblTotal.Visible=false;
    this.txtTotal.Text=(float.Parse(this.txtValor.Text)
    .ToString());
    SqlConnection conectar=new SqlConnection(this.con.conexion);
    SqlDataAdapter daVehiculosVista;
    SqlCommand cmdPrincipal;
    SqlDataReader drlector;
    int i=0;
    string semiClave="";
    string letra="";
    string desp=Session["despacho"].ToString();
    //sacar veh_id
    if(this.ddlUnidad.SelectedValue.ToString().Substring(0,3)==
    "maq")
    {
        this.claveVeh=this.ddlUnidad.SelectedValue.ToString();
    }
    else
    {
        this.claveVeh=this.ddlUnidad.SelectedValue.ToString()
        .Substring(6,this.ddlUnidad.SelectedValue.ToString()
        .Length-6).TrimStart();
    }
    conectar.Open();
    //controlar kilometraje
    if(this.txtKm.Text=="")
    {
        this.txtKm.Text="0";
    }
    cmdPrincipal=new SqlCommand("select * from
    tbl_ad_vehiculo where veh_id='"+claveVeh+"'",conectar);
    drlector=cmdPrincipal.ExecuteReader();
}

```

```

bool kilometraje=true;
if(drlector.Read())
{
if(drlector["veh_km"].ToString()=="")
{
if(float.Parse(this.txtKm.Text)<=0)
{
kilometraje=false;
}
}
else
{
if(float.Parse(this.txtKm.Text)<=float.Parse
(drlector["veh_km"].ToString()))
{
kilometraje=false;
}
}
}
drlector.Close();
conectar.Close();
if(kilometraje==true ||
this.ddlUnidad.SelectedValue.ToString().Substring(0,3)=="maq")
{
if(desp=="false")
{
conectar.Open();
//insertar despacho
cmdPrincipal = new SqlCommand("sp_ad_insertardespacho",
conectar);
cmdPrincipal.CommandType =
System.Data.CommandType.StoredProcedure;
SqlParameter p0 = new
SqlParameter("@dco_id",System.Data.SqlDbType.BigInt);
p0.Value = this.txtNumero.Text;
p0.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p0);
SqlParameter p1 = new
SqlParameter("@dco_fecha",System.Data.SqlDbType
.SmallDateTime);
p1.Value = f.ToShortDateString();
p1.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p1);
SqlParameter p3 = new
SqlParameter("@dco_precio_extra",System.Data.SqlDbType
.Float);
p3.Value = Math.Round(float.Parse(Session["precio_extra"]
.ToString()),2);
p3.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p3);
SqlParameter p4 = new
SqlParameter("@dco_precio_super",System.Data.SqlDbType
.Float);
p4.Value = Math.Round(float.Parse(Session["precio_super"]
.ToString()),2);
p4.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p4);
SqlParameter p5 = new
SqlParameter("@dco_precio_die",System.Data.SqlDbType

```

```

.Float());
p5.Value = Math.Round(float.Parse(Session[
"precio_diesel"].ToString()),2);
p5.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p5);
//seleccionar codigo de gasolinera
i=0;
semiClave="";
letra="";
string claveGas="";
semiClave=this.ddlGasolinera.SelectedValue.ToString();
while(letra!=" ")
{
    letra=semiClave.Substring(i,1);
    claveGas=claveGas+letra;
    i++;
    claveGas=claveGas.TrimEnd();
}
SqlParameter p2 = new SqlParameter("@dco_gasolinera"
, System.Data.SqlDbType.Int);
p2.Value = int.Parse(claveGas);
p2.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p2);
cmdPrincipal.ExecuteNonQuery();
conectar.Close();
Session["despacho"]="true";
}
//insertar despacho detalle
conectar.Open();
cmdPrincipal = new SqlCommand(
"SP_AD_INSERTARDESPACHO_DETALLE", conectar);
cmdPrincipal.CommandType = System.Data.CommandType
.StoredProcedure;
SqlParameter p6 = new SqlParameter("@dco_id", System
.Data.SqlDbType.BigInt);
p6.Value = this.txtNumero.Text;
p6.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p6);
SqlParameter p7 = new SqlParameter("@CLE_CI_RUC",
System.Data.SqlDbType.VarChar);
p7.Value = this.ddlChofer.SelectedValue.ToString();
p7.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p7);
SqlParameter p8 = new SqlParameter("@DCD_KM", System.Data
.SqlDbType.Float);
p8.Value = float.Parse(this.txtKm.Text);
p8.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p8);
SqlParameter p9 = new
SqlParameter("@DCD_CANTIDAD", System.Data.SqlDbType.Float);
p9.Value =
System.Math.Round(float.Parse(this.TextBox9.Text),2);
p9.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p9);
SqlParameter p10 = new
SqlParameter("@DCD_TIPO", System.Data.SqlDbType.VarChar);
p10.Value = this.ddlTipo.SelectedValue.ToString();
p10.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p10);

```

```

SqlParameter p11 = new
SqlParameter("@DCO_OTRO", System.Data.SqlDbType.VarChar);
p11.Value = "";
p11.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p11);
SqlParameter p12 = new
SqlParameter("@VEH_ID", System.Data.SqlDbType.VarChar);
p12.Value = claveVeh;
p12.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p12);
SqlParameter p13 = new
SqlParameter("@dcd_hora", System.Data.SqlDbType.VarChar);
p13.Value = "12:00";
p13.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p13);
SqlParameter p14 = new
SqlParameter("@dcd_valor", System.Data.SqlDbType.Float);
p14.Value = Math.Round(float.Parse(this.txtValor.Text), 2);
p14.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p14);
SqlParameter p15 = new
SqlParameter("@dcd_total", System.Data.SqlDbType.Float);
p15.Value = Math.Round(float.Parse(this.txtTotal.Text), 2);
p15.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p15);
cmdPrincipal.ExecuteNonQuery();
//MODIFICAR KILOMETRAJE
cmdPrincipal = new
SqlCommand("SP_AD_ACTUALIZAR_KILOMETRAJE", conectar);
cmdPrincipal.CommandType =
System.Data.CommandType.StoredProcedure;
SqlParameter p001 = new
SqlParameter("@VEH_id", System.Data.SqlDbType.VarChar);
p001.Value = claveVeh;
p001.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p001);
SqlParameter p002 = new
SqlParameter("@VEH_KM", System.Data.SqlDbType.Float);
p002.Value = this.txtKm.Text;
p002.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p002);
cmdPrincipal.ExecuteNonQuery();
this.ddlGasolinera.Enabled=false;
this.txtNumero.Enabled=false;
daVehiculosVista=new SqlDataAdapter("Select
unidad,hora,chofer,km,tipo,galones,usd,total from
w_ad_DESPACHO_DETALLE where
NUM='"+this.txtNumero.Text+"'", conectar);
DataSet dsVehiculos=new DataSet();
daVehiculosVista.Fill(dsVehiculos, "vw_ad_DESPACHO_DETALLE");
this.dgDetalle.DataSource=dsVehiculos;
this.dgDetalle.DataMember="vw_ad_DESPACHO_DETALLE";
this.dgDetalle.DataBind();
conectar.Close();
//blanquear edits
this.TextBox9.Text="0";
this.txtValor.Text="0";
this.txtTotal.Text="0";
this.lblErrorKm.Visible=false;

```

```

}
else
{
    this.lblErrorKm.Visible=true;
}
}
else
{
    this.lblTotal.Visible=true;
}
}
else
{
    this.lblErrorNumero.Visible=true;
}
}
}

```

- **frm\_ad\_solicitudvehiculo**

### Variables Globales de la página

```

clase_inicio con=new clase_inicio();
BiblioPermisos.Class1 permisos=new BiblioPermisos.Class1();

```

### Funciones del sistema

```

private void ibtnAceptar_Click(object sender,
System.Web.UI.ImageClickEventArgs e)
{
    if(this.txtCantidad.Text!=" " && this.txtNumero.Text!=" ")
    {
        //seleccionar codigo de gasolinera
        int i=0;
        string semiClave="";
        string letra="";
        string claveGas="";
        semiClave=this.ddlGasolinera.SelectedValue.ToString();
        while(letra!=" ")
        {
            letra=semiClave.Substring(i,1);
            claveGas=claveGas+letra;
            i++;
            claveGas=claveGas.TrimEnd();
        }
        SqlConnection conectar=new SqlConnection(this.con.conexion);
        SqlCommand cmdPrincipal;
        conectar.Open();
        //insertar despacho
        cmdPrincipal = new SqlCommand("SP_AD_INSERTARSOLICITUD",
conectar);
        cmdPrincipal.CommandType =
System.Data.CommandType.StoredProcedure;
        SqlParameter p1 = new SqlParameter
        ("@SCO_FECHA",System.Data.SqlDbType.SmallDateTime);
        p1.Value = Session["fecha"].ToString();
        p1.Direction = ParameterDirection.Input;
    }
}

```

```

cmdPrincipal.Parameters.Add(p1);
SqlParameter p3 = new
SqlParameter("@CLE_CI_RUC",System.Data.SqlDbType.VarChar);
p3.Value = this.ddlConductor.SelectedValue.ToString();
p3.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p3);
SqlParameter p4 = new
SqlParameter("@VEH_ID",System.Data.SqlDbType.VarChar);
p4.Value = Session["unidad"].ToString();
p4.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p4);
SqlParameter p5 = new SqlParameter("@SCO_DIRECTOR_
ADMINISTRATIVO",System.Data.SqlDbType.VarChar);
p5.Value = "yo";
p5.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p5);
SqlParameter p2 = new
SqlParameter("@GAS_ID",System.Data.SqlDbType.Int);
p2.Value = int.Parse(claveGas);
p2.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p2);
SqlParameter p6 = new
SqlParameter("@SCO_NUMERO",System.Data.SqlDbType.BigInt);
p6.Value = this.txtNumero.Text;
p6.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p6);
cmdPrincipal.ExecuteNonQuery();
//SELECCIONAR ID DE SOLICITUD
SqlDataReader drlector;
string f=(Session["fecha"].ToString());
cmdPrincipal=new SqlCommand("select * from
tbl_ad_SOLICITUD_COMBUSTIBLE where SCO_NUMERO
="+this.txtNumero.Text+" AND SCO_FECHA='"+f+"'",conectar);
string claveSCO="";
drlector=cmdPrincipal.ExecuteReader();
if(drlector.Read())
{
    claveSCO=drlector["SCO_ID"].ToString();
}
drlector.Close();
//seleccionar precio de combustible
string precioCombustible="";
if(this.ddlTipo.SelectedValue.ToString()=="Super")
{
    precioCombustible=this.lblSuper.Text;
}
else
{
    if(this.ddlTipo.SelectedValue.ToString()=="Extra")
    {
        precioCombustible=this.lblExtra.Text;
    }
    else
    {
        precioCombustible=this.lblDiesel.Text;
    }
}
//detalle solicitud
cmdPrincipal = new SqlCommand("SP_AD

```

```

_INSERTARSOLICITUDETALLE", conectar);
cmdPrincipal.CommandType =
System.Data.CommandType.StoredProcedure;
SqlParameter p7 = new
SqlParameter("@SCO_ID", System.Data.SqlDbType.VarChar);
p7.Value = claveSCO;
p7.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p7);
SqlParameter p8 = new SqlParameter
("@SCD_DESCRIPCION", System.Data.SqlDbType.VarChar);
p8.Value = "";
p8.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p8);
SqlParameter p9 = new
SqlParameter("@SCD_UNIDAD", System.Data.SqlDbType.VarChar);
p9.Value = this.ddlTipo.SelectedValue.ToString();
p9.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p9);
SqlParameter p10 = new
SqlParameter("@SCD_PRECIO", System.Data.SqlDbType.Float);
p10.Value = Math.Round(float.Parse(precioCombustible), 2);
p10.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p10);
SqlParameter p11 = new
SqlParameter("@SCD_CANTIDAD", System.Data.SqlDbType.Float);
p11.Value = float.Parse(this.txtCantidad.Text);
p11.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p11);
SqlParameter p12 = new
SqlParameter("@SCD_TOTAL", System.Data.SqlDbType.Float);
p12.Value =
Math.Round(float.Parse(this.txtCantidad.Text) * float.Parse(pr
ecioCombustible), 2);
p12.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p12);
cmdPrincipal.ExecuteNonQuery();
conectar.Close();
editarVisible(false);
this.dgVistaVehiculos.Visible=true;
this.lblErrorCantidad.Visible=false;
this.lblErrorNumero.Visible=false;
this.txtNumero.Text="";
this.txtCantidad.Text="";
this.lblTotal.Text="0";
}
else
{
    if(this.txtCantidad.Text=="")
    {
        this.lblErrorCantidad.Visible=true;
    }
    else
    {
        this.lblErrorCantidad.Visible=false;
    }
    if(this.txtNumero.Text=="")
    {
        this.lblErrorNumero.Visible=true;
    }
}
}

```

```

        else
        {
            this.lblErrorNumero.Visible=false;
        }
    }
}

```

- **frm\_ad\_vehSalvoconductos**

### **Variables Globales de la página**

```

clase_inicio con=new clase_inicio();
BiblioPermisos.Class1 permisos=new BiblioPermisos.Class1();

```

### **Funciones del programa**

#### **Función para ingresar un nuevo Salvoconducto**

```

private void ibtnAceptar_Click(object sender,
System.Web.UI.ImageClickEventArgs e)
{
    if(this.txtDestino.Text!=" " &&
this.txtNumero.Text!=" "&&this.txtProcedencia.Text!=" "&&this.txt
Razon.Text!=" ")
    {
        DateTime fechaini=new
DateTime(this.Calendar1.SelectedDate.Year,this.Calendar1.Sel
ectedDate.Month,this.Calendar1.SelectedDate.Day);
        DateTime fechafin=new
DateTime(this.Calendar2.SelectedDate.Year,this.Calendar2.Sel
ectedDate.Month,this.Calendar2.SelectedDate.Day);
        this.lblErrorDestino.Visible=false;
        this.lblErrorNumero.Visible=false;
        this.lblErrorRazon.Visible=false;
        this.lblErrorProcedencia.Visible=false;
        //realizar coneccion
        SqlConnection conectar=new SqlConnection(this.con.conexion);
        conectar.Open();
        //insertar salvoconducto
        SqlCommand cmdPrincipal = new
SqlCommand("SP_AD_ACT_VEHICULOLOSSALVOCONDUCTO",conectar);
        cmdPrincipal.CommandType =
System.Data.CommandType.StoredProcedure;
        SqlParameter p0 = new
SqlParameter("@ASC_FECHA_EMISION",System.Data.SqlDbType.Small
DateTime);
        p0.Value = fechaini.ToShortDateString();
        p0.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p0);
        SqlParameter p1 = new
SqlParameter("@ASC_FECHA_CADUCIDAD",System.Data.SqlDbType.Sm
allDateTime);
        p1.Value = fechafin.ToShortDateString();
        p1.Direction = ParameterDirection.Input;
        cmdPrincipal.Parameters.Add(p1);
        SqlParameter p2 = new
SqlParameter("@VEH_ID",System.Data.SqlDbType.VarChar);

```

```

p2.Value = Session["unidad"].ToString();
p2.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p2);
SqlParameter p3 = new SqlParameter
("@ASC_ACTIVIDAD", System.Data.SqlDbType.VarChar);
p3.Value = this.txtRazon.Text;
p3.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p3);
SqlParameter p4 = new SqlParameter
("@ASC_RESPONSABLE", System.Data.SqlDbType.VarChar);
p4.Value = Session["cle_ci_ruc"].ToString();
p4.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p4);
SqlParameter p5 = new
SqlParameter("@ASC_DIRECTOR_ADMINISTRATIVO
", System.Data.SqlDbType.VarChar);
p5.Value = "--";
p5.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p5);
SqlParameter p6 = new SqlParameter
("@ASC_PROCEDENCIA", System.Data.SqlDbType.VarChar);
p6.Value = this.txtProcedencia.Text;
p6.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p6);
SqlParameter p7 = new SqlParameter
("@ASC_DESTINO", System.Data.SqlDbType.VarChar);
p7.Value = this.txtDestino.Text;
p7.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p7);
SqlParameter p8 = new SqlParameter
("@ASC_FECHA", System.Data.SqlDbType.SmallDateTime);
p8.Value = DateTime.Parse
(Session["fechaSistema"].ToString()).ToShortDateString();
p8.Direction = ParameterDirection.Input;
cmdPrincipal.Parameters.Add(p8);
cmdPrincipal.ExecuteNonQuery();
conectar.Close();
this.ddlConductor.Items.Clear();
this.dgVistaVehiculos.Visible=true;
editarVisible(false);
}
else
{
    if(this.txtNumero.Text=="")
    {
        this.lblErrorNumero.Visible=true;
    }
    if(this.txtDestino.Text=="")
    {
        this.lblErrorDestino.Visible=true;
    }
    if(this.txtRazon.Text=="")
    {
        this.lblErrorRazon.Visible=true;
    }
    if(this.txtProcedencia.Text=="")
    {
        this.lblErrorProcedencia.Visible=true;
    }
}
}

```

```
}
```

- **frm\_ad\_reportes**

Función para llenar los datos de reportes según el parámetro enviado

```
private void llenar_grid()
{
    string idProv="";
    string cadena="";
    string letra="";
    if(this.ddlProveedores.SelectedValue.ToString()!="todos")
    {
        int k=0;
        while(letra!=" ")
        {
            letra=this.ddlProveedores.SelectedValue.Substring(k,1);
            cadena=cadena+letra;
            k=k+1;
        }
        cadena.TrimEnd();
        idProv=cadena;
    }
    this.lblProveedor.Text=this.
    lblProveedor.Text+this.ddlProveedores.SelectedValue.ToString();
    DateTime fechaini=new DateTime(this.Calendar1.SelectedDate
    .Year,this.Calendar1.SelectedDate.Month
    ,this.Calendar1.SelectedDate.Day);
    DateTime fechafin=new
    DateTime(this.Calendar2.SelectedDate.Year,this.Calendar2
    .SelectedDate.Month,this.Calendar2.SelectedDate.Day);
    //variables de seleccion de datos
    SqlConnection conectar=new SqlConnection(this.con.conexion);
    SqlCommand cmdPrincipal;
    SqlDataReader drlector;
    //variables de columnas y filas
    DataTable tablaCombustible = new DataTable("Mantenimiento");
    DataTable tablaGeneral = new DataTable("General");
    DataTable tablaGeneral2 = new DataTable("General");
    DataTable tablaGeneral3 = new DataTable("General");
    DataTable tablaGeneral4 = new DataTable("General");
    DataColumn dtCol;
    DataRow dtRow;
    //llenar tabla departamentos
    conectar.Open();
    cmdPrincipal = new SqlCommand("Select * from
    tbl_ad_departamentos", conectar);
    cmdPrincipal.CommandType = System.Data.CommandType.Text;
    drlector=cmdPrincipal.ExecuteReader();
    while(drlector.Read())
    {
        this.lbxDepId.Items.Add(drlector["dep_id"].ToString());
        this.lbxParNumero.Items.Add(drlector["dep_partida"]
        .ToString());
        this.lbxDepNombre.Items.Add(drlector["dep_nombre"]
        .ToString());
        this.lbxGenLubricantes.Items.Add("0");
        this.lbxGenManoObra.Items.Add("0");
    }
}
```

```

        this.lbxGenRepuestos.Items.Add("0");
        this.lbxGenOtros.Items.Add("0");
    }
    drlector.Close();
    //seleccionar vehiculos y departamentos
    cmdPrincipal = new SqlCommand("Select * from tbl_ad_vehiculo
    where veh_marca<>'maq00' or veh_marca is null", conectar);
    cmdPrincipal.CommandType = System.Data.CommandType.Text;
    drlector=cmdPrincipal.ExecuteReader();
    while(drlector.Read())
    {
        this.lbxVehiculos.Items.Add(drlector["veh_id"].ToString());
        this.lbxDepartamentos.Items.Add(drlector["dep_id"]
        .ToString());
    }
    drlector.Close();
    //seleccionar partidas
    for(int i=0;i<=this.lbxVehiculos.Items.Count-1;i++)
    {
        cmdPrincipal = new SqlCommand("Select * from tbl_ad_
        departamentos where dep_id="+this.lbxDepartamentos
        .Items[i].Text, conectar);
        cmdPrincipal.CommandType = System.Data.CommandType.Text;
        drlector=cmdPrincipal.ExecuteReader();
        while(drlector.Read())
        {
            this.lbxPartidas.Items.Add(drlector
            ["dep_partida"].ToString());
        }
        drlector.Close();
    }
    //seleccionar usd clasificados por tipo
    float sumaLubricantes=0;
    float sumaManoObra=0;
    float sumaRepuestos=0;
    float sumaOtros=0;
    float valorSuma=0;
    for(int i=0;i<=this.lbxVehiculos.Items.Count-1;i++)
    {
        if(this.ddlProveedores.Selected.Value.ToString()=="todos")
        {
            cmdPrincipal = new SqlCommand("SELECT TBL_AD_
            MANTENIMIENTO_DET.MAD_COSTO AS valor,MAD_ESTADO_ANTES
            as tipo,DEP_ID as dep FROM TBL_AD_MANTENIMIENTO INNER
            JOIN TBL_AD_MANTENIMIENTO_DET ON TBL_AD_MANTENIMIENTO
            .MAN_ID = TBL_AD_MANTENIMIENTO_DET.MAN_ID where
            dbo.TBL_AD_MANTENIMIENTO.man_fecha>='"+fechaini
            .ToShortDateString()+"' and BL_AD_MANTENIMIENTO.man_fecha
            <='"+fechafin.ToShortDateString()+"' and
            veh_id='"+this.lbxVehiculos.Items[i].Text+"'", conectar);
        }
        else
        {
            cmdPrincipal = new SqlCommand("SELECT
            TBL_AD_MANTENIMIENTO_DET.MAD_COSTO AS
            valor,MAD_ESTADO_ANTES as tipo,DEP_ID as dep FROM
            TBL_AD_MANTENIMIENTO INNER JOIN
            TBL_AD_MANTENIMIENTO_DET ON
            TBL_AD_MANTENIMIENTO.MAN_ID =

```

```

TBL_AD_MANTENIMIENTO_DET.MAN_ID where
dbo.TBL_AD_MANTENIMIENTO.man_fecha>='"+fechaini.ToShortDate
ToString()+"' and
TBL_AD_MANTENIMIENTO.man_fecha<='"+fechafin.ToShortDateSt
ring()+"' and
veh_id='"+this.lbxVehiculos.Items[i].Text+"' and
cle_ci_ruc='"+idProv+"'", conectar);
}
cmdPrincipal.CommandType = System.Data.CommandType.Text;
drlector=cmdPrincipal.ExecuteReader();
while(drlector.Read())
{
    if(drlector["valor"].ToString()=="")
    {
        valorSuma=0;
    }
    else
    {
        valorSuma=float.Parse(drlector["valor"].ToString());
    }
    if(drlector["tipo"].ToString().ToUpper()=="LUBRICANTES")
    {
        sumaLubricantes=sumaLubricantes+valorSuma;
    }
    else
    {
        if(drlector["tipo"].ToString().ToUpper()=="MANO
DE OBRA")
        {
            sumaManoObra=sumaManoObra+valorSuma;
        }
        else
        {
            if(drlector["tipo"].ToString().ToUpper()
=="REPUESTOS")
            {
                sumaRepuestos=sumaRepuestos+valorSuma;
            }
            else
            {
                sumaOtros=sumaOtros+valorSuma;
            }
        }
    }
}
for(int j=0;j<=this.lbxDepId.Items.Count-1;j++)
{
    if(this.lbxDepId.Items[j].Text
==drlector["dep"].ToString())
    {
        if(drlector["tipo"]. ToString().ToUpper()
=="LUBRICANTES")
        {
            this.lbxGenLubricantes.Items[j].Text
=(float.Parse(this.lbxGenLubricantes
.Items[j].Text)+valorSuma).ToString(
);
        }
    }
    else
    {

```

```

        if(drlector["tipo"].ToString().ToUpper()=="
        "MANO DE OBRA")
        {
            this.lbxGenManoObra.Items[j].Text
            =(float.Parse(this.lbxGenManoObra.It
            ems[j].Text)+valorSuma).ToString();
        }
        else
        {
            if(drlector["tipo"].ToString().ToUpp
            er()==" REPUESTOS")
            {
                this.lbxGenRepuestos.Items[j].
                Text=
                (float.Parse(this.lbxGenRepues
                tos.Items[j].Text)+valorSuma).
                ToString();
            }
            else
            {
                this.lbxGenOtros.Items[j].Text
                =
                (float.Parse(this.lbxGenOtros.
                Items[j].Text)+valorSuma).ToSt
                ring();
            }
        }
    }
}
drlector.Close();

this.lbxSumaLubricantes.Items.Add(sumaLubricantes.ToString()
);
this.lbxSumaManoObra.Items.Add(sumaManoObra.ToString());
this.lbxSumaRepuestos.Items.Add(sumaRepuestos.ToString());
this.lbxSumaOtros.Items.Add(sumaOtros.ToString());
sumaLubricantes=0;
sumaManoObra=0;
sumaRepuestos=0;
sumaOtros=0;
}
//columnas
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Unidad";
dtCol.Caption = "Unidad";
dtCol.ReadOnly = true;
tablaCombustible.Columns.Add(dtCol);
//2
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Mano de Obra";
dtCol.Caption = "Mano de Obra";
dtCol.ReadOnly = true;
tablaCombustible.Columns.Add(dtCol);
//3
dtCol = new DataColumn();

```

```

dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Lubricantes";
dtCol.Caption = "Lubricantes";
dtCol.ReadOnly = true;
tablaCombustible.Columns.Add(dtCol);
//4
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Repuestos";
dtCol.Caption = "Repuestos";
dtCol.ReadOnly = true;
tablaCombustible.Columns.Add(dtCol);
//5
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Otros";
dtCol.Caption = "Otros";
dtCol.ReadOnly = true;
tablaCombustible.Columns.Add(dtCol);
//6
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Total";
dtCol.Caption = "Total";
dtCol.ReadOnly = true;
tablaCombustible.Columns.Add(dtCol);
//6
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Partidas";
dtCol.Caption = "Partidas";
dtCol.ReadOnly = true;
tablaCombustible.Columns.Add(dtCol);
//filas
float ValorParcial=0;
float ValorTotal=0;
for(int i=0;i<=this.lbxVehiculos.Items.Count-1;i++)
{
    ValorParcial=float.Parse(this.lbxSumaManoObra.Items[i].Text)
    +float.Parse(this.lbxSumaLubricantes.Items[i].Text)
    +float.Parse(this.lbxSumaRepuestos.Items[i].Text)
    +float.Parse(this.lbxSumaOtros.Items[i].Text);
    dtRow = tablaCombustible.NewRow();
    dtRow["Unidad"] = this.lbxVehiculos.Items[i].Text;
    dtRow["Mano de Obra"] = this.lbxSumaManoObra.Items[i].Text;
    dtRow["Lubricantes"] =
        this.lbxSumaLubricantes.Items[i].Text;
    dtRow["Repuestos"] = this.lbxSumaRepuestos.Items[i].Text;
    dtRow["Otros"] = this.lbxSumaOtros.Items[i].Text;
    dtRow["Total"]=ValorParcial.ToString();
    dtRow["Partidas"] = this.lbxPartidas.Items[i].Text;
    tablaCombustible.Rows.Add(dtRow);
    ValorTotal=ValorTotal+ValorParcial;
}
//fila total
dtRow = tablaCombustible.NewRow();
dtRow["Unidad"] = "";
dtRow["Mano de Obra"] = "";
dtRow["Lubricantes"] = "";

```

```

dtRow["Repuestos"] = "";
dtRow["Otros"] = "***TOTAL***";
dtRow["Total"] = ValorTotal.ToString();
dtRow["Partidas"] = "";
tablaCombustible.Rows.Add(dtRow);
this.dgDetalle.DataSource=tablaCombustible;
this.dgDetalle.DataBind();
//data grid general lubricantes
//1
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Departamento";
dtCol.Caption = "Otros";
dtCol.ReadOnly = true;
tablaGeneral.Columns.Add(dtCol);
//2
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Total";
dtCol.Caption = "Total";
dtCol.ReadOnly = true;
tablaGeneral.Columns.Add(dtCol);
//3
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Partidas";
dtCol.Caption = "Partidas";
dtCol.ReadOnly = true;
tablaGeneral.Columns.Add(dtCol);
//filas
float ValorTotal1=0;
for(int i=0;i<=this.lbxDepId.Items.Count-1;i++)
{
    dtRow = tablaGeneral.NewRow();
    dtRow["Departamento"] =
        this.lbxDepNombre.Items[i].Text;
    dtRow["Total"] = this.lbxGenLubricantes.Items[i].Text;
    dtRow["Partidas"] = this.lbxParNumero.Items[i].Text;
    tablaGeneral.Rows.Add(dtRow);
    ValorTotal1=ValorTotal1+float.Parse
        (this.lbxGenLubricantes.Items[i].Text);
}
dtRow = tablaGeneral.NewRow();
dtRow["Departamento"] = "***TOTAL***";
dtRow["Total"] = ValorTotal1.ToString();
dtRow["Partidas"] = "";
tablaGeneral.Rows.Add(dtRow);
this.dgGeneralLubricantes.DataSource=tablaGeneral;
this.dgGeneralLubricantes.DataBind();
tablaGeneral.Clear();
//data grid general mano de obra
//1
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Departamento";
dtCol.Caption = "Otros";
dtCol.ReadOnly = true;
tablaGeneral2.Columns.Add(dtCol);
//2

```

```

dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Total";
dtCol.Caption = "Total";
dtCol.ReadOnly = true;
tablaGeneral2.Columns.Add(dtCol);
//3
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Partidas";
dtCol.Caption = "Partidas";
dtCol.ReadOnly = true;
tablaGeneral2.Columns.Add(dtCol);
//2
float ValorTotal2=0;
for(int i=0;i<=this.lbxDepId.Items.Count-1;i++)
{
    dtRow = tablaGeneral2.NewRow();
    dtRow["Departamento"] = this.lbxDepNombre.Items[i].Text;
    dtRow["Total"] = this.lbxGenManoObra.Items[i].Text;
    dtRow["Partidas"] = this.lbxParNumero.Items[i].Text;
    tablaGeneral2.Rows.Add(dtRow);
    ValorTotal2=ValorTotal2+float.Parse(this.lbxGenManoObra.Items[i].Text);
}
dtRow = tablaGeneral2.NewRow();
dtRow["Departamento"] = "***TOTAL***";
dtRow["Total"] = ValorTotal2.ToString();
dtRow["Partidas"] = "";
tablaGeneral2.Rows.Add(dtRow);
this.dgGeneralMantenimiento.DataSource=tablaGeneral2;
this.dgGeneralMantenimiento.DataBind();
tablaGeneral2.Clear();
//3
//data grid general lubricantes
//1
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Departamento";
dtCol.Caption = "Otros";
dtCol.ReadOnly = true;
tablaGeneral3.Columns.Add(dtCol);
//2
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Total";
dtCol.Caption = "Total";
dtCol.ReadOnly = true;
tablaGeneral3.Columns.Add(dtCol);
//3
dtCol = new DataColumn();
dtCol.DataType= System.Type.GetType("System.String");
dtCol.ColumnName = "Partidas";
dtCol.Caption = "Partidas";
dtCol.ReadOnly = true;
tablaGeneral3.Columns.Add(dtCol);
float ValorTotal3=0;
for(int i=0;i<=this.lbxDepId.Items.Count-1;i++)
{

```

```

        dtRow = tablaGeneral3.NewRow();
        dtRow["Departamento"] = this.lbxDepNombre.Items[i].Text;
        dtRow["Total"] = this.lbxGenRepuestos.Items[i].Text;
        dtRow["Partidas"] = this.lbxParNumero.Items[i].Text;
        tablaGeneral3.Rows.Add(dtRow);
        ValorTotal3=ValorTotal3+float.Parse(this.lbxGenRepuestos.Items[i].Text);
    }
    dtRow = tablaGeneral3.NewRow();
    dtRow["Departamento"] = "***TOTAL***";
    dtRow["Total"] = ValorTotal3.ToString();
    dtRow["Partidas"] = "";
    tablaGeneral3.Rows.Add(dtRow);
    this.dgGeneralRepuestos.DataSource=tablaGeneral3;
    this.dgGeneralRepuestos.DataBind();
    tablaGeneral3.Clear();
    //4
    //data grid general lubricantes
    //1
    dtCol = new DataColumn();
    dtCol.DataType= System.Type.GetType("System.String");
    dtCol.ColumnName = "Departamento";
    dtCol.Caption = "Otros";
    dtCol.ReadOnly = true;
    tablaGeneral4.Columns.Add(dtCol);
    //2
    dtCol = new DataColumn();
    dtCol.DataType= System.Type.GetType("System.String");
    dtCol.ColumnName = "Total";
    dtCol.Caption = "Total";
    dtCol.ReadOnly = true;
    tablaGeneral4.Columns.Add(dtCol);
    //3
    dtCol = new DataColumn();
    dtCol.DataType= System.Type.GetType("System.String");
    dtCol.ColumnName = "Partidas";
    dtCol.Caption = "Partidas";
    dtCol.ReadOnly = true;
    tablaGeneral4.Columns.Add(dtCol);
    float ValorTotal4=0;
    for(int i=0;i<=this.lbxDepId.Items.Count-1;i++)
    {
        dtRow = tablaGeneral4.NewRow();
        dtRow["Departamento"] = this.lbxDepNombre.Items[i].Text;
        dtRow["Total"] = this.lbxGenOtros.Items[i].Text;
        dtRow["Partidas"] = this.lbxParNumero.Items[i].Text;
        tablaGeneral4.Rows.Add(dtRow);
        ValorTotal4=ValorTotal4+float.Parse(this.lbxGenOtros.Items[i].Text);
    }
    dtRow = tablaGeneral4.NewRow();
    dtRow["Departamento"] = "***TOTAL***";
    dtRow["Total"] = ValorTotal4.ToString();
    dtRow["Partidas"] = "";
    tablaGeneral4.Rows.Add(dtRow);
    this.dgGeneralOtros.DataSource=tablaGeneral4;
    this.dgGeneralOtros.DataBind();
}

```

## 8.3 PRUEBAS

Las pruebas son un paso importante en el desarrollo de sistemas, en el caso de el sistema de control vehicular se ha destinado una semana para las pruebas necesarias que junto al personal de vehículos se han desarrollado con normalidad. En el transcurso de las pruebas se encontró pequeños problemas que en el mismo instante se los solucionó, no se produjeron problemas de gran magnitud lo que permitió que podamos seguir con la implantación del Sistema sin problemas.

### 8.3.1 FUNCIONES JAVA SCRIPT PARA EL CONTROL

Estas funciones son utilizadas con el fin de controlar el ingreso de información y la integridad de los datos:

Para no permitir al usuario ingresar datos erróneos no permitidos por la Base de Datos.

Con esto aseguramos que no existan errores de tipo sintáctico al momento de ingresar información.

**Función “atrapaenter”:** Esta función permite que al momento de teclear “Enter” dentro de un cuadro de texto se ejecute cierto evento, como por ejemplo on\_click de un botón.

```
function atrapaenter(btn,event,tipo)
{
    var tecla = window.event.keyCode;
    if (document.all)
    {
        if (tecla==13)
        {
            event.returnValue=false;
            event.cancel=true;
        }
    }
}
```

```

        if(tipo===nuevo)
        {
            window.ibtnAceptarNuevo.Button1.click();
        }
        if(tipo===editar)
        {
            window.ibtnAceptar.Button1.click();
        }
        if(tipo===eliminar)
        {
            window.ibtnAceptarEliminar.Button1.click();
        }
    }
}

```

**Función “solonumeros”:** esta función permite que en los cuadros de texto sólo se ingresen números enteros 0 al 9

```

function solonumeros(letra)
{
    var tecla = window.event.keyCode;
    var tecla1 = letra;
    if (tecla < 48 || tecla >57)
    {
        window.event.keyCode=0;
    }
}

```

**Función “sololetras”:** esta función permite que en los cuadros de texto sólo se ingresen letras minúsculas y mayúsculas

```

function sololetras(letra)
{
    var tecla = window.event.keyCode;
    var tecla1 = letra;
    if ((tecla < 65 || tecla >90)&&(tecla 122)&&(tecla 32))
    {
        window.event.keyCode=0;
    }
}

```

**Función “sololetrasnumeros”:** esta función permite que en los cuadros de texto sólo se ingresen números y letras, es decir no caracteres especiales.

```

function sololetrasnumeros(letra)
{
    var tecla = window.event.keyCode;
    var tecla1 = letra;
    if((tecla<65||tecla>90)&&(tecla122)&&(tecla<48||tecla>57)&&(tecla32)&
&(tecla45))
    {
        window.event.keyCode=0;
    }
}

```

**Función “solofloat”:** esta función permite que en los cuadros de texto sólo se ingresen números flotantes, es decir permite el ingreso del separador de decimales.

```

function solofloat()
{
    var tecla = window.event.keyCode;
    if ((tecla < 48 || tecla >57) && (tecla!=46))

```

```

    {
        window.event.keyCode=0;
    }
}

```

**Función “solocalculogalones”:** esta función permite calcular de manera automática la cantidad de galones consumidos al momento de ingresar el total de dinero consumido.

```

function solocalculogalones(valor,texto)
{
    var cantidad=" ";
    var total=0;
    var texto2;
    var tecla = window.event.keyCode;
    if ((tecla < 48 || tecla >57) && (tecla!=46))
    {
        window.event.keyCode=0;
    }
    if(texto=="galones")
    {
        texto2=window.Form1.txtTotal.value;
    }
    if(tecla==48)
    {
        cantidad=texto2+"0";
    }
    if(tecla==49)
    {
        cantidad=texto2+"1";
    }
    if(tecla==50)

```

```
{
    cantidad=texto2+"2";
}
if(tecla==51)
{
    cantidad=texto2+"3";
}
if(tecla==52)
{
    cantidad=texto2+"4";
}
if(tecla==53)
{
    cantidad=texto2+"5";
}
if(tecla==54)
{
    cantidad=texto2+"6";
}
if(tecla==55)
{
    cantidad=texto2+"7";
}
if(tecla==56)
{
    cantidad=texto2+"8";
}
if(tecla==57)
{
    cantidad=texto2+"9";
}
if(texto=="galones")
```

```

    {
        window.Form1.txtValor.value=cantidad/valor;
        window.Form1.TextBox9.value=eval(window.Form1.txtValor.valu
e);
    }
else
{
    window.Form1.txtTotal.value=eval(window.Form1.txtValor.value)
+eval( cantidad);
}
}

```

**Función “soloplaca”:** esta función permite que en el cuadros de texto de placa del vehículo se ingresen tres letras, luego se inserte automáticamente el símbolo “-“ y luego permita solo ingresar tres números.

```

function soloplaca(letra)
{
    var tecla = window.event.keyCode;
    var cantidad = Form1.txtPlaca.value.length;
    if(cantidad<3)
    {
        if ((tecla < 65 || tecla >90)&& (tecla 122))
        {
            window.event.keyCode=0;
        }
    }
else
{
    if ((tecla < 48 || tecla >57))
    {
        window.event.keyCode=0;
    }
}
}

```

```
    }  
    if(cantidad==3)  
    {  
        Form1.txtPlaca.value=Form1.txtPlaca.value+"-";  
    }  
} }  
}
```

## BIBLIOGRAFÍA

- OGATA, katsuko, Ingeniería de control moderna, Editorial Prentice – Hall Hispanoamericana S.A. tercera edición, México 1998.
- LEWIS, Paul, Sistemas de control en ingeniería, Prentice – Hall Iberia SRL, Madrid 1999.
- DELGADO, Albert, Microsoft SQL Server 2000, Pearson Education S. A. Madrid 2001.
- SHIMBERG David y BARNES Lisa, Estrategias Cliente – Servidor para el siglo XXI, Mexico 1998
- PAYNE, Chris, Aprendiendo ASP.NET en 21 lecciones avanzadas, Pearson Education, México 2002.
- PRESSMAN Roger S. Ingeniería del Software un enfoque práctico, Cuarta Edición, 1997.
- <http://www.monografias.com/trabajos6/sicox/sicox.shtml>
- <http://www.monografias.com/trabajos5/caudes/caudes.shtml>
- <http://personales.com/costarica/alajuela/humanos/def.htm>
- <http://es.wikipedia.org/wiki/Sistema>
- <http://www.monografias.com/trabajos11/teosis/teosis.shtml>
- [http://es.wikipedia.org/wiki/Vehículo#Veh.C3.ADculos\\_sobre\\_suelo\\_pavimentado\\_o\\_firme](http://es.wikipedia.org/wiki/Veh%C3%ADculo#Veh.C3.ADculos_sobre_suelo_pavimentado_o_firme)
- <http://elephas.wprldonline.es/valen-ar/glosario.html>

- <http://www.maestrosdelweb.com/editorial/sql2kinstal/>
- [http://www.programacion.net/bbdd/articulo/man\\_tablastemp/](http://www.programacion.net/bbdd/articulo/man_tablastemp/)
- <http://es.wikipedia.org/wiki/ASP.NET>
- [http://www.programacion.net/asp/articulo/aspnet\\_quees/](http://www.programacion.net/asp/articulo/aspnet_quees/)
- <http://es.wikipedia.org/wiki/Cliente-servidor>
- <http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml>
- <http://es.wikipedia.org/wiki/Intranet>
- <http://www.monografias.com/trabajos12/intrants/intrants.shtml>
- <http://www.monografias.com/trabajos6/sein/sein.shtml>

## 6.6.2 DICCIONARIO DE DATOS

### CONTROL DE VEHÍCULOS DE LA EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO DE AMBATO “EMAPA”

#### NIVEL 0

#### ENTIDADES

<b>ENTIDAD</b>	<b>DESCRIPCIÓN</b>	<b>FLUJOS PROPORCIONADOS</b>	<b>FLUJOS DE ENTRADA</b>
Encargado de área Vehicular	Persona que realiza los trámites y peticiones necesarias para el funcionamiento del software	<ul style="list-style-type: none"><li>- Solicitud de materiales</li><li>- Formulario de despacho</li><li>- datos de vehículo</li><li>- Solicitud de combustible</li></ul>	<ul style="list-style-type: none"><li>- datos erróneos</li></ul>
Jefe Departamental	Persona que recibe los reportes emitidos por el Software		<ul style="list-style-type: none"><li>- Informes de mantenimiento</li><li>- Informes de tanqueo</li><li>- Informes de vehículos de EMAPA</li></ul>
Departamento Financiero	Personas que aprueban las peticiones de mantenimiento y tanqueo requeridas	<ul style="list-style-type: none"><li>- aprobación de financiero</li></ul>	<ul style="list-style-type: none"><li>- Proforma de área vehicular</li></ul>
Lubricadoras –lavadoras – mecánicas	Personas que ejecutan mantenimiento para el vehículo	<ul style="list-style-type: none"><li>- Facturas de mantenimiento</li></ul>	<ul style="list-style-type: none"><li>- Informe de solicitud de materiales</li></ul>
Gasolineras	Personas que ejecutan el tanqueo del vehículo	<ul style="list-style-type: none"><li>- Factura de tanqueo</li></ul>	<ul style="list-style-type: none"><li>- Formulario de despacho</li></ul>

Tabla 8 (Diccionario de datos – Entidades)

## PROCESOS

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Sistema de Control Vehicular	Procesa información de los vehículos su rendimiento y gastos	<ul style="list-style-type: none"> <li>- Solicitud de materiales</li> <li>- Formulario de despacho</li> <li>- datos de vehículo</li> <li>- aprobación de financiero</li> <li>- Facturas de mantenimiento</li> <li>- Factura de tanqueo</li> </ul>	<ul style="list-style-type: none"> <li>- Informes de mantenimiento</li> <li>- Informes de tanqueo</li> <li>- Informes de vehículos de EMAPA</li> <li>- datos erróneos</li> <li>- Proforma de área vehicular</li> <li>- Informe de solicitud de materiales</li> <li>- Informe formulario de despacho</li> </ul>	Determina los costos que generan los vehículos para las diferentes áreas de la empresa, su rendimiento y su administración

Tabla 9 (Diccionario de datos – Procesos Nivel 0)

## ESTRUCTURA DE DATOS

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Formulario de despacho	Contiene información sobre el despacho de combustible realizado	Encargado de área Vehicular	Sistema de Control Vehicular	<ul style="list-style-type: none"> <li>- Número de despacho</li> <li>- Fecha del despacho</li> <li>- Gasolinera proveedora</li> <li>- Unidad vehicular</li> <li>- Conductor</li> <li>- Kilometraje</li> <li>- Tipo de Combustible</li> <li>- Cantidad en Galones de Combustible</li> <li>- Valor del despacho individual</li> </ul>	28/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Solicitud de combustible	Contiene información sobre la solicitud de combustible realizada	Encargado de área Vehicular	Sistema de Control Vehicular	<ul style="list-style-type: none"> <li>- Número de solicitud</li> <li>- Fecha de la solicitud</li> <li>- Gasolinera proveedora</li> <li>- Unidad vehicular</li> <li>- Tipo de Combustible</li> <li>- Cantidad en Galones de Combustible</li> <li>- Valor del despacho individual</li> </ul>	28/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos de vehículos	Contiene información detallada de cada vehículo	Encargado de área Vehicular	Sistema de Control Vehicular	<ul style="list-style-type: none"> <li>- Identificador de la Unidad</li> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Número de Motor</li> <li>- Número de Chasis</li> <li>- Marca</li> <li>- Clase</li> <li>- Modelo</li> <li>- Tipo</li> <li>- Año de Fabricación</li> <li>- Color</li> <li>- País de Origen</li> <li>- Chofer</li> </ul>	10/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Solicitud de materiales	Descripción de los mantenimientos requeridos en las unidades vehiculares	Encargado de área Vehicular	Sistema de Control Vehicular	<ul style="list-style-type: none"> <li>- Número de la solicitud</li> <li>- Fecha de la Solicitud</li> <li>- Departamento solicitante</li> <li>- Cantidad necesitada</li> <li>- Unidad de medida</li> <li>- Descripción del Artículo</li> <li>- Observaciones</li> </ul>	10/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Factura de tanqueo	Factura enviada por el proveedor de combustible	Gasolineras	Sistema de Control Vehicular	<ul style="list-style-type: none"> <li>- Número de Factura</li> <li>- Detalle de Factura</li> <li>- Valor de la Factura</li> </ul>	5/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Informe formulario de despacho	Informe de consumos enviados a los proveedores de combustible	Sistema de Control Vehicular	Gasolineras	<ul style="list-style-type: none"> <li>- Fecha de elaboración</li> <li>- Nombre de el proveedor</li> <li>- Unidad vehicular</li> <li>- Costo de Combustible consumido por unidad</li> <li>- Galones consumidos por unidad</li> <li>- Total consumido</li> </ul>	5/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Informe de solicitud de Materiales	Informe de consumos enviados a los proveedores de materiales y mecánicas	Sistema de Control Vehicular	Lubricadoras – lavadoras – mecánicas	<ul style="list-style-type: none"> <li>- Fecha de elaboración</li> <li>- Nombre de el proveedor</li> <li>- Unidad vehicular</li> <li>- Costo de mantenimiento por unidad</li> <li>- Total consumido</li> </ul>	10/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Facturas de mantenimiento	Factura enviada por el proveedor materiales o mano de obra	Lubricadoras – lavadoras – mecánicas	Sistema de Control Vehicular	<ul style="list-style-type: none"> <li>- Número de Factura</li> <li>- Detalle de Factura</li> <li>- Valor de la Factura</li> </ul>	10/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Proforma	Contiene información del valor de despacho de combustible o mantenimientos requeridos	Sistema de Control Vehicular	Departamento Financiero	<ul style="list-style-type: none"> <li>- Número de solicitud o despacho</li> <li>- Detalle</li> <li>- Valor Total</li> </ul>	40/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Informe vehículos EMAPA	Contiene Información detallada de los vehículos de la Empresa para ser presentados al Jefe Departamental	Sistema de Control Vehicular	Jefe Departamental	<ul style="list-style-type: none"> <li>- Identificador de la Unidad</li> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Número de Motor</li> <li>- Número de Chasis</li> <li>- Marca</li> <li>- Clase</li> <li>- Modelo</li> <li>- Tipo</li> <li>- Año de Fabricación</li> <li>- Color</li> <li>- País de Origen</li> <li>- Chofer</li> </ul>	10/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Informe de Mantenimiento		Sistema de Control Vehicular	Jefe Departamental	<ul style="list-style-type: none"> <li>- Fecha de elaboración</li> <li>- Nombre de el proveedor</li> <li>- Unidades vehiculares</li> <li>- Costo de Mano de obra</li> <li>- Costo de Repuestos</li> <li>- Costo de Lubricantes</li> <li>- Costos de Otros</li> <li>- Total consumido</li> <li>- Partida del vehículo</li> <li>- Observaciones</li> </ul>	10/mes

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Informe de tanqueo		Sistema de Control Vehicular	Jefe Departamental	<ul style="list-style-type: none"> <li>- Fecha de elaboración</li> <li>- Nombre de el proveedor</li> <li>- Unidades vehiculares</li> <li>- Costo de Combustible consumido por unidad</li> <li>- Galones consumidos por unidad</li> <li>- Kilometraje recorrido</li> <li>- Kilometraje promedio</li> <li>- Partida del vehículo</li> <li>- Total consumido</li> <li>- Valor del combustible a la fecha</li> </ul>	10/mes

Tabla 10 (Diccionario de datos – Estructuras Nivel 0)

#### ELEMENTOS DATO

ELEMENTO	DESCRIPCION	TIPO	LONGITUD	ALIAS	RANGO VALORES	LIMITE VALORES
Aprobación financiero	Informe de bodega junto a departamento financiero de aceptación económica para realizar el siguiente proceso	-	-	-		-

ELEMENTO	DESCRIPCION	TIPO	LONGITUD	ALIAS	RANGO VALORES	LIMITE VALORES
Datos erróneos	Mensaje que indica al usuario del sistema que la información ingresada es errónea	Varchar	37	-	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Identificador de la Unidad	Dato de identificación de cada unidad vehicular	Varchar	10	Veh_id	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Departamento al que pertenece	Dato que identifica al departamento que pertenece la unidad vehicular	Varchar	10	Dep_id	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Placas de la Unidad	Contiene el dato de la placa correspondiente a cada unidad	Varchar	10	Veh_placas	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Número de Motor	Dato que identifica al motor de la unidad vehicular	Varchar	50	Veh_motor	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Número de Chasis	Dato que identifica al chasis de la unidad vehicular	Varchar	50	Veh_chasis	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Marca	Contiene el dato de la marca de la unidad vehicular	Varchar	50	Veh_marca	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Clase	Contiene el dato de la clase a la que pertenece la unidad vehicular	Varchar	50	Veh_clase	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Modelo	Contiene el dato del modelo la unidad vehicular	Varchar	50	Veh_modelo	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Tipo	Contiene el dato del tipo de la unidad vehicular	Varchar	50	Veh_tipo	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Año de Fabricación	Contiene el dato del año de fabricación de la unidad vehicular	Varchar	5	Veh_año	(Desde año actual – 25)(hasta Año actual +1)	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Color	Contiene el dato del color de la unidad vehicular	Varchar	20	Veh_color	(Desde año actual – 25)(hasta Año actual +1)	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
País de Origen	Contiene el dato del país de origen de la unidad vehicular	Varchar	20	Veh_pais	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Chofer	Contiene el dato de la id (cédula o RUC) del chofer de la unidad vehicular	Varchar	15	cle_ci_ruc	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Número de despacho	Contiene el número de despacho ID del despacho realizado	Varchar	10	Ff_id	-	-

ELEMENTO	DESCRIPCION	TIPO	LONGITUD	ALIAS	RANGO VALORES	LIMITE VALORES
Fecha del despacho	Contiene la fecha en la que se realizó el despacho	Smalldateti me	-	_fecha	Fecha actual	-

ELEMENTO	DESCRIPCION	TIPO	LONGITUD	ALIAS	RANGO VALORES	LIMITE VALORES
Gasolinera proveedora	Contiene la id de la despachadora de combustible seleccionada	Varchar	10	Gas_id	-	-

ELEMENTO	DESCRIPCION	TIPO	LONGITUD	ALIAS	RANGO VALORES	LIMITE VALORES
Kilometraje	Contiene el kilometraje del vehículo al momento de tanquear	Bigint	7	_km	0 - 9'000.000	9'000.000

ELEMENTO	DESCRIPCION	TIPO	LONGITUD	ALIAS	RANGO VALORES	LIMITE VALORES
Tipo de Combustible	Contiene el tipo de combustible seleccionado (súper, extra, diesel)	Varchar	10	_tipo	-	-

ELEMENTO	DESCRIPCION	TIPO	LONGITUD	ALIAS	RANGO VALORES	LIMITE VALORES
Cantidad en Galones de Combustible	Contiene la cantidad de galones consumidos en la despachadora	bigint	7	_cantidad	0 - 9'000.000	9'000.000

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Valor del despacho individual	Contiene el valor en dólares consumido por cada vehículo	float	15	_total	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Número de solicitud	Contiene el número de solicitud o id de la solicitud	Varchar	10	Ff_id	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Fecha de la solicitud	Contiene la fecha en la que se realizó la solicitud	Smalldatetime	-	_fecha	Fecha actual	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Gasolinera proveedora	Contiene la id de la despachadora de combustible seleccionada	Varchar	10	Gas_id	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Número de la solicitud de materiales	Contiene el número de despacho ID de la solicitud realizada	Varchar	10	man_id	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Fecha de la Solicitud de materiales	Contiene la fecha en la que se realizó la solicitud	Smalldateti me	-	man_fecha	Fecha actual	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Departamento solicitante	Contiene la id del departamento que ha solicitado el mantenimiento	Varchar	15	Dep_id	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Kilometraje	Contiene el kilometraje del vehículo al momento del mantenimiento	Bigint	7	Man_km	0 - 9'000.000	9'000.000

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Descripción del Artículo	Contiene el artículo que necesita la unidad vehicular a dar mantenimiento	Varchar	50	Man_detalle	-	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Observaciones	Contiene las observaciones que se han presentado durante el mantenimiento	Varchar	200	Man_observaciones	-	-

Tabla 11 (Diccionario de datos – Elementos Dato Nivel 0)

## NIVEL 1

### PROCESOS

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Validar Datos (1)	Procesa información para luego ser validada o desechada.	- Datos de vehículos	- Datos erróneos - Kilometraje validado - Datos editables del vehículo	Determina si la información ingresada de los vehículos, kilometrajes fecha etc. Son correctos

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Modificar Kilometraje (2)	Procesa información de los kilometrajes ingresados para actualizar el campo respectivo de kilometraje de cada vehículo	- Kilometraje validado	- Informe nuevo kilometraje	Actualiza los kilometrajes respectivos de los vehículos

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Edición de vehículos (3)	Permite procesar la información ingresada de los vehículos y almacenarla	- Datos editables del vehículo	- Informes vehículos EMAPA	Nos permite crear, eliminar o Actualizar los datos de los vehículos de la empresa

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Añadir Tanqueo (4)	Procesa información del nuevas cargas de combustible realizadas a los vehículos	<ul style="list-style-type: none"> <li>- Operación de tanqueo encontrada</li> <li>- Formulario de despacho</li> <li>- Facturas de tanqueo</li> </ul>	<ul style="list-style-type: none"> <li>- Informe de formulario de despacho</li> <li>- Informe de tanqueo</li> </ul>	Permite ingresar nuevos despachos o solicitudes de combustible a realizar

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Añadir Mantenimiento (5)	Procesa información del nuevos mantenimientos requeridos por los vehículos	<ul style="list-style-type: none"> <li>- Operación de mantenimiento encontrada</li> <li>- Solicitud de materiales</li> <li>- Facturas de mantenimiento</li> <li>- Aprobación de financiero</li> </ul>	<ul style="list-style-type: none"> <li>- Proforma</li> <li>- Informe de mantenimiento</li> <li>- Informe de solicitud de materiales</li> </ul>	Permite ingresar e imprimir nuevas solicitudes de materiales requeridas en la empresa

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Evaluar Operaciones Necesarias (6)	Información y alerta de operaciones requeridas por los vehículos de la empresa	<ul style="list-style-type: none"> <li>- Informe nuevo kilometraje</li> </ul>	<ul style="list-style-type: none"> <li>- Operaciones de tanqueo encontrada</li> <li>- Operaciones de mantenimiento encontrada</li> </ul>	Ordenes emitidas por el operador del sistema para que se ejecuten operaciones necesarias

Tabla 12 (Diccionario de datos – Procesos Nivel 1)

## ESTRUCTURAS DE DATOS

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Datos editables de vehículos	Contiene información detallada y validada de cada vehículo	Validar Datos (1)	Edición de vehículos (3)	<ul style="list-style-type: none"> <li>- Identificador de la Unidad</li> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Número de Motor</li> <li>- Número de Chasis</li> <li>- Marca</li> <li>- Clase</li> <li>- Modelo</li> <li>- Tipo</li> <li>- Año de Fabricación</li> <li>- Color</li> <li>- País de Origen</li> <li>- Chofer</li> </ul>	10/mes

Tabla 13 (Diccionario de datos – Estructuras Nivel 1)

## ELEMENTOS DATO

ELEMENTO	DESCRIPCION	TIPO	LONGITUD	ALIAS	RANGO VALORES	LIMITE VALORES
Kilometraje Validado	Contiene el nuevo kilometraje del vehículo	Bigint	7	Veh_km	0 - 9'000.000	9'000.000

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Informe nuevo kilometraje	Grupo de datos de los kilometrajes actualizados de los vehículos	Bigint	7	Veh_km	0 - 9'000.000	9'000.000

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Operación de tanqueo encontrada	Dato de autorización para la realización de un despacho de combustible	Bool	-	-	True-false	-

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Operación de mantenimiento encontrada	Dato de autorización para la realización mantenimiento vehicular	Bool	-	-	True-false	-

Tabla 14 (Diccionario de datos – Elementos dato Nivel 1)

## NIVEL 2

Edición de vehículos (3)

### PROCESOS

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Buscar vehículo (3.1)	Procesa información del un vehículo requerido para realizar operaciones necesarias	- Datos editables de vehículos	- Datos de vehículo a modificar - Datos de vehículo a añadir - Datos de vehículo a eliminar	Encuentra los datos de un vehículo o sus valores predeterminados para efectuar una de las operaciones existentes

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Actualizar información de vehículos (3.2)	Procesa la nueva información del vehículo seleccionado y envía sus nuevos datos	- Datos de vehículo a modificar	- datos de vehículo actualizados	Permite actualizar los datos del vehículo para posteriormente ser almacenados

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Almacenar nuevo vehículo (3.3)	Procesa nueva información para un nuevo vehículo a añadir	- Datos de vehículo a añadir	- Datos nuevo vehículo	Permite el ingreso de nuevos datos de un vehículo para luego ser añadidos al almacén de datos

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Eliminar vehículo (3.4)	Recibe información de un vehículo y la procesa para ser eliminada	- Datos de vehículo a eliminar - Mensaje vehículo eliminado	- Datos vehículo a eliminar	Permite eliminar la información del vehículo que se eliminará

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Consultar vehículos (3.5)	Permite visualizar los detalles los datos de los vehículos	- Datos de vehículos actualizados	- Informe vehículos – EMAPA	Permite buscar todos los vehículos existentes para luego emitir un informe de estos.

Tabla 15 (Diccionario de datos – Procesos Nivel 2 Edición de datos)

**ALMACEN DE DATOS**

<b>ALMACEN</b>	<b>DESCRIPCION</b>	<b>FLUJO ENTRADA</b>	<b>FLUJO PROPORCIONADO</b>	<b>DESCRIPCION DE DATOS</b>	<b>VOLUMEN</b>	<b>ACCESO</b>
Tbl_ad_vehiculos	Guarda información de los vehículos de la empresa	- Datos nuevo vehículo - datos de vehículo actualizados - Datos vehículo a eliminar	- Mensaje vehículo eliminado - Datos actualizados de vehículos	- Identificador de la Unidad - Departamento al que pertenece - Placas de la Unidad - Número de Motor - Número de Chasis - Marca - Clase - Modelo - Tipo - Año de Fabricación - Color - País de Origen - Chofer	40/ mes	diario

Tabla 16 (Diccionario de datos – Almacenes Nivel 2 Edición de datos)

### ESTRUCTURAS DE DATOS

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Datos de vehículo a modificar	Contiene información del vehículo a modificar	Buscar vehículo (3.1)	Actualizar información de vehículo (3.2)	<ul style="list-style-type: none"> <li>- Identificador de la Unidad</li> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Número de Motor</li> <li>- Número de Chasis</li> <li>- Marca</li> <li>- Clase</li> <li>- Modelo</li> <li>- Tipo</li> <li>- Año de Fabricación</li> <li>- Color</li> <li>- País de Origen</li> <li>- Chofer</li> </ul>	20/mes

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Datos de vehículo a añadir	Contiene información de los valores por defecto que necesita el vehículo para ser añadido	Buscar vehículo (3.1)	Almacenar nuevo vehículo (3.3)	<ul style="list-style-type: none"> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Año de Fabricación</li> <li>- Chofer</li> </ul>	10/mes

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Datos de vehículo a eliminar	Contiene información del vehículo a eliminar	Validar Datos (1)	Edición de vehículos (3)	<ul style="list-style-type: none"> <li>- Identificador de la Unidad</li> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Número de Motor</li> </ul>	10/mes

				<ul style="list-style-type: none"> <li>- Número de Chasis</li> <li>- Marca</li> <li>- Clase</li> <li>- Modelo</li> <li>- Tipo</li> <li>- Año de Fabricación</li> <li>- Color</li> <li>- País de Origen</li> <li>- Chofer</li> </ul>	
<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos de vehículo a actualizados	Contiene información del vehículo modificada	Actualizar información de vehículo (3.2)	Tbl_ad_vehiculos	<ul style="list-style-type: none"> <li>- Identificador de la Unidad</li> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Número de Motor</li> <li>- Número de Chasis</li> <li>- Marca</li> <li>- Clase</li> <li>- Modelo</li> <li>- Tipo</li> <li>- Año de Fabricación</li> <li>- Color</li> <li>- País de Origen</li> <li>- Chofer</li> </ul>	20/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos nuevo vehículo	Contiene información del vehículo a modificar	Almacenar nuevo vehículo (3.3)	Tbl_ad_vehiculos	<ul style="list-style-type: none"> <li>- Identificador de la Unidad</li> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Número de Motor</li> </ul>	10/mes

				<ul style="list-style-type: none"> <li>- Número de Chasis</li> <li>- Marca</li> <li>- Clase</li> <li>- Modelo</li> <li>- Tipo</li> <li>- Año de Fabricación</li> <li>- Color</li> <li>- País de Origen</li> <li>- Chofer</li> </ul>	
--	--	--	--	---	--

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Datos vehículo a eliminar	Contiene información del vehículo a modificar	Eliminar vehículo (3.4)	Tbl_ad_vehiculos	- Identificador de la Unidad	10/mes

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Datos actualizados de vehículos	Datos actualizados de todos los vehículos de la empresa	Tbl_ad_vehiculos	Consultar vehículos (3.5)	<ul style="list-style-type: none"> <li>- Identificador de la Unidad</li> <li>- Departamento al que pertenece</li> <li>- Placas de la Unidad</li> <li>- Número de Motor</li> <li>- Número de Chasis</li> <li>- Marca</li> <li>- Clase</li> <li>- Modelo</li> <li>- Tipo</li> <li>- Año de Fabricación</li> <li>- Color</li> <li>- País de Origen</li> <li>- Chofer</li> </ul>	20/mes

Tabla 17 (Diccionario de datos – Estructuras Nivel 2 Edición de datos)

**ELEMENTOS DATO**

<b>ELEMENTO</b>	<b>DESCRIPCION</b>	<b>TIPO</b>	<b>LONGITUD</b>	<b>ALIAS</b>	<b>RANGO VALORES</b>	<b>LIMITE VALORES</b>
Mensaje vehículo eliminado	Mensaje de información al momento de eliminar una unidad	Varchar	50	-	-	-

Tabla 18 (Diccionario de datos – Elementos Dato Nivel 2 Edición de datos)

**NIVEL 2**

Añadir Tanqueo (4)

**PROCESOS**

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Consultar vehículos a tanquear (4.1)	Procesa información sobre los vehículos que necesitan el tanqueo	- Operación de tanqueo encontrada	- Datos de vehículos a tanquear	Registra los vehículos que necesitan el tanqueo emitido por el encargado de vehículos

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Ordenar Tanqueo (4.2)	Emite una orden para tranquear al grupo de vehículos que lo requieren	- Datos de vehículos a tanquear - Solicitud de combustible - Formulario de despacho	- Informe formulario de despacho - Orden enviada	Recibe formularios y solicitudes de combustible para ser emitidas al momento de tanquear

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Añadir tanqueo (4.3)	Procesa la información de los nuevo tanqueos emitidos por las despachadoras para ser almacenados	- Orden enviada - Facturas de tanqueo	- Datos de tanqueo	Permite ingresar información de los nuevos despachos que se realizaron en los vehículos de la empresa

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Validar datos tanqueo (4.4)	Procesa información del nuevo tanqueo y la valida	- Datos de tanqueo	- Datos de tanqueo validados	Recibe la información del nuevo tanqueo y la valida de acuerdo a reglas de validación

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Consultar tanqueos (4.5)	Realiza la búsqueda de los tanqueos requeridos para mostrarse	- Datos de tanqueo actualizados	- Informe de tanqueo	Encuentra los tanqueos deseados para poder emitir un informe de estos.

Tabla 19 (Diccionario de datos – Procesos Nivel 2 Combustible)

#### ALMACEN DE DATOS

<b>ALMACEN</b>	<b>DESCRIPCION</b>	<b>FLUJO ENTRADA</b>	<b>FLUJO PROPORCIONADO</b>	<b>DESCRIPCION DE DATOS</b>	<b>VOLUMEN</b>	<b>ACCESO</b>
tbl_ad_despacho_combustible	Guarda información de los tanqueos y solicitudes de tanqueo de la empresa	- Datos tanqueo validados	- Datos tanqueo actualizados	- Número de despacho - Fecha del despacho - Gasolinera proveedora - Unidad vehicular	60/ mes	diario

				<ul style="list-style-type: none"> <li>- Conductor</li> <li>- Kilometraje</li> <li>- Tipo de Combustible</li> <li>- Cantidad en Galones de Combustible</li> <li>- Valor del despacho individual</li> </ul>		
--	--	--	--	--	--	--

Tabla 20 (Diccionario de datos – Almacenes Nivel 2 Combustible)

**ESTRUCTURAS DE DATOS**

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos de vehículos a tanquear	Contiene información del los vehículos a tanquear	Consultar vehículos a tanquear (4.1)	Ordenar Tanqueo (4.2)	- Identificador de la Unidad	28/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos de tanqueo	Contiene información del nuevo tanqueo realizado a los vehículos	Añadir tanqueo (4.3)	Validar datos tanqueo (4.4)	<ul style="list-style-type: none"> <li>- Número de despacho</li> <li>- Fecha del despacho</li> <li>- Gasolinera proveedora</li> <li>- Unidad vehicular</li> <li>- Conductor</li> <li>- Kilometraje</li> <li>- Tipo de Combustible</li> <li>- Cantidad en Galones de Combustible</li> <li>- Valor del despacho individual</li> </ul>	28/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos tanqueo validados	Contiene información validada del nuevo tanqueo realizado a los vehículos	Validar datos tanqueo (4.4)	Tbl_ad_despacho_combustible	<ul style="list-style-type: none"> <li>- Número de despacho</li> <li>- Fecha del despacho</li> <li>- Gasolinera proveedora</li> <li>- Unidad vehicular</li> <li>- Conductor</li> <li>- Kilometraje</li> <li>- Tipo de Combustible</li> <li>- Cantidad en Galones de Combustible</li> <li>- Valor del despacho individual</li> </ul>	28/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos de tanqueo actualizados	Contiene información actualizada del nuevo tanqueo realizado a los vehículos	Tbl_ad_despacho_combustible	Consultar tanqueos (4.5)	<ul style="list-style-type: none"> <li>- Numero de despacho</li> <li>- Fecha del despacho</li> <li>- Gasolinera proveedora</li> <li>- Unidad vehicular</li> <li>- Conductor</li> <li>- Kilometraje</li> <li>- Tipo de Combustible</li> <li>- Cantidad en Galones de Combustible</li> <li>- Valor del despacho individual</li> </ul>	10/mes

Tabla 21 (Diccionario de datos – Estructuras Nivel 2 Combustible)

## NIVEL 2

Añadir mantenimiento (5)

### PROCESOS

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Consultar vehículos a mantener (5.1)	Procesa información sobre los vehículos que necesitan mantenimiento	- Operación de mantenimiento encontrada	- Datos de vehículos a mantener	Registra los vehículos que necesitan el mantenimiento emitido por el encargado de vehículos

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Ordenar Mantenimiento (5.2)	Emite una orden para dar mantenimiento al grupo de vehículos que lo requieren	- Aprobación financiero - Solicitud de materiales	- Proforma - Solicitud materiales	Recibe solicitudes de mantenimiento para ser emitidas al momento de dar mantenimiento a un vehículo

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Verificar mantenimiento (5.3)	Procesa información del nuevo mantenimiento	- Orden enviada - Factura de mantenimiento	- Informe mantenimiento exitoso	Recibe la información del nuevo mantenimiento.

PROCESOS	DESCRIPCIÓN	ENTRADAS	SALIDAS	RESUMEN
Añadir Mantenimiento (5.4)	Ingresa los datos e información del mantenimiento en general	- Informe mantenimiento exitoso	- Id_mantenimiento - Datos de mantenimiento	Registra los datos del mantenimiento que ha sido ejecutado

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Añadir Detalle Mantenimiento (5.5)	Ingresa los datos e información del mantenimiento detallado	- Id_mantenimiento	- Datos del detalle de mantenimiento	Registra el detalle del mantenimiento requerido y ejecutado

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Validar datos mantenimiento (5.6)	Revisa los datos del detalle y mantenimiento.	- Datos mantenimiento - Datos detalle mantenimiento	- Datos mantenimiento validados - Datos detalle validados	Recibe los datos tanto del mantenimiento junto con su detalle y los valida de acuerdo a las reglas de validación especificadas.

<b>PROCESOS</b>	<b>DESCRIPCIÓN</b>	<b>ENTRADAS</b>	<b>SALIDAS</b>	<b>RESUMEN</b>
Consultar Mantenimientos (5.7)	Realiza la búsqueda de los mantenimientos requeridos para mostrarse	- Datos mantenimiento - Datos detalle mantenimiento	- Informe de mantenimiento	Encuentra los registros de los mantenimientos deseados para poder emitir un informe de estos.

Tabla 22 (Diccionario de datos – Procesos Nivel 2 Mantenimiento)

#### ALMACEN DE DATOS

<b>ALMACEN</b>	<b>DESCRIPCION</b>	<b>FLUJO ENTRADA</b>	<b>FLUJO PROPORCIONADO</b>	<b>DESCRIPCION DE DATOS</b>	<b>VOLUMEN</b>	<b>ACCESO</b>
tbl_ad_mantenimiento	Guarda información de los tanques y solicitudes de mantenimiento de la empresa	- Datos mantenimiento validados	- Datos mantenimiento actualizados	- Man_id - Man_req_urg - Man-observaciones - Man_fecha - Dep_id	60/ mes	diario

ALMACEN	DESCRIPCION	FLUJO ENTRADA	FLUJO PROPORCIONADO	DESCRIPCION DE DATOS	VOLUMEN	ACCESO
tbl_ad_mantenimiento_det	Guarda información de los detalles de los mantenimientos de la empresa	- Datos de detalle validados	- Datos de detalle actualizados	- Mad_id - Mad_actividad - Mad_fecha_proa - Mad_fecha_eje - Mad_km - Cle_ci_ric - Mad_estado_antes - Mad_estado_después - Mad_costo - Mad_factura - Man_id - Veh_id - Cantidad - Unidad - claveParametro	60/ mes	diario

Tabla 23 (Diccionario de datos – Almacenes Nivel 2 Mantenimiento)

### ESTRUCTURAS DE DATOS

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Datos de vehículos a mantener	Contiene información de los vehículos a dar mantenimiento	Consultar vehículos a mantener (5.1)	Ordenar Mantenimiento (5.2)	- Identificador de la Unidad	28/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos de mantenimiento	Contiene información del nuevo mantenimiento	Añadir mantenimiento (5.4)	Validar datos mantenimiento (5.6)	- Man_id - Man_req_urg - Man-observaciones - Man_fecha - Dep_id	28/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos de detalle de mantenimiento	Contiene información del nuevo detalle de mantenimiento	Añadir detalle mantenimiento (5.5)	Validar datos mantenimiento (5.6)	- Mad_id - Mad_actividad - Mad_fecha_proa - Mad_fecha_eje - Mad_km - Cle_ci_ric - Mad_estado_antes - Mad_estado_después - Mad_costo - Mad_factura - Man_id - Veh_id - Cantidad - Unidad - claveParametro	28/mes

<b>DENOMINACION</b>	<b>DESCRIPCION</b>	<b>ORIGEN</b>	<b>DESTINO</b>	<b>ESTRUCTURA</b>	<b>VOLUMEN</b>
Datos de mantenimiento validados	Contiene información de los datos de mantenimiento	Validar datos de mantenimiento	Tbl_ad_mantenimiento	- Man_id - Man_req_urg - Man-observaciones	10/mes

	previamente validados	(5.6)		- Man_fecha - Dep_id	
--	-----------------------	-------	--	-------------------------	--

DENOMINACION	DESCRIPCION	ORIGEN	DESTINO	ESTRUCTURA	VOLUMEN
Datos de detalle validados	Contiene información de los datos del detalle del mantenimiento previamente validados	Validar datos de mantenimiento (5.6)	Tbl_ad_mantenimiento_det	- Mad_id - Mad_actividad - Mad_fecha_proa - Mad_fecha_eje - Mad_km - Cle_ci_ric - Mad_estado_antes - Mad_estado_después - Mad_costo - Mad_factura - Man_id - Veh_id - Cantidad - Unidad - claveParametro - Valor del despacho individual	10/mes

Tabla 24 (Diccionario de datos – Estructuras Nivel 2 Mantenimiento)



6. Se controla el kilometraje antes de permitir un tanqueo?	
7. Cada cuánto tiempo se ingresa la información de kilometraje?	
8. Cómo y quién determina la eliminación de vehículos?	
9. Cuándo se adquieren nuevos vehículos lo asignan a un departamento?	
10. Se permite que un vehículo se pueda asignar a un nuevo departamento?	
11. Cómo se designa el chofer a un vehículo?	
12. Se puede cambiar el chofer a un determinado vehículo?	
13. La empresa tiene almacenados lubricantes, neumáticos u otros elementos o se los adquiere el momento del mantenimiento?	
14. A qué departamento responde el área vehicular?	
15. Cada cuánto tiempo envían informes al departamento antes mencionado?	



ANEXO 3

DESPACHO DE COMBUSTIBLE

**EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO**  
AMBATO

**NOTA DE PEDIDO**      **Nº. 003009**

Ambato..... del 200.....

Señores..... Unidad Nº.....

---

.....	Galones SUPER	USD. ....	USD. ....
.....	Galones EXTRA	USD. ....	USD. ....
.....	Galones DIESEL	USD. ....	USD. ....
.....	.....	USD. ....	USD. ....
.....	.....	USD. ....	USD. ....
.....	.....	USD. ....	USD. ....

**TOTAL** \_\_\_\_\_

FIRMA AUTORIZADA                      VTO. BNO.                      RECIBI

---

# ANEXO 4

## SOLICITUD DE MATERIALES (MANTENIMIENTO)

**EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO**  
**SOLICITUD DE MATERIALES O EQUIPOS** N° 00210 D.A.  
AMBATO - ECUADOR

Unidad Solicitante: \_\_\_\_\_ Fecha: \_\_\_\_\_

Señor GUARDALMACEN:  
Sírvese proveer a la Unidad a mi cargo, los artículos siguientes:

CANTIDAD	UNIDAD	DESCRIPCION DEL ARTICULO

Observaciones: \_\_\_\_\_  
\_\_\_\_\_

Jefe de Unidad Solicitante \_\_\_\_\_

Gráficas A&B Telf.: 2422806-Ambato

ANEXO 5

ORDEN DE MOVILIZACIÓN (SALVOCONDUCTOS)

**EMPRESA MUNICIPAL DE AGUA POTABLE Y ALCANTARILLADO AMBATO**

**ORDEN DE MOVILIZACIÓN** N° **02319**

 El suscrito, Gerente de la Empresa Municipal de Agua Potable y Alcantarillado de Ambato, en uso de las facultades que le concede la Ley, confiere la presente Orden de Movilización.

Identificación del vehículo: Marca: Chevrolet Iuv

Tipo: D- Max

N° de Placas: Sin Placas - En trámite

Departamento Responsable: Financiero

Razón de la Comisión: Trámites Varios

Procedencia: Ambato Destino: Quito

Nombre del Conductor: Byron Vallejo C.I. 180210720-9

Tiempo de Duración de la Comisión: Un día

Desde: 01 de Agosto 2006 Hasta: XXXX XXXX XXXX

Ambato, a 31 de Julio del 2006

  
GERENTE DE EMAPA

Elaborado Por:

