



# Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas

Título: Sistema de Planificación del Presupuesto del Departamento de  
Transporte de la Unidad de Aseguramiento Aduanal.

**Autor: Esteban Leiseca Hernández.**

**Tutor: Ing. Julio Cesar Bravo Rodríguez.**

**Cuidad de la Habana, junio del 2010.**

**“Año 52 de la Revolución”**

## DECLARACIÓN DE AUTORÍA

Declaro ser autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

---

Autor

Esteban Leiseca Hernández

---

Tutor

Ing. Julio Cesar Bravo Rodríguez



*El éxito no se logra sólo con cualidades especiales. Es sobre todo un trabajo de constancia, de método y de organización.*

*J. P. Sergent*

### AGRADECIMIENTOS

*Agradezco en primer lugar a la Revolución cubana y a nuestro querido Cde. Fidel Castro Ruz por darme la oportunidad de estudiar esta carrera.*

*A mis padres por haber confiado en mí y brindarme su amor y su apoyo.*

*A mi tutor por toda la ayuda que me ha brindado.*

*A todas las amistades que tengo, de aquí y de allá, las que estuvieron, las que están y estarán, las que me han acompañado a lo largo de este camino, sin ustedes no hubiese sido posible la realización de este trabajo, a todos, de corazón, muchísimas gracias.*

DEDICATORIA

*A la memoria de mi abuelo Enrique, mis abuelas Eneida y Emelina, y tía abuela Norka, por todo el cariño y la educación que me dieron, suficiente para llevarlos eternamente en mi corazón.*

*A mis padres, por haber hecho de mí la persona que soy, por ser ejemplos a seguir en todo momento y quererme tanto.*

*A Tita, por ser mi otra mamá.*

*A mi abuelo Romelio y tíos abuelos Esteban y Cuco, por brindarme siempre su apoyo incondicional.*

*A mis tíos Julio y Romelito, siempre presentes cuando los necesito.*

### RESUMEN

En el Departamento de Transporte de la Unidad de Aseguramiento Aduanal (UAA) perteneciente a la Aduana General de la República (AGR) se maneja gran volumen de información en la planificación del presupuesto necesario para mantener organizada, planificada y controlada la explotación, mantenimiento, reparación y registro de su transporte tanto naval como terrestre. El manejo de esta información representa una pérdida considerable de tiempo y evidencia un insuficiente control sobre la misma. La documentación no se encuentra almacenada en formato digital lo que impide que el proceso de toma de decisiones se realice de forma oportuna ya que disminuye la velocidad de acceso a los datos y aumenta el riesgo de pérdida de la información. Con el objetivo de automatizar la gestión de dicha información se pretende desarrollar un Sistema de Gestión que controle el proceso de planificación del presupuesto del departamento de la UAA y permita que este se lleve a cabo con organización y rapidez.

<b>INTRODUCCIÓN .....</b>	<b>1</b>
<b>CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA .....</b>	<b>5</b>
<b>1.1. Introducción .....</b>	<b>5</b>
<b>1.2. ¿Qué es un presupuesto?.....</b>	<b>5</b>
1.2.1. Objetivos del presupuesto.....	7
1.2.2. Funciones del presupuesto .....	7
1.2.3. Importancia de los presupuestos .....	8
1.2.4. Ventajas que brindan los presupuestos .....	8
1.2.5. Desventajas que brindan los presupuestos .....	9
1.2.6. Limitaciones de los presupuestos .....	9
1.2.7. Clasificación de presupuesto .....	10
1.2.8. Planificación de los presupuestos.....	11
1.2.9. Cálculo de los presupuestos .....	12
1.2.10. Control presupuestario .....	13
1.2.11. La administración del presupuesto del estado cubano .....	13
1.2.12. El sistema del presupuesto del estado cubano .....	14
<b>1.3. Importancia de las aduanas para un país .....</b>	<b>16</b>
1.3.1. Aduana cubana .....	17
1.3.2. Unidad de aseguramiento aduanal perteneciente a la agr .....	17
1.3.3. Departamento de transporte perteneciente a la uaa .....	18
<b>1.4. Sistemas.....</b>	<b>20</b>
1.4.1. Sistemas de información.....	21
<b>1.5. Tecnologías de la información .....</b>	<b>21</b>
<b>1.6. ¿Por qué realizar este trabajo? .....</b>	<b>22</b>
<b>1.7. ¿Por qué la solución a través de una aplicación web? .....</b>	<b>23</b>
<b>1.8. Sistemas informatizados existentes.....</b>	<b>23</b>
<b>1.9. Descripción de la solución .....</b>	<b>26</b>
<b>1.10. Metodologías y herramientas de diseño utilizadas en el mundo.....</b>	<b>26</b>
1.10.1. Metodologías de diseño .....	27
1.10.2. Patrones de diseño .....	28
1.10.3. Herramienta de diseño .....	31
1.10.4. Herramientas escogidas para desarrollar este trabajo .....	31

<b>1.11. Herramientas utilizadas para desarrollar el sistema .....</b>	<b>32</b>
1.11.1. Lenguajes de programación.....	32
1.11.2. Frameworks.....	38
1.11.2.1. Frameworks javascript .....	38
1.11.2.2. Frameworks php.....	42
1.11.2.3. Sistemas gestores de bases de datos .....	46
<b>1.12. Conclusiones parciales.....</b>	<b>49</b>
 <b>CAPITULO 2: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA .....</b>	<b>50</b>
<b>2.1. Introducción .....</b>	<b>50</b>
<b>2.2. Diseño .....</b>	<b>50</b>
<b>2.3. Modelo del diseño.....</b>	<b>50</b>
<b>2.4. Modelado mediante estereotipos web.....</b>	<b>51</b>
2.4.1. Relaciones entre los elementos de diseño .....	51
<b>2.5. Diagramas de clases del diseño .....</b>	<b>52</b>
<b>2.6. Diagramas de interacción del diseño .....</b>	<b>53</b>
2.6.1. Diagramas de secuencia.....	54
2.6.2. Diagramas de colaboración .....	56
<b>2.7. Diseño de la base de dato.....</b>	<b>56</b>
<b>2.8. Principios de diseño que se aplican.....</b>	<b>57</b>
<b>2.9. Tratamiento de errores.....</b>	<b>58</b>
<b>2.10. Estándar de codificación .....</b>	<b>58</b>
<b>2.11. Comunicación entre capas .....</b>	<b>59</b>
2.11.1 Ejemplo de comunicación entre capas .....	60
<b>2.12. Modelo de implementación.....</b>	<b>63</b>
2.12.1 Diagrama de despliegue .....	64
2.12.2. Diagrama de componente.....	64
<b>2.13. Conclusiones parciales.....</b>	<b>65</b>
 <b>CONCLUSIONES GENERALES.....</b>	<b>66</b>

<b>RECOMENDACIONES.....</b>	<b>67</b>
<b>BIBLIOGRAFÍA .....</b>	<b>68</b>
<b>GLOSARIO DE TÉRMINOS.....</b>	<b>70</b>

## ÍNDICE DE FIGURAS

<b>Figura 1 Implementación del Patrón MVC por Symfony.....</b>	<b>29</b>
<b>Figura 2 Ejemplo de Implementación del patrón Decorator en Symfony.....</b>	<b>31</b>
<b>Figura 3 Composición de la estructura de Symfony por otros Frameworks.....</b>	<b>43</b>
<b>Figura 4 Relaciones entre los elementos de diseño.....</b>	<b>52</b>
<b>Figura 5 Diagrama de clases del diseño Gestionar Vehículo. ....</b>	<b>52</b>
<b>Figura 6 Diagrama de secuencia Insertar Vehículo.....</b>	<b>54</b>
<b>Figura 7 Diagrama de secuencia Modificar Vehículo.....</b>	<b>55</b>
<b>Figura 8 Diagrama de secuencia Mostrar Vehículo.....</b>	<b>55</b>
<b>Figura 9 Modelo entidad relación.....</b>	<b>57</b>
<b>Figura 10 Pantalla Principal.....</b>	<b>60</b>
<b>Figura 11 Pantalla Adicionar Vehículo.....</b>	<b>61</b>
<b>Figura 12 Código para cargar un combo mediante un objeto JSON .....</b>	<b>61</b>
<b>Figura 13 Mensaje de error para datos incorrectos.....</b>	<b>62</b>
<b>Figura 14 Pantalla con los datos correctos.....</b>	<b>62</b>
<b>Figura 15 Pantalla de espera al aceptar los datos de la demanda.....</b>	<b>63</b>
<b>Figura 16 Diagrama de despliegue.....</b>	<b>64</b>
<b>Figura 17 Diagrama de Componentes.....</b>	<b>65</b>

## INTRODUCCIÓN

La gestión de empresas enfrenta constantemente nuevos retos, y si le sumamos la situación del mundo actual, donde predomina la crisis económica, se convierte en una carrera de obstáculos. Aunque la estadística, la computación y la ingeniería están muy desarrolladas casi nunca se posee toda la información que se necesita para tomar decisiones, esto podría solucionarse con un sistema de gestión, que proporcione una plataforma de tecnología en la que las organizaciones puedan integrar y coordinar sus principales procesos internos de negocios.

Entre los desafíos que presentan las empresas actualmente está lograr una buena planificación de su presupuesto ya que parte de su éxito depende mucho de esta actividad.

El presupuesto es un plan de acción dirigido a cumplir una meta prevista, expresada en valores y términos financieros que, debe cumplirse en determinado tiempo y bajo ciertas condiciones previstas, utilizado como medio administrativo de determinación adecuada de capital, costos e ingresos necesarios en una organización. Son sumamente importantes ya que ayudan a minimizar el riesgo en las operaciones de la organización, mantiene el plan de operaciones de la empresa en límites razonables, sirve como mecanismo de la revisión de políticas y estrategias de la organización y direccionarla a lo que realmente se busca, así como la debida utilización de los recursos disponibles acorde con las necesidades de cada una de las unidades y/o departamentos. Este instrumento también sirve de ayuda para la determinación de metas que sean comparables a través del tiempo, coordinando así las actividades de los departamentos a la consecución de estas, evitando costos innecesarios y mala utilización de recursos. De igual manera permite a la administración conocer el desenvolvimiento de la empresa, por medio de la comparación de los hechos y cifras reales con los hechos y cifras presupuestadas y/o proyectadas para poder tomar medidas que permitan corregir o mejorar la actuación organizacional.

Un proceso presupuestario eficaz depende de muchos factores; sin embargo, existen dos que pueden tener la consideración de "requisitos imprescindibles". Así, por un lado, es necesario que la empresa tenga configurada una estructura organizativa clara y coherente, a través de la que se vertebrará todo el proceso de asignación y delimitación de responsabilidades. Un programa presupuestario será más eficaz en tanto se puedan asignar adecuadamente las responsabilidades, para lo cual, necesariamente, tendrá que contar con una estructura organizativa perfectamente definida. El otro requisito viene determinado por la

repercusión que, sobre el proceso presupuestario, tiene la conducta del potencial humano que interviene en el mismo; esto es, el papel que desempeñan dentro del proceso de planificación y presupuestario los factores de motivación y de comportamiento. El presupuesto, además de representar un instrumento fundamental de optimización de la gestión a corto plazo, constituye una herramienta eficaz de participación del personal en la determinación de objetivos, y en la formalización de compromisos con el fin de fijar responsabilidades para su ejecución. Esta participación sirve de motivación a los individuos que ejercen una influencia personal, confiriéndoles un poder decisorio en sus respectivas áreas de responsabilidad.

El proceso de planificación presupuestaria de la empresa varía mucho dependiendo del tipo de organización que se trate, sin embargo, con carácter general, se puede afirmar que consiste en un proceso secuencial. Permite a las empresas, los gobiernos, las organizaciones privadas y las familias establecer prioridades y evaluar la consecución de sus objetivos, así como ayudar en la toma de decisiones. En general, un sistema de gestión contable y financiera es una herramienta de apoyo que proporciona información en tiempo real con la cual se pueden tomar mejores decisiones. La informática bien aplicada ayuda a controlar y a reducir los gastos administrativos, sobre todo para organizaciones sin demasiados recursos. La implantación de soluciones de gestión específica, completa y avanzada, son capaces de aportar beneficios como:

- La reducción de tiempos en los cierres anuales y en la conciliación de cuentas bancarias.
- La comodidad en la introducción de asientos contables y de amortizaciones, facilidad a la hora de realizar las declaraciones.
- Mayor autonomía en la generación y gestión de la información, y eficacia en la toma de decisiones.
- La seguridad en la contabilización y el control de los cobros y pagos.
- Mejor planificación presupuestaria, trazabilidad de los documentos de cobro y pago, la optimización del beneficio financiero-fiscal, o la disponibilidad de múltiples informes etc.

Todo ello con la garantía de confidencialidad y seguridad de los datos, con protecciones por usuarios y grupos.

En la actualidad el departamento de transporte de la UAA no cuenta con un sistema propio de aplicaciones Web o cualquier otro sistema informatizado que controle de manera íntegra el proceso de planificación de su presupuesto, este proceso se realiza de forma manual, empleando como soporte el papel, lo cual puede acarrear la pérdida de datos y la falta de toda la información necesaria para la toma de decisiones oportunamente, además de que en múltiples ocasiones la velocidad de acceso a la información requerida no es la más adecuada; asimismo no se dispone del control absoluto de la totalidad del personal movilizado.

Por todas las ventajas mencionadas anteriormente que nos proporcionan los sistemas de gestión y teniendo en cuenta los problemas actuales que presenta la UAA , y por la necesidad de tener nuestro propio sistema de aplicaciones Web que gestione los procesos de la planificación del presupuesto, que sea configurable para muchas de las entidades cubanas y responda a los intereses y requerimientos de una economía de carácter socialista como la nuestra, se ha decidido realizar un análisis de la planificación del presupuesto del departamento de transporte perteneciente a la UAA, con el objetivo de lograr en un futuro muy próximo diseñar e implementar un sistema Web que logre la integridad de toda la información necesaria para dicha planificación y que además facilite la toma de decisiones de los directivos.

Debido a esto se identifica la siguiente situación problemática: En el Departamento de Transporte de la Unidad de Aseguramiento Aduanal (UAA) perteneciente a la Aduana General de la República (AGR) se maneja la información referente a la planificación del presupuesto necesario para mantener organizada, planificada y controlada la explotación, mantenimiento, reparación y registro de su transporte tanto naval como terrestre. El manejo de esta información representa una pérdida considerable de tiempo y evidencia un insuficiente control sobre la misma. La documentación no se encuentra almacenada en formato digital lo que impide que el proceso de toma de decisiones se realice de forma oportuna ya que disminuye la velocidad de acceso a los datos y aumenta el riesgo de pérdida de la información. Después de haber capturado los requisitos y hacer un análisis del sistema se necesita diseñar e implementar una aplicación la cual de solución a la situación antes expuesta.

Luego de hacer un análisis detallado de las dificultades existentes, se plantea el problema a resolver: Como desarrollar el diseño e implementación de un sistema informatizado que permita el control sobre la información referente a la planificación de presupuesto del Departamento de Transporte de la Unidad de

Aseguramiento Aduanal (UAA). En correspondencia con esto, el objeto de estudio está enmarcado hacia los procesos de diseño y la implementación.

El campo de acción abarca los procesos de diseño e implementación del módulo de planificación del presupuesto en el área de transporte de la Unidad de Aseguramiento Aduanal perteneciente a la Aduana General de la República.

Los objetivos generales de este trabajo son: Diseñar e implementar un Sistema que permita gestionar la información asociada a la Planificación del Presupuesto del Departamento de Transporte de la Unidad de Aseguramiento Aduanal.

Obteniendo así como posibles resultados: Un sistema que permita de forma automatizada la gestión de la planificación del presupuesto de departamento de transporte de la unidad de aseguramiento aduanal. Para poder darle cumplimiento a los objetivos planteados se proponen las siguientes Tareas de la Investigación:

- Estudio de las herramientas para darle cumplimiento a los objetivos.
- Modelar el diseño de la aplicación.
- Realizar el modelo de Datos.
- Implementar las vistas del Diseño.
- Implementar el modelo escogido.

## CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

### 1.1. Introducción

En una época de cambios en la gerencia moderna, la toma de decisiones exige presión y rapidez, y por lo tanto el factor de predicción y control de los presupuestos es de vital importancia como una eficiente herramienta administrativa.

En este capítulo se darán argumentos de por qué la motivación de realizar este trabajo. Se abordarán importantes conceptos necesarios para el entendimiento del mismo. Además se hace una valoración de los artefactos y herramientas que serán utilizados.

### 1.2. ¿Qué es un presupuesto?

El presupuesto es un plan de acción dirigido a cumplir una meta prevista, expresada en valores y términos financieros que, debe cumplirse en determinado tiempo y bajo ciertas condiciones previstas.

Los fundamentos teóricos y prácticos del presupuesto, como herramienta de planificación y control, tuvieron su origen en el sector gubernamental a finales del siglo XVIII cuando se presentaba al Parlamento Británico los planes de gastos del reino y se daban pautas sobre su posible ejecución y control.

Desde el punto de vista técnico de la palabra se deriva del francés antiguo bougette o bolsa. Dicha acepción intentó perfeccionarse posteriormente en el sistema inglés con el término budget de conocimiento común y que recibe en el idioma de la lengua española la denominación de presupuesto. En 1820 Francia adopta el sistema en el sector gubernamental y los Estados Unidos lo acogen en 1821 como elemento de control del gasto público y como base en la necesidad formulada por funcionarios cuya función era presupuestar para garantizar el eficiente funcionamiento de las actividades gubernamentales.

Entre 1912 y 1925, y en especial después de la Primera Guerra Mundial, el sector privado notó los beneficios que podía generar la utilización del presupuesto en materia de control de gastos, y destino recursos en aquellos aspectos necesarios para obtener márgenes de rendimiento adecuados durante un ciclo de operación determinado. En este período las industrias crecen con rapidez y se piensa en el empleo de métodos de planeación empresarial apropiados. En la empresa privada se habla intensamente

de control presupuestario, y en el sector público se llega incluso a aprobar una Ley de Presupuesto Nacional. La técnica siguió su continua evolución, junto con el desarrollo alcanzado por la contabilidad de costos.

En 1930 se celebra en Ginebra el Primer Simposio Internacional de Control Presupuestal, en que se definen los principios básicos del sistema.

En 1948 el Departamento de Marina de los Estados Unidos presenta el presupuesto por programas y actividades.

En 1961 el Departamento de Defensa de los Estados Unidos trabaja con un sistema de planificación por programas y presupuestos.

En 1965 el gobierno de los Estados Unidos crea el Departamento de Presupuesto e incluye en las herramientas de planeación y control del sistema conocido como “planeación por programas y presupuestos”.

El papel desempeñado por los presupuestos lo destacan prestigiosos tratadistas modernos quienes señalan lo vital de los pronósticos relacionados con el manejo de efectivo, por lo que puede considerarse una parte importante del clásico ciclo administrativo de planear, actuar y controlar o, más específicamente, como parte de un sistema total de administración que incluye:

- Formulación y puesta en práctica de estrategias.
- Sistemas de Planeación.
- Sistemas Presupuestales.
- Organización.
- Sistemas de Producción y Mercadotecnia.
- Sistemas de información y control.

En la práctica, los sistemas administrativos no se encuentran tan claramente definidos y a menudo se entrelazan con mayor o menor grado. En empresas pequeñas no es raro que el mismo grupo maneje tanto la contabilidad como los presupuestos.

Es evidente la íntima relación entre la planeación y el presupuesto y no es raro encontrar que utilizan indistintamente términos como presupuesto, plan anual de la empresa. Como se dijo, el presupuesto es resultado de algún tipo de plan o está basado en él, sea éste explícito o bien algo que se encuentra en las mentes de los directivos.

## **1.2.1. Objetivos del presupuesto**

- Planear integral y sistemáticamente todas las actividades que la empresa debe desarrollar en un periodo determinado.
- Controlar y medir los resultados cuantitativos, cualitativos y, fijar responsabilidades en las diferentes dependencias de la empresa para lograr el cumplimiento de las metas previstas.
- Coordinar los diferentes centros de costo para que se asegure la marcha de la empresa en forma integral.

## **1.2.2. Funciones del presupuesto**

Las funciones que desempeñan los presupuestos dependen en gran parte de la propia dirección de la empresa. Las necesidades y expectativas de los directivos y el uso que hagan de los presupuestos, están fuertemente influidos por una serie de factores relacionados con los antecedentes gerenciales y con el sistema empresarial. De manera ideal la dirección espera que la función presupuestal proporcione:

- Una herramienta analítica, precisa y oportuna.
- La capacidad para pretender el desempeño.
- El soporte para la asignación de recursos.
- La capacidad para controlar el desempeño real en curso.
- Advertencia de las desviaciones respecto a los pronósticos.

- Indicios anticipados de las oportunidades o de los riesgos venideros.
- Capacidad para emplear el desempeño pasado como guía o instrumento de aprendizaje.
- Concepción comprensible, que conduzca a un consenso y al respaldo de presupuesto anual.

### **1.2.3. Importancia de los presupuestos**

Las organizaciones hacen parte de un medio económico en el que predomina la incertidumbre, por ello deben planear sus actividades si pretenden sostenerse en el mercado competitivo, puesto que cuanto mayor sea la incertidumbre, mayores serán los riesgos por asumir.

Es decir, cuanto menor sea el grado de acierto de predicción o de acierto, mayor será la investigación que debe realizarse sobre la influencia que ejercerá los factores no controlables por la gerencia sobre los resultados finales de un negocio. Esto se constata en los países latinoamericanos que por razones de manejo macroeconómico en la década de los años ochenta experimentaron fuertes fluctuaciones en los índices de inflación y devaluación y en las tasas de interés.

El presupuesto surge como herramienta moderna del planteamiento y control al reflejar el comportamiento de indicadores económicos como los enunciados y en virtud de sus relaciones con los diferentes aspectos administrativos contables y financieros de la empresa.

### **1.2.4. Ventajas que brindan los presupuestos**

- De manera periódica se replantean las políticas si después de revisarlas y evaluarlas se concluye que no son adecuadas para alcanzar los objetivos propuestos.
- Ayuda a la planeación adecuada de los costos de producción.
- Se provoca optimizar resultados mediante el manejo adecuado de los recursos.
- Se crea la necesidad de idear medidas para utilizar con eficiencia los limitados recursos de la empresa, dado el costo de los mismos.

- Es el sistema más adecuado para establecer "costos promedios" y permite su comparación con los costos reales, mide la eficiencia de la administración en el análisis de las variaciones y sirve de incentivo para actuar con mayor efectividad.
- Facilita la vigilancia efectiva de cada una de las funciones y actividades de la empresa.

### **1.2.5. Desventajas que brindan los presupuestos**

- Sus datos al ser estimados estarán sujetos al juicio o la experiencia de quienes los determinaron.
- Su implantación y funcionamiento necesita tiempo, por tanto, sus beneficios se tendrán después del segundo o tercer periodo cuando se haya ganado experiencia y el personal que participa en su ejecución esté plenamente convencido de las necesidades del mismo.

### **1.2.6. Limitaciones de los presupuestos**

En la realidad vivida por las empresas pocas veces se alcanza el ideal. Los bruscos cambios en el medio ambiente, las nuevas disposiciones legales y los acontecimientos mercantiles inesperados tales como huelgas y accidentes pueden sacar de balance al proceso. El cuerpo directivo pasa la mayor parte de su tiempo "apagando fuegos", en vez de conducir a la empresa como es debido.

Las razones por las cuales los presupuestos y la planeación no tienen más éxito son, entre otras, las siguientes:

- La incapacidad de la dirección para comprender su sistema.
- Los directivos se preocupan por el detalle o tratan de controlar los factores equivocados.
- La falta de respaldo por parte de la dirección al sistema presupuestal.
- La dirección general no apoya a la actividad o no participa en esta con sentido.
- El compromiso excesivo del proceso presupuestal o la indebida dependencia en el mismo, es decir, tratar de sustituir el juicio por los presupuestos.

## 1.2.7. Clasificación de presupuesto

### Según la flexibilidad

- Rígidos, estáticos, fijos o asignados: Son aquellos que se elaboran para un único nivel de actividad y no permiten realizar ajustes necesarios por la variación que ocurre en la realidad. Dejan de lado el entorno de la empresa (económico, político, cultural etc.). Este tipo de presupuestos se utilizaban anteriormente en el sector público.
- Flexibles o variables: Son los que se elaboran para diferentes niveles de actividad y se pueden adaptar a las circunstancias cambiantes del entorno. Son de gran aceptación en el campo presupuestario moderno. Son dinámicos adaptativos, pero complicados y costosos.

### Según el periodo de tiempo

- A corto plazo: Son los que se realizan para cubrir la planeación de la organización en el ciclo de operaciones de un año. Este sistema se adapta a los países con economías inflacionarias.
- A largo plazo: Este tipo de presupuestos corresponden a los planes de desarrollo que, generalmente, adoptan los estados y grandes empresas.

### Según el campo de aplicación en la empresa

- De operación o económicos: Tienen en cuenta la planeación detallada de las actividades que se desarrollarán en el periodo siguiente al cual se elaboran y, su contenido se resume en un Estado de Ganancias y Pérdidas.

Entre estos presupuestos se pueden destacar:

- Presupuestos de Ventas: Generalmente son preparados por meses, áreas geográficas y productos.

- Presupuestos de Producción: Comúnmente se expresan en unidades físicas. La información necesaria para preparar este presupuesto incluye tipos y capacidades de maquinas, cantidades económicas a producir y disponibilidad de los materiales.
- Presupuesto de Compras: Es el presupuesto que prevé las compras de materias primas y/o mercancías que se harán durante determinado periodo. Generalmente se hacen en unidades y costos.
- Financieros: En estos presupuestos se incluyen los rubros y/o partidas que inciden en el balance. Hay dos tipos: 1) el de Caja o Tesorería y 2) el de Capital o erogaciones capitalizables.

## Según el sector de la economía en la cual se utiliza

- Presupuesto del Sector Público: Son los que involucran los planes, políticas, programas, proyectos, estrategias y objetivos del Estado. Son el medio más efectivo de control del gasto público y en ellos se contempla las diferentes alternativas de asignación de recursos para gastos e inversiones.

El sector público ejecuta una gran cantidad de operaciones de ingresos y gastos para desarrollar sus actividades, de modo que para estudiarlas y apreciar su significado es preciso darle forma a la multitud de datos estadísticos que reflejan su sentido y alcance. Para ordenar estas informaciones, existe un conjunto de métodos de clasificación presupuestaria única y universalmente aceptada, pues los enfoques del análisis fiscal pueden ser diversos y requerir distintas formas de distribución de los datos.

Lo anterior significa que es posible preparar distintas cuentas gubernamentales en las que aparecen clasificados los ingresos y gastos de acuerdo con diversos criterios.

- Presupuestos del Sector Privado: Son los presupuestos que utilizan las empresas particulares como instrumento de su administración.

### **1.2.8. Planificación de los presupuestos**

Los primeros pasos para la confección de los presupuestos pueden ser en principio: la cifra de venta a conseguir o el volumen de producción.

Todo ello dependerá de la situación económica de la empresa al principio de cada ejercicio económico. En condiciones de excesos de mercancías o productos almacenados, las ventas y no la capacidad de producción, se convierte en el factor crítico; sería todo lo contrario si existiese una cartera de pedidos no suministrados.

Las cifras que figuran en los presupuestos serán establecidas por los respectivos directores funcionales (Ventas, Producción y Administración). Cuando se establecen los presupuestos de ventas se deben preparar simultáneamente los de producción y fabricación en términos de utilización de los recursos propios de la empresa, como son: los equipos de producción, la mano de obra directa al proceso productivo y las materias primas necesarias para completar el programa de fabricación.

Es importante distinguir entre los costes sobre los cuales se le puede exigir a un director funcional ejercer un cierto grado de control y los costes que surgen de las decisiones tomadas por la Dirección General. Los directores de departamento deben presupuestar y controlar los salarios, incentivos, consumos energéticos, materiales y otros gastos de naturaleza similar, y es razonable que estas partidas sean incluidas exclusivamente dentro del presupuesto específico de cada departamento.

Sin embargo, otros costes indirectos a los productos y/o servicios que presta la empresa, como los referidos a la Dirección General, Marketing, Recursos Humanos, etc. no deben ser imputados de forma particular a ningún departamento funcional, sino al conjunto operativo de la empresa. (Gómez, 2001)

## **1.2.9. Cálculo de los presupuestos**

La planificación y cálculo del presupuesto se realiza mediante un cuidadoso estudio de todos los gastos que la empresa debe soportar para hacer efectiva y rentable su gestión. Un sistema contable claro y dinámico debe ser introducido para analizar los gastos producidos y cargar estos al centro presupuestario que le corresponda. Cuanto más grande es la empresa, es más difícil asegurar que gastos deben ser imputados al centro presupuestario correcto. En estos casos los documentos justificativos del gasto son codificados mediante un número o dígito que facilita su contabilización en un sistema muy similar a la clasificación de los libros en las bibliotecas (CDU. Código Decimal Universal).

Con las cifras financieras reflejadas en el presupuesto anual, el gerente se apoyará en los datos que le aporta la contabilidad para estar informado periódicamente sobre la evolución económica de la empresa.

Este procedimiento logra su mayor eficacia cuando aporta un desglose detallado por cada uno de los centros presupuestarios.

Analizando los gastos por departamento podremos detectar las variaciones que se vayan produciendo, sobre todo, aquellos gastos excesivos que puedan afectar a la cuenta de resultados. La ventaja de este método es que refleja claramente, a través de las desviaciones detectadas, hasta qué punto los márgenes de eficacia se ven influenciados por el cumplimiento en el logro del presupuesto.

Medir o comprobar los costos presupuestados se realizan con más detalle, si analizamos individualmente a cada una de las partidas de gastos asignadas a cada departamento.

### **1.2.10. Control presupuestario**

El concepto de control presupuestario permite que las actividades en la empresa sean planificadas con antelación y referidas a un periodo de tiempo determinado. Estos presupuestos estarán integrados por partidas de gastos que se consideran fijos y otros que deberán ser variables; los primeros por definición se consideran improbables que se vean alterados durante el periodo presupuestado, pero los segundos están sujetos a las fluctuaciones de los precios del mercado. Entre estos últimos estarían, por ejemplo, los precios de las materias primas, es decir, los procesos inflacionarios de costes.

En algunas industrias, por ejemplo, la elaboración de productos alimenticios, las fluctuaciones en los precios de origen de las materias primas pueden causar enormes trastornos en el control del presupuesto. Algunos de estos componentes están sujetos a las variaciones en calidad y cantidad, debido a los cambios climatológicos. Los precios pueden fluctuar al alza en porcentajes muy importantes y esto hace que sea casi imposible mantener un presupuesto válido para el departamento de producción. (Domínguez, 2006)

### **1.2.11. La administración del presupuesto del estado cubano**

El sistema presupuestario cubano se rige por el Decreto-Ley 192 del 8 de abril de 1999 "Administración Financiera del Estado." En el mismo se define que:

"La Administración Financiera del Estado es el conjunto de principios, normas, sistemas e instituciones y procedimientos administrativos que hacen posible la obtención de los recursos financieros públicos y su

aplicación a la consecución de los objetivos del estado, procurando la satisfacción de las necesidades colectivas de la sociedad."

En el artículo 9 de este documento se expresa:

"En el presupuesto del Estado se consigna el estimado de los recursos financieros y gastos previstos por el Estado para el año presupuestario destinados al desarrollo económico-social y al incremento del bienestar material. Los recursos financieros y los gastos figuran por separado y por sus montos íntegros, sin compensaciones entre sí".

Tal y como se define en el presupuesto, la creación de un fondo centralizado de recursos financieros dimana de las funciones inherentes al Estado cubano, las que originan la necesidad de la concentración de recursos y su utilización ulterior en el desarrollo económico-social del país. Se produce una distribución y redistribución centralizada de la renta nacional entre la esfera productiva e improductiva, entre las diferentes ramas de la economía, entre los diferentes territorios del país y entre los diferentes sectores sociales, es decir, entre el sector estatal, el cooperativo y el privado.

La redistribución a través del presupuesto está condicionada por la necesidad de transferir parte del ingreso nacional, o de los ingresos de las instituciones o individuos, a otras ramas o instituciones y entre territorios en correspondencia con el plan. Las proporciones de los ingresos que se distribuyen están determinadas por las diferentes tareas que el Estado tiene ante sí, por la no-uniformidad existente en la distribución territorial, la diferenciación de las riquezas materiales y naturales, la falta de correspondencia entre la magnitud de los ingresos y los gastos en la ampliación de la producción y de la sociedad en general.

El presupuesto, como plan financiero del país, tiene fuerza de ley, expresa las exigencias de las leyes económicas y en él se reflejan las directivas o lineamientos políticos y económicos que emanan de la dirección central y territorial del gobierno, además constituye el eslabón central del sistema financiero y está estrechamente vinculado a las finanzas de las empresas, de las ramas de la economía nacional, con el crédito bancario y con la circulación monetaria.

## **1.2.12. El sistema del presupuesto del estado cubano**

El Presupuesto del Estado está conformado por un sistema de presupuestos que incluye:

- El Presupuesto Central.
- El presupuesto para la Seguridad Social.
- Los presupuestos provinciales y el del municipio Isla de la Juventud.

En el presupuesto Central se centralizan los recursos financieros provenientes de las personas naturales o jurídicas cuya recaudación éste tenga derecho a percibir, en correspondencia con la política de distribución de estos recursos que se establezca legalmente. Además, cuenta con recursos financieros provenientes del crédito público, donaciones o dividendos.

Los recursos financieros de este Presupuesto se destinan al financiamiento del desarrollo económico y de actividades socio-culturales o científicas de carácter nacional, los tribunales, la fiscalía, la defensa y el orden interior.

El presupuesto de la Seguridad Social es un presupuesto independiente con recursos financieros y gastos específicos, sus recursos provienen mayoritariamente de la contribución a la seguridad social que efectúan los contribuyentes, de otros ingresos acorde a la legislación y puede recibir transferencias del Presupuesto Central para cubrir el desbalance entre sus gastos e ingresos.

El traslado de un conjunto de actividades productivas o de servicios a los órganos provinciales y municipales de gobierno, llevó aparejada la conformación de los presupuestos de cada nivel como base financiera esencial para que cada gobierno local diera cumplimiento a su papel de máxima autoridad estatal en los límites del territorio correspondiente.

El presupuesto de la Provincia es aquel del cual disponen los Consejos de la Administración Provincial (CAP) con el objetivo de financiar las actividades de carácter provincial, o sea, aquellas actividades que por su importancia o trascendencia no pueden enmarcarse dentro del presupuesto de un municipio específico. Este presupuesto es la suma del Presupuesto de la Provincia correspondiente y de los municipios que la componen.

Los presupuestos Municipales son aquellos de que disponen los Consejos de la Administración Municipal (CAM) para financiar el desarrollo económico y las actividades socio-culturales y científicas del municipio.

Los presupuestos de las Provincias y Municipios pueden contar con las siguientes fuentes de recursos financieros:

- Ingresos Cedidos.
- Participación en Ingresos del Presupuesto Central.
- Transferencias Directas.

Bajo la jurisdicción de la Asamblea Nacional del Poder Popular se encuentra el Presupuesto del Estado. A su vez, los presupuestos provinciales y municipales se encuentran jurídicamente dependientes de las Asambleas Provinciales y Municipales, respectivamente.

En el nivel central se concentra la función normativa de todo el sistema, el cual tiene la requerida maniobrabilidad para garantizar el respaldo financiero oportuno de tareas sociales, del desarrollo económico y social, y la necesaria equidad entre los diferentes territorios.

La concepción organizativa del Sistema Presupuestario condiciona su unidad, ya que cada presupuesto inferior se incluye en el superior formando un Presupuesto Estatal Único. (Villanueva)

### **1.3. Importancia de las aduanas para un país**

Los departamentos de Aduanas constituyen la piedra angular en la que se apoya la administración tributaria de muchos países. En algunos casos más del 50% de los presupuestos nacionales tienen su origen en las tasas arancelarias que recaudan en las aduanas de los pasos fronterizos. Desde ese punto de vista se comprenderá que la gestión de las Aduanas es de gran importancia estratégica para esos estados nacionales. En el mundo varias empresas se han dedicado a desarrollar herramientas informáticas para el control y administración de la gestión aduanera.

## **1.3.1. Aduana cubana**

La aduana cubana fue fundada hace 45 años y cuenta con más de 3000 trabajadores que dedican cada uno de sus días a proteger a nuestra sociedad y a facilitar el comercio con el extranjero.

La Aduana General de la República (AGR) es la encargada de regular el control aduanero aplicable a la entrada, el tránsito, el cabotaje, el trasbordo, el depósito y la salida del territorio nacional de mercancías, viajeros y sus equipajes, bienes y valores sujetos a regulaciones especiales, incluidas la flora y la fauna protegidas, así como los medios en que se transporten, además forma parte de la Administración del Estado y se subordina al Consejo de Ministros. Se designa a la Aduana como el órgano encargado de dirigir en materia aduanera, recaudar los derechos de aduanas y dar respuesta dentro de su jurisdicción y competencia a los hechos que incidan en el tráfico internacional de mercancías, viajeros, postal y los medios que los transportan, previniendo, detectando y enfrentando el fraude y el contrabando, así como contribuyendo a la protección nacional e internacional del medio ambiente.

La Aduana, para cumplir sus funciones se estructura y organiza verticalmente en un sistema que actúa bajo la dirección única de la Aduana General de la República y es independiente de todo órgano local. El Sistema de Órganos Aduaneros está integrado por la Aduana General de la República, las Delegaciones Territoriales de Aduanas y las Aduanas, cuyos niveles de dirección son:

- a) PRIMER NIVEL, integrado por el Jefe de la Aduana General de la República y los subjefes, quienes lo asisten participando en la dirección del Sistema.
- b) SEGUNDO NIVEL, en el que están comprendidos los Delegados Territoriales, los Jefes de las Aduanas Independientes, así como los Directores y los Jefes de Departamentos Independientes de la Aduana General de la República.
- c) TERCER NIVEL, en el que están comprendidos los Jefes de Aduanas y los Jefes de Departamentos de las Delegaciones Territoriales.

## **1.3.2. Unidad de aseguramiento aduanal perteneciente a la agr**

La Unidad Presupuestada denominada Unidad de Aseguramiento a la Aduana General de la República (UAA) ubicada en Ciudad de la Habana y subordinada a la Aduana General de la República se dedica a:

- Brindar servicios de abastecimientos técnico material al sistema de la Aduana General de la República, en moneda nacional.
- Ofrecer servicios de reparación de vehículos automotor al sistema de la Aduana General de la República, en moneda nacional.
- Prestar servicios de alojamiento no turístico y de recreación al sistema de la Aduana General de la República, en moneda nacional.
- Producir y comercializar de forma mayorista producciones agropecuarias, en moneda nacional al sistema de la Aduana General de la República.
- Brindar servicios de imprenta al sistema de la Aduana General de la República, en moneda nacional.
- Ofrecer servicios de comedor y realizar la venta minoristas de productos agropecuarios a sus trabajadores como parte de la atención al hombre, en moneda nacional.
- Prestar servicios de transportación de personal a los trabajadores, en moneda nacional.

### **1.3.3. Departamento de transporte perteneciente a la uaa**

El departamento de transporte perteneciente a la UAA tiene como misión organizar, planificar y controlar la explotación, mantenimiento, reparación y registro del Transporte Terrestre y Naval en el sistema de órganos aduaneros. Funciones del departamento de transporte:

- Organizar el sistema de explotación, mantenimiento y reparación del transporte terrestre y naval en el SOA, proponiendo las normas de utilización y consumo de los materiales durante el empleo de los equipos.
- Organizar el registro de medios técnicos del transporte terrestre y naval en el SOA, que permita conocer en todo momento los datos individuales de cada vehículo por marca, tipo y unidades.

- Organizar el sistema de registro y control, de la documentación del uso de la explotación del transporte terrestre y naval que posibilite la toma de decisión por parte de especialidad y la jefatura de la institución.
- Participar en la elaboración del presupuesto de la especialidad en el SOA, dirigiendo su objetivo a la consolidación del estado técnico de los vehículos.
- Recopilar la información necesaria a tareas de eventos, exposiciones y actividades con otras entidades afines que permitan la actualización con nueva técnica, equipamientos tecnológicos y otras modalidades de interés y posibilidades de la institución.
- Coordinar y tramitar con los organismos afines la obtención de la documentación necesaria según su designación para la legalización de su explotación.
- Analizar periódicamente con las unidades del SOA, de la necesidad de la tramitación de bajas de medios técnicos o equipamientos tecnológicos asesorándolo en su correcta elaboración.
- Realizar con la empresa de seguros nacionales el pago de la póliza de las flotas de vehículos sujetas a esta modalidad, tramitando con ASISTUR, las reclamaciones para el resarcimiento que se deriven durante la explotación de los equipos.
- Elaborar circulares y metodologías así como proponer indicaciones y resoluciones dirigiendo el mejoramiento del uso y conservación del transporte.
- Organiza y dirige el día de parqueo en la AGR y UAA y controla como se realiza este en resto de la unidad.
- Realizar visita de control y ayuda, así como controles sorpresivos al estado de la documentación y de conservación de los medios de transporte, activos fijos, e útiles y herramientas.
- Elaborar el plan anual y mensual de reparación, mantenimiento, copistería y pintura para el nivel central y UAA, insertando en este de ser necesario, vehículos de otras unidades del SOA.

- Organizar y dirigir la preparación técnica de las especialidades técnicas. Lograr que los choferes profesionales de la UAA, les tengan aprobado el examen psicológico y fisiológico establecido.
- Organizar la realización y participación de la especialidad en eventos sobre innovaciones racionalizaciones dirigida al mejoramiento del estado técnico del transporte para la cual cumplirá el cronograma que para esta actividad se consigna.
- Organizar que fluya la información necesaria de las unidades del SOA, afectadas con accidentes del tránsito, de forma tal que se permita elaborar la información complementaria de la institución. (AGR, 2006)

## 1.4. Sistemas

La palabra sistema tiene muchas acepciones. El análisis de sistemas tiene que ver sólo con un tipo concreto de sistemas, los sistemas de información, sistemas de información en un entorno empresarial y organizacional de gestión y dentro de ellos con los sistemas de información informatizados o basados en computadoras electrónicas.

### Conceptos de sistema

Se puede definir el concepto de Sistema de varias maneras:

- Conjunto de componentes que interactúan entre sí para lograr un objetivo común.
- Conjunto de elementos que interaccionan entre sí, orientados a la consecución de un objetivo común. Un sistema suele estar situado en un entorno o ambiente con el que interactúa, recibe entradas y produce salidas.
- Conjunto de personas, datos y procedimientos que trabajan coordinadamente. Para alcanzar el objetivo de apoyar el funcionamiento de la organización. Haciendo posible que la empresa mantenga contacto con el medio ambiente que la rodea, es decir con: Clientes, Proveedores, Organizaciones Estatales de contralor y recaudación de impuestos y aportes, Funcionarios, Accionistas o propietarios.

## 1.4.1. Sistemas de información

Cualquier organización o empresa es en sí un sistema. Su entorno es el sistema productivo en que se inserta, del que recibe una serie de entradas, en forma de demanda, y al que entrega una serie de salidas, u oferta. A su vez, dentro de él existen varios subsistemas, los distintos departamentos o áreas en que se divide la empresa. Para que los subsistemas que forman la organización funcionen coordinadamente, es necesario otro subsistema más, el sistema de información, que tiene como misión asegurar que la información necesaria fluya, dentro del sistema, de unos subsistemas a otros y que inevitablemente existe en cualquier empresa u organización.

Del buen funcionamiento de este subsistema depende en gran parte el éxito global del sistema. Los elementos de un sistema de información son muy variados; puede agruparse en: recursos físicos, como archivos, teléfonos, recursos humanos, documentación en general, y una serie de normas, procesos y procedimientos que determinan los flujos de información tanto internos como hacia y desde el exterior, así como el uso y administración de los recursos. Un sistema de información es un conjunto de elementos que interactúan entre sí con el fin de apoyar las actividades de una empresa o negocio.

Un sistema de información realiza cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información. Cuando un sistema de información cuenta entre sus recursos con computadoras electrónicas, en las que se basa parcial o totalmente, puede denominarse sistema de información informatizado.

## 1.5. Tecnologías de la información

Se denominan Tecnologías de la Información y las Comunicaciones (TIC) al conjunto de tecnologías que permiten la adquisición, producción, almacenamiento, tratamiento, comunicación, registro y presentación de informaciones, en forma de voz, imágenes y datos contenidos en señales de naturaleza acústica, óptica o electromagnética. Las TIC incluyen la electrónica como tecnología base que soporta el desarrollo de las telecomunicaciones, la informática y el audiovisual.

Las Tecnologías de la Información han sido conceptualizadas como la integración y convergencia de la computación, las telecomunicaciones y la técnica para el procesamiento de datos, donde sus principales componentes son: el factor humano, los contenidos de la información, el equipamiento, la infraestructura,

el software y los mecanismos de intercambio de información, los elementos de política y regulaciones, además de los recursos financieros.

Los Sistemas de Información (SI) y las Tecnologías de Información (TI) han cambiado la forma en que operan las organizaciones actuales. A través de su uso se logran importantes mejoras, pues automatizan los procesos operativos, suministran una plataforma de información necesaria para la toma de decisiones y, lo más importante, su implantación logra ventajas competitivas o reducir la ventaja de los rivales.

No parece haber ninguna duda sobre la importancia y efectividad de la informática como instrumento de modernización de cualquier organismo social; y la actividad aduanera no podría, por lo tanto, escapar a esta regla, ya sea en países donde sus aduanas, por diversos motivos, pasaron por un período de ostracismo o donde fueron mejor estructuradas y eficientes. En ambos casos, las tecnologías de información actúan como el principal y más efectivo instrumento para la modernización de estas entidades.

### **1.6. ¿Por qué realizar este trabajo?**

Es necesario en estos momentos un sistema informatizado que mejore el proceso de planificación del presupuesto del departamento de transporte de la UAA ya que este se realiza manual y representa una pérdida considerable de tiempo y un insuficiente control sobre la información. En estudios realizados no se ha encontrado un software internacional o nacional que se dedique específicamente a la planificación del presupuesto del transporte aduanero ni tampoco algún sistema que tenga esta funcionalidad dentro de uno de sus módulos, aunque esta es la causa principal por la cual se decide desarrollar este trabajo existen otras no menos importantes que también dan motivo a este proyecto como por ejemplo:

- Este sistema debe adaptarse a la economía cubana que en estos momentos tiene circulando doble monedas y esto no sucede en ningún otro país.
- Los sistemas internacionales cuestan una gran suma de dinero además de los pagos que hay que efectuar cada cierto tiempo por recibir el mantenimiento a la aplicación así como otras versiones más actualizadas.

- Es de vital importancia realizar un sistema que sea multiplataforma ya que el país está intentando migrar al software libre y la aduana ha sido la primera institución en dar este paso.

Estas son algunas de las razones por las que se pretende realizar el diseño e implementación de una aplicación Web que gestione el presupuesto que necesita para un año el departamento de Transporte de la UAA de Ciudad de la Habana.

## 1.7. ¿Por qué la solución a través de una aplicación web?

Una aplicación Web es aquella que los usuarios usan accediendo a un servidor Web a través de la red Internet o intranet. Las aplicaciones Web son populares debido a la practicidad del navegador como cliente ligero. La habilidad para actualizar y mantener aplicaciones Web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad. Dentro de sus principales ventajas están:

- Una empresa puede migrar de sistema operativo o cambiar el Hardware libremente sin afectar el funcionamiento de las aplicaciones de servidor.
- No se requieren complicadas combinaciones de Hardware/Software para utilizar estas aplicaciones. Solo un computador con un buen navegador Web.
- Se facilita el trabajo a distancia. Se puede trabajar desde cualquier PC o computador portátil con conexión a Internet.
- Actualizar o hacer cambios en el Software es sencillo y sin riesgos de incompatibilidades. Existe solo una versión en el servidor lo que implica que no hay que distribuirla entre los demás computadores. El proceso es rápido y limpio.
- Al funcionar en un navegador, se requiere un conocimiento básico de informática para utilizar una aplicación Web.

## 1.8. Sistemas informatizados existentes

En el plano internacional existen sistemas informáticos que se encargan de planificar presupuestos o especializados en la gestión de trasportes como por ejemplo:

**Presto** fue desarrollado por la empresa desarrolladora de productos informáticos Soft. Es el programa más utilizado para realizar presupuestos de edificación y obra civil. El usuario puede realizar fácilmente presupuestos y ofertas reutilizando información de proyectos anteriores o bases de datos de precios, permitiendo el ajuste y modificación de precios con diferentes sistemas automáticos. Es un sistema diseñado para Windows 2000/NT/XP ó Superior, con sus recursos más difundidos y avanzados:

- Edición simultánea de varios presupuestos o bases de datos de precios.
- Uso de arrastrar y soltar, copiar y pegar para traspasar información entre ventanas, entre presupuestos o con otros programas, como hojas de cálculo, editores de textos o sistemas CAD.
- Deshacer y rehacer ilimitados, múltiples sesiones y multiusuario.
- Configuración de ventanas, colores, fuentes, etc.
- Utilidades para copiar, pegar, renombrar, mover, eliminar, comprimir, descomprimir y enviar un presupuesto por correo electrónico sin abandonar Presto.
- Menús contextuales y barras de herramientas móviles con las opciones más utilizadas.

Este software a pesar de tener funcionalidades presupuestales no se ajusta a las necesidades que se requieren para resolver el problema de la planificación de presupuesto del departamento de transporte de la Unidad de Aseguramiento Aduanal de la Aduana General de la República, además que sus requisitos de instalación no cumplen con la política del país de migrar a plataforma libre para lograr la menor dependencia tecnológica.

**Wintrans** es una solución de software gratuito especializado en la gestión de transportes, creado para gestionar todas las operaciones de tráfico (aéreo, marítimo, terrestre), de aduanas y de facturación integrada a contabilidad. Contempla la gestión integral de cualquier estructura empresarial, desde la más compleja hasta la más simple.

El lenguaje de programación en que está desarrollado es VISUAL BASIC, uno de los más conocido por los profesionales de la informática, el mismo criterio se siguió para seleccionar la base de datos SQL

SERVER, complemento ideal de VISUAL BASIC, formando ambos productos uno de los entornos más fiables.

Aunque sea una aplicación útil para cualquier tipo de empresas, no es idónea para resolver el problema planteado porque además de presentar funcionalidades fuera de los objetivos, no cumple con el interés principal del departamento de transporte de la UAA de automatizar la planificación del presupuesto.

**Dig&See Quota** es una aplicación especializada en la creación de presupuestos y cotizaciones de obras, instalaciones y proyectos. Es capaz de ahorrarle una gran cantidad de cálculos y generar informes en un formato portable e imprimible.

### Requerimientos

Dig&See Quota está basado en tecnologías estándar Java TM, XML y es necesario que tenga instalado en su equipo el software siguiente:

- Soporte Java JRE 1.4.2 ó superior.

Permite visualizar la información de cualquier base de datos Access, Excel, MySQL, Oracle, Microsoft SQL Server o PostgreSQL de una manera sencilla, rápida e intuitiva, además de crear filtros para visualizar los datos que busca en menos tiempo. Sin olvidar la generación automática de las sentencias SQL que ahorrarán trabajo a los programadores que lo necesiten.

Esta aplicación aunque posibilita trabajar con funcionalidades para el manejo del presupuesto no es una herramienta que trabaje sobre plataforma libre por lo que nos hace dependiente tecnológicamente de las grandes transnacionales del software y no cumple los objetivos de la República de Cuba ni de su Aduana General de migrar a Software libre.

Y a nivel de aduana están:

El Sistema Aduanero Automatizado (SIDUNEA), Sistema de Información Aduanero (SIA), Sistema Informático Aduanero Uniforme Centroamericano (SIAUCA), Tecnología de la Información para el Control Aduanero (TICA), que implementan diversas funciones dentro del marco de la aduana como son prevenir, investigar y perseguir las infracciones a las legislaciones aduaneras, control y administración de la gestión

aduanera, controlar efectivamente la recaudación de los impuestos aduaneros, entre otras funcionalidades, sin embargo, ninguno aborda el tema de la planificación del presupuesto en el área de transporte.

Ninguno de estos Sistemas antes mencionado tanto a nivel de país como de Aduanas guarda semejanza con el trabajo que se quiere realizar en la Aduana General de la República de Cuba, que se basa en la planificación del presupuesto en el departamento de transporte. Siendo esa una de las razones por la que se hace necesario diseñar e implementar una aplicación que cumpla con los requisitos anteriormente planteados. (Rodríguez, 2008)

### **1.9. Descripción de la solución**

El polo productivo Sistema Tributarios y Aduanales está interesado en conjunto con el Departamento de Transporte de la Unidad de Aseguramiento Aduanal de la Aduana General de la República de Cuba en desarrollar una aplicación web que sea capaz de planificar el presupuesto en el departamento de transporte de la Unidad de Aseguramiento Aduanal.

### **1.10. Metodologías y herramientas de diseño utilizadas en el mundo.**

La evolución del diseño de software, como parte del proceso de desarrollo de software, es un proceso continuo que se ha ido produciendo durante las últimas tres décadas. Los primeros trabajos sobre diseño se centraron sobre los criterios para el desarrollo de programas modulares y los métodos para mejorar la arquitectura del software de una manera descendente. Los aspectos procedimentales de la definición del diseño evolucionaron hacia una filosofía denominada programación estructurada. Posteriores trabajos propusieron métodos para la traducción del flujo de datos o de la estructura de los datos, en una definición de diseño. Nuevos enfoques para el diseño proponen un método orientado a objetos para la obtención del diseño.

Ante un nuevo proyecto se pueden adoptar distintas formas de aproximación. A esto se le llama metodologías del diseño. Estas metodologías se pueden emplear de forma unitaria o combinándolas entre sí, para obtener un nuevo enfoque. Se debe tener en cuenta que también influye el ámbito de aplicación, y las disciplinas implicadas en el proceso del proyecto.

## 1.10.1. Metodologías de diseño

- **Diseño Axiomático**, cuando se emplean axiomas para tomar decisiones. Este enfoque se basa principalmente en las necesidades funcionales del cliente.
- **Diseño Orientado al Uso**, cuando el diseño se centra en las tareas asociadas al uso y sus objetivos. Este enfoque penaliza la usabilidad, ya que incrementa la curva de aprendizaje, pero logra solventar problemas complejos o altamente críticos.
- **Diseño Centrado en el Usuario** (UCD User Center Design) que es la metodología de diseño más habitual en el mundo. Este enfoque coloca todas las necesidades, deseos y limitaciones del usuario como núcleo del proceso de diseño. Por lo cual esta metodología conlleva por definición investigación y análisis del usuario. Así mismo implica que el proyecto debe ser realizado por fases para garantizar los resultados. Dentro del UCD hay metodologías específicas, como son KES (Kansei Engineering System) o QFD (Quality Function Deployment).
- **Enfoque Ascendente**, que consiste en partir de un elemento o de las características individuales de los elementos para desarrollar la totalidad del producto.
- **Enfoque descendente**, consistente en generar el producto desconociendo los elementos que lo configuran, esta técnica se emplea para desarrollar por ejemplo plantillas web.

Algunos principios que pueden resultar de ayuda a la hora de tomar decisiones son:

- **Navaja de Ockham**, consiste en eliminar los elementos innecesarios del diseño con el fin de obtener productos lo más sencillos posibles, de modo que se reduce la posibilidad de que existan incoherencias.
- **Principio KISS** (Keep It Short and Simple o Keep it simple, stupid) se emplea cuando la sencillez es un objetivo del diseño. El principio KISS no es en sí mismo una metodología sino una estrategia de diseño y sigue el reduccionismo metodológico de Ockham.
- **Timtowtdi**, (There is more than one way to do it) hace referencia a que existen múltiples soluciones a un problema, es un principio muy popular en programación. (Reyero, 2009)

## 1.10.2. Patrones de diseño

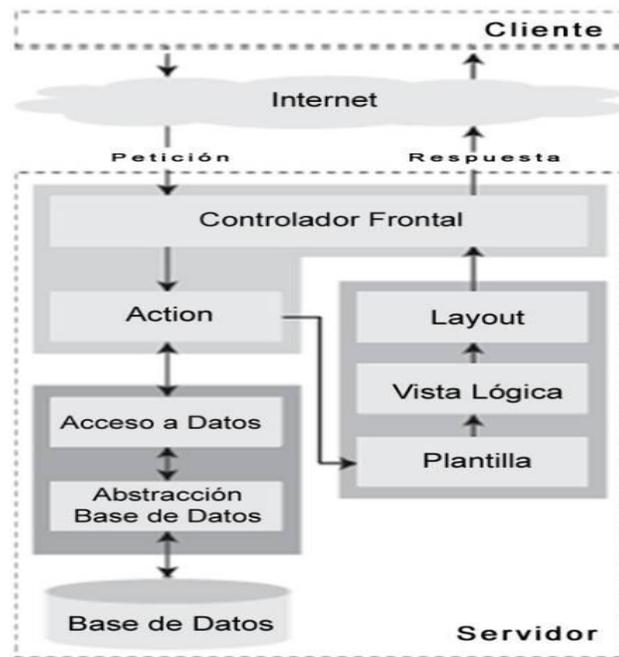
“Cada patrón describe un problema que ocurre una y otra vez en un entorno, para describir después el núcleo de la solución a ese problema, de tal manera, que esa solución pueda ser usada más de un millón de veces sin hacerlo siquiera dos veces de la misma forma”.

Un patrón de diseño es una solución estándar para un problema común de programación, una técnica para flexibilizar el código haciéndolo satisfacer ciertos criterios, una estructura de implementación que logra una finalidad determinada, manera práctica de describir aspectos de la organización de un programa.

Los patrones brindan la facilidad de reutilizar el conocimiento de desarrolladores, clasificando y describiendo problemas y soluciones a problemas que surgen con frecuencia durante el desarrollo, por lo que se convierten en un historial que no ayuda a no cometer errores ya descritos.

**Modelo Vista Controlador (MVC)**, es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. Symfony está basado en un patrón clásico del diseño web conocido como arquitectura MVC, donde la vista es la página HTML y el código que provee de datos dinámicos a la página, el controlador es el Sistema de Gestión de Base de Datos. Este patrón divide una aplicación interactiva en 3 áreas: procesamiento, salida y entrada. Para esto, utiliza las siguientes abstracciones:

- El modelo representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La vista transforma el modelo en una página web que permite al usuario interactuar con ella.
- El controlador se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista.



**Figura 1 Implementación del Patrón MVC por Symfony**

**Experto:** La responsabilidad de realizar una labor es de la clase que tiene o puede tener los datos involucrados (atributos). Una clase, contiene toda la información necesaria para realizar la labor que tiene encomendada.

**Creador:** Este patrón como su nombre lo indica es el que crea, el guía la asignación de responsabilidades relacionadas con la creación de objetos, se asigna la responsabilidad de que una clase B cree un Objeto de la clase A solamente cuando:

- B contiene A.
- B es una agregación (o composición) de A.
- B almacena A.
- B tiene los datos de inicialización de A (datos que requiere su constructor).
- B usa A.

A la hora de crear objetos se deben tener en cuenta las características de la clase.

**Bajo acoplamiento:** El acoplamiento es una medida de fuerza con que un elemento esta a, tiene conocimiento de, confía en, otros elementos. Este patrón es un principio que asigna la responsabilidad de controlar el flujo de eventos del sistema, a clases específicas. Esto facilita la centralización de actividades (validaciones, seguridad, etc.). El controlador no realiza estas actividades, las delega en otras clases con las que mantiene un modelo de alta cohesión.

Un error muy común es asignarle demasiada responsabilidad y alto nivel de acoplamiento con el resto de los componentes del sistema.

**Alta cohesión:** La cohesión es una medida de la fuerza con la que se relacionan las clases y el grado de focalización de las responsabilidades de un elemento, Cada elemento del diseño debe realizar una labor única dentro del sistema, no desempeñada por el resto de los elementos y auto identificable, una clase con baja cohesión hace muchas cosas no relacionadas o hace demasiado trabajo.

**Controlador:** Es un evento generado por actores externos. Se asocian con operaciones del sistema, operaciones del sistema como respuestas a los eventos del sistema, tal como se relacionan los mensajes y los métodos. Normalmente un controlador delega en otros objetos el trabajo que se necesita hacer, coordina o controla la actividad. No realiza mucho trabajo por sí mismo.

**Decorador:** Responde a la necesidad de añadir de forma dinámica funcionalidad a un Objeto. Esto nos permite no tener que crear sucesivas clases que hereden de la primera incorporando la nueva funcionalidad, sino otras que la implementan y se asocian a la primera:

- Añadir objetos individuales de forma dinámica y transparente.
- Responsabilidades de un objeto pueden ser retiradas.
- Cuando la extensión mediante la herencia no es viable.
- Hay una necesidad de extender la funcionalidad de una clase, pero no hay razones para extenderlo a través de la herencia.

- Hay la necesidad de extender de forma dinámica la funcionalidad de un objeto y quizás quitar la funcionalidad extendida.



*Figura 2 Ejemplo de Implementación del patrón Decorator en Symfony.*

**Factory method** (Método de fabricación): Centraliza en una clase constructora la creación de objetos de un subtipo de un tipo determinado, ocultando al usuario la casuística para elegir el subtipo que crear. (Larman, 1999)

### 1.10.3. Herramienta de diseño

**Herramienta:** Aplicación empleada para la construcción de otros programas o aplicaciones.

**Herramienta CASE** (Ingeniería Asistida por Computadora, CASE): Es la aplicación de tecnología informática a las actividades, las técnicas y las metodologías propias de desarrollo, su objetivo es acelerar el proceso para el que han sido diseñadas, en el caso de CASE para automatizar o apoyar una o más fases del ciclo de vida del desarrollo de sistemas.

El desarrollo de software ha logrado un gran avance en el mundo moderno, parte de este logro es gracias a las herramientas de modelado ya que es el medio donde se modela el sistema a desarrollar, guiándose por una metodología y utilizando algún lenguaje de modelado.

### 1.10.4. Herramientas escogidas para desarrollar este trabajo

**Visual Paradigm:** Es una herramienta CASE que permite tanto la ingeniería directa como inversa, la misma soporta varios lenguajes de programación para hacer la generación de código o la ingeniería inversa. También permite el diseño en el lenguaje UML (Unified Modeling Language), BPMN (Business Process Modeling Notation) y permite usar la metodología BPM (Business Process Management). Por otro

lado esta herramienta se integra en gran variedad de IDEs (Entornos de desarrollo integrados). (Coloma, 2007)

## Algunas características de Visual Paradigm

- Es profesional.
- Es amigable.
- Contiene facilidades para redactar especificaciones de casos de uso del sistema.
- Sincronización entre diagramas de entidad-relación y diagramas de clases.
- Generación de código / Ingeniería inversa.
- Soporte de UML versión 2.1.
- Interoperabilidad con otras aplicaciones.

### **1.11. Herramientas utilizadas para desarrollar el sistema**

En la actualidad la efervescencia de herramientas que ayudan a desarrollar aplicaciones está bastante marcada, si tenemos en cuenta que la tendencia al desarrollo de aplicaciones está encaminada a la tecnología web, pues tendríamos tela por donde cortar para decidir que entorno de desarrollo utilizar, pero la selección se reduce a los objetivos del polo de desarrollo Sistemas Tributarios y Aduanales de la Universidad de las Ciencias Informáticas.

#### **1.11.1. Lenguajes de programación**

En los últimos tiempos han surgido variedades de lenguaje de programación cada vez con más facilidades para el desarrollador y el educador, como java, php 5 o .NET, que son tecnologías más amigables a la hora de desarrollar aplicaciones de todas las envergaduras, pero para que un lenguaje atraiga a desarrolladores, vendedores y educadores debe resolver un problema particular tan bien que sea capaz de superar la resistencia de adoptarlo.

La programación orientada a objetos (POO) ha tenido un desarrollo de cerca de treinta años; es una técnica que se enfoca en los datos (objetos) y en la manera de llegar a ellos (interfaces), no en las herramientas que se utilizan para manejarlos, a fin de crear “módulos” que interactúen entre sí para lograr el objetivo del programa.

Un lenguaje de programación es un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Es utilizado para controlar el comportamiento físico y lógico de una máquina.

Entre los lenguajes más usados se encuentra **PHP** (acrónimo de Hipertexto Preprocessor) es un lenguaje "del lado del servidor" (esto significa que PHP funciona en un servidor remoto que procesa la página Web antes de que sea abierta por el navegador del usuario) especialmente creado para el desarrollo de páginas Web dinámicas.

Puede ser incluido con facilidad dentro del código HTML, y permite una serie de funcionalidades tan extraordinarias que se ha convertido en el favorito de millones de programadores en todo el mundo. (Vázquez, 2003)

## Características de PHP

- Gratuito. Al tratarse de software libre, puede descargarse y utilizarse en cualquier aplicación, personal o profesional, de manera completamente libre.
- Versatilidad. PHP puede usarse con la mayoría de sistemas operativos, ya sea basados en UNIX (Linux, Solares), como con Windows, el sistema operativo de Microsoft.
- Sencilla integración con múltiples bases de datos. Esencial para una página Web verdaderamente dinámica, es una correcta integración con base de datos. Aunque MySQL es la base de datos que mejor trabaja con PHP, puede conectarse también a PostgreSQL, Oracle o cualquier otra base de datos compatible con ODBC (Open DataBase Connectivity).

## Ventajas que brinda PHP

- Es un lenguaje multiplataforma.
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. PHP es Open Source, lo cual significa que el usuario no depende de una compañía específica para arreglar cosas que no funcionan, además no estás forzado a pagar actualizaciones anuales para tener una versión que funcione.
- Permite las técnicas de Programación Orientada a Objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- Tiene manejo de excepciones (desde PHP5).
- Rapidez. PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz. Esta completamente escrito en C, así que se ejecuta rápidamente utilizando poca memoria.
- Puede interactuar con muchos motores de bases de datos tales como MySQL, Oracle, PostgreSQL, y otros más.

## Desventajas de PHP

- Se necesita instalar un servidor web.
- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.

Otro de los más utilizados es **Java**, lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años noventa. El lenguaje en sí mismo toma mucha de su sintaxis de

otros como C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores. (Terra)

Entre noviembre de 2006 y mayo de 2007, Sun Microsystems liberó la mayor parte de sus tecnologías Java bajo la licencia GNU GPL, de acuerdo con las especificaciones del Java Community Process, de tal forma que prácticamente todo el Java de Sun es ahora software libre, aunque la biblioteca de clases de Sun que se requiere para ejecutar los programas Java aún no lo es.

## Características de Java

- Lenguaje simple: Posee una curva de aprendizaje muy rápida.
- Orientado a objetos: Fue diseñado como un lenguaje orientado a objetos desde el principio. Los objetos agrupan en estructuras encapsuladas tanto sus datos como los métodos (o funciones) que manipulan esos datos. La tendencia del futuro, a la que Java se suma, apunta hacia la programación orientada a objetos, especialmente en entornos cada vez más complejos y basados en red.
- Distribuido: Proporciona una colección de clases para su uso en aplicaciones de red, que permiten abrir sockets y establecer y aceptar conexiones con servidores o clientes remotos, facilitando así la creación de aplicaciones distribuidas.
- Interpretado y compilado a la vez: Es compilado, en la medida en que su código fuente se transforma en una especie de código máquina, los bytecodes, semejantes a las instrucciones de ensamblador. Por otra parte, es interpretado, ya que los bytecodes se pueden ejecutar directamente sobre cualquier máquina a la cual se hayan portado el intérprete y el sistema de ejecución en tiempo real (run-time).
- Robusto: Fue diseñado para crear software altamente fiable. Para ello proporciona numerosas comprobaciones en compilación y en tiempo de ejecución. Sus características de memoria liberan a los programadores de una familia entera de errores (la aritmética de punteros), ya que se ha prescindido por completo de los punteros, y la recolección de basura elimina la necesidad de liberación explícita de memoria.

- Indiferente a la arquitectura: Está diseñado para soportar aplicaciones que serán ejecutadas en los más variados entornos de red, desde Unix a Windows NT, pasando por estaciones de trabajo, sobre arquitecturas distintas y con sistemas operativos diversos. Para acomodar requisitos de ejecución tan variados, el compilador de Java genera bytecodes: un formato intermedio indiferente a la arquitectura, diseñado para transportar el código eficientemente a múltiples plataformas hardware y software. El resto de problemas los soluciona el intérprete de Java.
- Portable: La indiferencia a la arquitectura representa sólo una parte de su portabilidad. Además, Java especifica los tamaños de sus tipos de datos básicos y el comportamiento de sus operadores aritméticos, de manera que los programas son iguales en todas las plataformas.
- Multithreading: Hoy en día ya se ven como terriblemente limitadas las aplicaciones que sólo pueden ejecutar una acción a la vez. Java soporta sincronización de múltiples hilos de ejecución (multithreading) a nivel de lenguaje, especialmente útiles en la creación de aplicaciones de red distribuidas. Así, mientras un hilo se encarga de la comunicación, otro puede interactuar con el usuario mientras otro presenta una animación en pantalla y otro realiza cálculos.
- Dinámico: Java y su sistema de ejecución en tiempo real son dinámicos en la fase de enlazado. Las clases sólo se enlazan a medida que son necesitadas. Se pueden enlazar nuevos módulos de código bajo demanda, procedente de fuentes muy variadas, incluso desde la Red. (Java)

## Ventajas que brinda Java

- Orientado a objetos: Tiene todos los beneficios que ofrece esta metodología de programación.
- Multiplataforma: Es independiente de la plataforma donde se ejecuten sus aplicaciones.
- Seguro: El sistema de Java tiene ciertas políticas que evitan se puedan codificar virus con este lenguaje. Existen muchas restricciones, especialmente para los applets, que limitan lo que se puede y no puede hacer con los recursos críticos de una computadora.
- Multiproceso: En Java, la programación de multiproceso se ha integrado progresivamente.

- Existen dentro de su librería clases gráficas como awt y swing, las cuales permiten crear objetos gráficos comunes altamente configurables y con una arquitectura independiente de la plataforma.
- Elimina la complejidad de los lenguajes como "C" y da paso al contexto de los lenguajes modernos orientados a objetos.
- Como el código compilado de Java (conocido como bytecodes) es interpretado, un programa compilado de Java puede ser utilizado por cualquier computadora que tenga implementado el interprete de Java.
- Java no requiere que compile todas las clases de un programa para que este funcione. Si realizas una modificación a una clase Java se encarga de realizar un Dynamic Bynding o un Dynamic Loading para encontrar las clases.

## Desventajas de Java

- Los programas hechos en Java no tienden a ser muy rápidos, supuestamente se está trabajando en mejorar esto. Como los programas de Java son interpretados nunca alcanzan la velocidad de un verdadero ejecutable.
- La de poseer la tecnología de la máquina virtual, si se hace referencia a la velocidad y los ejecutables generados.

Debido a la robustez de Java, la facilidad de uso, capacidades de plataforma cruzada y características de seguridad, se ha convertido en un lenguaje de elección para ofrecer soluciones de Internet en todo el mundo.

Partiendo de que el sistema a implementar tiene que interactuar con SUA(Sistema Único de Aduanas) que es una aplicación PHP, se decidió que para el problema a resolver y con la idea de desarrollar una aplicación lo más libre de licencias posible y valorando que PHP cubre todas las necesidades se puede aceptar este lenguaje como el ideal para el desarrollo, lenguaje definido en la línea base de la arquitectura del polo de desarrollo Sistema Tributarios y Aduanales del que este trabajo está integrado y regido. (Rodríguez, 2008)

## 1.11.2. Frameworks

Un Framework es un conjunto de clases o estructuras que implementan los componentes de una aplicación genérica, así como también componentes concretos que cumplen a cabalidad tareas concretas. Para desarrollar programas completos, los desarrolladores buscan e instancian los componentes apropiados.

Realmente no existe una definición oficial de Framework, pero todos los autores coinciden en la utilización de un tema común: la reutilización. Una definición dada por R. E. Johnson, and B. Foote en 1988 en su publicación "Designing Reusable Classes. (Foote, 1988)

"Un Framework es un conjunto de clases que personifican un diseño abstracto para soluciones de una familia de problemas relacionados..."

### 1.11.2.1. Frameworks javascript

**Ext JS** es un Framework de JavaScript que permite realizar aplicaciones Web enriquecidas basándose en tecnología AJAX, JSON, DHTML y DOM. Empezó siendo un conjunto de librerías y extensiones para YUI (Yahoo User Interface). Con el tiempo se convirtió en un Framework independiente y a principios de 2007 se creó una compañía para comercializar y dar soporte del Framework Ext JS. De esta forma tiene dos tipos de licencias, LGPL y comercial. Básicamente, podéis usar Ext JS para vuestros desarrollos, pero para obtener soporte deberéis tener una licencia comercial. (JS)

#### Características de Ext JS

- Ext.Element: Representa un elemento del árbol DOM. Muchas de las funciones de manipulación de los elementos tienen un parámetro opcional que permite realizar el cambio mediante un efecto de animación. El parámetro de animación puede ser un dato booleano o un objeto que incluye las opciones de la animación.
- Ext.Border.Layout: Esta clase representa un diseño común para ser usado en aplicaciones de escritorio.

- Ext.Dom.Helper: Utilidades para trabajar plantillas o DOM. Suporta el uso de DOM o fragmentos de HTML de forma transparente.
- Ext.Tab.Panel: Un ligero contenedor de tabs.
- Ext.Update.Manager: Proporciona soporte para actualización AJAX de los objetos Element.

## Ventajas de Ext JS

- Posibilita utilizar un gran número de componentes visuales que mejoran considerablemente la calidad de las aplicaciones.
- Brinda la posibilidad de validaciones de formularios de todo tipo, basándose en expresiones regulares y tipos de datos.
- Trae implícitos componentes como vista en arboles, arrastrado y soltado, cambio de tamaño de imágenes, rejillas, paginado, agrupado de objetos, tabs, asistentes, entre otros muchos.
- Facilita la separación de las capas de la vista con la del controlador desde el punto de vista productivo ya que el código utilizado en la primera es solamente JavaScript y no es necesario utilizar ningún tipo de código PHP, así los desarrolladores pueden centrarse más en el aprendizaje de un solo lenguaje.
- Al soportar serialización de objetos mediante tecnología JSON permite que los datos enviados desde el controlador como respuesta a la vista contengan solo las propiedades de dichos objetos, pero no el comportamiento, minimizando los posibles errores de programación y los accidentes de que los objetos sean modificados erróneamente desde la vista.

## Desventajas de ExtJs

- En el caso específico de ExtJS existe un factor que hay que tener en cuenta y que nace del hecho de ser una librería JavaScript independiente de cualquier tecnología del lado servidor. No existe una forma fácil de realizar un vínculo entre los componentes visuales con el respectivo modelo, lo

cual genera que el programador tenga que escribir más código para validar y enlazar los formularios.

- Es difícil hacer que el servidor haga push (inserción) de información desde el servidor web hacia el navegador, haciendo que el cliente tenga que constantemente hacer pooling para obtener datos actualizados del servidor.
- La falta de un diseñador gráfico limita la difusión de la aplicación al no proveer una forma fácil y rápida de desarrollar en él.
- Su sistema de licenciamiento no contempla la licencia LGPL, o el código es 100% GPL o debes pagar por su licencia de desarrollo.

También se puede mencionar al framework **Dojo**, el cual es una librería escrita en JavaScript que busca facilitar problemas comunes que surgen al desarrollar aplicaciones web con ciertas características (AJAX). Se pueden usar los componentes que ofrece Dojo para hacer los sitios web más usables, eficientes y funcionales. En general, facilita el desarrollo de aplicaciones web. (RibosoMatic)

## Características de Dojo

- Múltiples puntos de entrada.
- Independencia del intérprete.
- Unifica estándares de codificación.
- Complementos: Los complementos de Dojo son componentes pre empaquetados de código JavaScript, HTML y CSS que pueden ser usados para enriquecer aplicaciones web.
- Comunicación asíncrona: Una característica importante de las aplicaciones AJAX es la comunicación asíncrona entre el navegador y el servidor. Tradicionalmente, se realizaba con el comando JavaScript XMLHttpRequest. Dojo provee de una capa de abstracción (dojo.io.bind) para varios navegadores web con la que se pueden usar otros transportes (como IFrames ocultos) y

diferentes formatos de datos. De esta forma podemos obtener los campos que se van a enviar como parámetros del formulario de una manera sencilla.

- Sistema de paquetes: Dojo provee de un sistema de paquetes para facilitar el desarrollo modular.
- Almacenamiento de datos en el cliente: Adicionalmente, ofrece funciones para leer y escribir cookies, proporcionando en el lado cliente una abstracción llamada Dojo Storage. Dojo Storage permite a la aplicación web almacenar datos en el lado cliente, persistencia y seguridad.
- Soporte para Adobe Integrated Runtime (AIR): Dojo permite usar aplicaciones Adobe AIR basadas en JavaScript.

## Ventajas del uso de Dojo

- Ahorra tiempo de desarrollo y diseño de interfaces complicadas y manejos de eventos complejos. Promueve la elaboración de pruebas unitarias y depuración.
- Dojo es una librería completa que contempla casi todos los componentes necesarios. Provee una infraestructura para la internacionalización de muy buena calidad. Está dentro del ranking de mejor desempeño junto a otras librerías. Es una comunidad abierta al público.

## Desventajas del uso de Dojo

- Requiere (aunque no es 100% necesario) el uso de entornos de desarrollo integrados o IDE (*Integrated Development Environment*) avanzados para el desarrollo del contenido web. No es fácil lograr que funcione en varios browsers a la vez.

Valorando que Ext JS brinda la posibilidad de utilizar un gran número de componentes visuales que mejoran considerablemente la calidad de las aplicaciones y facilita la separación de las capas de la vista con la del controlador desde el punto de vista productivo, y teniendo en cuenta otras características mencionadas anteriormente es considerado ideal para el desarrollo de la aplicación, además que está definido en la línea base de la arquitectura del polo de desarrollo Sistema Tributarios y Aduanales del que este trabajo está integrado y regido.

## 1.11.2.2. Frameworks php

**Symfony** es un Framework para PHP5 patentado bajo licencia MIT, es compatible con la mayoría de gestores de bases de datos, MySQL, PostgreSQL, Oracle y SQL Server de Microsoft entre otros en dependencia del tipo de abstracción de la Base de Datos que se utilice.

Symfony es un completo framework diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación web.

### Características de Symfony

- Proyecto de Symfony es un framework muy amplio, e incluye un verdadero ORM, de nombre Propel, que es otro proyecto de código abierto y, probablemente, una de las mejores soluciones ORM para PHP.
- Incluye Creole para la capa de abstracción de base de datos y Mojavi para la capa Model-View-Controller.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de que funciona correctamente en los sistemas Windows).
- Independiente del sistema gestor de bases de datos.
- Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.

- Basado en la premisa de "convenir en vez de configurar", en la que el desarrollador solo debe configurar aquello que no es convencional.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- Preparado para aplicación empresarial y adaptable a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros. (Zaninotto, 2008)



**Figura 3 Composición de la estructura de Symfony por otros Frameworks.**

## Ventajas de Symfony

- Los proyectos son autosuficientes. Todos los archivos que necesita un proyecto para funcionar se encuentran bajo el directorio raíz del proyecto. Además, como Symfony sólo utiliza rutas relativas, puedes mover el directorio del proyecto de un sitio a otro y todo seguirá funcionando correctamente sin necesidad de realizar ningún cambio. Por tanto, no es obligatorio que el directorio de producción sea el mismo que el directorio de la máquina de desarrollo.
- Su licencia de código abierto, su funcionamiento en diversas plataformas (como Windows y Linux) y su independencia del gestor de bases de datos, gracias a su capa de abstracción.

- Sigue las mejores prácticas y patrones de desarrollo para la web, como la separación estricta de código y diseño y el patrón Modelo Vista Controlador (MVC).
- Es recomendado para sistemas empresariales basados en la web donde se requiera flexibilidad para adaptarse a políticas y arquitecturas propias de cada empresa, así como un soporte lo suficientemente estable como para desarrollar aplicaciones escalables y de largo aliento.

## Desventajas de Symfony

- Consumo de memoria.
- Búsqueda de datos un poco lento. Esto es cuando se desea realizar una búsqueda de datos muy específica, se la debe hacer en forma manual ya que la utilización de la interfaces PROPEL genera código que puede ser lento.
- La integración entre diferentes aplicaciones de un mismo proyecto todavía no se encuentra muy depurada en Symfony. Se presentan varios inconvenientes si se desea utilizar un módulo que se encuentra en otra aplicación.

Otro muy popular es **Zend Frameworks**, un framework para desarrollo de aplicaciones Web y servicios Web con PHP que brinda soluciones para construir sitios web modernos, robustos y seguros. Además es Open Source y trabaja con PHP 5. (techtastico)

## Características de Zend Frameworks

- Es desarrollado por Zend que es la empresa que respalda comercialmente a PHP.
- Cuenta con módulos para manejar archivos PDF, canales RSS, etc.
- El Marco de Zend también incluye objetos de las diferentes bases de datos, por lo que es extremadamente simple para consultar su base de datos, sin tener que escribir ninguna consulta SQL.
- Completa documentación y pruebas de alta calidad.

- Robustas clases para autenticación y filtrado de entrada.
- Clientes para servicios web.
- Trabaja con MVC.

## Ventajas de Zend Frameworks

- Reduce el plazo de comercialización (time to market) de las aplicaciones, permitiendo ofrecer presupuestos más ajustados.
- Estandariza los procesos más frecuentes, dotándolos de gran robustez.
- Facilita el mantenimiento de las aplicaciones.
- Ofrece muchas facilidades para el acceso a recursos avanzados que de otro modo resultan más costosos de desarrollar.
- Tiene el respaldo de la propia ZEND, creadora de PHP, lo que asegura su continuidad futura tanto como la del propio lenguaje PHP.

## Desventajas de Zend Frameworks

- Ninguna protección por definición de la licencia BSD.
- No tiene mucha Independencia estratégica.

Utilizaremos el framework de php Symfony 1.1 porque es el más viable de acuerdo con sus características para desarrollar, es el más completo de los frameworks de php para el desarrollo de sistemas grandes, admite Object-Record-Mappe (ORM), tiene soporte integrado para ajax además de que tiene módulo integrado para manejar autenticación de usuario, todo esto a diferencia de Zend que es otro de los framework estudiados por el equipo, además de que el polo sistemas tributarios en su arquitectura base definió que este es el framework a utilizar. (Rodríguez, 2008)

### 1.11.2.3. Sistemas gestores de bases de datos

El SGBD **Oracle**, fabricado por Oracle Corporation, utiliza la arquitectura cliente/servidor. Ha incorporado en su sistema el modelo objeto-relacional, pero al mismo tiempo garantiza la compatibilidad con el tradicional modelo relacional de datos. Así ofrece un servidor de bases de datos híbrido. Es uno de los más conocidos y ha alcanzado un buen nivel de madurez y de profesionalidad. Se destaca por su soporte de transacciones, estabilidad y escalabilidad.

Los tipos objeto de Oracle son tipos de datos definidos por el usuario que permiten modelar entidades complejas del mundo real en una estructura que trata cada entidad como una unidad atómica simple en la base de datos. A partir de la versión 10 del año 2004, se añade a los servidores la capacidad de funcionar según el paradigma de “Grid” (o rejilla) y se ofrecen mejoras en la administración e integración de algunos elementos que previamente no funcionaban correctamente juntos. (Costilla, 2002)

#### Ventajas de Oracle

- Las entidades complejas del mundo real y la lógica se pueden modelar fácilmente, lo que permite reutilizar objetos para el desarrollo de base de datos de una forma más rápida y con mayor eficiencia.
- Los programadores de aplicaciones pueden acceder directamente a tipos de objetos Oracle, sin necesidad de ninguna capa adicional entre la base de datos y la capa cliente.
- Las aplicaciones que utilizan objetos de Oracle son fáciles de entender y mantener porque soportan las características del paradigma orientado a objetos.
- Tiene buen rendimiento y hace buen uso de los recursos.
- Posee un rico diccionario de datos.
- Brinda soporte a la mayoría de los lenguajes de programación.
- Es un sistema multiplataforma, disponible en Windows, Linux y Unix.

- Permite tener copias de la base de datos productiva en lugares lejanos a la ubicación principal. Las copias de la Base de Datos productiva pueden estar en modo de lectura solamente.

## Desventajas de Oracle

- Es un producto de elevado precio por lo que por lo general se utiliza en empresas muy grandes y multinacionales.
- Los costos de soporte técnico y mantenimiento son elevados.
- Vulnerabilidades en la seguridad de la plataforma, se hace necesario aplicar parches de seguridad.

Otro de los más usados es **PostgreSQL**, un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una sola empresa sino que es dirigido por una comunidad de desarrolladores y organizaciones comerciales las cuales trabajan en su desarrollo. Dicha comunidad es denominada el PGDG (PostgreSQL Global Development Group).

Algunas de sus principales características son, entre otras:

- Alta concurrencia.
- Mediante un sistema denominado MVCC (Acceso concurrente de varias versiones, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit. Esta estrategia es superior al uso de bloqueos por tabla o por filas común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.
- Claves ajenas también denominadas Llaves ajenas o Claves Foráneas (foreign keys).
- Disparadores (triggers).
- Vistas.
- Integridad transaccional.

- Herencia de tablas.
- Tipos de datos y operaciones geométricas.
- Soporte para transacciones distribuidas.

PostgreSQL ofrece muchas ventajas para su compañía o negocio respecto a otros sistemas de bases de datos:

Instalación ilimitada: Es frecuente que las bases de datos comerciales sean instaladas en más servidores de lo que permite la licencia. Algunos proveedores comerciales consideran a esto la principal fuente de incumplimiento de licencia. Con PostgreSQL, nadie puede demandarlo por violar acuerdos de licencia, puesto que no hay costo asociado a la licencia del software.

Esto tiene varias ventajas adicionales:

- Modelos de negocios más rentables con instalaciones a gran escala.
- No existe la posibilidad de ser auditado para verificar cumplimiento de licencia en ningún momento.
- Flexibilidad para hacer investigación y desarrollo sin necesidad de incurrir en costos adicionales de licenciamiento.
- Extensible: El código fuente está disponible para todos sin costo.
- Multiplataforma.
- Diseñado para ambientes de alto volumen.

## Desventajas de PostgreSQL

Hay muy pocas desventajas, en comparación con MySQL, PostgreSQL es más lento en inserciones y actualizaciones porque cuenta con cabeceras de transacción que no tiene MySQL, además la velocidad de respuesta que ofrece PostgreSQL con bases de datos relativamente pequeñas, puede parecer un poco deficiente.

Por las características antes expuestas se decidió utilizar para la aplicación el SGBD Oracle además que está definido en la línea base de la arquitectura del polo de desarrollo Sistema Tributarios y Aduanales del que este trabajo está integrado y regido.

## **1.12. Conclusiones parciales**

En este capítulo se presentaron los conceptos más significativos para la correcta comprensión de este trabajo. Se realizó un estudio en el que se demuestra que el desarrollo de aplicaciones Web hoy en día está regida por la utilización de Framework implementados bajo el patrón arquitectónico MVC que faciliten el diseño e implementación de los sistemas y el estudio realizado por los Framework para PHP más significativos demuestra que el que más se ajusta a las necesidades de las aplicaciones empresariales de gestión que requieran de Base de Datos Oracle 11 y PostgreSQL es Symfony utilizando Propel para manejar la persistencia de los datos.

## CAPITULO 2: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

### 2.1. Introducción

En el presente capítulo se incluyen temas relacionados con el desarrollo de los flujos de trabajo diseño e implementación y algunos conceptos fundamentales para la comprensión del sistema, teniendo como principal objetivo los diagramas de clases del diseño, diagrama de entidad relación y diagramas de secuencia para guiar al equipo de desarrollo en la implementación de la aplicación con mayor precisión y en el menor tiempo posible además de obtener los diagramas de despliegue y de componente para posibilitar el mantenimiento y reutilización del código de manera más eficiente.

### 2.2. Diseño

En el diseño se modela el sistema, incluyendo la arquitectura, de forma que soporte todos los requisitos, el diseño no es más que el refinamiento del análisis, la forma como cumple el sistema sus objetivos. Es el centro de atención al final de la fase de elaboración y el comienzo de las iteraciones de construcción. Para lograr un sistema implementado sin ambigüedades es necesario un diseño eficiente.

### 2.3. Modelo del diseño

El Modelo del Diseño es un modelo de objetos que describe la realización física de los casos de usos centrándose en como los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas en el entorno de implementación, tienen un impacto en el sistema a considerar y sirve de abstracción a la implementación y al código fuente del sistema. Contiene protocolo, cápsula, realización de casos de usos, señales, eventos, subsistema de diseño, paquetes de diseño, interfaces, clases del diseño, clases de prueba, diseño de pruebas.

Este artefacto constituye la entrada fundamental utilizada para el correcto desarrollo de las actividades de implementación. Entre los propósitos del Modelo de Diseño se encuentran:

- Adquirir una comprensión de los aspectos relacionados con los requisitos no funcionales y restricciones relacionadas con los lenguajes de programación, componentes reutilizables, sistemas operativos, tecnologías de distribución y concurrencia y tecnologías de interfaz de usuario.

- Crear una entrada apropiada y un punto de partida para actividades de implementación, capturando los requisitos o subsistemas individuales, interfaces y clases.
- Descomponer los trabajos de implementación en partes más manejables que puedan ser llevadas a cabo por diferentes equipos de desarrollo.

### 2.4. Modelado mediante estereotipos web

Para la realización de los Diagramas de Clases del Diseño será utilizada la extensión de UML para el modelado de aplicaciones Web, (28). Esta extensión presenta como elementos más significativos a tres clases UML: Server Page (Página Servidora), Client Page (Página Cliente) y Form (Formulario) empleadas para el código servidor, código cliente y formularios respectivamente, permitiendo además representar ficheros contenedores de sentencias script. A continuación se brinda una explicación de cómo son usados estos estereotipos en el diseño de la propuesta del sistema y qué representa cada cual:



<<Server Page>>: Representa la clase que tiene código que se ejecuta en el servidor, la cual se encarga de construir (build) o generar el resultado HTML y/o realizar peticiones a la capa inferior.



<<Client Page>>: Es una página Web con formato XHTML. Mezcla de datos, presentación y lógica. Son interpretadas por el navegador. Sus atributos son las variables declaradas dentro del script que son accesibles para cualquier función dentro de la página. Cada página cliente es construida por una sola página de servidor.



<<FormHTML>>: Es una colección de elementos de entrada que están contenidos en la página cliente. Sus atributos son los elementos de entrada del formulario (input boxes, radio buttons, check boxes, hidden fields, entre otros). No tienen operaciones, el método para el paso de los parámetros es \$\_POST y se comunican con las páginas servidores mediante submit.

#### 2.4.1. Relaciones entre los elementos de diseño

- Entre las páginas servidoras pueden existir relaciones de inclusión (<<include>>).
- Las páginas servidoras construyen el resultado XHTML que conforma el código cliente (<<build>>).
- Los formularios forman parte del resultado XHTML (<<aggregation/ aggregation by>>).

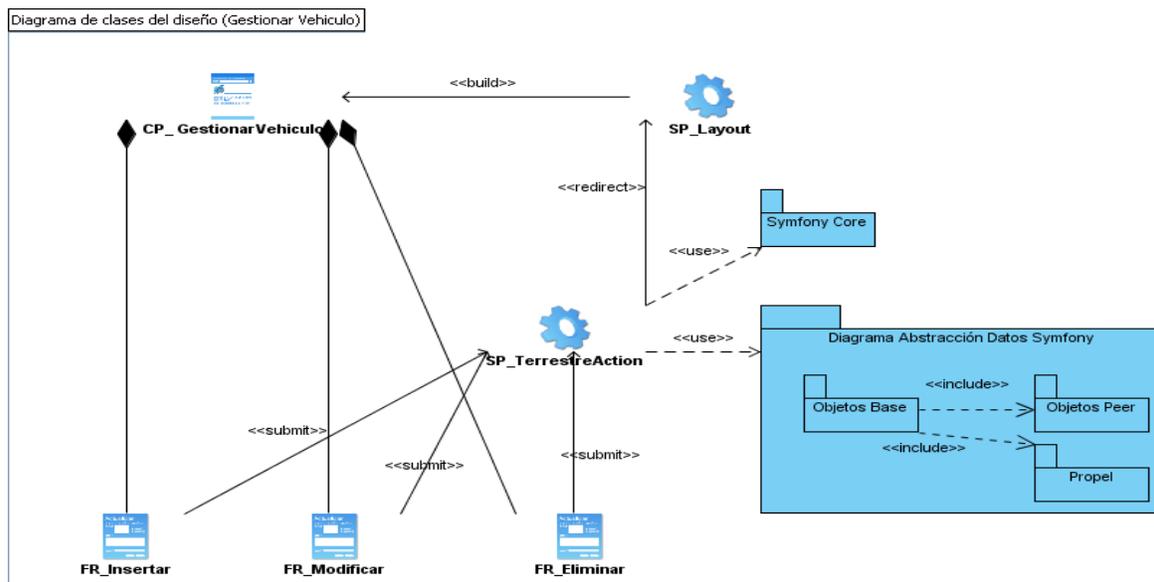
- Los formularios envían los datos al código servidor para su procesamiento (<<submit>>).
- Entre las páginas clientes pueden existir vínculos (<<link>>) o redirecciones (<<redirect>>).
- Las paginas clientes pueden incluir ficheros script (<<include>>).

DESDE/HASTA	Client Page	Form	Server Page
Client Page	<<link>> <<redirect>>	aggregation	<<link>>
Form	aggregation by		<<submit>>
Server Page	<<build>>		<<include>>

**Figura 4 Relaciones entre los elementos de diseño.**

## 2.5. Diagramas de clases del diseño

Los diagramas de clases del diseño son diagramas estáticos que describen la estructura de un sistema mostrando sus clases, atributos y las relaciones entre ellos. Los diagramas de clases son utilizados durante el proceso de diseño de un sistema para crear el diseño conceptual de la información que se maneja en el mismo y los componentes que se encargaran su funcionamiento así como la relación entre ellos como se muestra en la Figura 5.



**Figura 5 Diagrama de clases del diseño Gestionar Vehículo.**

El diseño debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades, y su principal objetivo es la elaboración de los diagramas de clases de diseño, que muestra las clases participantes en la realización en un caso de uso con todos sus atributos.

### 2.6. Diagramas de interacción del diseño

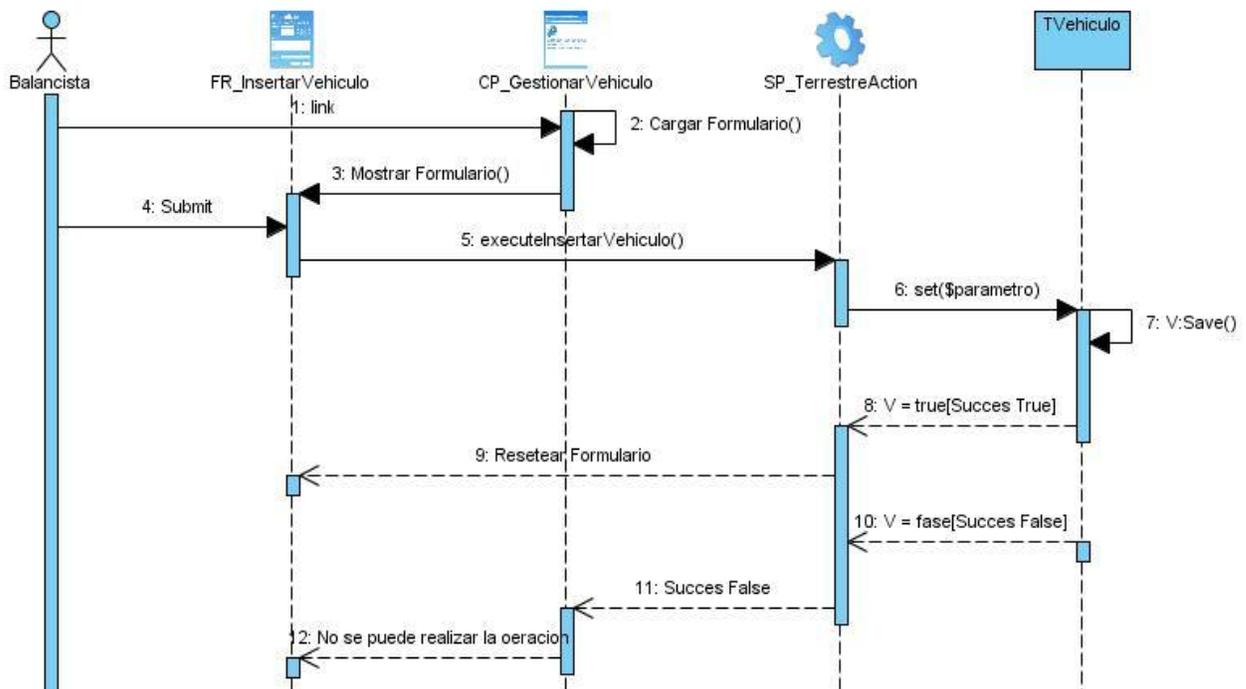
Los diagramas de interacción son modelos que describen la manera en que colaboran grupos de objetos para cierto comportamiento, en otras palabras, son los artefactos que UML brinda para expresar las interacciones entre objetos para cumplir con los requerimientos del sistema. Un diagrama de interacción capta el comportamiento de un solo caso de uso. El diagrama muestra cierto número de ejemplos de objetos y los mensajes que se pasan entre estos dentro del caso de uso. Entre los aspectos más importantes a considerar para la realización de este tipo de diagramas se encuentra:

- Se clasifican en 2 tipos: Diagramas de Secuencia y Diagramas de Colaboración.
- Modelan aspectos dinámicos del sistema.
- Se utilizan para realizar una traza de la ejecución de un escenario.
- A cada escenario le corresponde un diagrama de interacción.
- Una interacción es un conjunto de objetos y sus relaciones, incluyendo los mensajes mediante los cuales pueden establecer comunicación.
- Un diagrama de secuencia destaca la ordenación temporal de los mensajes.
- Un diagrama de colaboración destaca la organización estructural de los objetos que envían y reciben mensajes.
- Un diagrama de interacción contiene:
  - Objetos.
  - Enlaces.
  - Mensajes.
- Un mensaje de interacción puede contener:
  - Notas.
  - Restricciones.

## 2.6.1. Diagramas de secuencia

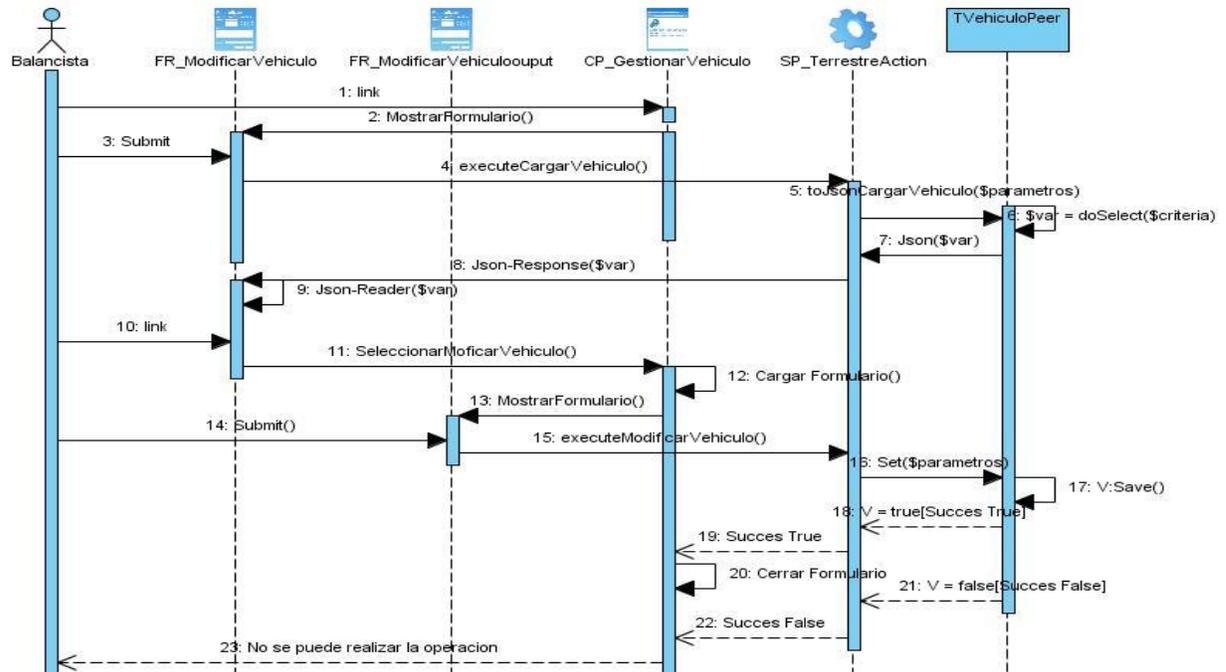
Los diagramas de secuencia son aquellos diagramas de interacción que destacan la ordenación temporal de los mensajes mediante los cuales pueden establecer comunicación los objetos representados en el mismo. Las características más distintivas de los diagramas de secuencia son las que se enuncian a continuación:

- Se realiza un diagrama de secuencia por cada escenario.
- Permite en las fases iniciales del diseño:
  - Analizar detalladamente el comportamiento de un escenario.
  - Obtener nuevas clases y objetos en un escenario (enriquecimiento del diccionario de clases).
  - Detectar las funcionalidades de las clases para llevar a cabo la tarea encomendada en el escenario.
- Se utilizan en las fases de prueba para validar la codificación.

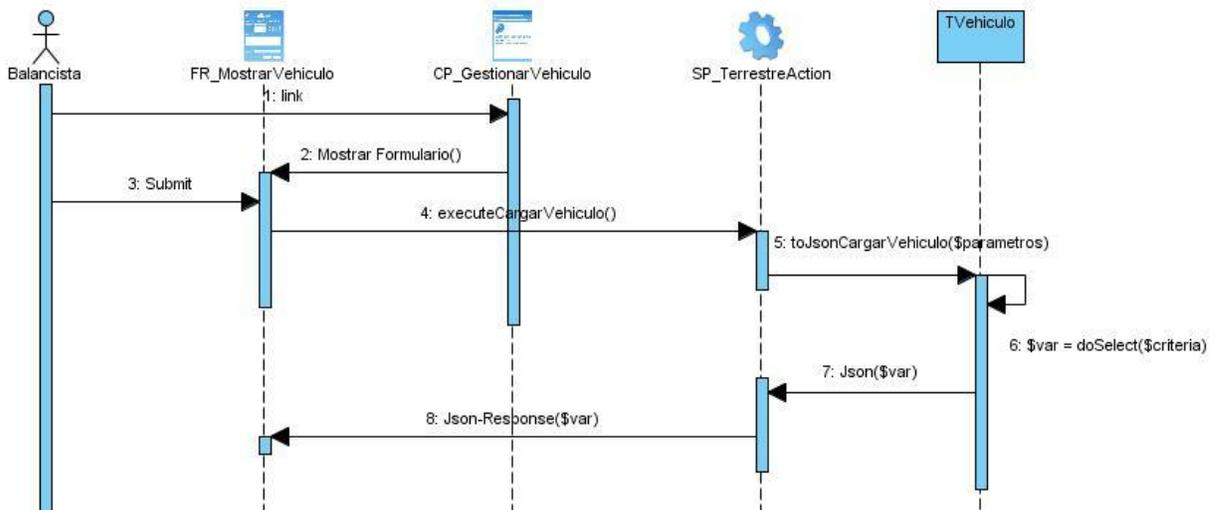


**Figura 6 Diagrama de secuencia Insertar Vehículo.**

## CAPÍTULO 2: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA



**Figura 7 Diagrama de secuencia Modificar Vehículo.**



**Figura 8 Diagrama de secuencia Mostrar Vehículo.**

### 2.6.2. Diagramas de colaboración

En los diagramas de colaboración, los objetos ejemplo se muestran como íconos. Las flechas indican, como en los diagramas de secuencia, los mensajes enviados dentro del caso del uso dado. Sin embargo, la secuencia se indica numerando los mensajes.

El numerar los mensajes dificulta más ver la secuencia que poner las líneas verticales en la página. La disposición espacial del diagrama permite mostrar otras cosas mejor. Se puede mostrar cómo se vincula entre ellos los objetos y emplear la disposición para sobreponer paquetes u otra información.

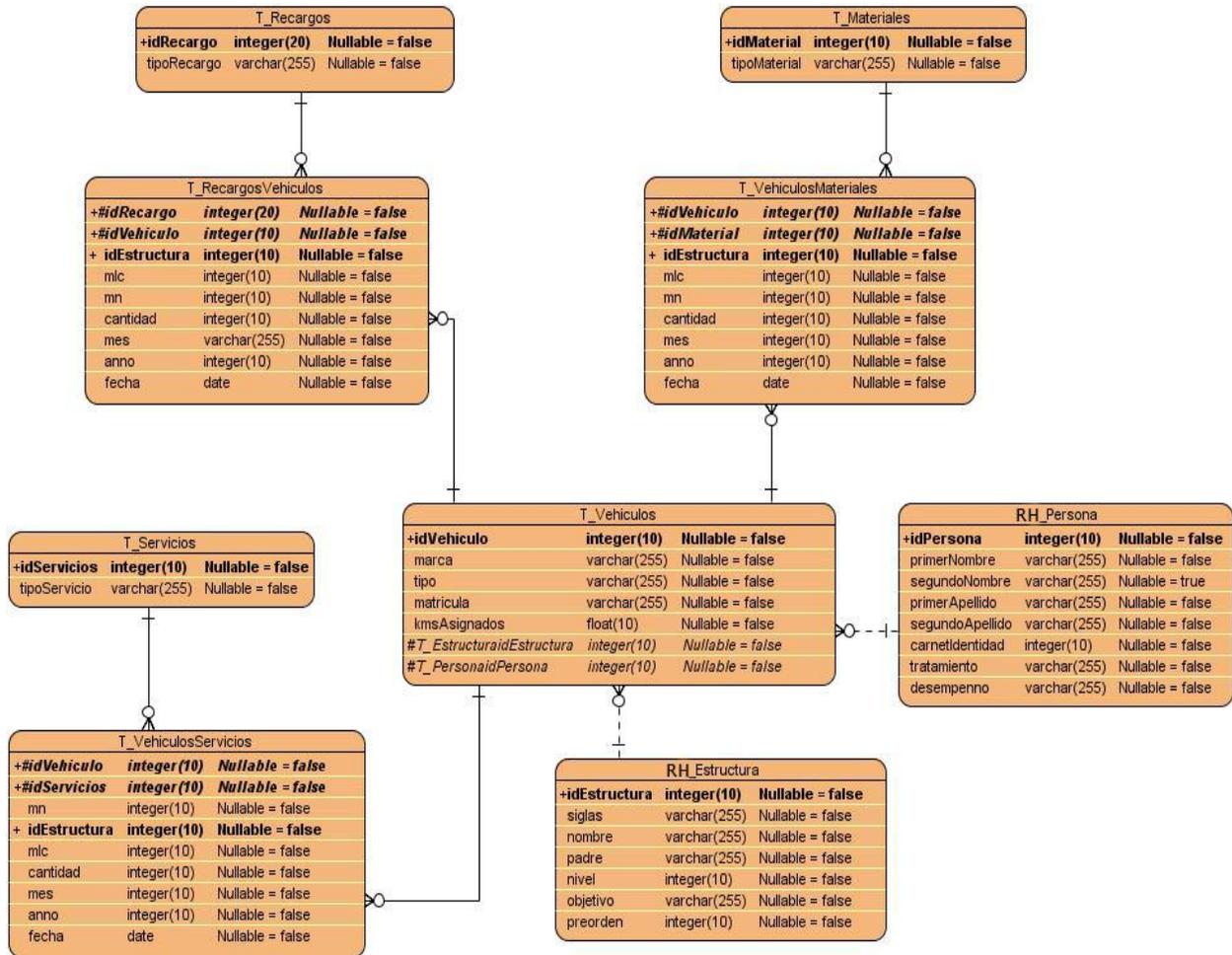
### 2.7. Diseño de la base de dato

Uno de los pasos cruciales en la construcción de una aplicación que maneje una base de datos, es sin duda, el diseño de la base de datos.

Son muchas las consideraciones a tomar en cuenta al momento de hacer el diseño de la base de datos, quizá las más fuertes sean:

- La velocidad de acceso.
- El tamaño de la información.
- El tipo de la información.
- Facilidad de acceso a la información.
- Facilidad para extraer la información requerida.
- El comportamiento del manejador de bases de datos con cada tipo de información.

La base de dato en el desarrollo de este trabajo es de vital importancia ya que se trabaja sobre clases generadas por el framework utilizado, a continuación se muestra el diagrama entidad relación.



**Figura 9 Modelo entidad relación.**

## 2.8. Principios de diseño que se aplican

El diseño del sistema está orientado directamente hacia los usuarios finales, por lo que se considera que es una parte fundamental dentro del proceso de desarrollo, pues es la parte del sistema con que el usuario interactúa y que le facilita además el acceso a los recursos. En general los usuarios finales no están familiarizados con las ciencias de la computación, por lo que se puede decir que no están interesados en la parte interna de la aplicación (el código), sino en cómo se le muestra y cómo usarla. De acuerdo con lo anterior y en concordancia con las características de los usuarios se trazaron los siguientes principios de diseño:

- Mostrar la menor cantidad de elementos posibles en cada interfaz, evitando siempre que sea posible que el usuario tenga que desplazarse horizontalmente dentro de la interfaz mostrada.
- Se seguirán patrones de diseño para los elementos que conforman las interfaces, en cuanto a colores, tamaño, formas; además no se utilizarán colores fuertes ni brillantes.
- El diseño debe proporcionar un entorno agradable que contribuya al entendimiento por parte del usuario de la información presentada.
- La interfaz ha de ser simple, intuitiva, fácil de entender y de aprender; independientemente de la experiencia, conocimientos, nivel cultural, capacidades físicas y mentales de los usuarios finales del sistema.

### **2.9. Tratamiento de errores**

Para garantizar el correcto funcionamiento de cualquier sistema es imprescindible identificar y controlar los posibles errores que se pueden presentar a la hora de interactuar con el software.

En el sistema propuesto se tratan estos errores de forma tal que las interacciones con la base de datos (inserción, eliminación, modificación, etc.) se realicen de forma correcta. Para lograr esto se establecieron mecanismos de validación que comprueben la corrección de los datos a tratar; además en los formularios se insiste en que el usuario introduzca la menor cantidad posible de datos, evitando así incoherencias e incorrecciones en los mismos, en el caso de la entrada de datos por parte del usuario se implementarán funciones que validen dicha entrada para que, de existir errores, se muestren mensajes que ilustren la incorrecta inserción, modificación o mala manipulación de datos en general.

### **2.10. Estándar de codificación**

Las convenciones o estándares de codificación son pautas de programación que no están enfocadas a la lógica del programa, sino a su estructura y apariencia física para facilitar la lectura, comprensión y mantenimiento del código.

Actualmente se hallan estándares de codificación para la mayoría de los lenguajes de programación existentes. El uso de ellos partiendo de las convenciones definidas permite una mejor comunicación entre

los programadores creando condiciones para la reusabilidad y mantenimiento de los sistemas. Para definir el estilo de codificación a seguir en la aplicación se utilizó la notación estándar establecida para aplicaciones desarrolladas en PHP (PHP Coding Standard), que mayormente está basada en el estándar de código para aplicaciones en C++ (C++ Coding Standard). Las etiquetas de apertura y cierre del lenguaje serán de la forma `<?php ¿>`, ya que siempre están disponible en cualquier configuración. Para nombrar las variables se seguirá la regla de escribir los identificadores con letras minúsculas y en español, no se usará separador entre palabras tratando de usar nombres sugerentes a la acción de la variable. Los campos id (identificadores) son palabras que identifiquen el campo. Los arreglos empezarán con el identificador array y las palabras no se separarán.

En el caso de las clases se pondrá delante el identificador del módulo. Ejemplo: “ENombreClase” y para los métodos las palabras continuas deben comenzar con mayúsculas. Ejemplo: “InsertarServicio”.

Para comentar el código se utilizará, en el caso de una línea, al final de la misma el carácter “//”y seguido el comentario y en el caso de un bloque se utilizarán los caracteres “/\* \*/”. Se usará una identificación en el código de cuatro espacios para facilitar la lectura de éste. Las llaves se usarán poniendo la llave inicial en una línea para ella sola, y en su respectiva columna la llave final también en una línea. Los nombres de las tablas se escribirán en minúscula con el identificador del módulo separado por línea abajo “\_”. En el caso de ser un nomenclador que sea la conexión al esquema del SUA se mantendrá el mismo formato con que está definido en el polo.

### **2.11. Comunicación entre capas**

La comunicación entre las capas en las que se divide la arquitectura está dada por mantener los paradigmas de la Programación Orientada a Objetos y la posibilidad de realizar sistemas con mayor portabilidad se utiliza entre el cliente y el Controlador Frontal, y viceversa, tecnología JSON, con esta se serializan los datos pasados por el usuario en Objetos PHP. Esto permite utilizar una interfaz de cualquier tipo para interactuar con el servidor de aplicación además de la seguridad que brinda este tipo de comunicación en materia de desarrollo, ya que los implementadores de interfaz de usuario no tienen acceso al comportamiento de los objetos enviados desde la capa controladora. Siempre que esto sea posible la comunicación será utilizada en combinación con AJAX, para permitir mayor velocidad y

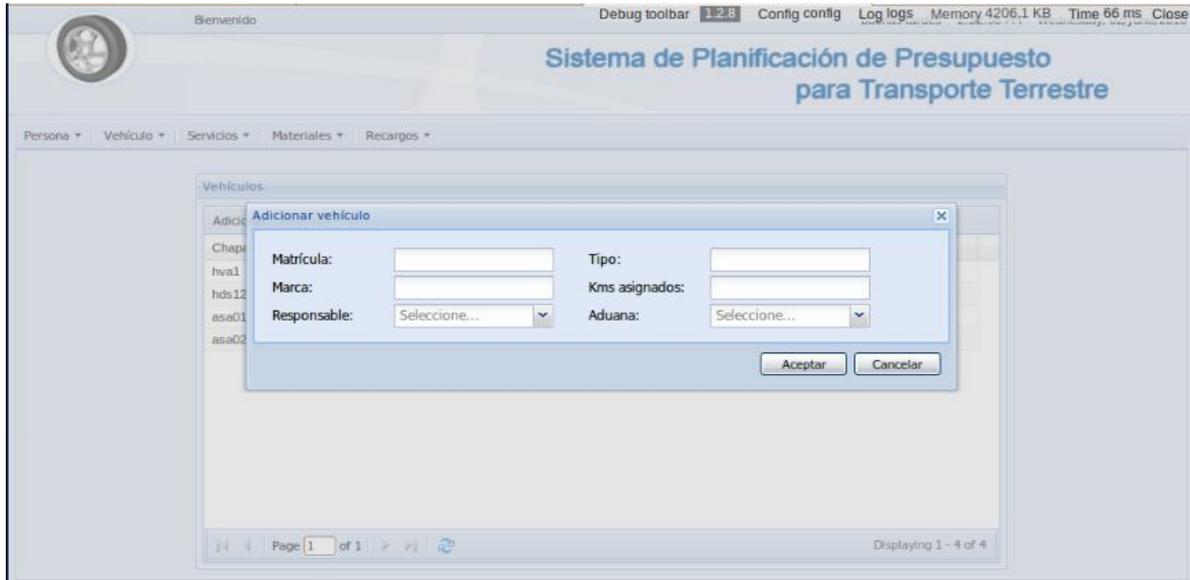
dinamismo en la interacción de los sistemas por vía Web. La utilización de AJAX con comunicación basada en XML, se utilizará para los casos en los que sea necesario por no ser factible utilizar JSON. La comunicación entre las capas del Controlador y el Modelo, gracias a la implementación del ORM, está dada por el envío de objetos que contienen la información necesaria para manejar las peticiones del usuario o las acciones a realizar. En caso de que los arreglos de datos a utilizar contengan múltiples registros serán manejados mediante arreglos que contienen los Objetos referentes a las clases del Modelo. Para manejar la comunicación con la Base de Datos se utiliza Creole, librería que trae el Propel 1.2 para la abstracción de la BD.

### 2.11.1 Ejemplo de comunicación entre capas

El usuario interactúa con la página principal (ver figura 10) del sistema donde está la barra de menú con las funcionalidades que brinda la aplicación, el usuario selecciona la acción a realizar, el sistema muestra la pantalla correspondiente, para ejemplificar mejor esta comunicación se muestra el caso de uso Gestionar Vehículo.



**Figura 10 Pantalla Principal.**



**Figura 11 Pantalla Adicionar Vehículo.**

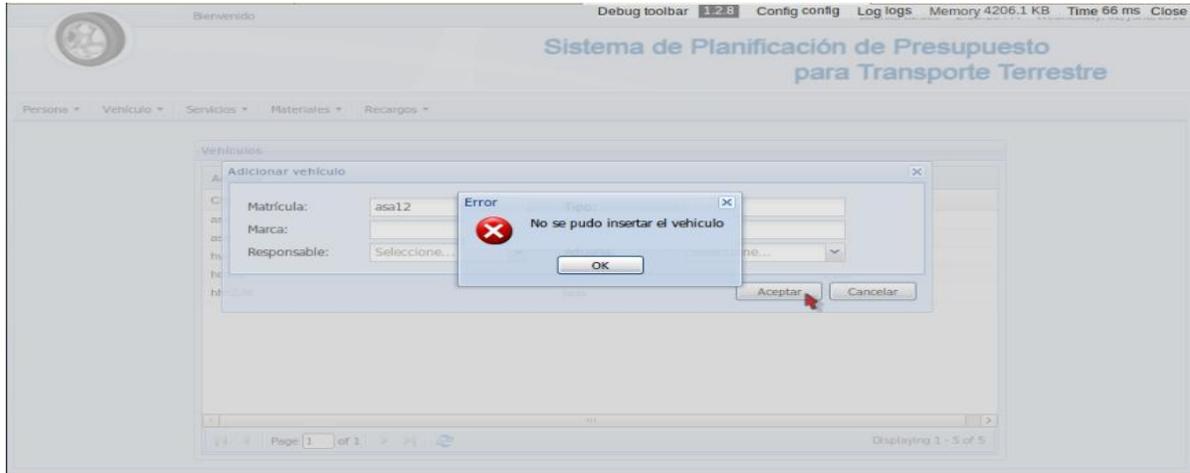
Para insertar el vehículo el usuario debe cargar el formulario con los datos correspondientes, con este fin se realizan peticiones para llenar los campos Responsable y Aduana, además que se utiliza la creación de JSON para cargar los campos mostrables al usuario. En la siguiente imagen se muestra el código de petición a la controladora para cargar un combo pasándole un objeto JSON.

```
new Ext.form.ComboBox({
    fieldLabel: 'Aduana',
    width: 130,
    id: 'aduana',
    typeAhead: true,
    emptyText: 'Seleccione...',
    triggerAction: 'all',
    displayField: 'aduana',
    valueField: 'id_aduana',
    hiddenName: 'id_aduana',
    lazyRender: true,
    mode: 'remote',
    store: new
    Ext.data.JsonStore({
        storeId: 'llenarAduana',
        url: 'terrestre/CargarAduana',
        root: 'agr',
        fields: ['id_aduana', 'aduana']
    })
})
```

**Figura 12 Código para cargar un combo mediante un objeto JSON**

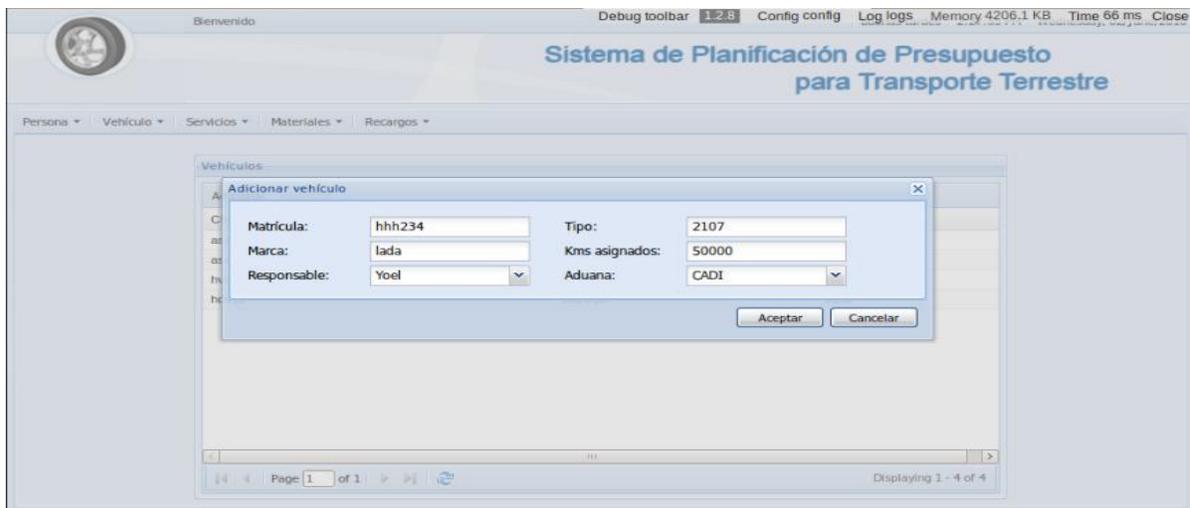
## CAPÍTULO 2: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA

Si el usuario no introduce los datos correspondientes en la pantalla el sistema muestra un mensaje de error lanzado desde el JavaScript como se muestra en la Figura 13.



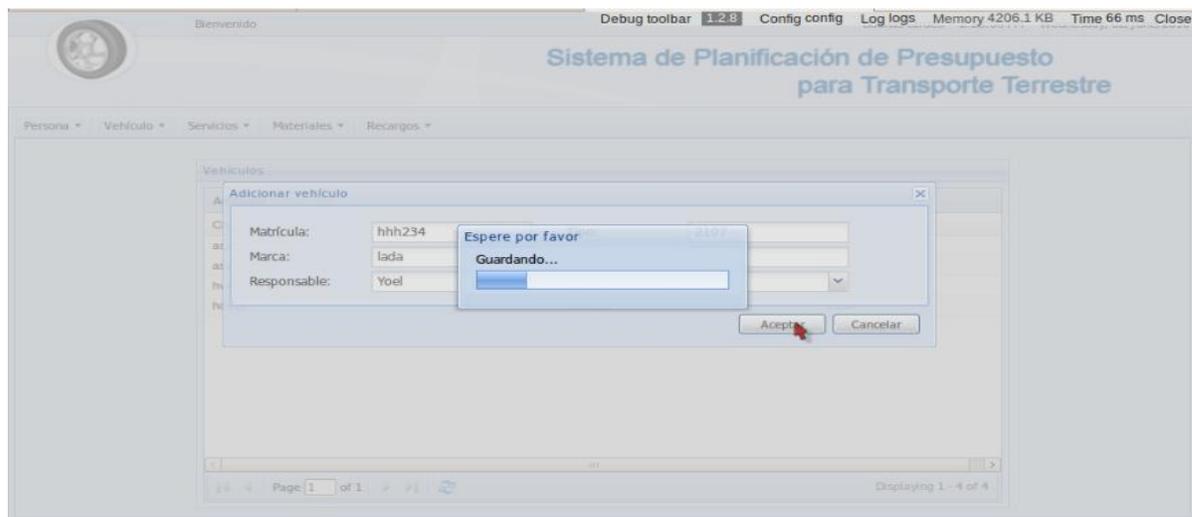
**Figura 13 Mensaje de error para datos incorrectos.**

Después que el usuario introduce los datos de la pantalla satisfactoriamente como se muestra en la Figura 14 y acepta, estos son enviados a la controladora TerrestreActions en el evento submit del formulario InsertarVehiculo por el método POST donde se muestra la dirección del action que recibirá los datos en la controladora a través del atributo url donde se muestra el evento submit del formulario.



**Figura 14 Pantalla con los datos correctos.**

Luego de ser enviado los datos a la clase controladora esta los recibe en el action especificado con el método `getRequestParameter('parámetro')`; pasándole por parámetro el nombre con que es enviado desde la cliente los atributo que se quieren recibir, aquí la controladora se crea el objeto que se desea insertar en este caso un vehículo, `$vehiculo= new TVehiculos()`; se setean los parámetros y se llama al método `$vehiculo->save()`; de la clase `BaseTVehiculos.php` generada por symfony al mapear la base de dato. Mientras esto ocurre el sistema muestra al usuario la pantalla que se visualiza en la Figura 15.



**Figura 15** Pantalla de espera al aceptar los datos de la demanda.

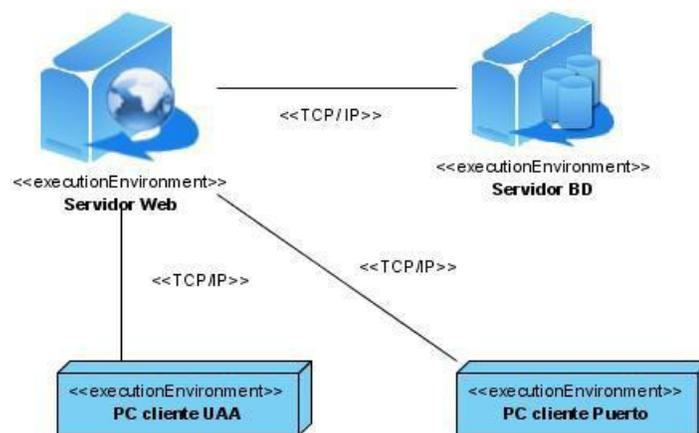
### 2.12. Modelo de implementación

En la implementación se empieza con el resultado del diseño y se implementa el sistema en términos de componentes, es decir, ficheros de código fuente, ficheros de código binario, ejecutables y similares. La mayor parte de la arquitectura del sistema es capturada durante el diseño, siendo el propósito fundamental de la implementación el desarrollar la arquitectura y el sistema como un todo. Además se implementan las clases encontradas durante el diseño y se prueban individualmente todos los componentes, integrándolos, compilándolos y enlazándolos en uno o más ejecutables. Se realizan los release y las pruebas necesarias al sistema una vez concluida todas sus funcionalidades. En el flujo de trabajo de diseño se propone crear un plano del modelo de implementación, por lo que sus últimas actividades están vinculadas a la creación del modelo de despliegue. El flujo de trabajo de implementación describe cómo los elementos del modelo del diseño se implementan en términos de componentes y cómo

estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue. Los diagramas de despliegue y componentes, que son artefactos generados en este flujo de trabajo, conforman lo que se conoce como un modelo de implementación, al describir los componentes a construir y su organización y dependencia entre nodos físicos en los que funcionará la aplicación.

### 2.12.1 Diagrama de despliegue

Los Diagramas de Despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. Se utiliza como entrada fundamental en las actividades de diseño e implementación, debido a que la distribución del sistema tiene una influencia principal en su diseño. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria.



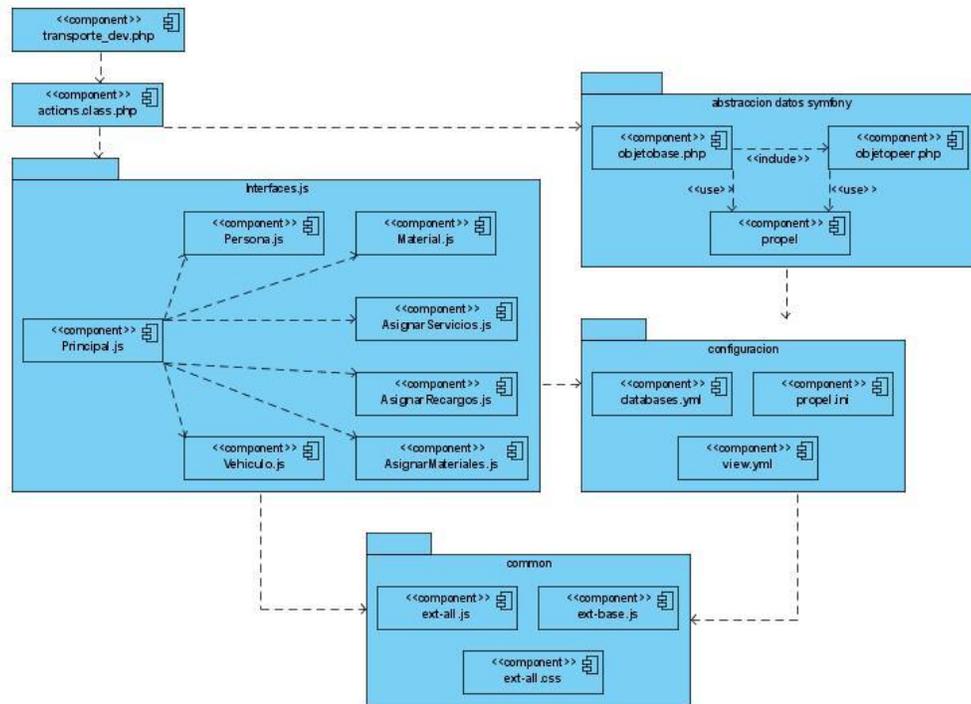
*Figura 16 Diagrama de despliegue.*

### 2.12.2. Diagrama de componente

Los componentes constituyen la parte modular del sistema, encapsulan implementación y un conjunto de interfaces y proporciona la realización de los mismos. Un componente típicamente contiene clases y puede ser implementado por uno o más artefactos (ficheros ejecutables, binarios, etc.).

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de implementación. Para dar

solución a la propuesta del sistema se desarrollaron una serie de componentes, los cuales se reflejan en el Diagrama de Componentes que se presenta a continuación: Un componente es el empaquetamiento físico de los elementos de un modelo, como son las clases del modelo de diseño; los mismos son creados, modificados o eliminados en el proceso de implementación y constituyen la versión del producto.



**Figura 17 Diagrama de Componentes.**

### 2.13. Conclusiones parciales

En este capítulo se presentaron los diagramas más significativos para la correcta comprensión de este trabajo, exponiéndose los de clases del diseño, interacción particularmente los de secuencia, así como el diseño de la base de dato, incluyendo los de despliegue y componente, además de que se describe la comunicación entre las capas con un ejemplo práctico del sistema.

### CONCLUSIONES GENERALES

Al culminar el estudio sobre la planificación de presupuesto que se realiza actualmente en el Departamento de Transporte de la Unidad de Aseguramiento Aduanal de la Aduana General de la República de Cuba, con la visión clara y segura de mejorar a gran escala la gestión de dicha planificación y valorando las tareas y objetivos propuestos, se llegó a las siguientes conclusiones:

- Se realizó el diseño de la aplicación a partir de los requerimientos identificados.
- Se desarrolló el Sistema de Planificación de Presupuesto para Transporte Terrestre usando las herramientas definidas en el polo.
- Se dio cumplimiento a cada uno de los objetivos planteados.

### **RECOMENDACIONES**

Se recomienda que la aplicación pase por el equipo de calidad del polo Sistemas Tributarios y Aduanales y que se exploten las facilidades que brinda el sistema, que se termine el modelo de planificación de presupuesto para embarcaciones y además agregarle a la aplicación un manual de usuario.

## BIBLIOGRAFÍA

MASTERMAGAZINE. [En línea] <http://www.mastermagazine.info>.

AduaCuba. [En línea] <http://www.aduana.co.cu/>.

**AGR. 2006.** *MANUAL DE ORGANIZACIÓN DE LA UNIDAD DE ASEGURAMIENTO ADUANAL*. Ciudad de la Habana : s.n., 2006.

aminfo. [En línea] <http://www.prestosoftware.cl/>.

Avivair - Servicios General - Servicios - Directorio de empresas- Dig & See Quota . [En línea] [Citado el: ] <http://www.guiadeprensa.com/servicios/servicios/avivair.html>.

**Coloma, Ramon. 2007.** Blog de Ramon Coloma. [En línea] 26 de noviembre de 2007. <http://bikermon-informatico.blogspot.com/2007/11/visual-paradigm-la-herramienta.html>.

**Costilla, Carmen. 2002.** *Características Objeto-Relacionales del Sistema de Gestión de Bases de Datos Oracle*. 2002.

**Domínguez, Pedro Rubio. 2006.** *Introducción a la gestión Empresarial*. s.l. : Edición electrónica, 2006.

**Foote, Johnson y. 1988.** *Designing Reusable Classes*. 1988.

**Gomez, Jorge Burbano Ruiz & Alberto Ortiz. 2001.** *Presupuestos*. s.l. : Mc Graw Hill, 2001.

**Larman, Craig. 1999.** *UML y patrones*. Mexico : Prentice-Hall Hispanoamericana, 1999.

**Rachlin., H.W Allen Sweeny & Robert.** *Manual de Presupuestos*. México : McGraw Hill.

**Reyero, Eusebio. 2009.** Will Web For Food. [En línea] 15 de January de 2009. <http://wwff.thespacer.net/blog/metodologias-de-diseno>.

**Rodriguez, Jose Antonio Cobo. 2008.** *Línea Base Arquitectónica para el Polo Sistemas Tributarios y de Aduanas*. . 2008.

**Ruiz, J.E. Burbano. 1992.** *Presupuestos : enfoque moderno de planeación y control de recursos.* México : McGraw-Hill, 1992.

**Vazquez, Jose Antonio Gallego. 2003.** *Desarrollo de web con PHP y MySQL.* 2003.

**Villanueva, Dr. Omar Everlery Pérez.** La administración del presupuesto del estado cubano, una valoración. [En línea]

[http://www.nodo50.org/cubasigloXXI/economia/villanueva4\\_301102.pdf](http://www.nodo50.org/cubasigloXXI/economia/villanueva4_301102.pdf).

Wintrans 4.0. [En línea] <http://www.wintrans.net/index.php/es/mnuinfowintrans>.

**Zaninotto, Fabien Potencier y François. 2008.** *Symfony la guía definitiva.* 2008.

**Terra.** Terra. *Tecnologías.* [En línea] [www.terra.es](http://www.terra.es).

**Java.** Manual de Java. [En línea] [manual-java.com/manualjava/caracteristicas-java.html](http://manual-java.com/manualjava/caracteristicas-java.html).

**Buenmaster.** [En línea] [buenmaster.com](http://buenmaster.com).

**JS, Ext.** [En línea] [extjs.com](http://extjs.com).

**RibosoMatic.** [En línea] [ribosomatic.com](http://ribosomatic.com).

**techtastico.** [En línea] [techtastico.com](http://techtastico.com).

### GLOSARIO DE TÉRMINOS

**Adobe AIR:** Adobe Integrated Runtime es un entorno de ejecución multiplataforma para la construcción de aplicaciones RIA (Rich Internet Applications) utilizando Adobe Flash, Adobe Flex, HTML y AJAX, las cuales pueden usarse como aplicaciones de escritorio.

**Ajax:** Acrónimo de *Asynchronous JavaScript And XML* (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o **RIA** (Rich Internet Applications)

**JSON:** (JavaScript Object Notation - Notación de Objetos de JavaScript) es un formato ligero de intercambio de datos.

**Open DataBase Connectivity:** Es un estándar de acceso a Bases de datos desarrollado por Microsoft Corporation, el objetivo de ODBC es hacer posible el acceder a cualquier dato desde cualquier aplicación, sin importar qué Sistema Gestor de Bases de Datos (DBMS por sus siglas en inglés) almacene los datos.

**Pool:** Manejo de una colección de conexiones abiertas a una base de datos de manera que puedan ser reutilizadas al realizar múltiples consultas o actualizaciones.

**RSS** (Really Simple Syndication) es un formato para la sindicación de contenidos de páginas web.

**Software libre:** El software libre es la libertad de los usuarios de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software.

**URL:** Acrónimo de *Uniform Resource Locator*, es decir, localizador uniforme de recurso. Es una secuencia de caracteres, de acuerdo a un formato estándar, que se usa para nombrar recursos.

**XML:** Siglas en inglés de *Extensible Markup Language* (lenguaje de marcas extensibles), es un metalenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium.