



UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA

UNAN-MANAGUA

FACULTAD DE CIENCIAS E INGENIERÍAS

DEPARTAMENTO DE COMPUTACIÓN



**Seminario de Graduación**

**METODOLOGÍAS ÁGILES PARA EL DESARROLLO DE SOFTWARE**

Autores:

- Br. Luis Adolfo Henríquez
- Br. Elizabeth de los ángeles Pérez Gutiérrez
- Br. Ivette Deyanira Rodríguez Gadea

Tutor:

- Msc. Ainoa Calero Castro

Managua, Junio de 2014



**TEMA:**

Metodologías ágiles para el desarrollo de Software.

**SUBTEMA:**

Metodología ágil de desarrollo de software basado en rasgos o funcionalidades (FDD).



## **DEDICATORIA:**

El esfuerzo de este trabajo de Graduación se lo dedicamos primero a Dios, nuestro señor, quien nos ha colmado de bendiciones brindándonos la oportunidad de poder culminar nuestros estudios, quien nos ha dado el maravilloso don de la vida, la sabiduría, el entendimiento y las fuerzas necesarias para concluir nuestra.

A nuestros padres que con sus sacrificios económicos, no cesaron en ninguno de los momentos difíciles, en los que con tanta tensión para la entrega a tiempo de nuestra tesis, nos apoyaron incondicionalmente en nuestra preparación académica. Por eso, este trabajo va por ustedes en retribución además por sus valiosos consejos, amor y perseverancia.

A nuestra apreciada tutora la maestra, Ingeniera Ainoa Calero, por habernos orientado, coordinado, apoyado e impulsado a seguir, dándonos ánimo y confianza para la realización de este arduo trabajo, y a todos nuestros maestros por sus experiencias y asesorías brindadas.



## **AGRADECIMIENTO**

Le agradecemos a Dios sobre todas las cosas por ser quien nos ha dado la oportunidad y bendición de ser lo que hoy somos, quien nos ha iluminado en nuestras vidas y nos ha dado la vida, salud y fortaleza y sabiduría.

A nuestros padres quienes nos han estimulado a continuar nuestros estudios, por sus consejos, su confianza depositada en nosotros y por la comprensión, por la paciencia, el amor y todo el apoyo que nos han brindado en cada momento de nuestras vidas, a quienes le debemos hoy también el ser profesionales.



## ÍNDICE DE CONTENIDO

RESUMEN .....	8
I. INTRODUCCIÓN.....	9
1.2 JUSTIFICACIÓN.....	10
1.3 OBJETIVOS: .....	11
1.3.1 Objetivos Generales:.....	11
1.3.2 Objetivos Específicos: .....	11
II. Marco Teórico .....	12
2.1 METODOLOGÍAS PARA EL DESARROLLO DE SOFTWARE: .....	12
2.1.1 HISTORIA DE METODOLOGIAS ÁGILES: .....	13
2.1.2 EL MANIFIESTO ÁGIL: .....	14
2.1.3 METODOLOGÍAS ÁGILES: .....	16
2.2 Control de Calidad en Procesos .....	22
2.3 Dirección o Administración de Operaciones .....	23
2.4 ESTRATEGIA DE OPERACIONES .....	23
2.5 LA TECNOLOGÍA.....	24
2.6 MANTENIMIENTO MECÁNICO.....	25
2.6.1 Tipos de Mantenimiento:.....	25
2.6.2 Aplicaciones Del Mantenimiento:.....	27



2.6.3 Mantenimiento en Motocicletas .....	28
2.6.4 CATEGORÍAS FUNDAMENTALES DEL MANTENIMIENTO .....	32
III. DISEÑO METODOLÓGICO:.....	33
3.1 Actividades de la metodología ágil FDD .....	35
3.1.1 Desarrollo de un modelo global.....	35
3.1.2 Construcción de una lista de rasgos o requerimientos.....	37
3.1.3 Planeación por rasgos: .....	37
3.1.4 Diseño y construcción por rasgos .....	40
3.2 RESULTADOS:.....	42
3.2.1 Primera Iteración (Anexo 2, figura 4.5) .....	42
3.2.2 Segunda Iteración (Anexo 3, Fig. 4.6).....	48
3.2.3 Tercera Iteración (Anexo 3, Fig. 4.7) .....	49
3.3 Diagrama de Casos de uso .....	52
3.4 Pruebas Unitarias.....	53
4.2 CONCLUSIONES: .....	63
4.3 RECOMENDACIONES .....	64
IV. MATERIALES CONSULTADOS:.....	65
4.1 Bibliografía: .....	65
V. ANEXOS .....	67
Anexo 1: .....	67



Anexo 2 .....	69
Anexo 3 .....	70
Anexo 4 Fig. 4.7 carta entrega-aceptación tercera iteración .....	71



## RESUMEN

El presente proyecto está orientado a desarrollar un sistema que dará solución a un problema socio laboral y que permita llevar el control de gestiones de mantenimiento y reparación en un taller de motocicletas,(MORIAH) automatizando el flujo de procesos con la aplicación de la metodología de desarrollo ágil FDD (Feature Driven Development –Desarrollo Basado en funcionalidades).

El marco de trabajo de FDD, se centra más en obtener resultados de alta importancia para el cliente, que en la documentación excesiva de otras metodologías, se corre el riesgo de no reflejar la verdadera necesidad con la que se construye el sistema, convirtiéndose en una metodología novedosa y fácil de implementar. A como lo describe FDD, el proceso de desarrollo, desde el levantamiento de requisitos, análisis, diseño e implementación del software, se realizó en iteraciones cortas, llevando el control de versiones de todos los artefactos generados, y estimulando la participación activa del usuario al entregarle resultados funcionales en tiempos cortos. La aplicación práctica de esta metodología se llevó a cabo desarrollando el sistema de control de gestiones de mantenimiento y reparación en un taller de motocicletas”, siguiendo todos los lineamientos y buenas prácticas que indica la metodología para cumplir con los objetivos requeridos por el taller y entregando el software a la medida de las necesidades del proceso de control de gestiones que emplean actualmente.

El producto de software resultante es un sistema, que utilizó como lenguaje de programación Visual Basic.Net y empleando SQL server 2008 como motor de base de datos.





## I. INTRODUCCIÓN

En la actualidad existe un sinnúmero de metodologías de desarrollo de software en las cuáles siempre se busca optimizar el proceso de trabajo, tratando de cumplir a cabalidad cada requerimiento de los usuarios sin descuidar la calidad en el diseño del software.

Feature Driven Development (FDD) es una metodología que se enfoca en la entrega ágil de los requerimientos, ocupándose de la calidad del software, pero más enfocada en la entrega continua, cuidando detalles como: Exceso de presupuesto, fallas en el sistema a través de entregas cortas y periódicas favoreciendo en su gran mayoría al cliente.

Para la realización del siguiente proyecto se han aplicado tanto conocimiento científico adquiridos durante la etapa de estudios universitarios, así como el conocimiento adquirido durante el ejercicio de nuestros diferentes trabajos, los que han sido implementados en el presente trabajo

Este trabajo está constituido de cinco capítulos: Introducción, marco teórico, diseño metodológico, materiales consultados y anexos que han sido desarrollados en toda la etapa que duró la realización del mismo, cada capítulo ha sido elaborado en base a estudios realizados.



## 1.2 JUSTIFICACIÓN

Mayor Servicios de Nicaragua, S.A (MASESA), fue fundada el 06 de agosto de 2004 , e inició operaciones en el país el 1 de Noviembre del mismo año, es una empresa que se dedica a la venta de motocicletas y partes de las mismas, se encuentra distribuida en todo el territorio nacional y fuera de éste, posee un gran número de clientes y además tiene un taller de mantenimiento para el chequeo de las motocicletas, debido a la enorme cantidad de clientes, el taller está saturado de gestiones y trabajan de una manera desordenada y todo es manual; además, todos los procesos se hacen en físico y esto representa mucha carga para los usuarios y por ello se observa gran lentitud en el trabajo.

Los servicios que el taller ha venido brindando a estos clientes son:

Mantenimiento preventivo, mantenimiento correctivo, mantenimiento predictivo, mantenimiento mejorativo o rediseños, que consiste en la modificación o cambio de las condiciones originales del equipo o instalación de partes y mantenimiento selectivo.

Debido a esta gran demanda, la empresa quiere implementar un sistema de control de gestiones que optimice la atención y agilice los procesos internos en el área del taller de MASESA, además generará reportes de clientes, marcas, modelos por marca y ordenes por cliente.

El presente proyecto tiene como propósito primordial brindar apoyo innovador en procesos de ejecución de tareas al sector que representan los pequeños talleres de motocicletas, procurando contribuir con soluciones viables para un mejor desempeño en el trabajo que estos realizan.



## **1.3 OBJETIVOS:**

### **1.3.1 Objetivos Generales:**

- Desarrollar un sistema de control de gestiones de mantenimiento y reparación en un taller de motocicletas, automatizando el flujo de procesos con la aplicación de la metodología de desarrollo ágil FDD.

### **1.3.2 Objetivos Específicos:**

- a) Conocer los diferentes tipos de metodologías ágiles de desarrollo de software más utilizados en la actualidad.
- b) Brindar herramientas al lector para que les ayuden a la toma de decisiones sobre cuándo es factible aplicar estas metodologías.
- c) Afianzar los conocimientos de desarrollo de software.
- d) Examinar los fundamentos formales, la relevancia metodológica y el alcance práctico de los métodos ágiles de desarrollo de software, específicamente de la metodología FDD.



## II. Marco Teórico

El presente marco teórico expresa las bases sobre las cuales se enmarca la elaboración de este seminario de graduación y contiene los aportes bajo los que se desarrolló la misma, así mismo aborda los tópicos sobre el desarrollo y aplicación de las metodologías ágiles de desarrollo de software, así como el desarrollo de un sistema de control de gestiones para un taller de motocicletas.

### 2.1 METODOLOGÍAS PARA EL DESARROLLO DE SOFTWARE:

Una metodología de desarrollo de software se refiere a un marco de trabajo que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información.

A lo largo del tiempo, una gran cantidad de métodos han sido desarrollados diferenciándose por su fortaleza y debilidad.

El marco de trabajo para metodología de desarrollo de software consiste en:

- Una filosofía de desarrollo de programas de computación con el enfoque del proceso de desarrollo de software.
- Herramientas, modelos y métodos para asistir al proceso de desarrollo de software.

Estos marcos de trabajo son a menudo vinculados a algún tipo de organización, que además desarrolla, apoya el uso y promueve la metodología.

La metodología es a menudo documentada en algún tipo de documentación formal.



### 2.1.1 HISTORIA DE METODOLOGIAS ÁGILES:

A principio de la década de los 90s surgió un enfoque que fue bastante revolucionario para su momento ya que iba en contra de toda la creencia de que mediante procesos altamente definidos se iba a lograr obtener software en tiempo, costo y con la requerida calidad.

El enfoque fue planteado por primera vez por Martin y se dio a conocer en la comunidad de ingeniería de software con el nombre RAD, o Rapid Application Development, el cual consistía en un entorno de desarrollo altamente productivo, en el que participaban grupos pequeños de programadores utilizando herramientas que generaban código en forma automática tomando como entradas sintaxis de alto nivel.

En general, se considera que este fue uno de los primeros hitos en pos de la agilidad en los procesos de desarrollo.

Es así como que este tipo de metodologías fueron inicialmente llamadas “metodologías livianas”, sin embargo, aún no contaban con una aprobación. pues se le consideraba por muchos desarrolladores como meramente intuitiva. Luego, con el pasar de los años, en febrero de 2001, tras una reunión celebrada en Utah-EEUU, nace formalmente el término “**ágil**” aplicado al desarrollo de software.



### 2.1.2 EL MANIFIESTO ÁGIL:

El Manifiesto Ágil comienza enumerando los principales valores del desarrollo ágil. Según el Manifiesto, se valora:

Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno, muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

- Desarrollar software que funciona más que conseguir una buena documentación.
- La colaboración con el cliente más que la negociación de un contrato.
- Responder a los cambios más que seguir estrictamente un plan.

Los valores anteriores inspiran los doce principios del manifiesto. Son características que diferencian un proceso ágil, de uno tradicional. Los dos primeros principios son generales y resumen gran parte del espíritu ágil. El resto tienen que ver con el proceso a seguir y con el equipo de desarrollo, en cuanto metas a seguir y organización del mismo. Los principios son:

- I. La prioridad es satisfacer al cliente mediante tempranas y continuas entregas de software que le aporte un valor.



- II. Dar la bienvenida a los cambios. Se capturan los cambios para que el cliente tenga una ventaja competitiva.
- III. Entregar frecuentemente software que funcione desde un par de semanas a un par de meses, con el menor intervalo de tiempo posible entre entregas.
- IV. La gente del negocio y los desarrolladores deben trabajar juntos a lo largo del proyecto.
- V. Construir el proyecto en torno a individuos motivados. Darles el entorno y el apoyo que necesitan y confiar en ellos para conseguir finalizar el trabajo.
- VI. El diálogo cara a cara es el método más eficiente y efectivo para comunicar información dentro de un equipo de desarrollo.
- VII. El software que funciona es la medida principal de progreso.
- VIII. Los procesos ágiles promueven un desarrollo sostenible. Los promotores, desarrolladores y usuarios deberían ser capaces de mantener una paz constante
- IX. La atención continua a la calidad técnica y al buen diseño mejora la agilidad.
- X. La simplicidad es esencial.



- XI. Las mejores arquitecturas, requisitos y diseños surgen de los equipos organizados por sí mismos.
  
- XII. En intervalos regulares, el equipo reflexiona respecto a cómo llegar a ser más efectivo, y según esto ajusta su comportamiento.

Un proceso es ágil cuando el desarrollo de software es **incremental** (entregas pequeñas de software, con ciclos rápidos), **cooperativo** (cliente y desarrolladores trabajan juntos constantemente con una cercana comunicación), **sencillo** (el método en sí mismo es fácil de aprender y modificar, bien documentado), y **adaptable** (permite realizar cambios de último momento).

### 2.1.3 METODOLOGÍAS ÁGILES:

Entre las metodologías ágiles más usadas están:

#### 1. XP(Extreme Programming)

Este tipo de programación es un enfoque formulado por Kent Beck, autor del primer libro sobre esta materia : Embrace Change (1999), en el que menciona que es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, además especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.





Una característica saltante de XP, es que el código siempre se produce en parejas, parejas que van cambiando constantemente para lograr así que todo el equipo sepa y pueda modificar según necesidades el código generado, esto logra en el equipo que los integrantes aprendan entre sí y compartan todo el código.

Los valores originales de la programación extrema son: simplicidad, comunicación, retroalimentación y coraje.

## 2. Scrum

Esta metodología fue desarrollada por Ken Schwaber, Jeff Sutherland y Mike Beedle. Y define un marco para la gestión de proyectos, que se ha utilizado con éxito durante los últimos 10 años, está especialmente indicada para proyectos con un rápido cambio de requisitos.

Sus principales características se pueden resumir en dos.

- El desarrollo de software se realiza mediante iteraciones, denominadas *sprints*, con una duración de 30 días. El resultado de cada *sprint* es un incremento ejecutable que se muestra al cliente.
- La segunda característica importante son las reuniones a lo largo proyecto. Éstas son las verdaderas protagonistas, especialmente la reunión diaria de 15 minutos del equipo de desarrollo para coordinación e integración.



### 3. Metodologías Crystal:

La metodología crystal trata de un conjunto de metodologías para el desarrollo de software caracterizadas por estar centradas en las personas que componen el equipo (de ellas depende el éxito del proyecto) y la reducción al máximo del número de artefactos producidos.

Han sido desarrolladas por Alistar Cockburn, y que manifiestan que el desarrollo de software se considera un juego cooperativo de invención y comunicación, limitado por los recursos a utilizar. El equipo de desarrollo es un factor clave, por lo que se deben invertir esfuerzos en mejorar sus habilidades y destrezas, así como tener políticas de trabajo en equipo definidas. Estas políticas dependerán del tamaño del equipo, estableciéndose una clasificación por colores

#### ➤ **Crystal Clear**

Crystal Clear no es una metodología en sí misma sino una familia de metodologías con un “**código genético**” común.

El nombre Crystal deriva de la caracterización de los proyectos según 2 dimensiones, tamaño y complejidad (como en los minerales, color y dureza).



Por ejemplo.

- Clear es para equipos de hasta 8 personas o menos.
- Amarillo para equipos entre 10 a 20 personas
- Naranja para equipos entre 20 a 50 persona.
- Roja para equipos entre 50 a 100 personas.
- Azul para equipos entre 100 a 200 personas.

#### 4. **Dynamic Systems Development Method(DSDM)**

Define el marco para desarrollar un proceso de producción de software, nace en 1994 con el objetivo de crear una metodología RAD unificada.

Sus principales características son:

Es un proceso iterativo e incremental y el equipo de desarrollo y el usuario trabajan juntos. Propone cinco fases: estudio viabilidad, estudio del negocio, modelado funcional, diseño y construcción, y finalmente implementación. Las tres últimas son iterativas, además de existir realimentación a todas las fases.

#### 5. **Adaptive Software Development(ASD)**

El impulsor de esta metodología fue Jim Highsmith, sus principales características son: iterativo, orientado a los componentes de software más que a las tareas y tolerante a los cambios.



El ciclo de vida se compone de tres fases esenciales: especulación, colaboración y aprendizaje. En la primera de ellas se inicia el proyecto y se planifican las características del software; en la segunda desarrollan las características y finalmente en la tercera, se revisa su calidad, y se entrega al cliente. La revisión de los componentes sirve para aprender de los errores y volver a iniciar el ciclo de desarrollo.

## 6. Feature-Driven Development (FDD)

Los impulsores de esta metodología son Jeff De Luca y Peter Coad y definen esta metodología como un proceso iterativo que consta de 5 pasos. Las iteraciones son cortas (hasta 2 semanas). Se centra en las fases de diseño e implementación del sistema partiendo de una lista de características que debe reunir el software.

### **ACTIVIDADES DEL PROCESO FDD:**

FDD, propone un ciclo de vida del software que consta de cinco procesos:

- **Desarrollo de un modelo global:**

Cuando esta fase comienza, los expertos del dominio ya tienen una idea del contexto y los requerimientos del sistema. Es probable que el documento de especificación de requerimientos ya exista. Sin embargo, FDD no hace énfasis en la recolección y



administración de los requerimientos. Los expertos del dominio presentan un informe llamado guía o manual (walkthrough) en el cual los miembros del equipo y el Arquitecto jefe son informados a través de una descripción de alto nivel del sistema.

- **Construcción de una lista de rasgos:**

Los manuales o guías, el modelo de objetos y los requerimientos existentes ofrecen una buena base para construir una lista de características que resuma la funcionalidad del sistema a ser desarrollado. En dicha lista, el equipo de desarrolladores presenta cada una de las funcionalidades evaluadas por el cliente. Las funcionalidades son presentadas por cada área del dominio y éstas forman un Lista de rasgos, Dicha lista es dividida en subconjuntos en base a la funcionalidad.

- **Planeación por rasgos**

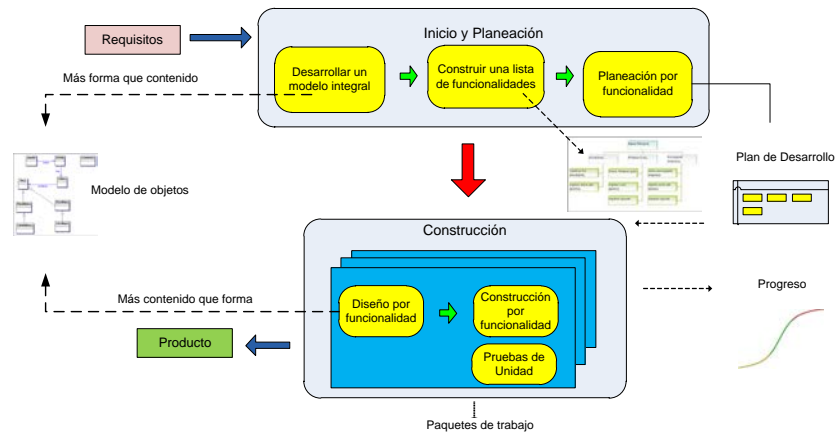
En esta etapa se incluye la creación de un plan de alto nivel en el cual la lista de características es ordenada en base a la prioridad y a la dependencia entre cada característica además, las clases identificadas en la primera etapa son asignadas a cada programador.

- **Diseño y construcción por rasgos:**

Un conjunto de rasgos es seleccionado de la lista de rasgos. El diseño y construcción de la funcionalidad es un proceso iterativo durante el cual las características seleccionadas son producidas. Una iteración puede llevar desde unos pocos días a un máximo de dos semanas. Este proceso iterativo incluye tareas como inspección del diseño, codificación, prueba unitaria, integración e inspección del código. Luego que la iteración llega a su fin se realiza



una construcción principal de la funcionalidad en el cual se integra la funcionalidad. Dicha construcción principal se realiza mientras la siguiente iteración comienza.



**Fig. # 1. Esquema de iteración de la metodología FDD**

**Fuente:** (<http://fi.ort.edu.uy/innovaportal/file/2021/1/metodologiafdd.pdf>)

Donde las dos últimas fases se realizan tantas veces como iteraciones se planifiquen en el Desarrollo.

## 2.2 Control de Calidad en Procesos

El control del proceso consiste en aplicar la calidad al proceso de fabricación de un producto  
Calidad del Proceso. (2010).kelvin Pinales ([http://www.gestiopolis.comi/Calidad\\_del\\_proceso](http://www.gestiopolis.comi/Calidad_del_proceso)).

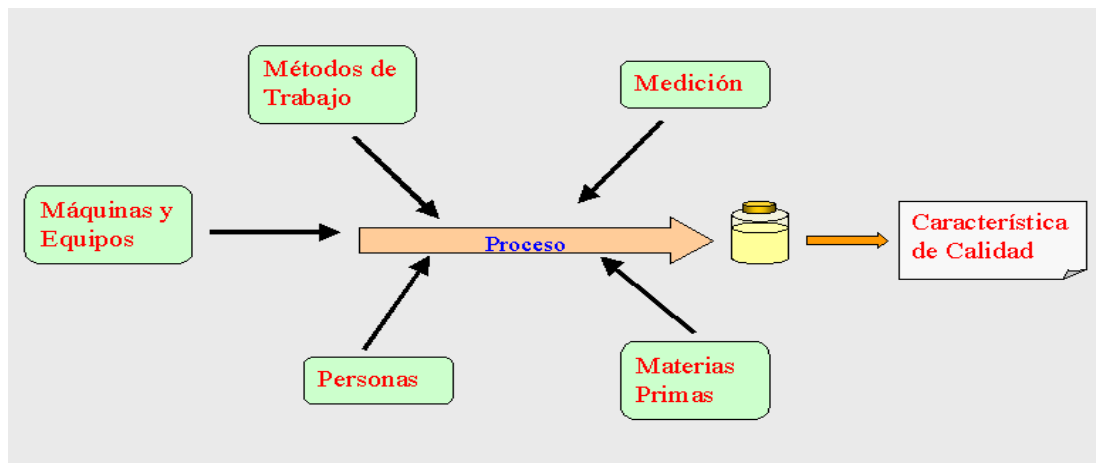
Control de la Calidad es el proceso de asegurarse de que los productos o servicios se hacen a un nivel elevado. Control de la Calidad se puede aplicar a una empresa en todas sus partes:

- La fabricación.
- El servicio al cliente.
- El rendimiento de los empleados.



- La formación de los trabajadores.
- La simplificación de todos los procesos.
- La limpieza de los lugares de trabajo.

En la figura 1.1 se muestra el diagrama de control de proceso.



**fig. 1.1 Diagrama de control de proceso**

**Fuente:**(<http://www.google.com/imgres?imgurl=http://html.rincondelvago.com/000356952.png&imgrefurl>).

### **2.3 Dirección o Administración de Operaciones**

Según Jorge E. Pereira, la administración de operaciones estudia la toma racional de decisiones en el ámbito del subsistema de operaciones para lograr los objetivos asignados. Los responsables de este subsistema orientan sus decisiones a conseguir la mayor eficacia y eficiencia de la empresa y no a la optimización de una de sus partes.

### **2.4 ESTRATEGIA DE OPERACIONES**

La estrategia de operaciones se define como un plan a largo plazo para el subsistema de operaciones en el que se recoge los objetivos a lograr y los cursos de acción para alcanzarlos.



La estrategia de operaciones tiene cuatro prioridades competitivas las que se describen a continuación:

- **Coste de la producción:** Es el valor monetario de los bienes y servicios consumidos por la empresa en su actividad de transformación incluyendo los costes de la mano de obra (MO), de los materiales y de los costes indirectos.
  
- **La calidad:** Es la idoneidad o la adecuación al uso, es decir, la bondad del producto definida por su valor, prestigio y utilidad.
  
- **Entrega:** Es la competencia basada en el tiempo que trata de lograr entregas rápidas y entregas en fecha.
  
- **Flexibilidad:** Es la habilidad de la empresa para desplegar y replegar de forma eficiente y eficaz sus recursos en respuesta a condiciones cambiantes.

## 2.5 LA TECNOLOGÍA

La tecnología es el proceso empleado por las empresas para convertir las entradas en resultados.

Todos los procesos de producción tienen una tecnología. La selección de tecnología por parte de las empresas es una cuestión que tiene importantes repercusiones.





La innovación tecnológica, brinda ciertas ventajas y beneficios de orden económico, social, y tecnológico, pudiéndose resaltar las siguientes:

- Se asegura una mejora en la calidad del trabajo del operador y en el desarrollo del proceso, esta dependerá de la eficiencia del sistema implementado.
- Se obtiene una reducción de costos, puesto que se racionaliza el trabajo, se reduce el tiempo y dinero dedicado al mantenimiento.
- Existe una reducción en los tiempos de procesamiento de información.

## **2.6 MANTENIMIENTO MECÁNICO**

Conjunto de acciones oportunas, continuas y permanentes dirigidas a prever y asegurar el funcionamiento normal, la eficiencia y la buena apariencia de sistemas, edificios, equipos y accesorios.

### **2.6.1 Tipos de Mantenimiento:**

Dentro de los principales tipos de mantenimiento tenemos los siguientes:

- Mantenimiento Preventivo
- Mantenimiento Correctivo
- Mantenimiento Predictivo



- Mantenimiento Mejorativo o Rediseños
- Mantenimiento Selectivo.

### **Mantenimiento Preventivo:**

Son los servicios de inspección, control, conservación y restauración de un ítem con la finalidad de prevenir, detectar o corregir defectos, tratando de evitar fallas. Este mantenimiento se realiza con una frecuencia dependiendo de la criticidad del equipo.

### **Mantenimiento Correctivo:**

Son los servicios de reparación en ítems con falla, es decir el mantenimiento, y se realiza cuando se detecta la falla o cuando ya ocurrió.

### **Mantenimiento Predictivo:**

Son los servicios de seguimiento del desgaste de una o más piezas o componente de equipos prioritarios a través de análisis de síntomas, o estimación hecha por evaluación estadística, tratando de extrapolar el comportamiento de esas piezas o componentes y determinar el punto exacto de cambio.

El mantenimiento predictivo se da por la confiabilidad o la forma sistemática de como preservar el rendimiento requerido basándose en las características físicas, la forma como se



utiliza, especialmente de cómo puede fallar y evaluando sus consecuencias para así aplicar las tareas adecuadas de mantenimiento (preventivas o correctivas).

### **Mantenimiento mejorativo o Rediseños:**

Consiste en la modificación o cambio de las condiciones originales del equipo o Instalación.

### **Mantenimiento Selectivo:**

Son los servicios de cambio de una o más piezas o componentes de equipos prioritarios, de acuerdo con recomendaciones de fabricantes o entidades de investigación.

### **2.6.2 Aplicaciones Del Mantenimiento:**

El principio del mantenimiento es asegurar que todo activo continúe desempeñando las funciones deseadas. Con el objetivo de asegurar la competitividad de la empresa por medio de:

- Garantizar la disponibilidad y confiabilidad planeadas de la función deseada.
- Satisfacer todos los requisitos del sistema de calidad de la empresa.
- Cumplir todas las normas de seguridad y medio ambiente, y maximizar el beneficio global.



### 2.6.3 Mantenimiento en Motocicletas

Durante la vida de una motocicleta las operaciones de mantenimiento básico son cruciales para el buen funcionamiento del conjunto y para su larga duración. Lo usual es que en los concesionarios oficiales pagan un precio a su garantía y son los únicos que pueden dársela y además, son los que tienen acceso a toda la información acerca de su motocicleta mediante el servicio postventa de la marca.

#### **Mantenimiento Básico:**

El manual del propietario de una motocicleta indica las operaciones de mantenimiento recomendado y los intervalos con los que se debe hacer pero casi todos coinciden en las operaciones básicas que son:

- a) Cambio de aceite cada 6000 km.
- b) Cambio de filtro de aceite cada 1200 km.
- c) Cambio de bujías cada 12000 km.
- d) Cambio o limpieza del filtro de aire (cada 12000 km para el cambio y 6000 km para la limpieza).
- e) Engrase general.



**a) Cambio aceite cada 6000 km:**

En la figura 1.2 se muestra el procedimiento de cambio cada 6000 km de aceite, se procede localizando el tornillo de vaciado del cárter. Colocar el recipiente para recoger aceite bajo el tornillo y desenroscarlo, dejar escurrir el aceite sobre el recipiente una hora o más.



**Fig. 1.2. Cambio de aceite**

**Fuente:** ([http://www.gassattack.com/articulos\\_tecnicos/mantenimiento%20basico.pdf](http://www.gassattack.com/articulos_tecnicos/mantenimiento%20basico.pdf))

**b) Cambio de filtro de aceite cada 1200 km:**

En la figura 1.3 se muestra el cambio de filtro de aceite, el cual usualmente necesita un cambio cada 1200 km, los más puristas prefieren hacerlo cada cambio de aceite.



**Fig. 1.3 Filtro de aceite**

**Fuente:** ([http://www.gassattack.com/articulos\\_tecnicos/mantenimiento%20basico.pdf](http://www.gassattack.com/articulos_tecnicos/mantenimiento%20basico.pdf))

**c) Cambio de bujías cada 12000 km:**

En la figura 1.4 se muestra el cambio de bujías, lo más adecuado y fácil es proceder retirando la caja del filtro de aire (airbox) y desde allí proceder.



**Fig. 1.4. Cambio de bujías**

**Fuente:** ([http://www.gassattack.com/articulos\\_tecnicos/mantenimiento%20basico.pdf](http://www.gassattack.com/articulos_tecnicos/mantenimiento%20basico.pdf))



#### d) Limpieza o cambio del filtro de aire:

En la figura 1.5 se muestra el procedimiento y la limpieza o cambio del filtro de aire.

**Limpieza:** Si es filtro de papel sacarlo del airbox, eliminar todos los residuos que allí se encuentren, con la ayuda de un compresor de aire soplar en dirección en la que circula el aire, hacerlo al revés puede obstruirlo.

Si es un filtro de aceite, lavar con agua y jabón, luego impregnar en aceite y apretarlo para eliminar el aceite sobrante.

**Cambio:** Solamente se tiene que cambiar el viejo por una nuevo.



**Fig. 1.5. Filtro de aire**

**Fuente:**([http://www.gassattack.com/articulos\\_tecnicos/mantenimiento%20basico.pdf](http://www.gassattack.com/articulos_tecnicos/mantenimiento%20basico.pdf))

#### e) Engrase general

Antes de engrasar las piezas descritas a continuación, hay que limpiarlas de grasa, suciedad y óxidos.



Engrasar con aceite para motor:

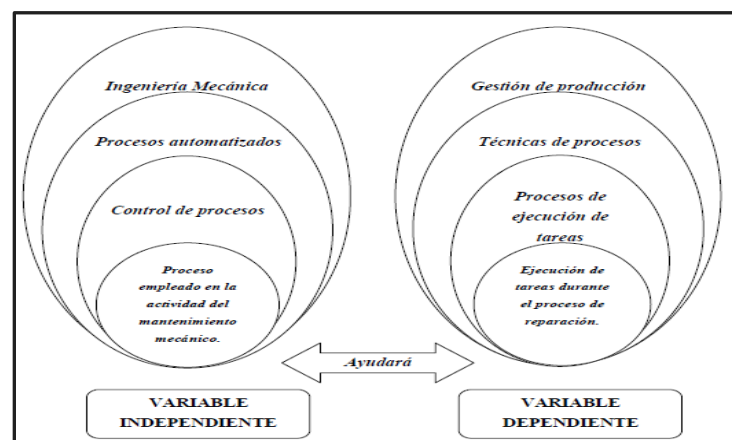
- Caballete central
- Manija de embrague
- Manija de freno
- Pata de cabra

f) **Engrasar con grasa negra:**

- Extremo superior del cable del embrague
- Extremo inferior del cable del embrague
- Extremos del cable del acelerador

## 2.6.4 CATEGORÍAS FUNDAMENTALES DEL MANTENIMIENTO

En la figura 1.5 se muestran las categorías fundamentales del mantenimiento de motocicletas.



Fuente:(<http://repo.uta.edu.ec/bitstream/handle/123456789/2316/Maestria V. T. 66>)

**Fig. # 1.6 Categorías fundamentales del mantenimiento de motocicletas.**





### III. DISEÑO METODOLÓGICO:

Para llevar a cabo el desarrollo del presente trabajo se ha implementado la metodología ágil de desarrollo de software, Desarrollo basado en componentes o características (FDD).

Entiéndase por FDD con sus siglas en inglés (Feature Driven Development) como un enfoque ágil para el desarrollo de sistemas. Dicho enfoque no hace énfasis en la obtención de los requerimientos sino en cómo se realizan las fases de diseño y construcción. Sin embargo, fue diseñado para trabajar con otras actividades de desarrollo de software y no requiere la utilización de ningún modelo de proceso específico.

A pesar de ser una metodología ágil la metodología FDD tiene muy bien definido los roles dentro de la misma, los cuales pueden ser divididos en roles claves, roles de soporte y roles adicionales.

#### Roles Claves

- **Director del proyecto:** Es el líder administrativo y financiero del proyecto, protege al equipo de situaciones externas.
- **Arquitecto Jefe:** Diseña el modelo global del sistema también tiene la última palabra sobre las decisiones del diseño que se puedan presentar sobre todo el sistema, se puede dividir en Arquitecto Dominio y Arquitecto Técnico.
- **Director de desarrollo:** lleva diariamente las actividades del desarrollo, resuelve conflictos que se pueden presentar dentro de las propiedades de las clases y resuelve problemas referentes a recursos, las tareas se pueden combinar con las del Director del proyecto y el Arquitecto Jefe.



- **Programador jefe:** Hace diseño del proyecto, analiza todos los requerimientos que pueda tener el sistema y selecciona de cada fase cuales son las funcionalidades a las que se le va a dar una solución, debe ser un desarrollador con experiencia ya que es quien identifica las clases y propietarios para que se trabaje en cada iteración.

### Roles de Soporte

- **Gestor de Dominio:** Su principal función es resolver cada una de las diferencias de opinión que se presenten sobre los requerimientos del sistema.

- **Gestor de Liberación:** Presenta resultados obtenidos al Director del proyecto y al cliente.

- **Gurú del lenguaje:** Responsable en tener un vasto conocimiento en un lenguaje o tecnología, muy importante cuando se trabaja en una nueva tecnología.

- **Ingeniero de construcción:** sus funciones son reparar, mantener y correr el proceso de construcción, realiza mantenimiento de las versiones del sistema y publicación de documentación.

### Roles Adicionales

**Inspector:** Verifica que el sistema cumpla con lo requerido, puede ser una persona independiente al equipo del proyecto.

**Escritor Técnico:** Prepara documentación por usuario, en caso de que una persona se una al equipo a mediados del proyecto, el escritor técnico se encarga de que la persona pueda integrarse rápidamente al desarrollo del mismo.



**Propietario de clases:** Son los responsables de los desarrollos de clases que les han asignado como propias, participan en la decisión de que clase va a ser incluida en la lista de funcionalidades de la próxima iteración, además trabaja con la guía del programador jefe en las tareas de diseño, codificación, prueba y documentación.

**Expertos del dominio:** Debe ser una persona con conocimiento en los requerimientos del sistema, su función principal es pasar conocimientos a los desarrolladores para asegurar la entrega de un sistema completo.

Esta metodología tiene 5 procesos o actividades, los cuales describen tareas que se detallan a continuación

### **3.1 Actividades de la metodología ágil FDD**

#### **3.1.1 Desarrollo de un modelo global**

El cliente está listo para comenzar con la construcción del sistema. Además posee una lista de requerimientos especificada de alguna forma y un diagrama de clase del sistema.

A continuación se detallan los requerimientos solicitados por el cliente, MASESA.

- El sistema debe permitir la realización de transacciones (agregar nuevo cliente, ingresar nueva orden, buscar, eliminar y actualizar), y almacenarlas en la base de datos.
- El sistema debe permitir, agregar un nuevo usuario y el estado que este tendrá, entiéndase por, administrador, cajero.
- El sistema debe ser seguro, también debe genera reporte de clientes después de haber recibido el servicio de manteamiento a las motocicletas.



- **Diagrama de clases:**

El diagrama de clases (figura 1.6) muestra la estructura general del sistema, así como las propiedades relacionadas y de comportamiento.

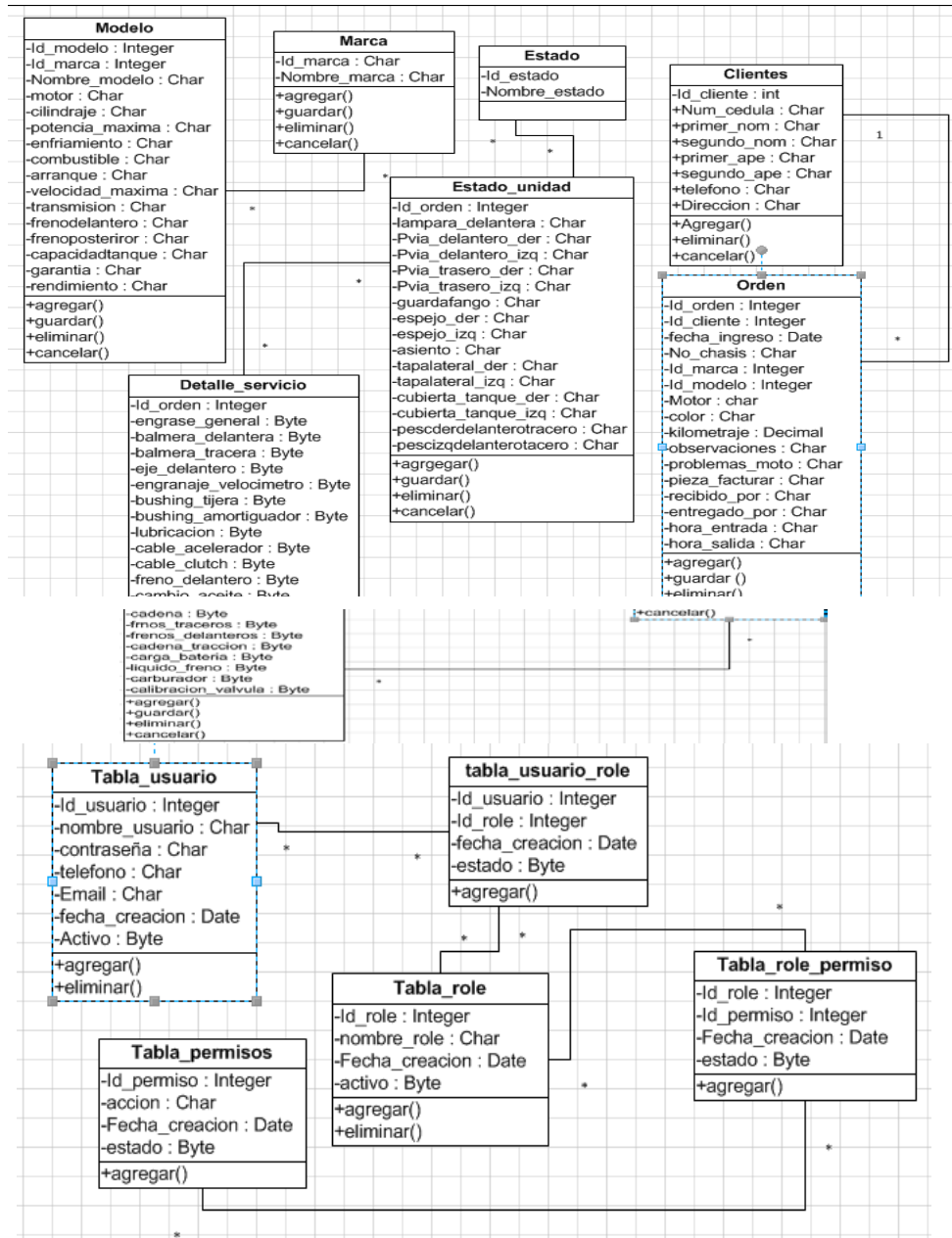


Fig. # 1.7 Diagrama de clases general del sistema



### **3.1.2 Construcción de una lista de rasgos o requerimientos.**

A través del proceso de reuniones con el equipo se determinó una lista de rasgos como se lista a continuación:

- Ingreso de nuevo cliente.
- Ingreso de nueva orden.
- Registros de clientes que llegan al taller y almacenarlos en una base de datos.
- Control y registro de las motocicletas y sus características, para ser valoradas por el taller.
- Llevar un control de los trabajos que son realizados a las motocicletas al momento del chequeo mecánico.
- Llevar un control del estado en que son ingresadas las motocicletas.
- Generar reporte al cliente después de haber recibido el servicio de mantenimiento a su motocicleta.
- Generar a diario reporte de clientes a los cuales la administración debe llamar, tomando como parámetro la fecha del último mantenimiento, ya que la frecuencia de chequeo es continua.
- Buenas interfaces de usuario (ingreso nuevo cliente, ingreso nueva orden, nuevo usuario).

### **3.1.3 Planeación por rasgos:**

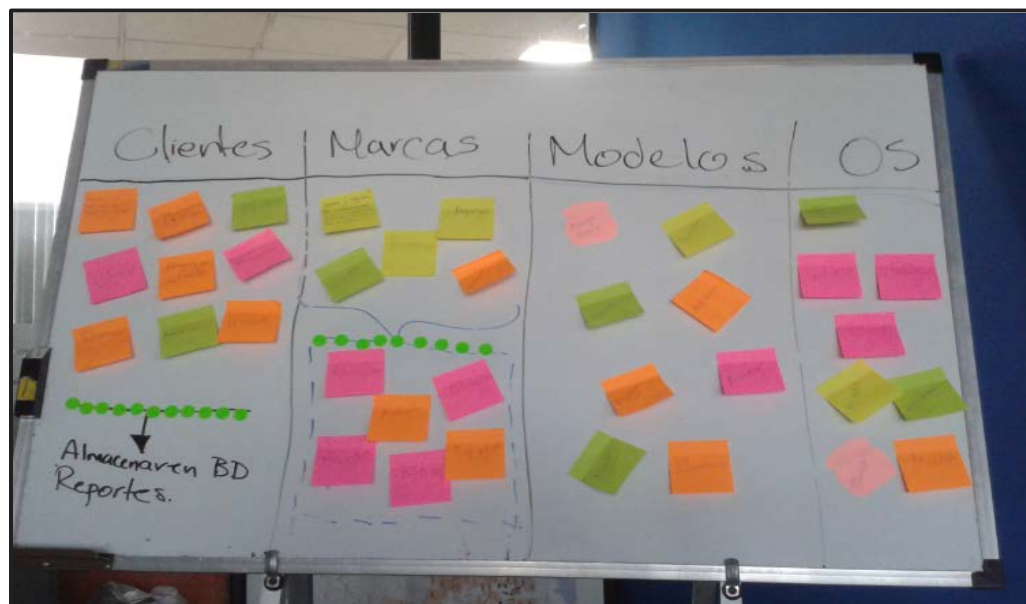
En esta fase el equipo de planificación es formada por: El administrador del proyecto el administrador de desarrollo y jefes de programadores.



Esta fase incluye la creación de un plan de alto nivel, los miembros colaboran para obtener un análisis y diseño completo a bajo nivel en el que los conjuntos de rasgos se ponen en secuencia conforme a su prioridad y dependencia, y se asigna a los programadores jefes.

- Sistema de consultas y transacciones y la correspondiente actualización de la base de datos y del sistema de contabilidad.
- Interfaces de usuario, para todas las consultas y las transacciones y su correspondiente integración al sistema.
- Sistema de seguridad, que incluya las restricciones del sistema y protección contra accesos indebidos y su integración al sistema

**La Figura 1.7 muestra la fase de elaboración de la planeación por rasgos.**



**Fig. 1.8**

**Fuente: Propia**

**Diagrama descriptivo de la fase de elaboración de plan por rasgos de FDD**



Se han ordenado los grupos de funcionalidades, según su prioridad y la dependencia y a cada una de ellas se le asignó un responsable:

➤ **Luis Adolfo Henriquez**

**Cronograma:**

En la tabla # 1 se detalla las actividades planificadas y realizadas, durante la construcción del software.

**Tabla # 1 Cronograma de Actividades**

	ACTIVIDAD	DURACION	TIEMPO ESTIMADO		TIEMPO REAL DE EJECUCIÓN		OBSERVACION
			FECHA INICIO	FECHA FIN	FECHA INICIO	FECHA FIN	
DESARROLLO DE UN MODELO GLOBAL	PRIMERA VISITA A LA EMPRESA	1 DIA	26-Aug-13	26-Aug-13	26-Aug-13	26-Aug-13	RECOLECCION DE DATOS DE LA EMPRESA y ENTREVISTA CON GERENTE DE LA
	ANALISIS DE LOS DATOS RECOPIADOS	15 DIAS	31-Aug-13	7-Sep-13	31-Aug-13	7-Sep-13	-
	REQUERIMIENTOS DEL SISTEMA A CONSTRUIR	15 DIAS	14-Sep-13	21-Sep-13	14-Sep-13	21-Sep-13	-
	DIAGRAMA DE CLASES GLOBAL DEL SISTEMA	3 DIAS	28-Sep-13	30-Sep-13	28-Sep-13	2-Oct-13	-
CONSTRUCCION DE LISTA DE RASGOS	ELABORACION DE LISTA DE REQUERIMIENTOS	1 SEMANA	5-Oct-13	12-Oct-13	5-Oct-13	12-Oct-13	
	PRESENTACION DE LISTA DE AGRUPACION DE FUNCIONALIDADES SEGUN AFINIDAD Y	15 DIAS	14-Oct-13	18-Oct-13	14-Oct-13	22-Oct-13	RETRASOS POR REVISION Y ACEPTACION DE REQUERIMIENTOS POR PARTE DEL
	ASIGNACION DE FUNCIONALIDADES A CADA MIEMBRO EL DISEÑO Y CONSTRUCCION DE	1 MES	26-Oct-13	26-Nov-13	26-Oct-13	2-Dec-13	
PLANEACION POR RASGOS	ASIGNACION DE FUNCIONALIDADES A CADA MIEMBRO EL DISEÑO Y CONSTRUCCION DE	1 SEMANA	7-Dec-13	14-Dec-13	7-Dec-13	14-Dec-13	SE DIVIDIÓ EQUITATIVAMENTE LAS FUNCIONALIDADES A REALIZAR ENTRE LOS MIEMBROS DEL EQUIPO
	DISEÑO Y CONSTRUCCION DE	1 MES	18-Jan-14	22-Feb-14	18-Jan-14	22-Feb-14	
DISEÑO Y CONSTRUCCION POR	INSPECCION DE DISEÑO	1 SEMANA	24-Feb-14	1-Mar-14	24-Feb-14	1-Mar-14	
	CODIFICACION	2 MESES	3-Mar-14	3-May-14	3-Mar-14	3-May-14	
	CASOS DE USO	3 DIAS	5-May-14	8-May-14	5-May-14	8-May-14	
	PRUEBAS UNITARIAS	1 SEMANA	10-May-14	17-May-14	10-May-14	17-May-14	
	INTEGRACION E INSPECCION DE CODIGO	1 SEMANA	19-May-14	24-May-14	19-May-14	24-May-14	

Fuente: Propia



### 3.1.4 Diseño y construcción por rasgos

Estas dos fases, implican un proceso iterativo, que comienza con el diseño y termina con la prueba de la funcionalidad implementada, pasando por la codificación, su evaluación y la integración al sistema y pruebas unitarias del mismo.

#### ✓ Descripción de clases y métodos

**Clase Modelo:** la clase modelo, registra todas la características específicas de la carrocería o hardware de cada modelo de motocicleta, y tiene como procesos para ejecutar esta acción, agregar, guardar, eliminar y cancelar, los cuales estas acciones están definidas de acuerdo a los permisos asignados a cada usuario.

**Clase Marca:** La clase marca registra el nombre de la marca de la motocicleta, y tiene como procesos, agregar, eliminar, guardar y cancelar, los cuales estas acciones están definidas de acuerdo a los permisos asignados a cada usuario.

**Clase Estado:** Registra el estado en que se encuentra la motocicleta, y tiene como proceso agregar y eliminar.

**Clase Clientes:** La clase cliente registra todos los datos de los clientes, y tiene como procesos, agregar, eliminar, guardar y cancelar, los cuales estas acciones están definidas de acuerdo a los permisos asignados a cada usuario.





**Clase Detalle de Servicio:** registra el Id de orden de cada cliente, y el tipo de servicio a brindar a cada motocicleta, y tiene como procesos, agregar, eliminar, guardar y cancelar, los cuales estas acciones están definidas de acuerdo a los permisos asignados a cada usuario.

**Clase Estado de la unidad:** La clase estado de la unidad, registra el estado en que la motocicleta ingresa al taller, y tiene como procesos, agregar, eliminar, guardar y cancelar, los cuales estas acciones están definidas de acuerdo a los permisos asignados a cada usuario.

**Clase Orden:** La clase orden registra el número de orden de cada cliente, la hora en que ingresa la motocicleta al taller, y la hora en que sale del mismo, asimismo se registra en la orden, la marca y modelo de la motocicleta, las piezas a facturar después del chequeo, además se registra, el nombre de la persona del taller quien recibe y entrega la motocicleta, y tiene como procesos, agregar, eliminar, guardar y cancelar, los cuales estas acciones están definidas de acuerdo a los permisos asignados a cada usuario.

**Clase Usuario:** Registra los datos de los usuarios que van a hacer uso del sistema, y tiene como procesos, agregar, eliminar, guardar y cancelar, estas acciones están habilitadas solo para los administradores del sistema.

**Clase permisos:** La clase permisos registra los tipos de usuarios que podrán hacer uso del sistema, y el estado en que se encuentra el usuario, estos son: activos e inactivos, estas acciones están habilitadas solo para los administradores del sistema.



**Clase usuario-role:** la clase usuario-role registra los roles que tiene asignado cada usuario de acuerdo a sus permisos, estas acciones están habilitadas solo para los administradores del sistema.

**Clase role:** la clase role registra las acciones o roles que podrá realizar cada usuario de acuerdo a sus permisos asignado, estas acciones están habilitadas solo para los administradores del sistema.

**Clase role- permiso:** la clase role permiso registra el Id del permiso, la fecha de creación de este, y el estado en que se encuentra, activo e inactivo, estas acciones están habilitadas solo para los administradores del sistema.

## **3.2 RESULTADOS:**

### **3.2.1 Primera Iteración (Anexo 2, figura 4.5)**

El sistema fue desarrollado en:

- Microsoft Visual Basic.Net.
- SQL Server 2008.
- implementado sobre Windows 7, y cumple con las funciones solicitadas por el cliente.

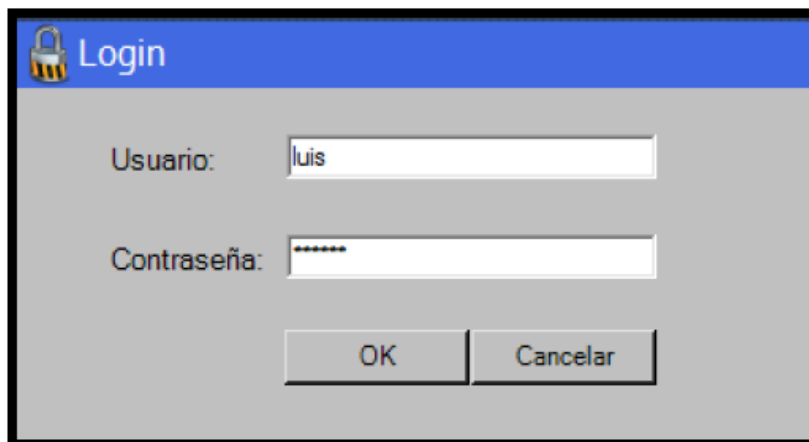


En la figura 1.9 se muestra la pantalla principal del sistema de control y mantenimiento de motocicletas.



**Figura 1.9** Interfaz gráfica del sistema de control y mantenimiento

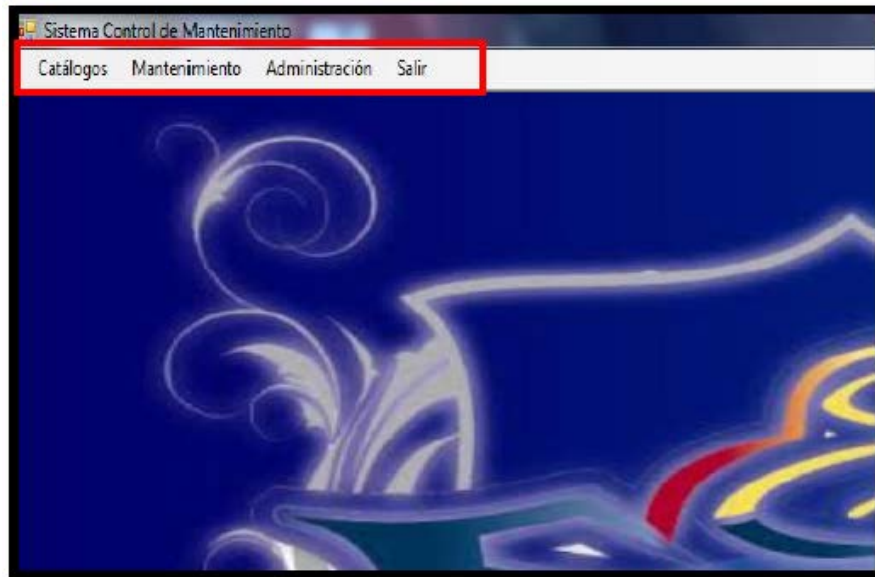
En la figura 2.1 se muestra la ventana del acceso al sistema, donde el operario accede al sistema, a través de un nombre de usuario y contraseña.





**Figura # 2.1 Inicio sistema de información**

En la figura # 2.2 se muestra el menú principal del sistema.



**Figura # 2.2 Formulario general del sistema**

En la figura 2.3 se muestra el menú “catálogos” en donde se encuentra el submenú, clientes, marcas y modelo.

**Catálogos / Clientes:**





**Figura 2.3 Ingreso al formulario clientes:**

En la figura 2.4 se muestra el formulario clientes, con todos los clientes registrados, bajo el encabezado de cada columna se presenta un cuadro de texto que funciona como buscador, el formulario posee las operaciones siguientes: Agregar registro, editar registro, Guardar cambios, cancelar y eliminar.

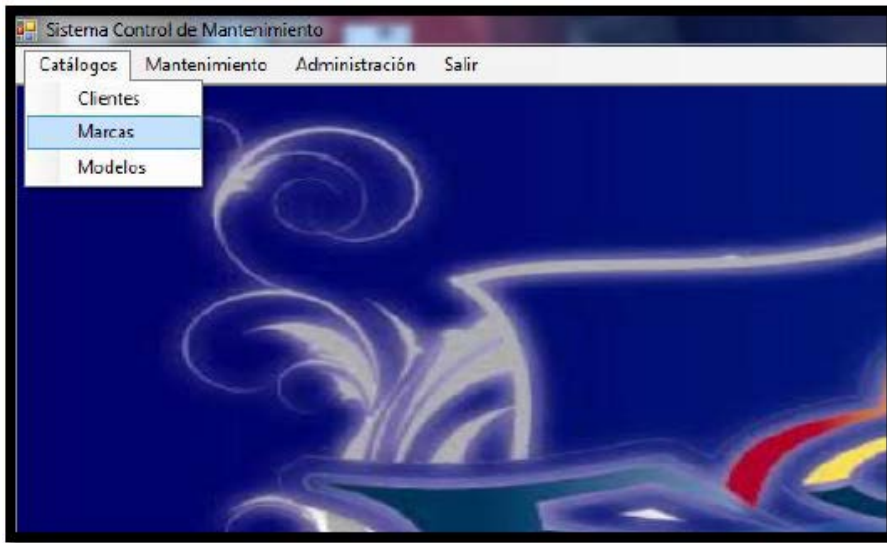
El formulario 'Clientes' muestra una tabla con los siguientes datos:

Id Cliente	No. Escuela	Primer Nombre	Segundo Nombre	Primer Apellido	Segundo Apellido	Telefono
11	001-23004-0013L	AMANDA	DEL SOLORRO	ESQUIVEL	GONZALEZ	89730364
8	001-221083-0049H	LUIS	ADOLFO	HENRIQUEZ	FARIÑA	83288917
10	004-130170-0001A	VERONICA	DE JESUS	MUÑOZ	TORRES	89715428
9	001-270471-0050V	JOSE	GABRIEL	QUINTANILLA	ESQUIVEL	86604423
12	001-260795-0025F	LESTER	ANTONIO	RAYO	FONSECA	89872655
13	001-191283-0000L	KARLA	PATRICIA	REYES	MARTINEZ	82403057
7	281-230690-0002M	ROBERTO	ANTONIO	SOLIS	SOZA	86907823

**Figura 2.4 formulario clientes**

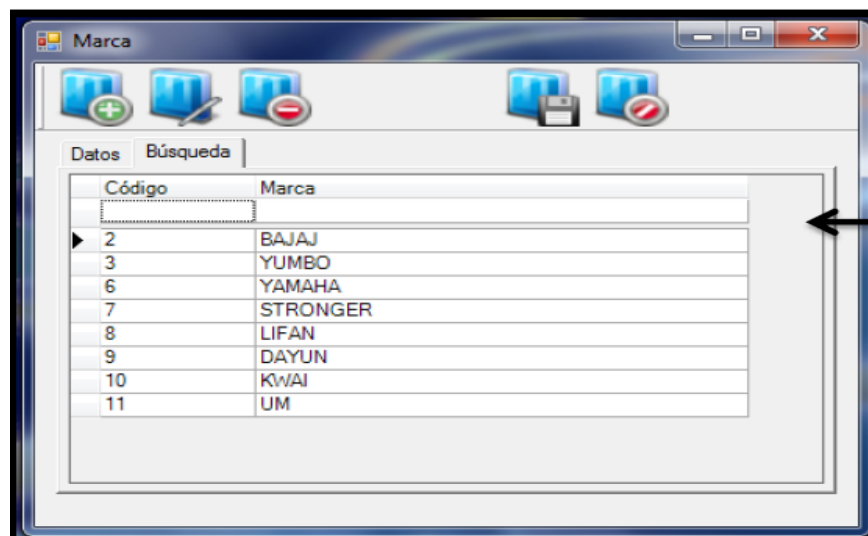


**Catálogos / Marcas** En la figura 2.5 se muestra el ingreso al formulario marcas



**Figura 2.5 formulario Ingreso a marcas**

En la figura 2.6 se muestra se muestran los registros de marcas de motocicletas que el taller puede recibir.

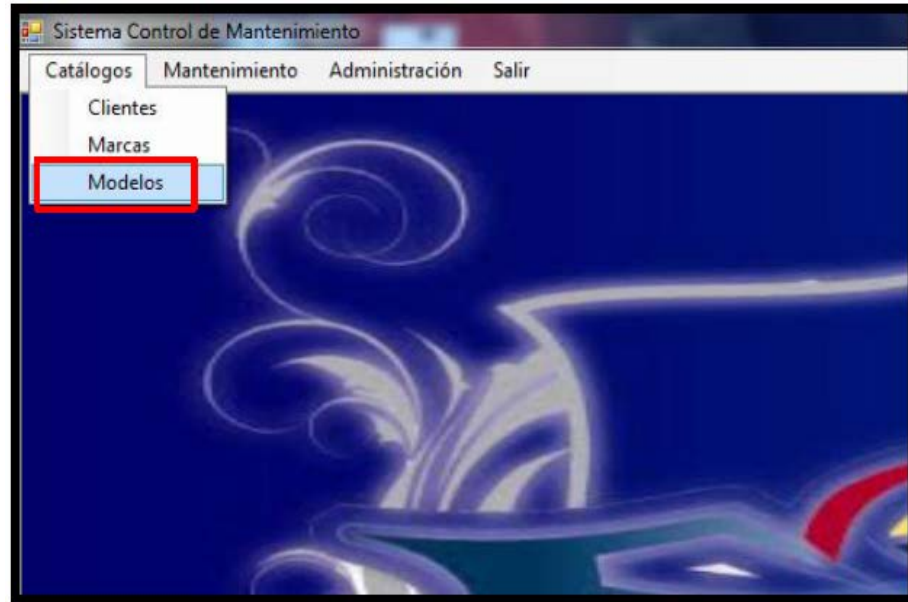


**Figura 2.6 Formulario marcas**



## Catálogos / Modelos

En la figura 2.7 se muestra el acceso al formulario modelos.



**Figura 2.7 formulario Ingreso a modelos**

En la figura 2.8 se muestra la descripción de todos los modelos existentes en el taller y sus especificaciones.

Código	Modelo	Marca	Enfriamiento	Pot. Máxima	Cilindraje	Motor	Combustible	Arranque
10	PULSAR 135	BAJAJ	Por Aire	13,5 hp a 9.000 R	135cc	4 tiempos dts	Gasolina Regular	Electrico y Patada
12	PULSAR 180	BAJAJ	Por Aire	16,5 hp a 8.000 R	180 cc	4 tiempos dts-l	Gasolina Regular	Electrico y de Pata
13	PULSAR 200	BAJAJ	Por Refrigerante	21,04 hp a 8.500 R	200 cc	4 tiempos dts-l	Gasolina Regular	Electrico
14	PULSAR 150	BAJAJ	Por Aire	13,5 hp a 9.000 R	150 cc	4 tiempos dts-l	Gasolina Regular	Electrico y Patada
15	DISCOVER 125	BAJAJ	Por Aire	11,5 hp a 8.000 R	125 cc	4 tiempos dts-l	Gasolina Regular	Electrico y de Pata
16	DISCOVER 100	BAJAJ	Por aire	7,6 hp a 7.500 RP	94,39 cc	4 tiempos dts-l	Gasolina Regular	Electrico y de Pata
17	DISCOVER 125 S	BAJAJ	Por Aire	6,04 hp a 5000 RP	223 cc	4 tiempos dts-l	Gasolina Regular	Electrico y de Pata
18	BOXER 100	BAJAJ	Por Aire	8,2 HP A 7.500 RP	100 cc	4 tiempos dts-l	Gasolina Regular	De Patada unicam
19	BOXER 150	BAJAJ	Por Aire	8,2 hp a 7.500 RP	150 cc	4 tiempos dts-l	Gasolina Regular	Electrico y de Pata
20	PLATINA 125	BAJAJ	Por Aire	9,5 hp a 7.000 RP	125 cc	4 tiempos dts-l	Gasolina Regular	Electrico y de Pata
11	BIT 125	YUMBO	Por Aire	8,85 hp a 7.500 R	125 cc	4 tiempos dts-l	Gasolina Regular	Electrico y de P
21	TRAKIP 300	YUMBO	Por Aire	14,3 hp a 8.000 R	200 cc	4 tiempos dts-l	Gasolina Regular	Electrico y de Pata

**Figura 2.8 Formulario modelos**



### 3.2.2 Segunda Iteración (Anexo 3, Fig. 4.6)

En la figura 2.9 se muestra una única opción de submenú, levantar orden de recepción y servicio.



**Figura 2.9 Ingreso al submenú Orden de recepción y servicio**

En la figura 3.1 se muestran todos los datos necesarios para levantar una orden de servicio.

El formulario 'Orden de Recepción y Trabajo' está dividido en varias secciones:

- Detalle Orden:** Incluye campos para 'Fecha Ingreso' (con calendario), 'Cliente', 'Marca', 'Modelo', 'Chasis', 'Motor', 'Km.' y 'Color'.
- Estado de la Unidad:** Una cuadrícula de 4x3 de botones desplegables con el valor 'Bueno' en cada uno, para registrar el estado de componentes como: Lempare Delantero, P/Via Delantero Der./Izq., P/Via Trasero Der./Izq., Guardafango, Espejo Der./Izq., Asiento, Tapa Lateral Der./Izq., Cubierta Tanque Der./Izq., Pies Der./Izq. Delant./Tracer., y Pies Izq. Delant./Tracer.
- Detalle del Servicio:** Una lista de casillas de verificación para seleccionar los tipos de servicios a realizar, como: Engrase General, Balinera Delantero, Eje Delantero, Balinera Trascera, Engranaje Velocimetro, Bushing Tijero, Bushing Amortiguador, Lubricacion, Cable Acelerador, Cable Clutch, Freno Delantero, Cambio Aceite, Codeno, Cadena Traccion, Frenos Trasceros, Frenos Delanteros, Carga Baterias, Liquido de freno, Carburador, and Calibracion Valvula.

**Figura 3.1 Formulario Orden de recepción de servicio**





### 3.2.3 Tercera Iteración (Anexo 3, Fig. 4.7)

En la figura 3.2 se muestra la pantalla de acceso a los reportes

## Reportes

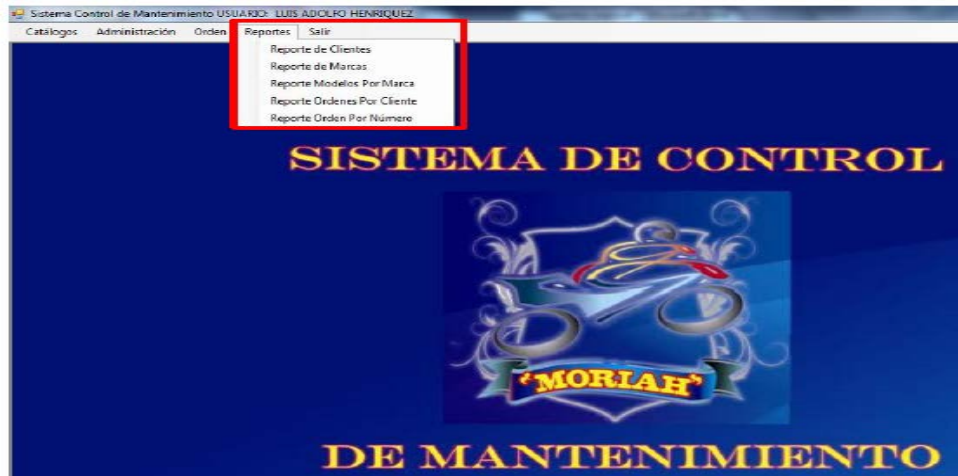


Figura 3.2 Ingreso a reportes

## Reporte/ Clientes

En la figura 3.3 se muestra el reporte de clientes.

No Cédula	Primer Nombre	Segundo Nombre	Primer Apellido	Segundo Apellido
0012210830049H	LUIS	ADOLFO	HENRIQUEZ	FARIÑA
0012704710050V	JOSE	GABRIEL	QUINTANILLA	ESQUIVEL
0041301700001A	VERONICA	DE JESUS	MUÑOZ	TORRES
0012302430013L	AMANDA	DEL SOCORRO	ESQUIVEL	GONZALEZ
0011912630000L	KARLA	PATRICIA	REYES	MARTINEZ
0012211890034F	MARTHA	EMILIA	SANCHEZ	VILLAVIGENCIO
0001118640000A	DENNIS	URIEL	TORRES	PEREZ
00030908840004T	JUAN	SIO	CUADRA	FLORES
0010101830000J	FRANCIS	GUADALUPE	OCON	CRUZ
0010102710084M	EDUARDO		MORA	DESHON
0010102920044F	MAXIMO	CELEDONIO	MEDINA	
0010103820032D	FABRICIO	ALMANZOR	HERRERA	CACERES
1242503790001T	ROSIBEL		ALANIS	ROSTRAN
12345008900024R	MARIO	ERNESTO	CANIZALES	RIVAS
16121088200005R	FREDDY	LUIS	GUADAMUZ	COREA
00129119000061E	CRISTOPHER	DEJESUS	HERNANDEZ	OROZCO
1231204890002W	JUANA	AZUCENA	CAMPOS	FLORES
0012007680003R	WALTER	MARTIN	SOTO	DAVILA

Figura 3.3 Reporte clientes



## Reporte / Marcas

En la figura 3.4 se presenta todas las marcas existentes.

Nombre Marca
BAJAJ
YUNEO
YAMAHA
STRONGER
LIFAN
DAYUN
LIM
KAYAK MONTANERA
CHINITA

Figura 3.4 Reporte marcas

## Reporte / Modelos por marca

En la figura 3.5 se muestra el reporte de modelos por marca.

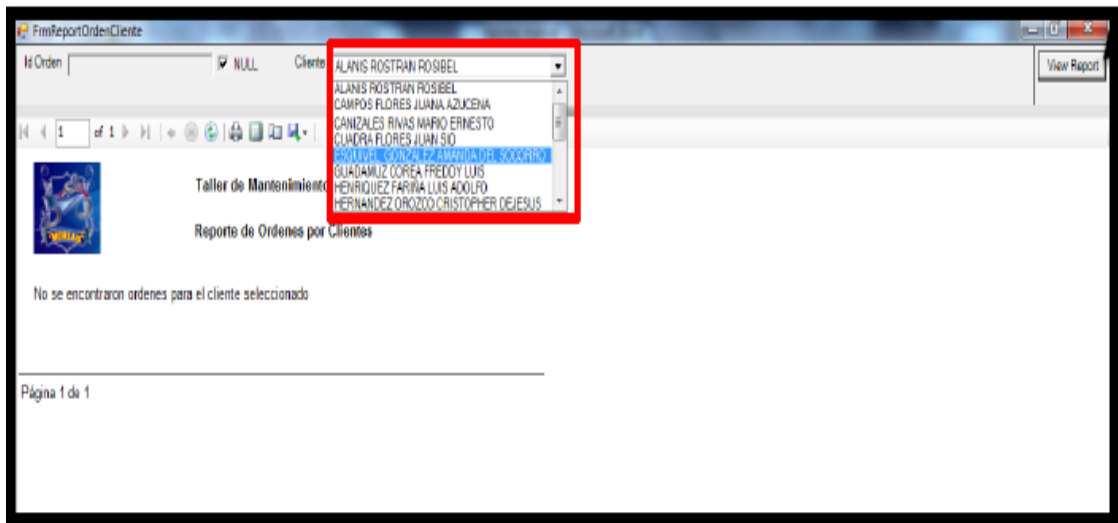
Modelo	Motor	Cilindraje	Potencia Máxima
PULSAR 200	4 tiempos dts-l	180 cc	16,5 hp a 8.000 RPM
PULSAR 180	4 tiempos dts-l	200 cc	21,04 hp a 8.500 RPM
DISCOVER 100	4 tiempos dts-l	94,38 cc	7,6 hp a 7.500 RPM
DISCOVER 125 ST	4 tiempos dts-l	223 cc	9,04 hp a 5000 RPM
BOXER 100 BAJAJ	4 tiempos dts-l	100 cc	8,2 HP A 7.500 RPM
PLATINA 125	4 tiempos dts-l	125 cc	9,5 hp a 7.000 RPM

Figura 3.5 Reporte modelos por marca



## Reporte / Órdenes por cliente

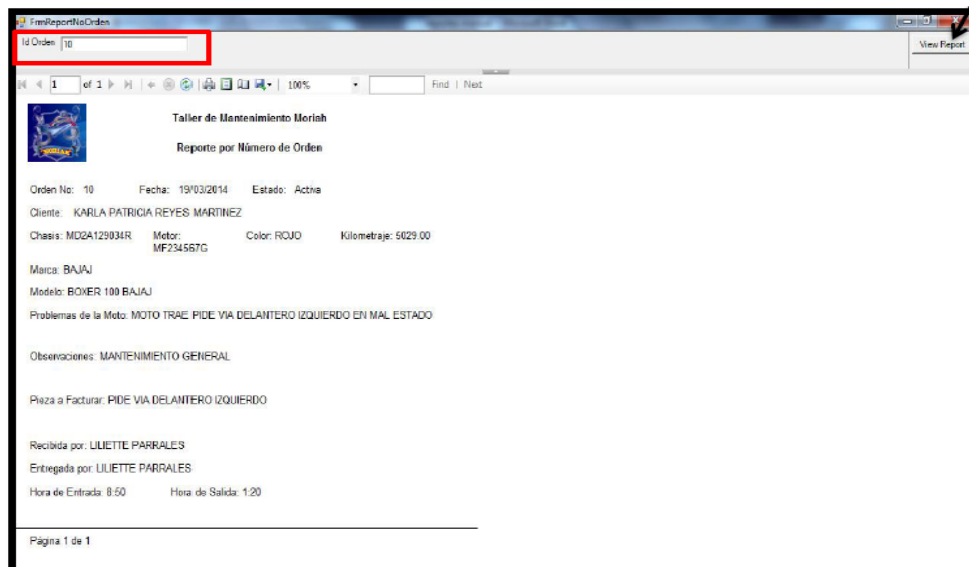
En la figura 3.6 se muestra el reporte de órdenes por cliente



**Figura 3.6 Reporte órdenes por cliente**

## Reporte /órdenes por numero

En la figura 3.7 se muestra el reporte de órdenes por número.

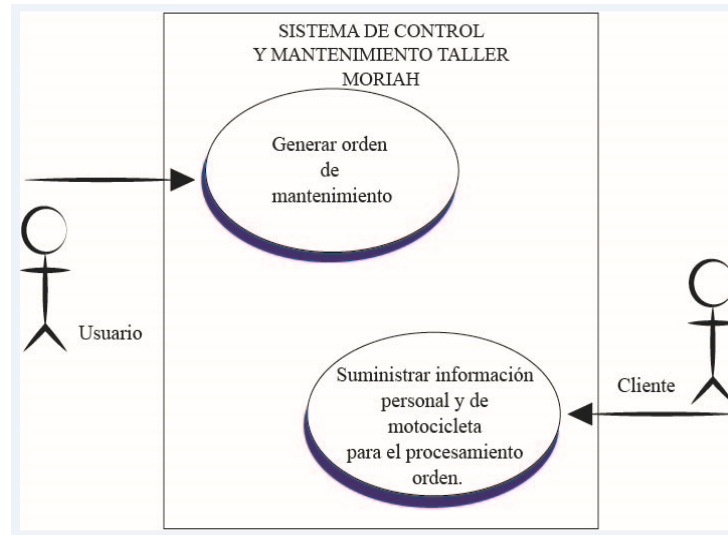


**Figura 3.7 Reporte ordenes por numero**

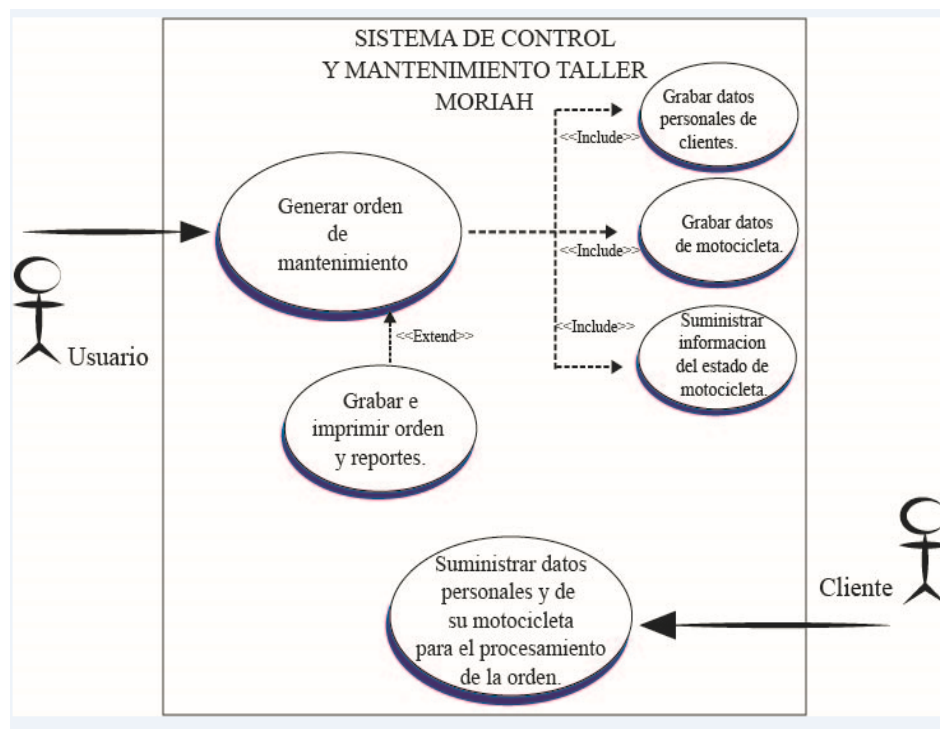


### 3.3 Diagrama de Casos de uso

#### Diagrama general del sistema



#### Diagrama de caso de uso “atención al cliente”





### 3.4 Pruebas Unitarias

Se realizaron pruebas unitarias a los siguientes métodos:

1. **Public Shared Function** Existe Cédula (cédula **AsString**) **AsBoolean**

En la figura 3.8 se muestra la prueba unitaria del numero de cédula, este método recibe como parámetro un número de cédula y regresa true en caso que ya exista un número de cédula asignado a un cliente y se utiliza para validar que no existan dos clientes con el mismo número de cédula.

Definición de la prueba: el número de cédula 1242503790001T ya existe en la base de datos, por tanto al pasárselo como parámetro al método el resultado que se espera es true.

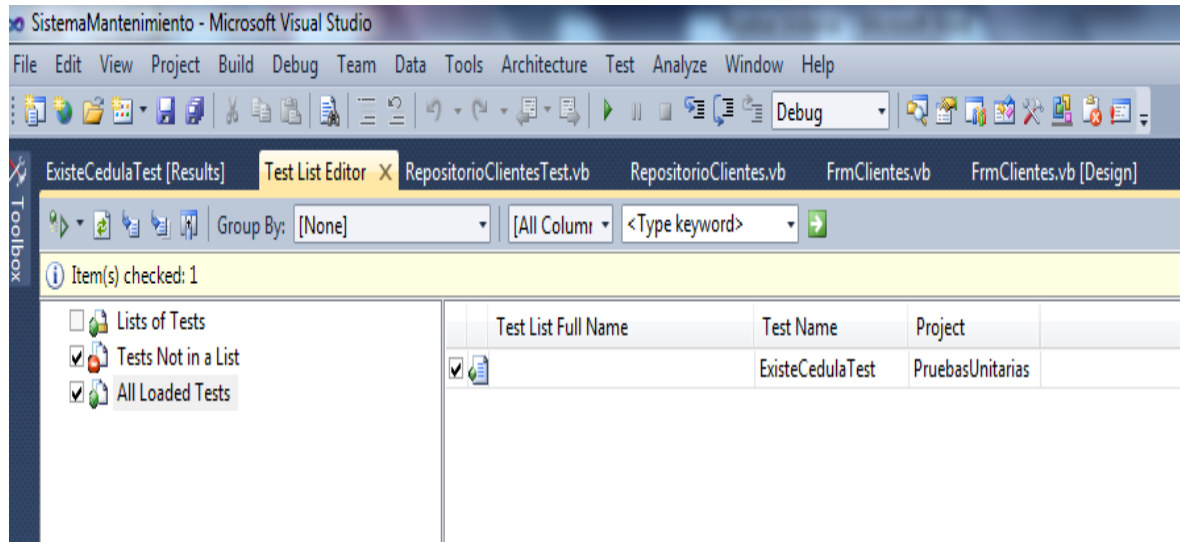
```
'''<summary>
'''A test for ExisteCedula
'''</summary>
<TestMethod()> _
Public Sub ExisteCedulaTest()
    Dim cedula As String = "1242503790001T" ' TODO: Initialize to an appropriate value
    Dim expected As Boolean = True ' TODO: Initialize to an appropriate value
    Dim actual As Boolean
    actual = RepositorioClientes.ExisteCedula(cedula)
    Assert.AreEqual(expected, actual)

End Sub
End Class
```

**Fig. 3.8 Prueba Unitaria Numero de cedula**

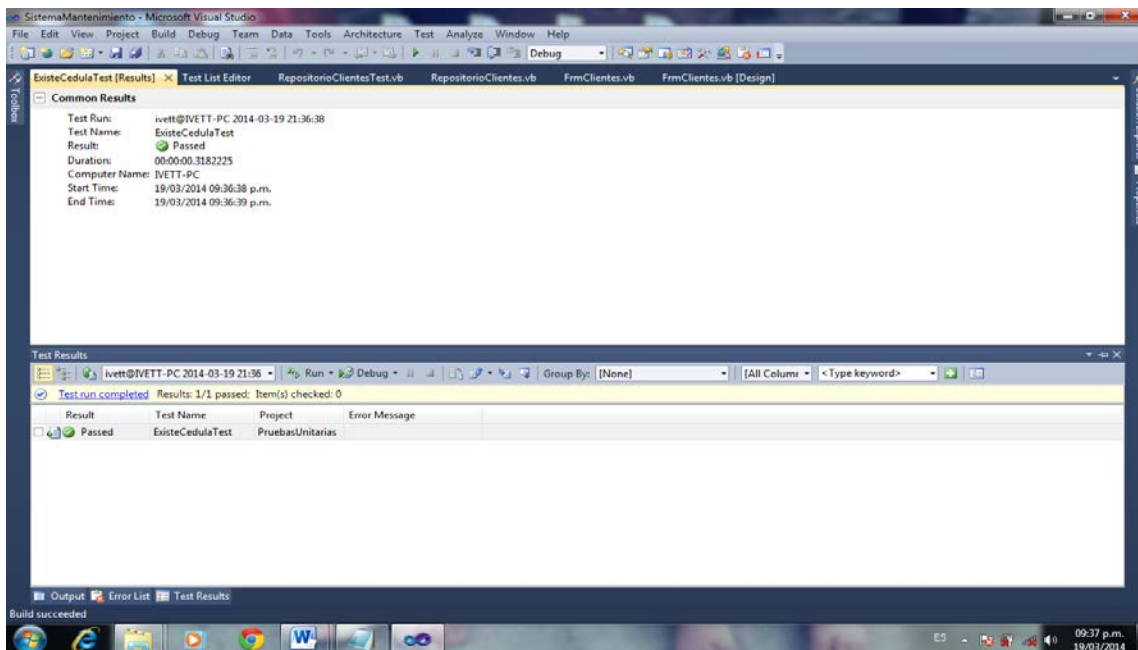


En la figura 3.9 se muestra el resultado de la prueba unitarias que al ser ejecutada el resultado es true.



**Fig. 3.9 resultado de la prueba unitaria Numero de cedula.**

En la figura 4.1 se muestra el detalle del resultado de la prueba unitaria, numero de cédula.



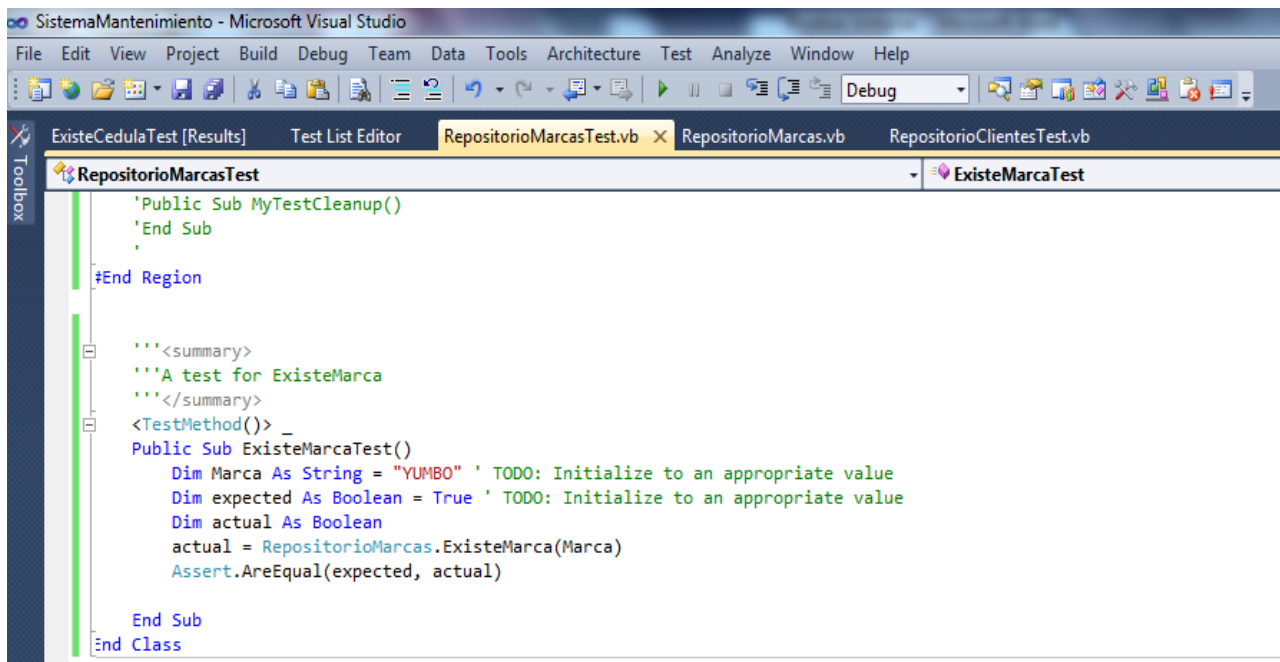
**Fig. 4.1 Detalle de resultado de la prueba unitaria número de cédula**



## 2. `PublicSharedFunction` ExisteMarca (Marca `AsString`) `AsBoolean`

En la figura 4.2 se muestra la ejecución de la prueba unitaria para la clase marca, este método recibe como parámetro un nombre de marca y regresa true en caso que ya exista el nombre de marca que se esté intentando guardar.

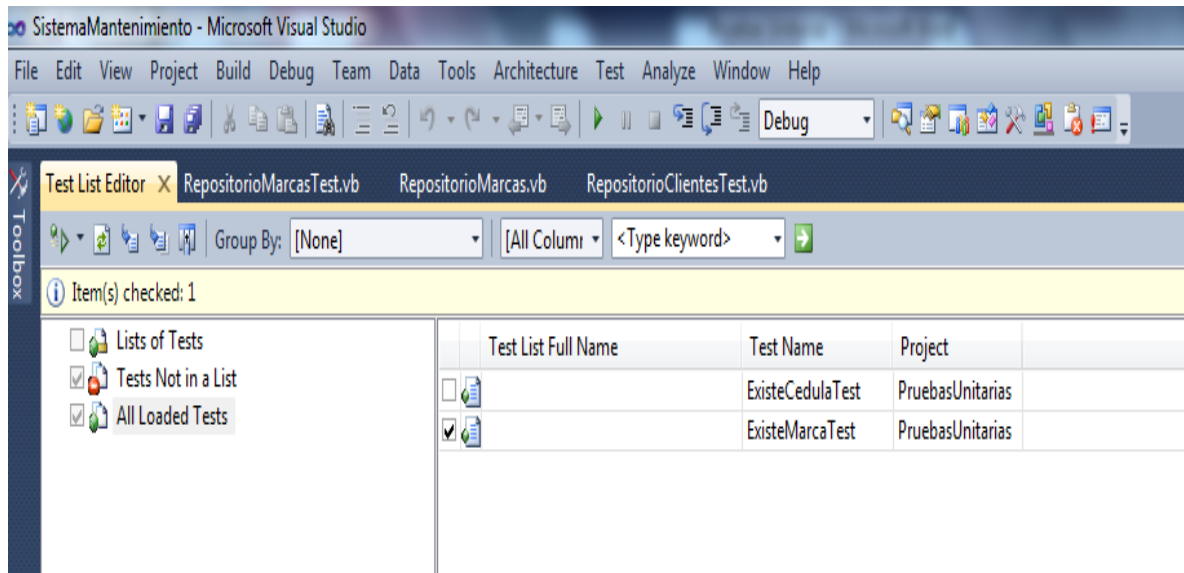
Definición de la prueba: la marca yumbo ya existe en la base de datos, por tanto al pasárselo como parámetro al método el resultado que se espera es true.



```
SistemaMantenimiento - Microsoft Visual Studio
File Edit View Project Build Debug Team Data Tools Architecture Test Analyze Window Help
ExisteCedulaTest [Results] Test List Editor RepositorioMarcasTest.vb X RepositorioMarcas.vb RepositorioClientesTest.vb
RepositorioMarcasTest
  ExisteMarcaTest
    Public Sub MyTestCleanup()
    End Sub
  #End Region
  <summary>
  A test for ExisteMarca
  </summary>
  <TestMethod() _
  Public Sub ExisteMarcaTest()
    Dim Marca As String = "YUMBO" ' TODO: Initialize to an appropriate value
    Dim expected As Boolean = True ' TODO: Initialize to an appropriate value
    Dim actual As Boolean
    actual = RepositorioMarcas.ExisteMarca(Marca)
    Assert.AreEqual(expected, actual)
  End Sub
End Class
```

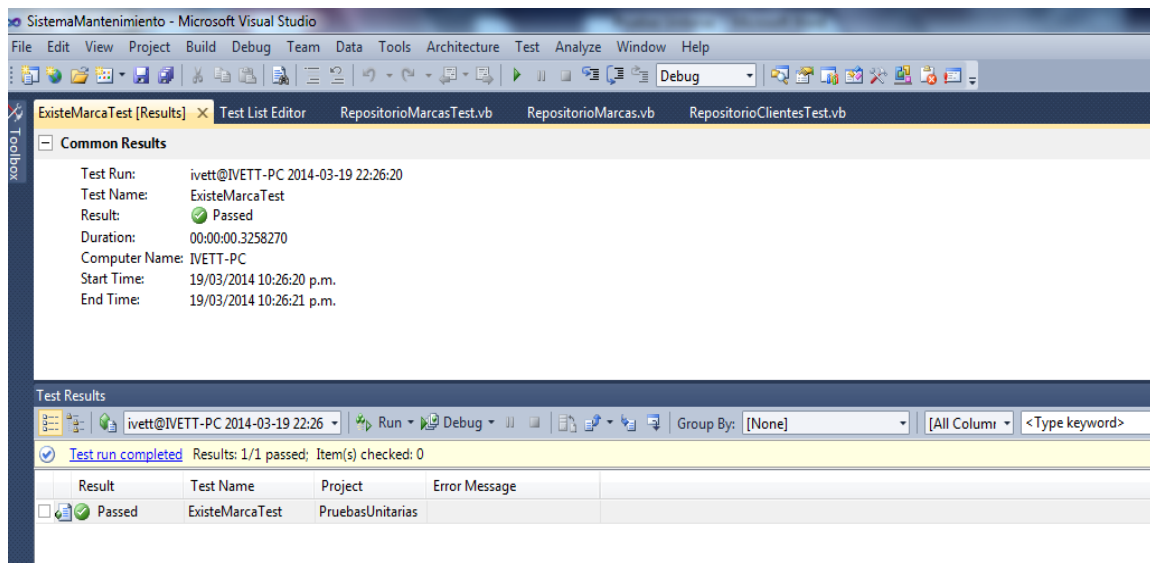
**Fig. 4.2 Prueba unitaria clase marca**

En la figura 3.8 se muestra el resultado de la prueba unitaria de la clase marca, y el resultado es true.



**Fig 4.3 resultado de la prueba unitarias clase marca**

En la figura 4.4 se muestra el detalle del resultado de la prueba unitaria de la clase marca.



**Fig. 4.4 Detalle de resultado de la Clase marca.**





## Plan de recolección de información

En el presente cuadro se explica las técnicas, los instrumentos y los actores quienes serán objetos de estudio y servirán para la recolección de datos en el presente trabajo:

<b>Técnica</b>	<b>Instrumento</b>	<b>Actores</b>
Observación	- Cuaderno de notas  - Estructuración de una ficha  De información	- Actores relacionados con los procesos de trabajo del taller
Encuesta	- Cuestionario	- Actores relacionados con los procesos de trabajo del taller

## Procesamiento y análisis de la información

El procedimiento para procesar y analizar la información se lo realizara de la siguiente manera:

- Revisión crítica de la información recogida, es decir limpieza y depuración de información defectuosa, incomprensible, contradictoria, incompleta, no pertinente, etc.
- Tabulación o representación gráfica de los datos y resultados según sea el caso.



- Manejo de información debe ser de forma clara y precisa para que no sea un obstáculo para el desarrollo del estudio.

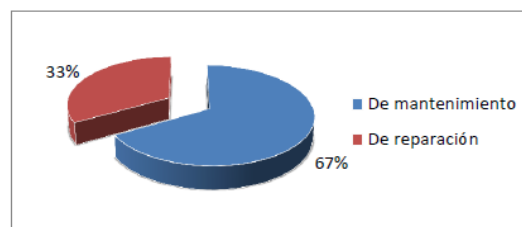
### **Encuesta:**

(Anexo 1)

**Pregunta 1.** ¿Qué tipos de servicio son los más frecuentes que Ud. realiza en el taller “MORIAH”?

<b>Respuesta</b>	<b>No. Personas</b>	<b>Porcentaje</b>
De mantenimiento	6	67 %
De reparación	3	33%
Total	9	100%

**Fuente: Investigador**



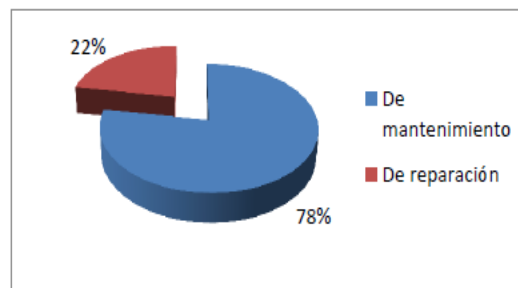
**Gráfico No. 1** Estadística grafica- Pregunta No 1.

**Pregunta 2.** ¿De los tipos de servicio antes mencionados, cuál es el que más problemas le ha presentado durante su proceso?



Respuesta	No. Personas	Porcentaje
De mantenimiento	7	78 %
De reparación	2	22%
Total	9	100%

**Fuente: Investigador**



**Grafico No. 2** Estadística grafica- Pregunta No 2.

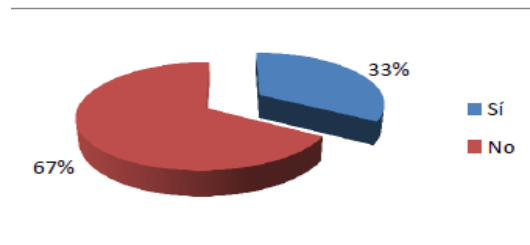
**Pregunta 3.** ¿El taller se encuentra debidamente equipado para realizar sin problemas todos los servicios mantenimiento en una motocicleta?

Respuesta	No. Personas	Porcentaje
Si	3	33%
No	6	67%
Total	9	100%

**Fuente: Investigador**



**Grafica No. 3**

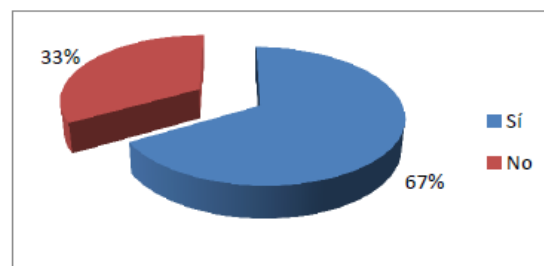


Estadística grafica – Pregunta No. 3

**Pregunta 4.** ¿Cree que los procesos empleados en el mantenimiento de motocicletas, son determinantes para la aceptación de los clientes?

Respuesta	No. Personas	Porcentaje
Si	6	67%
No	3	33%
Total	9	100%

**Fuente: Investigador**



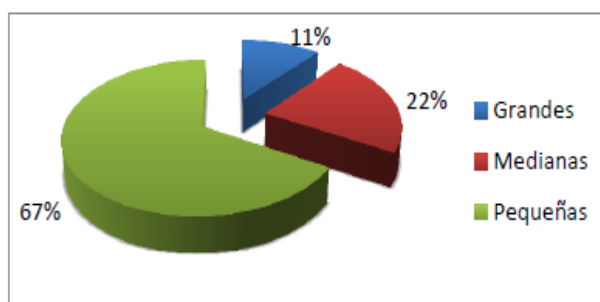
**Grafico No 4** Estadística Gráfica – Pregunta No 4.

**Pregunta 5.** ¿De forma general cual es el tipo de motocicletas con la que ha tenido problemas durante la realización de su trabajo?



Respuesta	No. Personas	Porcentaje
Grandes	1	11 %
Medianas	2	22 %
Pequeñas	6	67%
Total	9	100%

**Fuente: Investigador**



**Gráfico No 5** Estadística grafica – Pregunta No 5.

### **INTERPRETACIÓN DE LOS RESULTADOS:**

#### **RESULTADOS DE LA ENCUESTA**

**Pregunta 1:** En los talleres encuestados se puede verificar un mayor porcentaje de trabajos de mantenimiento en motocicletas, siendo los que mayor frecuencia se los realiza.

**Pregunta 2.:** El personal involucrado dentro de las actividades de mantenimiento y reparación de los talleres, asegura tener mayores dificultades de realizar sus tareas en la actividad de mantenimiento por ser la que mayor frecuencia se presenta.



**Pregunta 3:** El personal encuestado en los talleres considera que no siempre un taller puede estar equipado en un 100%, y que en todo momento se ve la necesidad de realizar implementaciones o readecuaciones dentro del taller.

**Pregunta 4:** Para que un servicio sea aceptable dentro de un campo competitivo depende de la calidad del servicio que se preste, para los encuestados, la mayoría es consciente que la aceptación de sus servicios depende de la forma como se los realiza, siendo un aspecto a considerar dentro una estabilidad laboral y económica para cualquier sector productivo.

**Pregunta 5:** En esta pregunta se hace visible la dificultad que se hace presente al realizar cualquier tipo de actividad mecánica en motocicletas de características pequeñas, por la complejidad que estas representan al momento de manipular sus elementos constitutivos.



## 4.2 CONCLUSIONES:

Las metodologías para el desarrollo de software imponen un proceso disciplinado sobre el desarrollo de estos con el fin de hacerlo más predecible y eficiente además de hacer más eficaz la producción y lograr alta calidad de una forma costeable.

Después del estudio de esta metodología, queda claro que FDD es una metodología que disminuye el riesgo de los proyectos, pues gracias a sus entregas tangibles y al constante monitoreo de su calidad se asegura el firme avance del mismo, en particular para aquellos en constante cambio y con una necesidad de retroalimentación constante por parte del cliente.

Por otra parte, se trata de una metodología que ayuda al equipo a producir resultados periódicos y tangibles y con un enfoque en aspectos de calidad durante todo el proceso, por lo que al contrario de otras metodologías, FDD afirma ser conveniente para el desarrollo de sistemas críticos.

FDD, permite dejar satisfechos a los desarrolladores, gerentes y clientes sin afectar al proyecto. Esto gracias a un buen manejo de las actividades, a la disminución del riesgo del proyecto y al aseguramiento de la calidad del mismo respectivamente, sin embargo es una metodología que necesita de un grupo de líderes con vasta experiencia en el desarrollo de proyectos ya que dado al poco énfasis en la formulación de los requerimientos puede surgir un modelo de sistema incorrecto que conlleve a una gran cantidad de fallas.



### 4.3 RECOMENDACIONES

Se recomienda a los lectores que trabajan en desarrollo de software y que aún no han tenido la experiencia de emplear metodologías ágiles tener un acercamiento a estos enfoques modernos porque necesitan un proceso que pueda responder de manera eficiente a los cambios en los productos en desarrollo. Las metodologías ágiles permiten una mayor flexibilidad que las metodologías tradicionales de desarrollo, que se bloquean muy pronto en los detalles del proyecto y son menos capaces de ajustarse a las cambiantes necesidades de los accionistas, del mercado y de los desafíos imprevistos que plantea la tecnología.

No existe una metodología universal para hacer frente con éxito a cualquier proyecto de desarrollo de software. Toda metodología debe ser adaptada al contexto del proyecto, exigiendo un esfuerzo considerable para ser adaptadas, sobre todo en proyectos pequeños y con requisitos muy cambiantes. Las metodologías ágiles ofrecen una solución casi a medida para una gran cantidad de proyectos que tienen estas características. Una de las cualidades más destacables en una metodología ágil es su sencillez, tanto en su aprendizaje como en su aplicación, reduciéndose así los costos de implantación en un equipo de desarrollo. Esto ha llevado hacia un interés creciente en las metodologías ágiles.

Al aplicar una metodología ágil no podemos perder de vista ni un momento los doce principios del manifiesto Ágil, ya que son características que diferencian un proceso ágil de uno tradicional.





#### **IV. MATERIALES CONSULTADOS:**

##### **4.1 Bibliografía:**

- 1. SERRANO, Nicolás** (2008) “NEUMÁTICA”, 5ª edición, España.
- 2. Mc, Cormac**, “DISEÑO DE ESTRUCTURAS METÁLICAS”, 4ª Edición.
- 3. TORRES, M. SERAUTO’S** Manual Básico de Mantenimiento Automotriz. Ecuador.  
1996.
- 4. HERRERA E. Luis. MEDINA F. Arnaldo. NARANJO L. Galo.** “TUTORÍA DE LA INVESTIGACIÓN CIENTÍFICA”.
- 5. SHIGLEY, Joseph** “DISEÑO EN INGENIERÍA MECÁNICA”, sexta edición, 2002.
- 6. Canós, José H.** HISTORIA DE LAS METODOLOGÍAS ÁGILES PARA EL DESARROLLO DE SOFTWARE.



## Web grafía:

Mantemiento básico:

(2000)[http://www.empresariorural.com/index.php?option=com\\_content&view](http://www.empresariorural.com/index.php?option=com_content&view)

Ingeniería en soporte (1995)[http://www.guillesime.galeon.com/index\\_archivos/Page736.htm](http://www.guillesime.galeon.com/index_archivos/Page736.htm)

Compresores (1988)[http://www.elprisma.com/apuntes/ingenieria\\_mecanica/compresores/](http://www.elprisma.com/apuntes/ingenieria_mecanica/compresores/)

Tipos de compresores (2002)[http://www.elprisma.com/apuntes/ingenieria\\_mecanica/compresores/](http://www.elprisma.com/apuntes/ingenieria_mecanica/compresores/)

Cilindros (2000)<http://electrotelex.net16.net/guia%20-%20cilindros%20neumaticos.pdf>

aplicaciones de la neumática

(1990)[http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lep/hernandez\\_b\\_ii/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lep/hernandez_b_ii/capitulo2.pdf)

Aplicaciones de la neumática 2

(2001)[http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lep/hernandez\\_b\\_ii/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lep/hernandez_b_ii/capitulo2.pdf)

Operación de mantenimiento básico

(1990)[http://www.gassattack.com/articulos\\_tecnicos/mantenimiento%20basico.pdf](http://www.gassattack.com/articulos_tecnicos/mantenimiento%20basico.pdf)

(2012) consejos para asegurar el éxito. (<http://www.empresariorural.com/index.php?option=com>)

(1990)JorgeE.Pereira.(<http://html.rincondelvago.com/gestion-de-la-produccion-y-los-servicios.html>)).



## V. ANEXOS

### Anexo 1:

Formato de encuesta para la obtención de información.

Universidad Nacional Autónoma de Nicaragua

Facultad de Ciencias e Ingenierías

Encuestadores: Elizabeth Pérez Gutiérrez

Ivette Rodríguez Gadea

### Encuesta:

Nombre del taller .....

Dirección.....

**Objetivo:** La presente encuesta está enfocada al fortalecimiento laboral en el taller de motocicletas, “MORIAH”.

**Marque con una X la opción que usted elija.**

1. ¿Qué tipos de servicio son los más frecuentes que Ud. realiza en el taller?

De mantenimiento (...)

De reparación (...)



2. ¿De los tipos de servicio antes mencionados, cuál es el que más problemas le ha presentado durante su proceso?

De mantenimiento (...)

De reparación (...)

3. ¿El taller se encuentra debidamente equipado para realizar sin problemas todos los servicios mantenimiento en una motocicleta?

SI (...)

NO (...)

4. ¿Cree que los procesos empleados en el mantenimiento de motocicletas, son determinantes para la aceptación de los clientes?

SI (...)

NO (...)

5. ¿De forma general cuál es el tipo de motocicletas con la que ha tenido problemas durante la realización de su trabajo?

Grandes (...)

Medianas (...)

Pequeñas (...)



## Anexo 2

Universidad Nacional Autónoma de Nicaragua



### ACTA DE ENTREGA Y ACEPTACION

|

Referente al acuerdo establecido con el distribuidor autorizado Taller de Mantenimiento y Reparación Moriah de la empresa MASESA S.A relativo al servicio de desarrollo de un sistema de control de gestiones de mantenimiento y reparación, se elabora la presente acta de entrega – aceptación el día 11 de enero del año 2014.

Por medio de la presente y conforme a lo acordado se hace

entrega a la dirección administrativa del taller los siguientes entregables:

- Formulario Clientes
- Formulario Marca
- Formulario Modelo
- Formulario de Usuarios

ENTREGA	RECIBE
FECHA: 11 de Enero 2014	FECHA: 11 de Enero 2014
Entregado: Luis Adolfo Henriquez Desarrollador de Software	Recibido a satisfacción por: Liliette Parrales Gerente Administrativa
Firma: 	Firma 

**Fig. 4.5 Carta entrega-aceptación primera Iteración**



### Anexo 3

## Universidad Nacional Autónoma de Nicaragua



### ACTA DE ENTREGA Y ACEPTACION

Referente al acuerdo establecido con el distribuidor autorizado Taller de Mantenimiento y Reparación Moriah de la empresa MASESA S.A relativo al servicio de desarrollo de un sistema de control de gestiones de mantenimiento y reparación, se elabora la presente acta de entrega – aceptación el día 24 de Marzo del año 2014.

Por medio de la presente y conforme a lo acordado se hace

Entrega a la dirección administrativa del taller los siguientes entregables:

- Formulario de orden de servicio

ENTREGA	RECIBE
FECHA: 24 de Marzo del año 2014	FECHA: 24 de Marzo del año 2014
Entregado: <b>Luis Adolfo Henriquez/Elizabeth Perez/ Ivette Rodriguez</b> Desarrolladores	Recibido a satisfacción por: <b>Liliette Parrales</b> Gerente Administrativa
Firma: 	Firma 

**Fig. 4.6 Carta entrega- aceptación segunda iteración**



## Anexo 4

Universidad Nacional Autónoma de Nicaragua



### ACTA DE ENTREGA Y ACEPTACION

Referente al acuerdo establecido con el distribuidor autorizado Taller de Mantenimiento y Reparación Moriah de la empresa MASESA S.A relativo al servicio de desarrollo de un sistema de control de gestiones de mantenimiento y reparación, se elabora la presente acta de entrega – aceptación el día 12 de Abril del año 2014.

Por medio de la presente y conforme a lo acordado se hace

Entrega a la dirección administrativa del taller los siguientes entregables:

Reportes:

1. Clientes
2. Marca
3. Modelo
4. Orden de servicio

ENTREGA	RECIBE
FECHA: 12 de Abril del año 2014	FECHA: 12 de Abril del año 2014
Entregado: <b>Luis Adolfo Henriquez/Elizabeth Perez/ Ivette Rodriguez</b> Desarrolladores	Recibido a satisfacción por: <b>Liliette Parrales</b> Gerente Administrativa
Firma: 	Firma: 

**Fig. 4.7 carta entrega-aceptación tercera iteración**