

# **ESCUELA POLITÉCNICA NACIONAL**

## **ESCUELA DE FORMACIÓN DE TECNÓLOGOS**

### **SISTEMA WEB PAR EL CONTROL DE PRODUCCIÓN Y TIEMPO PERDIDO EN LA PLANTA DE PINTURA (GM)**

#### **PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN ANÁLISIS DE SISTEMAS INFORMÁTICOS**

**PILACUÁN TEPUD JOSÉ ARMANDO**

armandos8612@hotmail.com

**DIRECTOR: ING. LUZ MARINA VINTIMILLA JARAMILLO**

marina.vintimilla@epn.edu.ec

**Quito, Abril, 2014**

## DECLARACIÓN

Yo, José Armando Pilacuán Tepud, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

---

José Armando Pilacuán Tepud

## CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por José Armando Pilacúan Tepud, bajo mi supervisión.

---

**Ing. Luz Marina Vintimilla**  
**DIRECTOR DE PROYECTO**

## **AGRADECIMIENTO**

Primero agradezco a Dios por ser mi guía y brindarme la sabiduría y actitud para alcanzar uno de los objetivos propuesto en mi vida.

Agradezco a mis Padres Victoriano y Germania quienes son el pilar fundamental en mi vida, seres que me brindaron su apoyo incondicionalmente y siempre están para guiarme por un buen camino, lo más importante me enseñaron el verdadero sentido y valor de la vida. Me enseñaron que todo se consigue con perseverancia y sacrificio.

Agradezco a mis hermanos Adriana, Fernando y Maricela por siempre brindarme su apoyo.

Un agradecimiento especial a mi Directora del proyecto Ing. Luz Marina Vintimilla por todo el apoyo y la confianza depositada en mí. Quien es una gran maestra y persona.

Agradezco a mis amigos con quienes compartimos aulas y experiencias, pero sobre todo una gran amistad.

Finalmente agradezco a la Escuela Politécnica Nacional por brindarme la oportunidad de culminar mis estudios en tan prestigiosa Institución.

Armando.

## DEDICATORIA

Este trabajo está dedicado a las personas más importantes en mi vida Mis Padres y mis hermanos, que con sus consejos me ayudaron a siempre seguir adelante.

A mis padres quienes siempre me brindan su cariño y comprensión, personas que siempre me apoyan en todo momento con sus sabios consejos y experiencias vividas.

A mi Padre una persona excelente, que con sus consejos me ha sabido guiar por el camino correcto y sobre todo con su ejemplo me ha demostrado que todo en la vida se consigue con esfuerzo y que nada es imposible.

A mi Madre un persona maravillosa, llena de virtudes quien me enseñó a valorar a las personas que me brindan su apoyo, me enseñó que la vida está llena de retos pero también de logros y que con perseverancia se llega al objetivo propuesto.

A mis hermanos quienes siempre me brindaron su apoyo y sus consejos.

## TABLA DE CONTENIDO

<b>CAPÍTULO 1. ANTECEDENTES</b> .....	<b>8</b>
1.1 ÁMBITO DEL PROBLEMA.....	8
1.2 IDENTIFICACIÓN DEL PROBLEMA.....	9
1.3 FORMULACIÓN Y SISTEMATIZACIÓN DEL PROBLEMA .....	11
1.3.1 FORMULACIÓN .....	11
1.3.2 SITEMATIZACIÓN .....	11
1.4 OBJETIVOS .....	11
1.4.1 OBJETIVO GENERAL .....	11
1.4.2 OBJETIVOS ESPECÍFICOS .....	11
1.5 JUSTIFICACIÓN DEL PROYECTO .....	12
1.6 ANÁLISIS DE COSTOS DEL SISTEMA .....	12
<b>CAPÍTULO 2. ASPECTOS TEÓRICOS</b> .....	<b>14</b>
2.1 INGENIERÍA DE SOFTWARE .....	14
2.1.1 OBJETIVOS DE LA INGENIERÍA DE SOFTWARE .....	14
2.1.2 PROCESO DE DESARROLLO DE SOFTWARE.....	15
2.2 APLICACIONES WEB.....	15
2.2.1 ESTRUCTURA DE UNA APLICACIÓN WEB.....	15
2.3 DESARROLLO DE APLICACIONES WEB .....	16
2.3.1 ARQUITECTURA WEB.....	16
2.3.2 SERVIDOR WEB.....	19
2.3.3 NAVEGADOR WEB.....	19
2.3.4 APLICACIONES MULTINIVEL .....	20
2.3.5 DISEÑO WEB .....	22
2.4 HERRAMIENTAS DE DESARROLLO .....	23
2.4.1 HERRAMIENTA DE TIPO FRONT-END .....	24
2.4.2 HERRAMIENTA DE TIPO BACK-END .....	27
2.4.3 PHP .....	32
2.4.4 HTML5 .....	36
2.4.5 JAVASCRIPT.....	37
2.4.6 CSS.....	38
<b>CAPÍTULO 3. METODOLOGÍA RUP</b> .....	<b>39</b>
3.1 PARADIGMA EN ESPIRAL.....	39
3.1.1 VENTAJAS.....	39
3.1.2 DEVENTAJAS .....	39
3.2 METODOLOGÍA RUP .....	40
3.2.1 RUP COMO PROCESO DE DESARROLLO.....	40
3.3 FASES DE DESARROLLO DEL RUP.....	40
3.3.1 MODELADO DEL NEGOCIO .....	41
3.3.2 REQUISITOS .....	41
3.4 FASE DE ELABORACIÓN .....	41
3.4.1 ANÁLISIS Y DISEÑO .....	42
3.5 FASE DE CONTRUCCIÓN .....	42
3.5.1 IMPLEMENTACIÓN.....	42
3.5.2 PRUEBAS.....	42
3.5.3 ETAPA DE TRANSICIÓN .....	43
3.5.4 DESPLIEGUE.....	43
3.5.5 VENTAJAS.....	44
3.6 UML.....	44
3.6.1 PRINCIPALES BENEFICIOS DEL UML .....	44
3.6.2 MODELOS .....	45
3.7 CONCLUSIONES Y RECOMENDACIONES .....	62
3.7.1 CONCLUSIONES .....	62
3.7.2 RECOMENDACIONES .....	62

BIBLIOGRAFÍA .....	63
MANUAL TÉCNICO .....	67
3.8 ANÁLISIS DE REQUERIMIENTOS.....	67
3.8.1 IDENTIFICACIÓN DE ACTORES .....	67
3.9 DISEÑO .....	67
3.9.1 DIAGRAMA ADMINISTRADOR DEL SISTEMA.....	67
3.9.2 DIAGRAMA DEI USUARIO OPERADOR .....	68
3.9.3 DIAGRAMA DEI USUARIO INVITADO .....	68
3.9.4 DICCIONARIO DE CASOS DE USO .....	69
3.9.5 MÓDELO FÍSICO .....	72
3.9.6 MÓDELO LÓGICO.....	73
3.9.7 DIAGRAMA DE CLASES.....	74
3.10 DIAGRAMA DE SECUENCIAS.....	75
3.10.1 DIAGRAMA ADMINISTRADOR DEL SISTEMA.....	75
3.10.2 DIAGRAMA DE USUARIO OPERADOR .....	76
3.11 DIAGRAMA DE ACTIVIDADES .....	77
3.11.1 DIAGRAMA DE REGISTRO DE DOWNTIME.....	77
3.11.2 DIAGRAMA DE REGISTRO DE PRODUCCIÓN.....	77
3.11.3 DIAGRAMA DE REGISTRO DE UNIDADES REZAGADAS .....	78
3.12 COSTRUCCIÓN .....	78
3.12.1 BACK END .....	78
3.13 PRUEBAS.....	82
3.13.1 ESTRATEGIA DE PRUEBAS.....	82
MANUAL DE USUARIO .....	88

## **CAPÍTULO 1. ANTECEDENTES**

General Motors OBB del Ecuador es una compañía ecuatoriana que contribuye industrial y socialmente al desarrollo del país ofreciendo soluciones de movilidad y transporte con alta satisfacción a los clientes de la marca y en armonía con el ambiente y los grupos de interés externos e internos.

Su visión es diseñar construir y vender los mejores vehículos del Mundo.

### **1.1 ÁMBITO DEL PROBLEMA**

La empresa General Motors se encuentra ubicada en la Av. Galo Plaza Lasso #OE1-134 y Enrique Guerrero Portilla (Quito - Ecuador) Norte de la ciudad de Quito. La empresa desde su inicio ha tenido gran acogida por el crecimiento del parque automotor y por la necesidad de ensamblar autos con manos ecuatorianas.

La empresa General Motors posee una Planta interna de Pintura la cual se encarga del proceso del pintado de autos. El mismo que se inicia con la recepción de autos desde el área de ensamble y siguen los siguientes pasos:

1. Elevador Primer.- Se realiza una revisión minuciosa del mismo de la unidad lo que ayuda a determinar si es apto para iniciar el proceso de pintado, de lo contrario se lo marca como defectuoso y se lo aparta de la línea de producción hasta corregir el error.
2. Primer punto de control (Elpo).- Se realiza un tratamiento de limpieza de toda la unidad y se impregna la primera capa de pintura la cual se la conoce como curación de la unidad.
3. Segundo punto de control (Primer Status 30).- Se realiza un proceso de secado para luego pasar a cubrir las aberturas entre el metal, con una mezcla especial este proceso se lo conoce como Calafateo.



4. Tercero punto de control (Esmalte o Status 40).- Se realiza la recepción de la carrocería para ingresar a las cabinas de pintura donde se realiza el proceso de pintado. Para luego pasar a un minucioso control de calidad el cual se lo realiza con artefactos electrónicos que miden la densidad de la pintura.
5. Cuarto punto de Control (Finesse o Status 50).- Se realiza el control de calidad y se corrigen defectos pequeños.
6. Quinto punto de control (Status 60).- Luego de pasar los controles de calidad se adhiere a la carrocería el código de identificación (PVI).

La planta de pintura procesa diariamente 250 autos aproximadamente, para lo cual existen 3 turnos los cuales se detallan así:

- 1 Turno de 7:05:00 am - 15:00:00 pm.
- 2 Turno de 15:05:00 pm - 23:00:00 pm.
- 3 Turno de 23:00:00 pm - 7:00:00 am.

El proceso se lo realiza de lunes a viernes en los horarios descritos anteriormente.

## **1.2 IDENTIFICACIÓN DEL PROBLEMA**

Actualmente en la planta de pintura se hace un registro de producción y tiempo perdido (Downtime) en cinco puntos de control (Área o status) y en tres turnos de trabajo (1 Turno, 2 Turno, 3 Turno) en los cuales se tiene la necesidad de automatizar este proceso de registro de información. Debido a que se lo realiza de una forma manual así:

1. Una persona de cada área al inicio del turno retira un libro de producción para registrar la información durante el turno de trabajo, el cual fue diseñado tomando en cuenta que el manejo de la información es similar con diferencias tales como:
  - ATT.- Tiempo al que corre la línea de producción.
  - JPH.- Unidades reales por hora producidas. Entre otras.

2. En el sitio de trabajo la persona encargada escribe los datos informativos en el libro de producción tales como: fecha, Líder de grupo (Quien es responsable del grupo de trabajo), responsable (Es la persona quien realiza el registro de información, en el turno de trabajo), ausentismo (se verifica si las personas que conforman el equipo están presentes de lo contrario se realiza el registro).
3. El registro de información (Downtime, Producción y unidades perdidas) se realiza al final de cada hora lo determina si se cumplió con el objetivo propuesto. Si el objetivo fue cumplido quiere decir que en la planta de pintura no tuvo ningún problema, de lo contrario se deberá hacer las respectivas anotaciones detallando la siguiente información:
  - CAUSA / DESCRIPCIÓN.- Se describe el problema que genero pérdida de tiempo.
  - LUGAR FALLA.- Describe el lugar exacto donde se produjo el inconveniente detallando si fue por falla humana o mecánica.
  - RESPONSABLE.- Se determina los responsables de la pérdida de tiempo, pueden ser equipos de trabajo (Personas), proveedores o maquinaria mecánica.

En algunos casos la persona encargada de realizar el registro no conoce muy bien el funcionamiento de la planta de pintura, por lo que no se podrá llevar un buen control de problemas que se tienen en la línea de producción. Por lo que se toma la decisión de implementar un sistema web donde se pueda mantener un estándar de información y posibles problemas.

4. Al final del turno se realiza la verificación y sumatoria de unidades producidas, el resultado obtenido se compara con el objetivo diario. El siguiente paso es realizar la sumatoria de tiempo perdido con lo que se justificara la diferencia de unidades.
5. Al final de cada turno la persona encargada realiza la entrega del el libro de control la controlador de producción quien ingresa de información a una base de datos diseñada en Access 2007 en el año 2005, la misma que se encuentra actualizada. Sobre esta base se han realizado modificaciones a

los datos para homologar con los puntos de control de la planta, lo que genera inconvenientes al momento de obtener los reportes de producción y productividad.

6. El controlador de producción obtiene reportes los mismos que son enviados para la validación de objetivos e indicadores de productividad. Si los resultados son negativos se deberá realizar un plan de seguimiento lo que ayudara a mejorar la producción y evitar el downtime.

## **1.3 FORMULACIÓN Y SISTEMATIZACIÓN DEL PROBLEMA**

### **1.3.1 FORMULACIÓN**

¿Es posible que mediante la implementación del Sistema Control de Producción se pueda apoyar a solucionar los inconvenientes, optimizando recursos, tiempo, dinero?

### **1.3.2 SISTEMATIZACIÓN**

¿Qué módulos se requieren para la solución del proyecto?

¿Qué herramientas se utilizarán para el desarrollo y manejo de datos?

¿Qué información nos brindaría el sistema?

¿Qué tipo de reportes se obtendría?

¿Qué beneficio nos brindaría el sistema?

## **1.4 OBJETIVOS**

### **1.4.1 OBJETIVO GENERAL**

Desarrollar un sistema web para el control de producción y tiempo perdido en la Planta Pintura (GM).

### **1.4.2 OBJETIVOS ESPECÍFICOS**

- Obtener la información de las necesidades de cada una de las áreas.
- Analizar la forma de registro de datos en los libros de producción
- Analizar los diferentes partícipes en cada área.

- Plantear una solución precisa para los requerimientos establecidos.
- Desarrollar los módulos requeridos para el proyecto.
- Integrar el sistema y datos.

## 1.5 JUSTIFICACIÓN DEL PROYECTO

El proyecto plantea facilitar la administración de la información generada por los problemas en la planta de pintura GM, así como también permitirá realizar una búsqueda rápida en los inconvenientes suscitados dentro de la planta.

Con los reportes obtenidos del sistema se podrá evaluar el rendimiento de la planta de pintura GM y observar el desempeño interno de cada una de las áreas. Con la información almacenada es posible hacer un seguimiento a los problemas de mayor frecuencia, así como también podría tomar de decisiones en tiempo real.

## 1.6 ANÁLISIS DE COSTOS DEL SISTEMA

Recursos físicos y Software.

<b>Costos de Hardware</b>	
<b>Detalle</b>	<b>Costo</b>
Computador Core i3	\$800
Impresora Lexmark	\$90

z600	
Total	\$930

<b>Costo de Software</b>	
<b>Detalle</b>	<b>Costo</b>
Paquete DreamWeaver CS5	\$100
PHP y MySQL (Libre)	-----
Total	\$100

<b>Costo Indirecto</b>	
Suministros de oficina	\$30
Agua, Luz, Teléfono	\$60
Costo Total	\$90

<b>Recurso Humano</b>				
<b>Detalle</b>	<b>(2 personas)</b>	<b>Costo</b>	<b>Tiempo en horas</b>	<b>Total</b>
Analista *		290	6	\$1740
Programador *		290	6	\$1740
				\$3480

El costo total es de \$4600

Este costo asumirá General Motors

(\*).- Personas necesarias para la estudio y desarrollo del sistema.

## CAPÍTULO 2. ASPECTOS TEÓRICOS

### 2.1 INGENIERÍA DE SOFTWARE

La Ingeniería de Software es la rama de la ingeniería que aplica los principios de la ciencia de la computación y las matemáticas para lograr soluciones costo-efectivas (eficaces en costo o económicas) a los problemas de desarrollo de software, permite elaborar consistentemente productos correctos, utilizables y costo-efectivos<sup>1</sup>.

Según la definición del IEEE, software es la suma total de los programas de computadora, procedimientos, reglas, la documentación asociada y los datos que pertenecen a un sistema de cómputo. Un producto de software es un producto diseñado para un usuario<sup>2</sup>.

#### 2.1.1 OBJETIVOS DE LA INGENIERÍA DE SOFTWARE

En la construcción y desarrollo de proyectos se aplican métodos y técnicas para resolver los problemas, la informática aporta herramientas y procedimientos sobre los que se apoya la ingeniería de software.

- Mejorar la calidad de los productos de software
- Aumentar la productividad y trabajo de los ingenieros del software.
- Facilitar el control del proceso de desarrollo de software.
- Suministrar a los desarrolladores las bases para construir software de alta calidad en una forma eficiente.
- Definir una disciplina que garantice la producción.
- Mantenimiento de los productos software desarrollados en el plazo fijado y dentro del costo estimado<sup>3</sup>.

---

<sup>1</sup> Cota A. 1994 "Ingeniería de Software". Soluciones Avanzadas. Julio de 1994. pp. 5-13.

<sup>2</sup> Lewis G. 1994. "What is Software Engineering?" DataPro (4015). Feb 1994. pp. 1-10.

<sup>3</sup> <http://www.monografias.com/trabajos5/inso/inso.shtml#estra>

### **2.1.2 PROCESO DE DESARROLLO DE SOFTWARE**

Es un conjunto de conceptos, una metodología y un lenguaje propio, se le llama el ciclo de vida del software que comprende cuatro grandes fases:

- La concepción define el alcance del proyecto y desarrolla un caso de negocio.
- La elaboración define un plan del proyecto, especifica las características y fundamenta la arquitectura.
- La construcción crea el producto.
- La transición transfiere el producto a los usuarios.

El proceso de desarrollo de software es aquel en que las necesidades del usuario son traducidas en requerimientos de software, estos requerimientos transformados en diseño y el diseño implementado en código, el código es probado, documentado y certificado para su uso operativo<sup>4</sup>.

## **2.2 APLICACIONES WEB**

Una aplicación web es un conjunto de páginas que interactúan unas con otras y con diversos recursos en un servidor web, incluidas bases de datos. Esta interacción permite implementar características en su sitio como catálogos de productos virtuales y administradores de noticias y contenidos. Adicionalmente podrá realizar consultas a bases de datos, registrar e ingresar información, solicitudes, pedidos y múltiples tipos de información en línea en tiempo real<sup>5</sup>.

### **2.2.1 ESTRUCTURA DE UNA APLICACIÓN WEB**

Una aplicación web esta normalmente estructurada como una aplicación de tres-capas. El navegador web ofrece la primera capa y un motor capaz de usar alguna

---

<sup>4</sup> Jacobson, I. 1998. "Applying UML in The Unified Process" Presentación. Rational Software. Presentación disponible en <http://www.rational.com/uml> como UMLconf.zip

<sup>5</sup> [http://www.suronline.net/nuevo\\_sitio/beneficios-funcionamiento-aplicaciones-web.asp](http://www.suronline.net/nuevo_sitio/beneficios-funcionamiento-aplicaciones-web.asp)

tecnología web dinámica por ejemplo PHP, constituye la capa intermedia. Por último una base de datos constituye la tercera y última capa.

El navegador web manda peticiones a la capa media que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario<sup>6</sup>.

### **2.3 DESARROLLO DE APLICACIONES WEB**

El desarrollo de aplicaciones web consiste en implementar necesidades, objetivos o ideas en Internet utilizando las nuevas tecnologías, las mismas que pueden ser accedidas en portales de internet o intranets.

#### **2.3.1 ARQUITECTURA WEB**

Es una típica arquitectura cliente/servidor, en el cual de un lado se encuentra el cliente que está compuesto de browsers web, capaces de mostrar y solicitar documentos sobre una red. Opcionalmente, el cliente puede estar acompañado por aplicaciones externas usando una presentación del documento, o parte de este. El otro lado de la arquitectura web es un servidor, cuya función es atender los pedidos del cliente web por documentos almacenados en el sistema de archivos de la plataforma donde se encuentra instalado.

Una característica importante de la web es que fue proyectada para funcionar en topología de Internet, está compuesta por una gran variedad de computadoras que interactúan entre sí.

Internet tiene un espacio global y se comunica por canales públicos de comunicación sin restricción cuando es continuo. De esta forma la web incorpora naturalmente la característica de ser un ambiente distribuido y multiplataforma. La web es que está basada en tres patrones abiertos que son:

---

<sup>6</sup> <http://unidad6aplicacionesweb1.blogspot.com/2012/10/estructura-de-las-aplicaciones-web.html>



**Transferir información:** para desarrollar el transporte de información entre el servidor y el cliente web fue propuesto un protocolo de información denominado http, cuya principal característica es ser un protocolo abierto y especializado en la transmisión de documentos web sobre Internet. Este protocolo proporciona la eficiencia y velocidad que necesita el sistema de información distribuido en la web.

**Descripción de la información:** para la prestación de documentos en la web se utiliza el lenguaje HTML (HiperText Markup Language) que permite estructurar los documentos web, usando vínculos hacia otros documentos y recursos de Internet. Los documentos pueden ser interpretados por browsers y estructurados según las características de cada plataforma en que son exhibidas. El lenguaje html está compuesto por un conjunto de elementos denominados tags que permiten al usuario especificar la estructura de un documento web. Este lenguaje se basa en el lenguaje SGML (Standard Generalized Markup Language) el cual es independiente de la plataforma en la que es mostrado, lo que posibilita la creación de formularios que contengan iconos y campos para envío de datos por los usuarios, esto proporciona un mayor dinamismo, pues permite que las páginas web funcionen para aplicaciones que necesiten interacción con el usuario.

**Localización de información:** para la identificación y localización de documentos web distribuidos por Internet se utiliza el formato URL (Uniform Resource Locator) cuya sintaxis es la siguiente: (<Protocolo>://<Host>/<Path>/<Doc>Localizacion>). Donde el protocolo indica el tipo de recurso de Internet que debe ser usado para la conexión con el servidor, que puede ser entre otros HTTP, FTP, NEWS, WAIS y FILE.

- **El host.-** Es el nombre de la maquina a la cual desea conectarse.
- **El path.-** Es la lista de directorios separados por barras.
- **Doc.-** Es el nombre del documento o programa a ser ejecutado.
- **Localización.-** Es una marca textual opcional de posición del documento.

**Desarrollo y portabilidad:**

En lo que se refiere al ambiente de desarrollo y portabilidad de aplicaciones web-bases de datos los siguientes puntos merecen ser tenidos en cuenta:

- Las aplicaciones web bases de datos deben ser fácilmente extendidas a las nuevas versiones HTML. Adicionalmente estas deben ser flexibles por si suceden cambios
- Es aconsejable la existencia de mecanismos eficientes para transferir variable de entrada del cliente web para las consultas en el servidor de bases de datos. Este es un punto particularmente importante en virtud del hecho de que la transferencia de datos de un formulario HTML, para el Common Gateway vía Interface CGI (es una interfaz en la que se tiene la posibilidad de ejecutar de aplicaciones externas al servidor web, es una manera más natural de desarrollar aplicaciones en la web, ya que son aplicaciones externas al ambiente cliente servidor).
- La estructura para el desarrollo de aplicaciones web bases de datos debe ser flexibles, poco dependiente del esquema de bases de datos, con un diseño generado para el ambiente web, sin grandes conocimientos de interfaces, como por ejemplo CGI APIs de servidores web o programación de bases de datos.
- La falta de metodología para el desarrollo de aplicaciones es una característica del ambiente web actualmente, por ser una tecnología muy reciente. La metodología aquí es entendida como un procedimiento formal para convertir las necesidades del usuario en código de aplicación, lo que incluye en general, modelo, proyecto, texto y documentación. Particularmente el modelo de una aplicación web bases de datos debe ser mejor estudiada. Los atributos en las bases de datos pueden incorporar características genéricas de formación de entradas y salidas en el formato HTML de forma que pueda facilitar el desarrollo<sup>7</sup>.

---

<sup>7</sup> [http://luisf.wikispot.org/Arquitectura\\_web](http://luisf.wikispot.org/Arquitectura_web), 2007

### **2.3.2 SERVIDOR WEB**

Un servidor web puede ser tanto un ordenador de grandes dimensiones y capacidad como un programa informático que utiliza el protocolo de comunicaciones http para recibir las peticiones de información de un programa cliente (navegador) en el ordenador del usuario.

Una vez interpretada la solicitud que el cliente hace, el servidor web recupera la información solicitada de sus carpetas y la envía al programa cliente para su visualización.

Cuando los servidores web utilizan protocolos de seguridad para la transmisión de datos entre ellos y los clientes, hablamos de servidores web seguros<sup>8</sup>.

### **2.3.3 NAVEGADOR WEB**

Un navegador web es un programa informático que facilita al usuario el navegar por la red, significa el poder visualizar e interactuar las páginas web de cualquier red. Su nombre viene de su capacidad para poder moverse, desplazarse de una página a otra mediante utilidades como los hipervínculos, que conectan las páginas situadas en distintos lugares del mundo.

El navegador, también llamado explorador o browser (en inglés) permite leer la información que se encuentra en servidores (remotos o locales, según el tipo de red), y del mismo modo que un programa de imágenes procesa la información y las muestra, obtener la información necesaria para que las páginas puedan ser mostradas.

---

<sup>8</sup> Agosto 2009, <http://www.consumoteca.com/telecomunicaciones/internet/servidor-web/>

Las páginas web suelen estar codificadas según un lenguaje de programación (en general HTML), que el navegador debe ser capaz de decodificar para interpretarlo. El navegador se encarga de comunicarse con el servidor mediante un protocolo determinado para evitar pérdidas de datos y problemas. Ese protocolo suele ser HTTP, si bien algunas páginas usan protocolos propios o de seguridad<sup>9</sup>.

#### **2.3.4 APLICACIONES MULTINIVEL**

Los sistemas típicos cliente/servidor pertenecen a la categoría de las aplicaciones de dos niveles. La aplicación reside en el cliente mientras que la base de datos se encuentra en el servidor. En este tipo de aplicaciones el peso del cálculo recae en el cliente, mientras que el servidor hace la parte menos pesada, y eso que los clientes suelen ser máquinas menos potentes que los servidores. Además, está el problema de la actualización y el mantenimiento de las aplicaciones, ya que las modificaciones a la misma han de ser trasladada a todos los clientes. Para solucionar estos problemas se ha desarrollado el concepto de arquitecturas de tres niveles: interfaz de presentación, lógica de la aplicación y los datos.

La capa intermedia es el código que el usuario invoca para recuperar los datos deseados. La capa de presentación recibe los datos y los formatea para mostrarlos adecuadamente. Esta división entre la capa de presentación y la de la lógica permite una gran flexibilidad a la hora de construir aplicaciones, ya que se pueden tener múltiples interfaces sin cambiar la lógica de la aplicación. La tercera capa consiste en los datos que gestiona la aplicación. Estos datos pueden ser cualquier fuente de información como una base de datos o documentos XML.

---

<sup>9</sup> <http://www.mastermagazine.info/termino/6055.php>

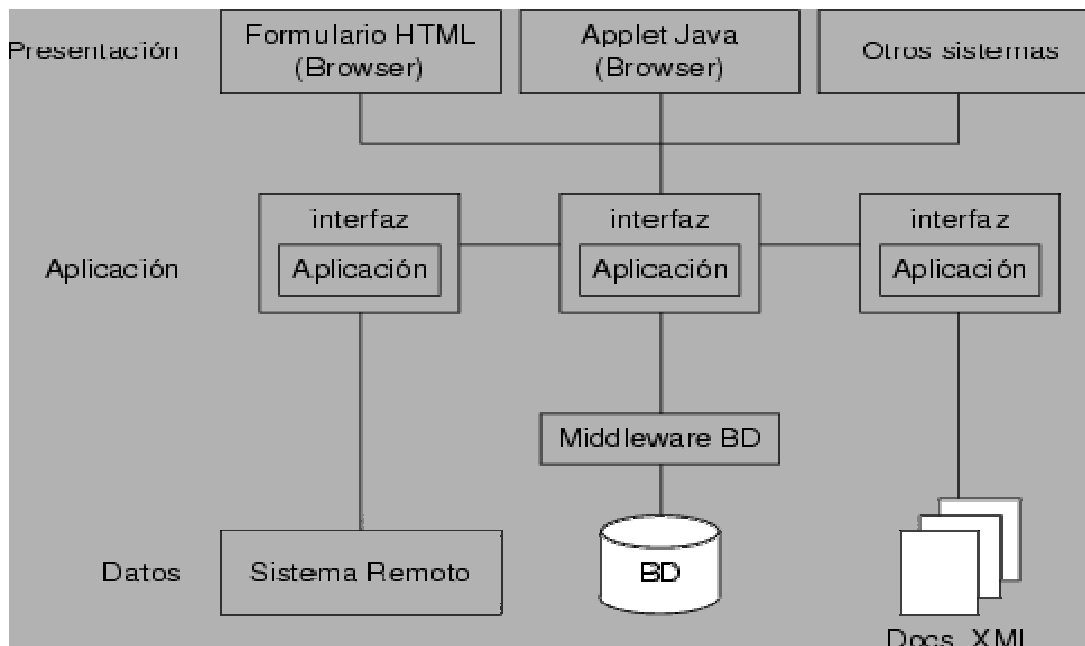


Figura1. Arquitectura Multinivel

Una Web típica consta de tres niveles

**Primer nivel.-** Consiste en la capa de presentación que incluye no sólo el navegador, sino también el servidor web que es el responsable de dar a los datos un formato adecuado.

**Segundo nivel.-** Está referido habitualmente a algún tipo de programa o script.

**Tercer nivel.-** Proporciona al segundo los datos necesarios para su ejecución.

Una aplicación Web típica recogerá datos del usuario (primer nivel), los enviará al servidor, que ejecutará un programa (segundo y tercer nivel) y cuyo resultado será formateado y presentado al usuario en el navegador (primer nivel otra vez)<sup>10</sup>.

---

<sup>10</sup> <http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node21.html>

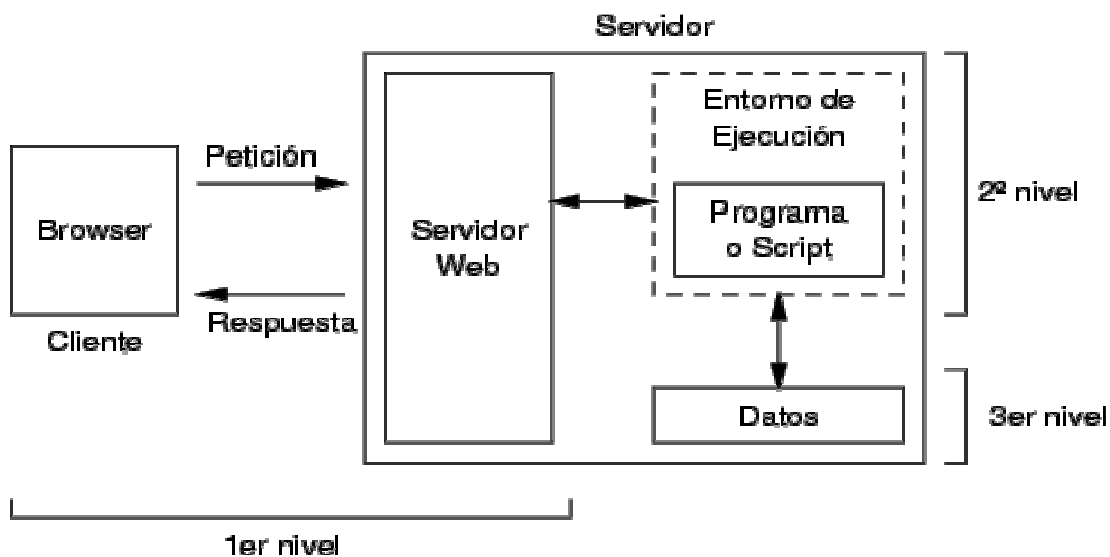


Figura 2. Arquitectura Web de tres niveles

### 2.3.5 DISEÑO WEB

El diseño web es una actividad que consiste en la planificación, diseño e implementación de sitios web. No es simplemente una aplicación de diseño convencional, ya que requiere tener en cuenta la navegabilidad, interactividad, usabilidad, arquitectura de la información y la interacción de medios como el audio, texto, imagen, enlaces y vídeo.

La unión de un buen diseño con una jerarquía bien elaborada de contenidos, aumenta la eficiencia de la web como canal de comunicación e intercambio de datos, que brinda posibilidades como el contacto directo entre el productor y el consumidor de contenidos.

El diseño web ha visto amplia aplicación en los sectores comerciales de Internet especialmente en la World Wide Web. La web se utiliza como medio de expresión, artistas y creadores hacen de las páginas en Internet un medio más para ofrecer sus producciones y utilizarlas como un canal de difusión.

### 2.3.5.1 Etapas del Diseño Web

Para el diseño de páginas web debemos tener en cuenta tres etapas:

- El diseño visual de la información que se desea editar. En esta etapa se trabaja distribuyendo el texto, los gráficos, los vínculos a otros documentos y otros objetos multimedia que se consideren pertinentes. Es importante que antes de trabajar sobre el computador se realice un boceto o prediseño sobre el papel. Esto facilitará tener un orden claro sobre el diseño.
- Estructura y relación jerárquica de las páginas del sitio web. Una vez que se tiene el boceto se pasa a escribir la página web. Para esto se creó el lenguaje de marcación de hipertexto o HTML. La importancia de la estructura y arborescencia web radica en que los visitantes no siempre entran por la página principal o inicial y en ese caso el sitio debe darle la respuesta a lo que busca rápido, además permitirle navegar por el sitio.
- Posicionamiento en buscadores o SEO. Ésta consiste en optimizar la estructura del contenido para mejorar la posición en que aparece la página en determinada búsqueda.

El HTML consta de una serie de elementos que estructuran el texto y son presentados en forma de hipertexto por agente de usuario o navegadores. Esto se puede hacer con un simple editor de textos (debe guardarse como texto plano, sin ningún tipo de formato y con extensión .html o .htm)<sup>11</sup>.

## 2.4 HERRAMIENTAS DE DESARROLLO

En el diseño de software las herramientas utilizadas son de dos tipos el front-end y el back-end la primera (front-end) es la parte del software que interactúa con el o los usuarios, mientras que la segunda (back -end) es la parte que procesa la entrada desde el front-end. La separación del sistema en "front ends" y "back ends" es un tipo de abstracción que ayuda a mantener las diferentes partes del

---

<sup>11</sup> [http://es.wikipedia.org/wiki/Dise%C3%B1o\\_web#Etapas](http://es.wikipedia.org/wiki/Dise%C3%B1o_web#Etapas) Septiembre, 2013

sistema separadas. La idea general es que el front-end sea el responsable de recolectar los datos de entrada del usuario, que pueden ser de muchas y variadas formas, y procesarlas de una manera conforme a la especificación que el back-end pueda usar. La conexión del front-end y el back-end es un tipo de interfaz.

#### **2.4.1 HERRAMIENTA DE TIPO FRONT-END**

Las herramientas de “tipo front-end”, automatizan las primeras actividades del proceso de desarrollo de sistema.

Entre muchos los aspectos que se toman en cuenta las herramientas para esta fase, se hallan las técnicas de soporte para ayudar a validar las descripciones del sistema con el objeto de determinar su consistencia, la evolución de los requerimientos de la aplicación en características que formen parte del sistema que finalmente será implantado.

Las herramientas, de tipo front-end proporcionan soporte para el desarrollo de modelos gráficos de sistemas y procesos. Los diagramas de flujos de datos son representativos de este tipo de herramienta. Los diagramas de flujo de datos representan en forma gráfica los procesos y flujo de datos del sistema, tenemos las siguientes herramientas:

- **Herramientas de consulta:** usan consultas predefinidas y las capacidades de información incorporadas para que los usuarios tenga accesos a los datos.
- **Aplicaciones de usuarios:** Muchos programas de aplicación comunes como Microsoft Excel pueden proporcionar acceso “front-end” a bases de datos de apoyo.
- **Herramientas de desarrollos de programas:** Muchas instalaciones cliente-servidor necesitan aplicaciones “front-end” especiales personalizados para sus tareas de obtención de datos.



#### **2.4.1.1 Adobe Dreamweaver CS5**

Es una aplicación en forma de estudio (basada en la forma de estudio de Adobe Flash) que está destinada a la construcción, diseño y edición de sitios, videos y aplicaciones Web basados en estándares. Creado inicialmente por Macromedia (actualmente producido por Adobe Systems) es el programa más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como Adobe Flash y, recientemente, por su soporte de los estándares del World Wide Web Consortium.

La gran ventaja de este editor sobre otros es su gran poder de ampliación y personalización del mismo, puesto que en este programa, sus rutinas (como la de insertar un hipervínculo, una imagen o añadir un comportamiento) están hechas en Javascript-C, lo que le ofrece una gran flexibilidad en estas materias.

Dreamweaver permite al usuario utilizar la mayoría de los navegadores Web instalados en su ordenador para pre visualizar las páginas web. También dispone de herramientas de administración de sitios. Permite reemplazar líneas de texto y código por cualquier tipo de parámetro especificado, hasta el sitio web completo.

Con la llegada de la versión MX, Macromedia incorporó herramientas de creación de contenido dinámico en Dreamweaver. En lo fundamental de las herramientas HTML WYSIWYG, también permite la conexión a Bases de Datos como MySQL y Microsoft Access, para filtrar y mostrar el contenido.

Un aspecto de alta consideración de Dreamweaver es su funcionalidad permite el uso de "Extensiones". Las extensiones, son pequeños programas, que cualquier desarrollador web puede escribir (normalmente en HTML y Javascript) y que

cualquiera puede descargar e instalar, ofreciendo así funcionalidades añadidas a la aplicación<sup>12</sup>.

#### **2.4.1.2 Ventajas CS5**

- Popularidad.- CS5 y Dreamweaver en general, está muy extendido a través de Internet. Gran cantidad de conocimiento que hay en el programa.
- Gestión Integral.- Dreamweaver no es sólo una herramienta para el desarrollo de una sola página. También se pueden desarrollar sitios web completos, la integración de todos ellos, junto con una variedad de herramientas de integración diferentes.
- Velocidad e Integración.- Es conocido por tener una buena velocidad y la integración web. Trabajar con imágenes en la web es más fácil que en versiones anteriores

CS5 también se integra bien con el resto de la suite de Adobe, por lo que es más fácil ir y venir entre los distintos programas para un mayor control total sobre los diversos procedimientos en el equipo.

#### **2.4.1.3 Desventajas**

- Precio.- El costo es bastante elevado por lo cual esto hace más de un programa diseñado para las grandes empresas.
- Problemas de interfaz.- CS5 sufre de algunos problemas de interfaz que puede hacer que sea difícil de tratar, ya que no es específicamente diseñado desde un entorno Windows o Mac<sup>13</sup>.

---

<sup>12</sup> Julio 2013, [http://es.wikipedia.org/wiki/Adobe\\_Dreamweaver](http://es.wikipedia.org/wiki/Adobe_Dreamweaver)

<sup>13</sup> <http://www.mitecnologico.com/Main/HerramientasFrontEnd>

## 2.4.2 HERRAMIENTA DE TIPO BACK-END

Las herramientas tienen como finalidad ayudar al analista a formular la lógica del programa, los algoritmos de procesamiento y la descripción física de datos, también ayudan a la interacción con los dispositivos (QIS), estas actividades convierten los diseños lógicos del software en código de programación que es el que finalmente da existencia a la aplicación, su empleo está destinado al desarrollo del software, este tipo de también se conoce como herramientas para programación asistida por computador.

MySQL es una herramienta back-end muy utilizada para la descripción física de los datos y uno de los lenguajes más conocidos en el área de implementación es PHP (Hypertext Preprocessor) sencillo, de sintaxis cómoda para el desarrollo de cualquier tipo de aplicación.

En diseño de software el front-end es la parte del software que interactúa con el o los usuarios y el back-end es la parte que procesa la entrada desde el front-end. La separación del sistema en "front ends" y "back ends" es un tipo de abstracción que ayuda a mantener las diferentes partes del sistema separadas. La idea general es que el front-end sea el responsable de recolectar los datos de entrada del usuario, que pueden ser de muchas y variadas formas, y procesarlas de una manera conforme a la especificación que el back-end pueda usar. La conexión del front-end y el back-end es un tipo de interfaz.

En el desarrollo web se hace referencia a la visualización del usuario navegante el mismo que hace uso del front-end, y el administrador del sitio con sus respectivos sistemas que son visualizados en el back-end.

Muchos métodos conocidos de interactuar con computadoras pueden ser conceptualizados en términos de "front-end" y "back-end".

### 2.4.2.1 Apache Web Server

Es un servidor de páginas web HTTP de código abierto para plataformas Unix, Microsoft Windows, Macintosh que en los últimos años se ha convertido en el más popular.

Apache es desarrollado y mantenido por una comunidad abierta de desarrolladores bajo el auspicio de la Apache Software Foundation. Su misión es crítica, ya que es el encargado de aceptar las peticiones de páginas o recursos en general que provienen de los visitantes que acceden al sitio web y gestionar su entrega o denegación, de acuerdo a las políticas de seguridad establecidas. Las principales funcionalidades son:

- Atender de manera eficiente, ya que puede recibir un gran número de peticiones HTTP, incluyendo una ejecución multitarea ya que pueden darse peticiones simultáneas.
- Restricciones de acceso a los ficheros que no se quieran exponer, gestión de autenticaciones de usuarios o filtrado de peticiones según el origen de éstas.
- Manejar los errores por páginas no encontradas, informando al visitante y/o redirigiendo a páginas predeterminadas.
- Gestión de la información a transmitir en función de su formato e informar adecuadamente al navegador que está solicitando dicho recurso.
- Gestión de logs, es decir almacenar las peticiones recibidas, errores que se han producido y en general toda aquella información que puede ser registrada y analizada posteriormente para obtener las estadísticas de acceso al sitio web<sup>14</sup>.

---

<sup>14</sup> Digital Learning, Que hace un servidor Web como Apache, diciembre 2011, <http://www.digitalllearning.es/blog/apache-servidor-web-configuracion-apache2-conf/>

### 2.4.2.2 Arquitectura Apache Web Server

La arquitectura del Servidor Apache es modular, es decir está compuesto de módulos o partes que se utilizan de acuerdo a las necesidades que se presentan.

Los módulos de apache se los puede clasificar en tres categorías:

- **Módulos Base:** módulos con funciones básicas de apache.
- **Módulos Multiprocesos:** son responsables de la unión con los puertos de la máquina, aceptando las peticiones y enviando a los hijos a atender a las peticiones.
- **Módulos Adicionales:** cualquier otro modulo que le añade una funcionalidad al servidor.<sup>15</sup>

Las funcionalidades más elementales se encuentran en el módulo base el mismo que requiere un módulo multiproceso en donde se manejan las peticiones, el resto de módulos lo que hacen es agregar funcionalidades al servidor.

### 2.4.2.3 MySQL

Es la base de datos de código fuente abierto más usada del mundo. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar. La extensiva reutilización del código dentro del software y una aproximación minimalista para producir características funcionalmente ricas, ha dado lugar a un sistema de administración de la base de datos incomparable en velocidad, compactación, estabilidad y facilidad de despliegue.

La versión que se utilizara para el desarrollo del proyecto es MySQL5, que brindara facilidad en el manejo de datos al igual que la administración.

---

<sup>15</sup> Desarrolloweb.com, Arquitectura en Módulos Apache, enero 2013, <http://www.desarrolloweb.com/articulos/1112.php>

#### 2.4.2.4 Componentes Principales de MySQL

El servidor MySQL opera en un ambiente de red donde inter-operan clientes con servidores. Los clientes y servidores funcionan o son hospedados en ordenadores a los cuales se los llama anfitriones.

El anfitrión es el conjunto de componentes electrónicos que conforman el ordenador lo cuales son controlados por un sistema operativo. Los componentes clientes y servidores son los programas que operan en los anfitriones para darnos un servicio. En este caso, los servidores y clientes nos proporcionan el servicio de una base de datos.

La base de datos MySQL contiene los siguientes componentes:

- **Mysqld:** es el corazón de MySQL. Es el programa servidor el cual proporciona las bases de datos que se encuentran en memoria o en el disco duro. En los sistemas operativos Windows NT, 2000 o XP el servidor MySQL son los programas `mysqld-nt` o `mysql-max-nt`. Entre los clientes tenemos:
- **Mysqlcc:** ésta interfaz gráfica permite controlar todos los aspectos del servidor MySQL. Permite crear bases de datos, tablas, usuarios, cambiar permisos, acceder y cambiar la información almacenada, de una forma muy intuitiva y muy fácil. Tiene todas las opciones y tiene mejoras de la interfaz de texto `mysql`.
- **Mysql:** ésta interfaz de texto permite también controlar todos los aspectos del servidor pero la interfaz es solamente texto. Permite leer instrucciones del usuario por medio del teclado como también puede leer archivos que contienen instrucciones. Ésta es la aplicación más usada por los programadores para controlar la base de datos. Ésta y otras aplicaciones son aplicaciones de texto que funcionan en la consola.
- **Mysqlimport:** permite importar datos a través de archivos de texto. Provee una interfaz de texto para los comandos `LOAD`, `DATA`, `INFILE`.

- Mysqldump: permite hacer copias o respaldos de la información almacenada para restaurarlos en el mismo servidor o para exportarlo a otros servidores.
- Mysqladmin: permite administrar el servidor con una interfaz gráfica y de una forma muy sencilla
- Mysqlcheck: permite revisar la salud de la base de datos. Permite también reparar dichas bases si fuera necesario.

#### **2.4.2.5 Características de MYSQL**

Las principales características de este gestor de bases de datos son las siguientes:

- El principal objetivo de MySQL es velocidad y robustez.
- Soporta gran cantidad de tipos de datos para las columnas.
- Gran portabilidad entre sistemas, puede trabajar en distintas plataformas y sistemas operativos.
- Cada base de datos cuenta con 3 archivos: Uno de estructura, uno de datos y uno de índice y soporta hasta 32 índices por tabla.
- Aprovecha la potencia de sistemas multiproceso, gracias a su implementación multihilo.
- Flexible sistema de contraseñas y gestión de usuarios, con un muy buen nivel de seguridad en los datos.
- El servidor soporta mensajes de error en distintas lenguas.

#### **2.4.2.6 Ventajas**

- Acceso a las bases de datos de forma simultánea por varios usuarios y/o aplicaciones.
- Seguridad: en forma de permisos y privilegios, determinados usuarios tendrán permiso para consulta o modificación de determinadas tablas. Esto permite compartir datos sin que peligre la integridad de la base de datos o protegiendo determinados contenidos.

- **Potencia:** es un lenguaje muy potente para consulta de bases de datos, usar un motor nos ahorra una enorme cantidad de trabajo.
- **Portabilidad:** es también un lenguaje estandarizado, de modo que las consultas hechas usando SQL son fácilmente portables a otros sistemas y plataformas.

#### **2.4.2.7 Desventajas de MYSQL**

- **SELECT INTO TABLE:** esta característica propia de Oracle, todavía no está implementada.
- **Triggers y Procedures:** se tiene pensado incluir el uso de procesos almacenados en la base de datos, pero no el de triggers, ya que los triggers reducen de forma significativa el rendimiento de la base de datos, incluso en aquellas consultas que no los activan.
- **Transacciones:** las últimas versiones ya hay soporte para transacciones.
- **Integridad referencial:** aunque sí que admite la declaración de claves ajenas en la creación tablas, internamente no las trata de forma diferente al resto de campos<sup>16</sup>.

#### **2.4.3 PHP**

Es el acrónimo de Hipertext Preprocesor. Es un lenguaje de programación del lado del servidor gratuito e independiente de plataforma, rápido, con una gran librería de funciones y mucha documentación.

Es un lenguaje del lado del servidor es aquel que se ejecuta en el servidor web, justo antes de que se envíe la página a través de Internet al cliente. Las páginas que se ejecutan en el servidor pueden realizar accesos a bases de datos, conexiones en red, y otras tareas para crear la página final que verá el cliente.

---

<sup>16</sup> <http://dSPACE.espoch.edu.ec/bitstream/123456789/96/1/18T00372.pdf> Pag39



El cliente solamente recibe una página con el código HTML resultante de la ejecución de la PHP. Como la página resultante contiene únicamente código HTML, es compatible con todos los navegadores<sup>17</sup>.

#### **2.4.3.1 Funcionamiento de PHP**

Cuando el cliente hace una petición al servidor para que le envíe una página web, el servidor ejecuta el intérprete de PHP. Éste procesa el script solicitado que generará el contenido de manera dinámica por ejemplo obteniendo información de una base de datos. El resultado es enviado por el intérprete al servidor, quien a su vez se lo envía al cliente.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, PostgreSQL, Oracle, ODBC, DB2, Microsoft SQL Server.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como Unix, Linux o Mac OS X y Microsoft Windows, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI ()

#### **2.4.3.2 Php 5**

Para el desarrollo del proyecto se utilizó la versión de PHP5, el mismo que utiliza el motor Zend-2, programación orientada a objetos mucho más completo, que permite que PHP proporcione un mejor rendimiento en las aplicaciones web<sup>18</sup>.

---

<sup>17</sup> <http://www.desarrolloweb.com/articulos/392.php>

<sup>18</sup> [http://www.blogdellamas.vacau.com/curso\\_php5/lecciones/lecc01\\_php5\\_introduccion.html](http://www.blogdellamas.vacau.com/curso_php5/lecciones/lecc01_php5_introduccion.html)

### 2.4.3.3 Características de PHP

- Puede ser utilizado en cualquiera de los principales Sistemas Operativos del mercado, incluyendo Linux, muchas variantes Unix, Microsoft Windows, Max OS X.
- Soporta la mayoría de servidores Web, Apache, Microsoft Internet Information Server, Personal Web Server, Netscape e iPlanet, O'Reilly Website Pro server, Caudium, Xitami, OmniHTTPd y muchos otros.
- Tiene módulos disponibles para la mayoría de los servidores, para aquellos otros que soporten el Estándar CGI, PHP puede usarse como procesador CGI.
- Tiene la posibilidad de usar programación procedimental o programación orientada a objetos.
- La característica más potente es que tiene soportes para una gran cantidad de base de datos.

### 2.4.3.4 Tareas Principales de PHP

PHP se va convirtiendo en un lenguaje que permite hacer de todo. En un principio diseñado para realizar poco más que un contador y un libro de visitas, PHP ha experimentado en poco tiempo una verdadera revolución con sus funciones, se pueden realizar una multitud de tareas útiles para el desarrollo web:

- Funciones de correo electrónico.
- Gestión de bases de datos.
- Gestión de archivos.
- Tratamiento de imágenes.

### 2.4.3.5 Funciones de PHP

Las funciones integradas en PHP son muy fáciles de utilizar. Tan sólo hemos de realizar la llamada de la forma apropiada y especificar los parámetros y/o variables necesarios para que la función realice su tarea.

- LOS PARÁMETROS.- son los datos que reciben las funciones y que utilizan para realizar las operaciones de la función.

- TIPOS DE DATOS.- todos los lenguajes de programación tienen un sistema que especifican los diferentes tipos de datos que pueden aparecer en los programas.
- CONTROL DE FLUJO.- es casi imposible hacer programas útiles si no pudiésemos hacer que la ejecución del programa dependiese de determinados valores. Este tipo de ejecución requiere de estructuras de control que indican que partes del código deben ejecutarse en distintas situaciones<sup>19</sup>.

#### **2.4.3.6 Ventajas de PHP**

- Es un lenguaje multiplataforma.
- Completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.
- El código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable.
- Puede interactuar con muchos motores de bases de datos tales como MySQL, MSSQL, Oracle, Informix, PostgreSQL, y otros muchos.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos o extensiones.
- Rapidez PHP generalmente es utilizado como módulo de Apache, lo que lo hace extremadamente veloz. Se ejecuta rápidamente utilizando poca memoria.
- PHP es Open Source<sup>20</sup>.

---

<sup>19</sup> <http://dspace.esPOCH.edu.ec/bitstream/123456789/96/1/18T00372.pdf> Pag 21

<sup>20</sup> [http://administraciondesistemas.pbworks.com/f/Manual\\_PHP5\\_Basico.pdf](http://administraciondesistemas.pbworks.com/f/Manual_PHP5_Basico.pdf) Pag.14

#### 2.4.3.7 Desventajas de PHP

- El código fuente no pueda ser ocultado<sup>21</sup>.
- El manejo de errores es diferente ante otros lenguajes de programación.

#### 2.4.4 HTML5

HTML5 es un lenguaje diseñado para organizar contenido Web. Tiene por objeto facilitar el diseño y el desarrollo Web, mediante la creación de una IU estandarizada e intuitiva para lenguaje de marcación.

Proporciona los medios para diseccionar y compartimentar sus páginas, y le permite crear componentes discretos que no sólo están diseñados para organizar su sitio lógicamente, sino también para darle a su sitio capacidades de sindicación.

##### 2.4.4.1 Características de HTML5

Las herramientas para la administración efectiva de datos, dibujo, video y audio. Facilita el desarrollo de aplicaciones para diferentes navegadores para la Web, así como para dispositivos portátiles. HTML5 es una de las tecnologías que está impulsando los avances de los servicios de computación móvil en nube, gracias a que permite mayor flexibilidad, permitiendo así el desarrollo de sitios Web emocionantes e interactivos. También introduce nuevas etiquetas y mejoras, incluyendo una elegante estructura, controles de formulario, APIs, multimedia, soporte de bases de datos, y una velocidad de procesamiento significativamente más rápida<sup>22</sup>.

---

<sup>21</sup> Abril de 2011, <http://klarimartinezbenjumea.blogspot.com/2011/04/ventajas-y-desventajas.html>

<sup>22</sup> Junio 2011, <http://www.ibm.com/developerworks/ssa/web/library/wa-html5fundamentals/>

## 2.4.5 JAVASCRIPT

Javascript es un lenguaje con muchas posibilidades, utilizado para crear pequeños programas que luego son insertados en una página web y en programas más grandes, orientados a objetos mucho más complejos. Con Javascript podemos crear diferentes efectos e interactuar con nuestros usuarios.

Este lenguaje posee varias características, entre ellas podemos mencionar que es un lenguaje basado en acciones que posee menos restricciones. Además, es un lenguaje que utiliza Windows y sistemas X-Windows, gran parte de la programación en este lenguaje está centrada en describir objetos, escribir funciones que respondan a movimientos del mouse, aperturas, utilización de teclas<sup>23</sup>.

### 2.4.5.1 JQuery

jQuery es una biblioteca de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

jQuery es software libre y de código abierto, permitiendo su uso en proyectos libres y privativos. Ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio<sup>24</sup>.

---

<sup>23</sup> Julio 2013, <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>

<sup>24</sup> Septiembre 2013, <http://es.wikipedia.org/wiki/JQuery>

#### 2.4.6 CSS

CSS son las siglas de Cascading Style Sheet (Hojas de Estilo en Cascada). Es un lenguaje de programación web, usado para dar mayor personalización a nuestras páginas web y mantener el mismo estilo en múltiples páginas. La idea es separar el diseño del contenido y así lograr una mayor independencia a la hora de cambiar o renovar un diseño web<sup>25</sup>.

---

<sup>25</sup> noviembre de 2011, [http://unksexplora.blogspot.com/2011\\_11\\_01\\_archive.html](http://unksexplora.blogspot.com/2011_11_01_archive.html)

## CAPÍTULO 3. METODOLOGÍA RUP

### 3.1 PARADIGMA EN ESPIRAL

El desarrollo en espiral es un modelo de ciclo de vida del software definido por primera vez por Barry Boehm en 1986, utilizado generalmente en la Ingeniería de software.

Las actividades de este modelo se conforman en una espiral, en la que cada bucle o iteración representa un conjunto de actividades. Las actividades no están fijadas a ninguna prioridad, sino que las siguientes se eligen en función del análisis de riesgo, comenzando por el bucle interior<sup>26</sup>.

#### 3.1.1 VENTAJAS

El análisis del riesgo se hace de forma explícita y clara.

- Reduce riesgos del proyecto.
- Incorpora objetivos de calidad.
- Integra el desarrollo con el mantenimiento.

Además es posible tener en cuenta mejoras y nuevos requerimientos sin romper con la metodología, ya que este ciclo de vida no es rígido ni estático.

#### 3.1.2 DEVENTAJAS

- Mucho tiempo en el desarrollo del sistema.
- Modelo costoso.
- Requiere experiencia en la identificación de riesgos.

---

<sup>26</sup> abril 2013, [http://es.wikipedia.org/wiki/Desarrollo\\_en\\_espiral](http://es.wikipedia.org/wiki/Desarrollo_en_espiral)

## **3.2 METODOLOGÍA RUP**

El Proceso Unificado Racional, Rational Unified Process en inglés, y sus siglas RUP, es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.

El RUP no es un sistema con pasos firmemente establecidos, sino que trata de un conjunto de metodologías adaptables al contexto y necesidades de cada organización, donde el software es organizado como una colección de unidades atómicas llamados objetos, constituidos por datos y funciones, que interactúan entre sí<sup>27</sup>.

### **3.2.1 RUP COMO PROCESO DE DESARROLLO**

- RUP es explícito en la definición de software y su trazabilidad, es decir, contempla en relación causal de los programas creados desde los requerimientos hasta la implementación y pruebas.
- RUP identifica claramente a los profesionales (actores) involucrados en el desarrollo del software y sus responsabilidades en cada una de las actividades.

## **3.3 FASES DE DESARROLLO DEL RUP**

Se identifican los principales casos de uso y se identifican los riesgos. Se concreta la idea, la visión del producto, como se enmarca en el negocio, el alcance del proyecto. El objetivo en esta etapa es determinar la visión del proyecto.

---

<sup>27</sup> <http://es.scribd.com/doc/31440864/Metodologia-RUP>



### **3.3.1 MODELADO DEL NEGOCIO**

El equipo se conoce más al funcionamiento de la empresa, sobre conocer sus procesos.

- Entender la estructura y la dinámica de la organización para la cual el sistema va ser desarrollado.
- Entender el problema actual en la organización objetivo e identificar potenciales mejoras.
- Asegurar que clientes, usuarios finales y desarrolladores tengan un entendimiento común de la organización objetivo.

### **3.3.2 REQUISITOS**

En esta línea los requisitos son el contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar los requisitos que especifiquemos.

- Establecer y mantener un acuerdo entre clientes, sobre lo que el sistema podría hacer.
- Proveer a los desarrolladores un mejor entendimiento de los requisitos del sistema.
- Definir el ámbito del sistema.
- Proveer una base para estimar costos y tiempo de desarrollo del sistema.
- Definir una interfaz de usuarios para el sistema, enfocada a las necesidades y metas del usuario.

## **3.4 FASE DE ELABORACIÓN**

Se realiza el plan de proyecto, donde se completan los casos de uso y se mitigan los riesgos. Planificar las actividades necesarias y los recursos requeridos, especificando las características y el diseño de la arquitectura. En esta etapa el objetivo es determinar la arquitectura óptima.

### **3.4.1 ANÁLISIS Y DISEÑO**

En esta actividad se especifican los requerimientos y se describen cómo se van a implementar en el sistema.

- Transformar los requisitos al diseño del sistema.
- Desarrollar una arquitectura para el sistema.
- Adaptar el diseño para que sea consistente con el entorno de implementación.

## **3.5 FASE DE CONTRUCCIÓN**

Es la elaboración de un producto totalmente operativo y en la elaboración del manual de usuario. Construir el producto, la arquitectura y los planes, hasta que el producto está listo para ser probado por los usuarios. En esta etapa el objetivo es llevar a obtener la capacidad operacional inicial.

### **3.5.1 IMPLEMENTACIÓN**

Se implementan las clases y objetos en ficheros fuente, binarios, ejecutables y demás. El resultado final es un sistema ejecutable.

- Planificar qué subsistemas deben ser implementados y en qué orden deben ser integrados, formando el Plan de Integración.
- Cada implementador decide en qué orden implementa los elementos del subsistema.
- Si encuentra errores de diseño, los notifica.
- Se integra el sistema siguiendo el plan.

### **3.5.2 PRUEBAS**

Este flujo de trabajo es el encargado de evaluar la calidad del producto que estamos desarrollando, pero no para aceptar o rechazar el producto al final del proceso de desarrollo, sino que debe ir integrado en todo el ciclo de vida.

- Encontrar y documentar defectos en la calidad del software.

- Generalmente asesora sobre la calidad del software percibida.
- Provee la validación de los supuestos realizados en el diseño y especificación de requisitos por medio de demostraciones concretas.
- Verificar las funciones del producto de software según lo diseñado.
- Verificar que los requisitos tengan su apropiada implementación.

### **3.5.3 ETAPA DE TRANSICIÓN**

En esta etapa se realiza la instalación del producto en el cliente y se procede al entrenamiento de los usuarios. Realizar la transición del producto a los usuarios, lo cual incluye: manufactura, envío, entrenamiento, soporte y mantenimiento del producto, hasta que el cliente quede satisfecho, por tanto en esta fase suelen ocurrir cambios.

### **3.5.4 DESPLIEGUE**

Esta actividad tiene como objetivo producir con éxito distribuciones del producto y distribuirlo a los usuarios. Las actividades implicadas incluyen:

- Probar el producto en su entorno de ejecución final.
- Empaquetar el software para su distribución.
- Distribuir el software.
- Instalar el software.
- Proveer asistencia y ayuda a los usuarios.
- Formar a los usuarios y al cuerpo de ventas.
- Migrar el software existente o convertir bases de datos<sup>28</sup>.

---

<sup>28</sup> Diciembre 2010, <http://fabianbermeop.blogspot.com/2010/12/metodologia-rup-desarrollo-de-software.html>

### 3.5.5 VENTAJAS

- RUP se puede utilizar para proyectos grandes, medianos y pequeños.
- Tiene explícito todo lo que se debe hacer dentro del proceso de desarrollo de software.
- Progreso visible en las etapas tempranas.
- Los usuarios están involucrados continuamente.

## 3.6 UML

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos.

Se utiliza para definir un sistema, para detallar los artefactos en el sistema, para documentar y construir. Es el lenguaje en el que está descrito el modelo<sup>29</sup>.

### 3.6.1 PRINCIPALES BENEFICIOS DEL UML

- Mejores tiempos de desarrollo
- Modelar sistemas utilizando conceptos orientados a objetos.
- Establecer conceptos y artefactos ejecutables.
- Encaminar el desarrollo del escalamiento en sistemas complejos.
- Crear un lenguaje de modelado utilizado por humanos y máquinas.
- Mejor soporte en la planeación y al control de proyectos.
- Reutilización y minimización de costos<sup>30</sup>.

---

<sup>29</sup> Octubre 2013, [http://es.wikipedia.org/wiki/Lenguaje\\_Unificado\\_de\\_Modelado](http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado)

<sup>30</sup> <http://profesores.fi-b.unam.mx/carlos/aydoo/uml.html>

### 3.6.2 MODELOS

Los Modelos representan un sistema desde una perspectiva específica, cada modelo permite visualizar distintos aspectos del sistema.

#### Modelo Estático (Estructural)

- Diagrama de Despliegue
- Diagrama de Clases
- Diagrama de Objetos
- Diagrama de Componentes

#### Modelo Dinámico (Comportamiento)

- Diagrama de estados
- Diagrama de actividades
- Diagrama de secuencia
- Diagrama de colaboración
- Diagrama de casos de uso

#### 3.6.2.1 Modelo Estático

Es aquel en el que se describen las clases y los objetos. Se denomina estático porque muestran todas las relaciones posibles a lo largo del tiempo<sup>31</sup>.

##### 3.6.2.1.1 Diagrama de despliegue

Un Diagrama de Despliegue modela la arquitectura en tiempo de ejecución de un sistema. Esto muestra la configuración de los elementos de hardware (nodos) y muestra cómo los elementos y artefactos del software se trazan en esos nodos<sup>32</sup>.

---

<sup>31</sup> Ingeniería del software, Escrito por Benet Campderrich Falgueras, Pag 38

<sup>32</sup> [http://www.sparxsystems.com.ar/resources/tutorial/uml2\\_deploymentdiagram.html](http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html)

**Componentes:**

- **Nodo.-** Es un objeto físico en tiempo de ejecución que representa un recurso computacional, generalmente con memoria y capacidad de procesamiento. Un Nodo es un elemento de hardware o software.
- **Instancia de nodo.-** Se puede distinguir desde un nodo por el hecho de que su nombre esta subrayado y tiene dos puntos antes del tipo de nodo base. Una instancia puede o no tener un nombre antes de los dos puntos.
- **Estereotipo de nodo.-** Son cosas u objetos que se repiten sin variación. El estereotipo de un nodo es la manera de poder verificar que tipo de nodo es el que se está observando.
- **Artefactos.-** Es un producto del proceso de desarrollo de software, que puede incluir los modelos del proceso (modelos de Caso de uso, modelos de Diseño, etc.). Donde un artefacto es un conjunto de componentes.
- **Asociación.-** Representa una ruta de comunicación entre los nodos. Donde esta asociación va incluida con misma dependencia del diagrama de componentes<sup>33</sup>.

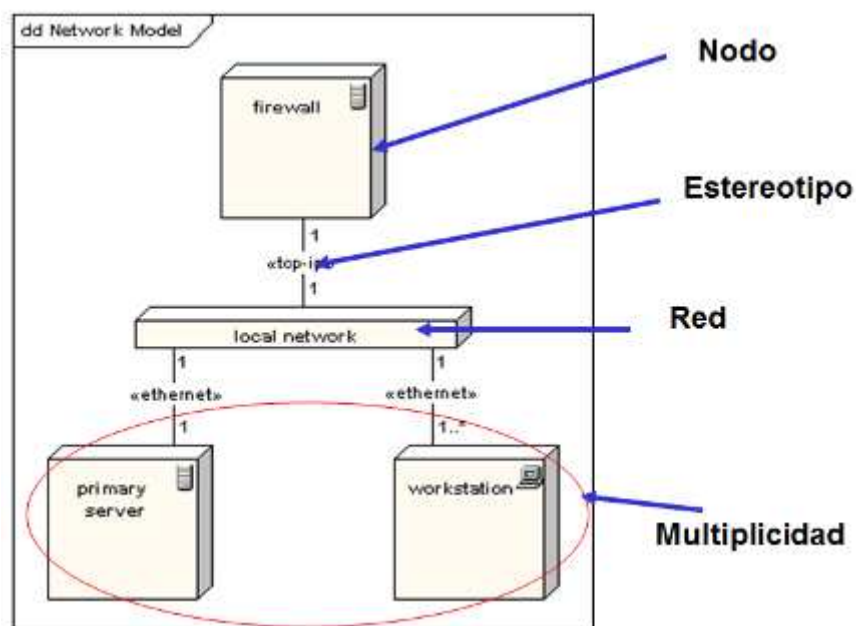
**Comunicación entre nodos**

Los nodos se conectan mediante asociaciones de comunicación, estas asociaciones indican:

La ruta de comunicación entre los nodos en donde intercambian objetos o envían mensajes a través de su ruta.

---

<sup>33</sup> [http://www.ecured.cu/index.php/Diagrama\\_de\\_despliegue](http://www.ecured.cu/index.php/Diagrama_de_despliegue)



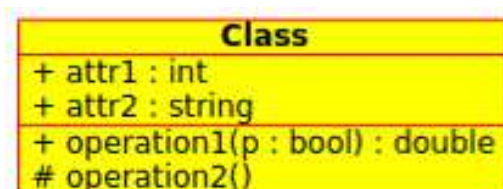
**Diagrama de Despliegue**

### 3.6.2.1.2 Diagrama de Clases

Un diagrama de clases es un tipo de diagrama estático que describe la estructura de un sistema mostrando sus clases, orientados a objetos.

#### Clase

Una clase define los atributos y los métodos de una serie de objetos. Todos los objetos de esta clase (instancias de esa clase) tienen el mismo comportamiento y el mismo conjunto de atributos (cada objeto tiene el suyo propio). En ocasiones se utiliza el término «tipo» en lugar de clase, pero recuerde que no son lo mismo, y que el término tipo tiene un significado más general.



**Estructura de una Clase**

## Atributos

En UML, los atributos se muestran al menos con su nombre, y también pueden mostrar su tipo, valor inicial y otras propiedades. Los atributos también pueden ser mostrados visualmente:

- + Indica atributos públicos
- # Indica atributos protegidos
- - Indica atributos privados

## Operaciones

Las operaciones (métodos) también se muestran al menos con su nombre, y pueden mostrar sus parámetros y valores de retorno. Las operaciones, al igual que los atributos, se pueden mostrar visualmente:

- + Indica operaciones públicas
- # Indica operaciones protegidas
- - Indica operaciones privadas

## Plantillas

Las clases pueden tener plantillas, un valor usado para una clase no especificada o un tipo. El tipo de plantilla se especifica cuando se inicia una clase (es decir cuando se crea un objeto).

### Asociaciones de clases

Las clases se pueden relacionar con otras de diferentes maneras:

## Generalización

**La herencia.-** Es uno de los conceptos fundamentales de la programación orientada a objetos, en la que una clase «recoge» todos los atributos y operaciones de la clase de la que es heredera, y puede alterar/modificar algunos de ellos, así como añadir más atributos y operaciones propias.

En UML, una asociación de generalización entre dos clases, coloca a estas en una jerarquía que representa el concepto de herencia de una clase derivada de la



clase base. En UML, las generalizaciones se representan por medio de una línea que conecta las dos clases, con una flecha en el lado de la clase base<sup>34</sup>.

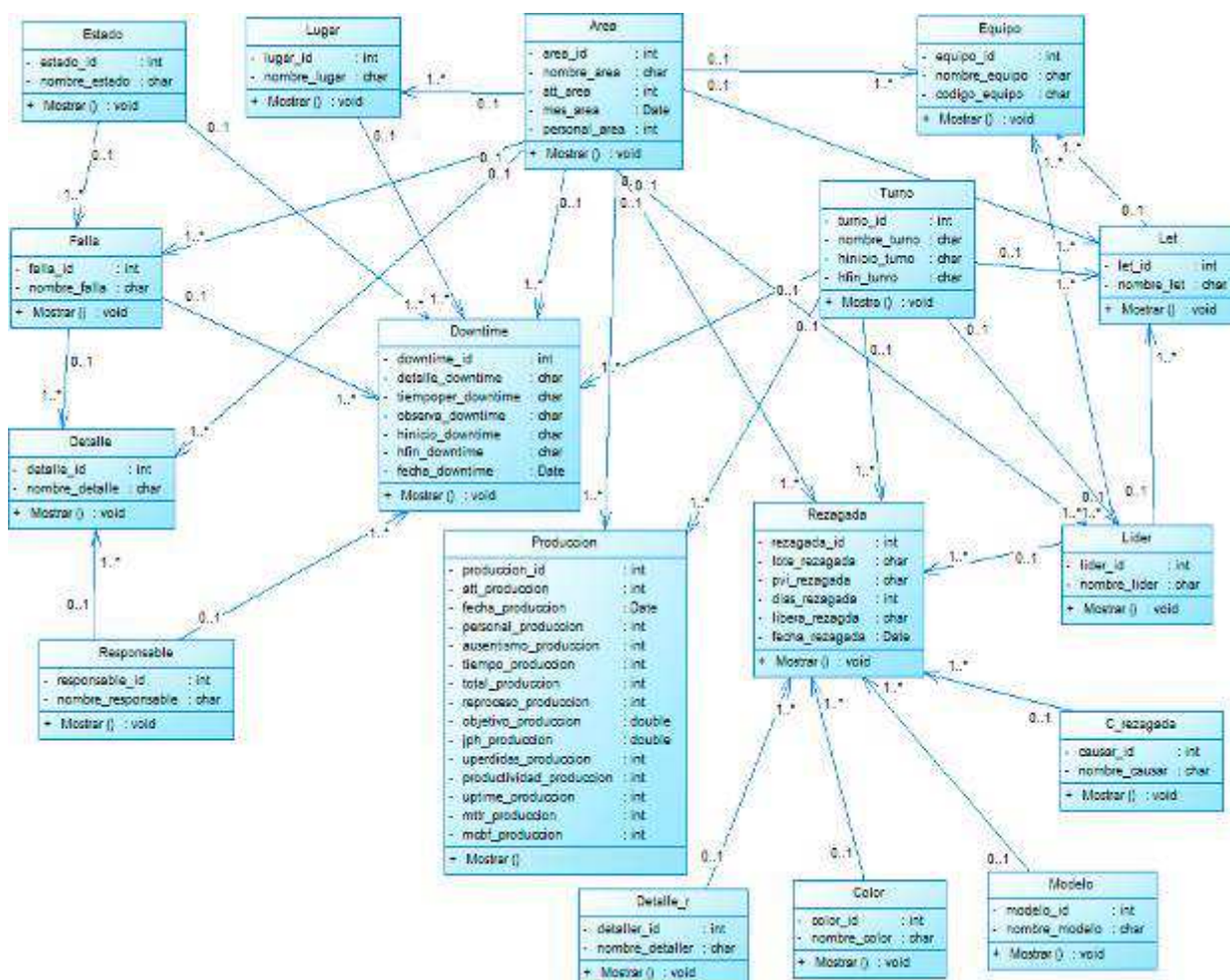


Diagrama de Clases

<sup>34</sup> <http://docs.kde.org/stable/es/kdesdk/umbrello/uml-elements.html>

### 3.6.2.1.3 Diagrama de objetos

Los diagramas de objetos son utilizados durante el proceso de Análisis y diseño de los sistemas informáticos en la metodología UML.

Se puede considerar un caso especial de un diagrama de clases en el que se muestran instancias específicas de clases (objetos) en un momento particular del sistema. Los diagramas de objetos utilizan un subconjunto de los elementos de un diagrama de clase. Los diagramas de objetos no muestran la multiplicidad ni los roles, aunque su notación es similar a los diagramas de clase.

Una diferencia con los diagramas de clase es que el compartimiento de arriba va en la forma Nombre de objeto: Nombre de clase<sup>35</sup>.

### 3.6.2.1.4 Diagrama de componentes

Un diagrama de componentes representa cómo un sistema de software es dividido en componentes y muestra las dependencias entre estos componentes.

Los componentes físicos incluyen archivos, cabeceras, bibliotecas compartidas, módulos, ejecutables, o paquetes. Los diagramas de Componentes prevalecen en el campo de la arquitectura de software pero pueden ser usados para modelar y documentar cualquier arquitectura de sistema<sup>36</sup>.

## 3.6.2.2 Modelo Dinámico

Describe los aspectos de un sistema que cambian con el tiempo. EL modelo dinámico se utiliza para especificar e implementar los aspectos de control del sistema.<sup>37</sup>

---

<sup>35</sup> marzo 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_objetos](http://es.wikipedia.org/wiki/Diagrama_de_objetos)

<sup>36</sup> marzo 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_componentes](http://es.wikipedia.org/wiki/Diagrama_de_componentes)

<sup>37</sup> <http://es.scribd.com/doc/30208081/MODELO-DINAMICO>

### 3.6.2.2.1 Diagrama de Estados

Muestran los diferentes estados de un objeto durante su vida, y los estímulos que provocan los cambios de estado en un objeto.

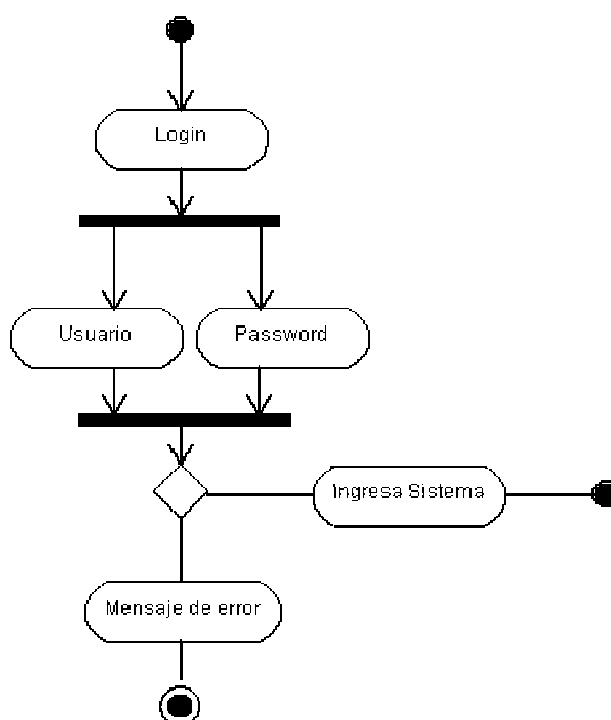
Los diagramas de estado ven a los objetos como máquinas de estado o autómatas finitos que pueden estar en un conjunto de estados finitos y que pueden cambiar su estado a través de un estímulo perteneciente a un conjunto finito.

#### Elementos:

- **Estado.-** Identifica un periodo de tiempo del objeto (no instantáneo) en el cual el objeto está esperando alguna operación, tiene cierto estado característico o puede recibir cierto tipo de estímulos.
- **Eventos.-** Es una ocurrencia que puede causar la transición de un estado a otro de un objeto
- **Envío de mensajes.-** además de mostrar y transición de estados por medio de eventos, puede representarse el momento en el cual se envían mensajes a otros objetos.
- **Transición simple.-** Una transición simple es una relación entre dos estados que indica que un objeto en el primer estado puede entrar al segundo estado y ejecutar ciertas operaciones, cuando un evento ocurre y si ciertas condiciones son satisfechas.
- **Transición interna.-** Es una transición que permanece en el mismo estado, en vez de involucrar dos estados distintos. Representa un evento que no causa cambio de estado. Se denota como una cadena adicional en el compartimiento de acciones del estado.
- **Sub-estados.-** Un estado puede descomponerse en sub-estados, con transiciones entre ellos y conexiones al nivel superior. Las conexiones se ven al nivel inferior como estados de inicio o fin, los cuales se suponen conectados a las entradas y salidas del nivel inmediatamente superior.
- **Transición compleja.-** Una transición compleja relaciona tres o más estados en una transición de múltiples fuentes y/o múltiples destinos.

Representa la subdivisión en threads del control del objeto o una sincronización.

- **Transición a estados anidados:** Una transición de hacia un estado complejo (descrito mediante estados anidados) significa la entrada al estado inicial del subdiagrama. Las transiciones que salen del estado complejo se entienden como transiciones desde cada uno de los subestados hacia afuera (a cualquier nivel de profundidad).



**Diagrama de Estado**

### 3.6.2.2.2 Diagrama de Actividades

Representa los flujos de trabajo paso a paso de negocio y operacionales de los componentes en un sistema. Un diagrama de actividades muestra el flujo de control general<sup>38</sup>.

Diagrama de actividades

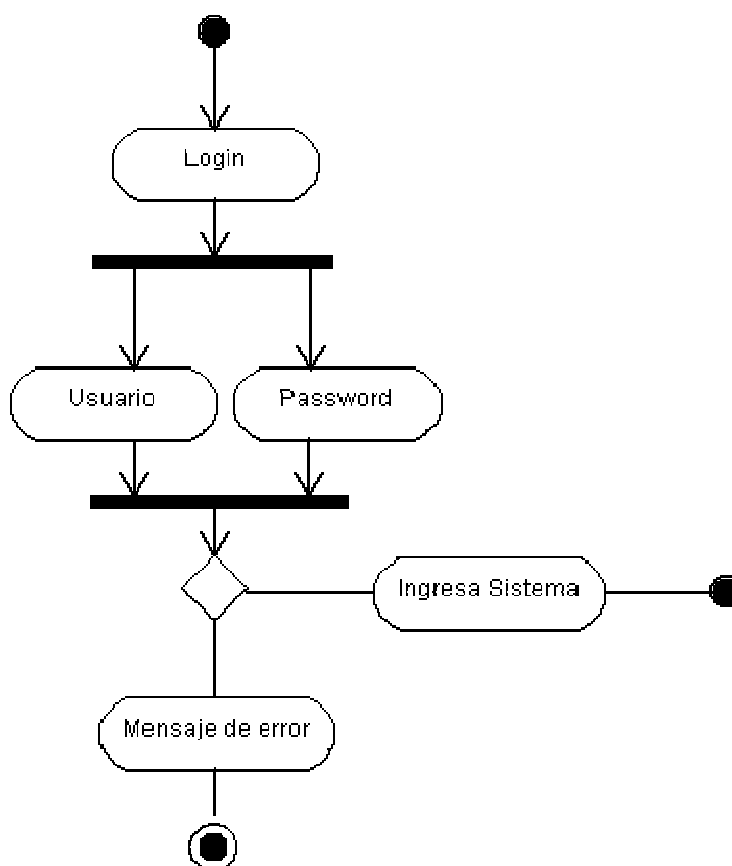


Diagrama Registro de Downtime

---

<sup>38</sup> Octubre 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_flujo](http://es.wikipedia.org/wiki/Diagrama_de_flujo)

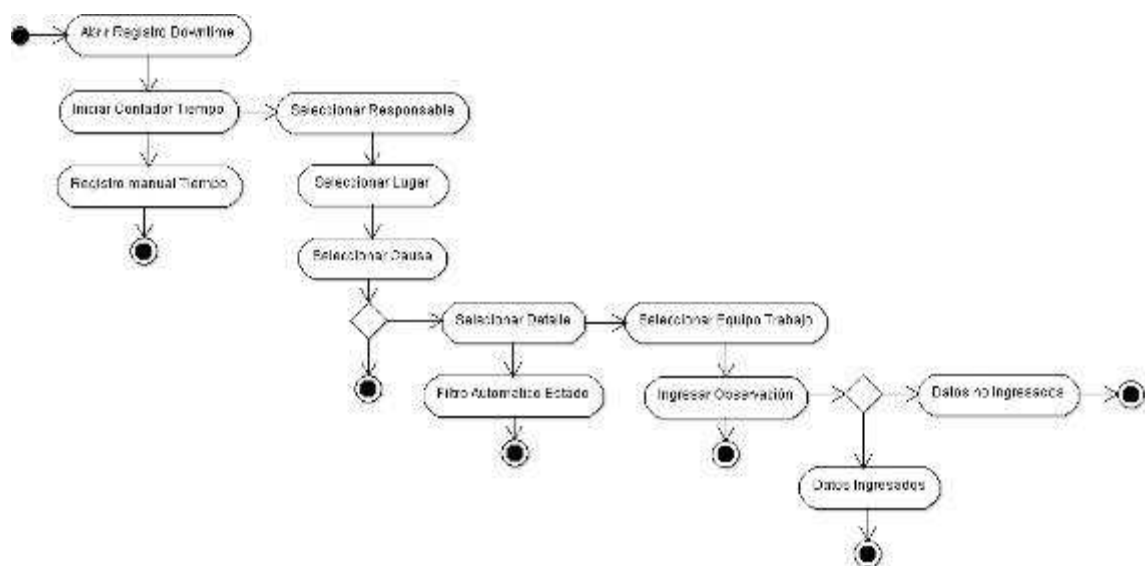


Diagrama Registro de Producción

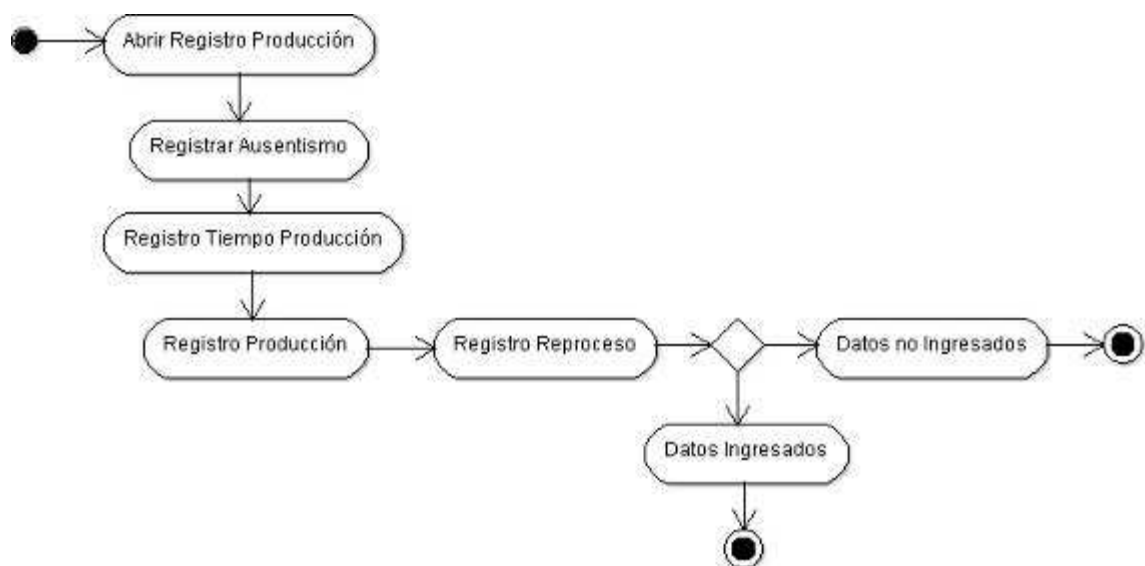
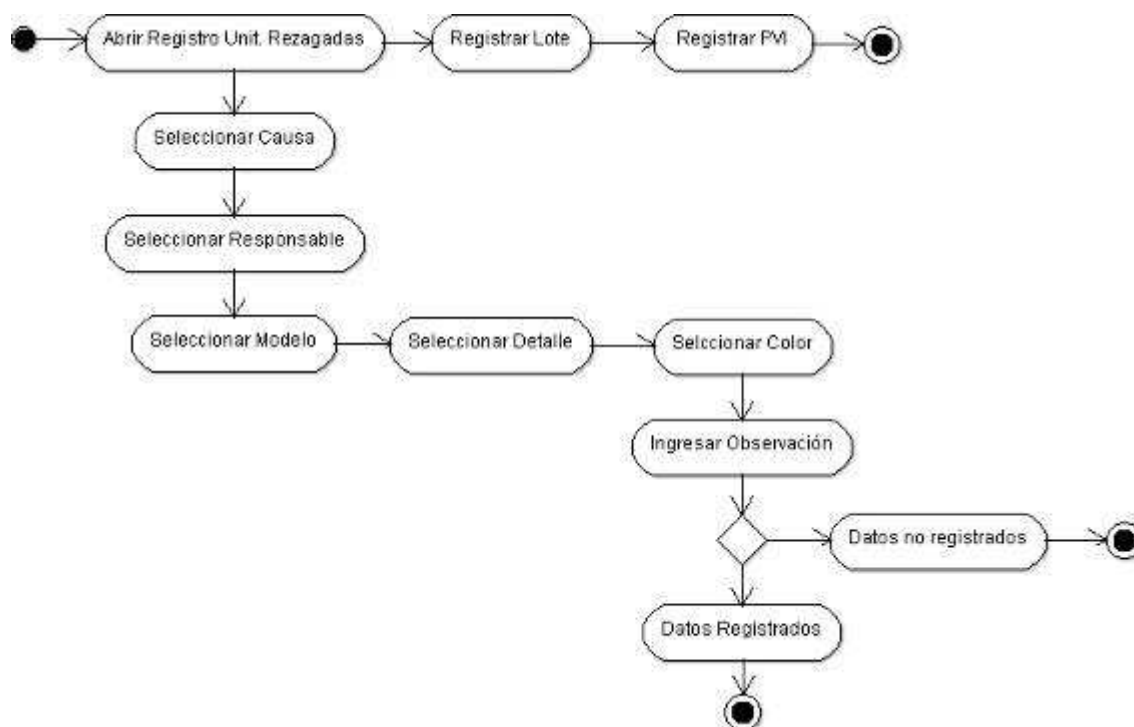


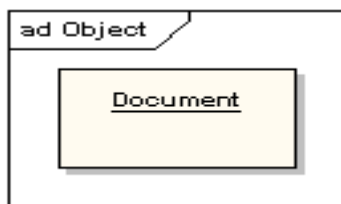
Diagrama Registro de Unidades Rezagadas



### Elementos del Diagrama de Actividades:

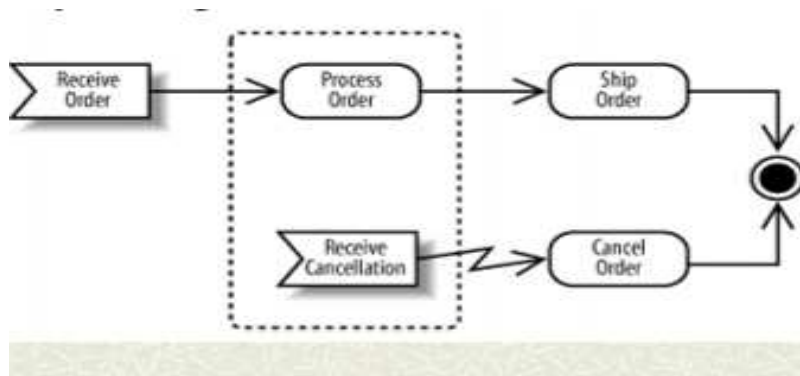
- **Acciones.-** Una acción es un paso del proceso, bajo la premisa se inicia para ser terminado.
- **Actividades.-** Conjunto de acciones que modelan el proceso.
- **Flujo de Control.-** Permite visualizar el flujo de Control de una acción a otra. La notación es una línea con una flecha punteada.
- **Nodo Inicial.-** El primero es el comienzo de una actividad y se describe con un punto negro.
- **Nodo Final.-** Existen dos tipos de nodos final:
  - El nodo final de la actividad que se define con un círculo con un punto negro en la mitad.
  - Nodo final de Flujo se define con un círculo con una cruz en el centro del mismo.

- **Flujo de Objetos.-** Es la ruta por el que pueden pasar objetos o datos.



- **Nodos de Decisión y Combinación.-** Los nodos de decisión y combinación tienen la misma notación: una forma de diamante. Los dos se pueden nombrar. Los flujos de control que provienen de un nodo de decisión la misma que tendrán condiciones que permitirán el control para fluir si la condición de guarda se realiza.
- **Nodos de Combinación y Unión.-** Estos nodos tienen una notación tanto de una barra horizontal como una vertical
  - **Unión.-** Sincroniza dos flujos de entrada y produce un solo flujo de salida, en donde el flujo de salida desde una unión no se puede ejecutar hasta que todos los flujos se hayan recibido.
  - **Combinación.-** Si dos o más flujos de entrada se reciben por un símbolo de combinación, la acción a la que el flujo de salida apunta se ejecuta dos o más veces.
- **Región de Actividad Interrumpible.-** Una región de actividad interrumpible rodea un grupo de acciones que se pueden interrumpir.





Región de Actividad Interrumpible<sup>39</sup>

- **Partición de una Actividad:** son representados como calles horizontales o verticales, es decir se usan para separar acciones dentro de una actividad.

### 3.6.2.2.3 Diagrama de Secuencia

Muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada caso de uso. Mientras que el diagrama de casos de uso permite el modelado de una vista negocios del escenario, el diagrama de secuencia contiene detalles de implementación del escenario, incluyendo los objetos y clases que se usan para implementar el escenario y mensajes intercambiados entre los objetos<sup>40</sup>.

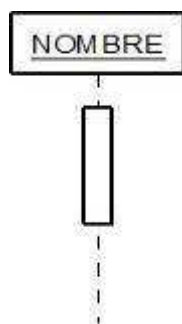
### Elementos del Diagrama de Secuencia

- **Objetos.-** Se representan de modo usual: rectángulo con nombre, mensajes entre los objetos representados por líneas continuas con una punta de flecha y el tiempo representado como una progresión vertical. Los objetos se colocan cerca de la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen el diagrama.

<sup>39</sup> Mayo 2012, [http://www.ctr.unican.es/asignaturas/procodis\\_3\\_II/Doc/stateDiagram.pdf](http://www.ctr.unican.es/asignaturas/procodis_3_II/Doc/stateDiagram.pdf)

<sup>40</sup> Septiembre 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_secuencia](http://es.wikipedia.org/wiki/Diagrama_de_secuencia)

La extensión que está debajo (en forma descendente) de cada objeto será una línea discontinua conocida como la línea de vida de un objeto, junto con la línea de vida de un (objeto rectángulo) se le conoce como activación, el cual una operación que realiza el objeto la interpreta como la duración de la activación.<sup>41</sup>



- **Línea de Vida.-** Son líneas de puntos verticales e indican la presencia de objetos durante el tiempo
- **Mensaje.-** Son flechas que representan comunicación entre objetos

Flecha	Tipo de mensaje
	Simple
	Sincrónico
	Asincrónico
	Rechazado
	Time out

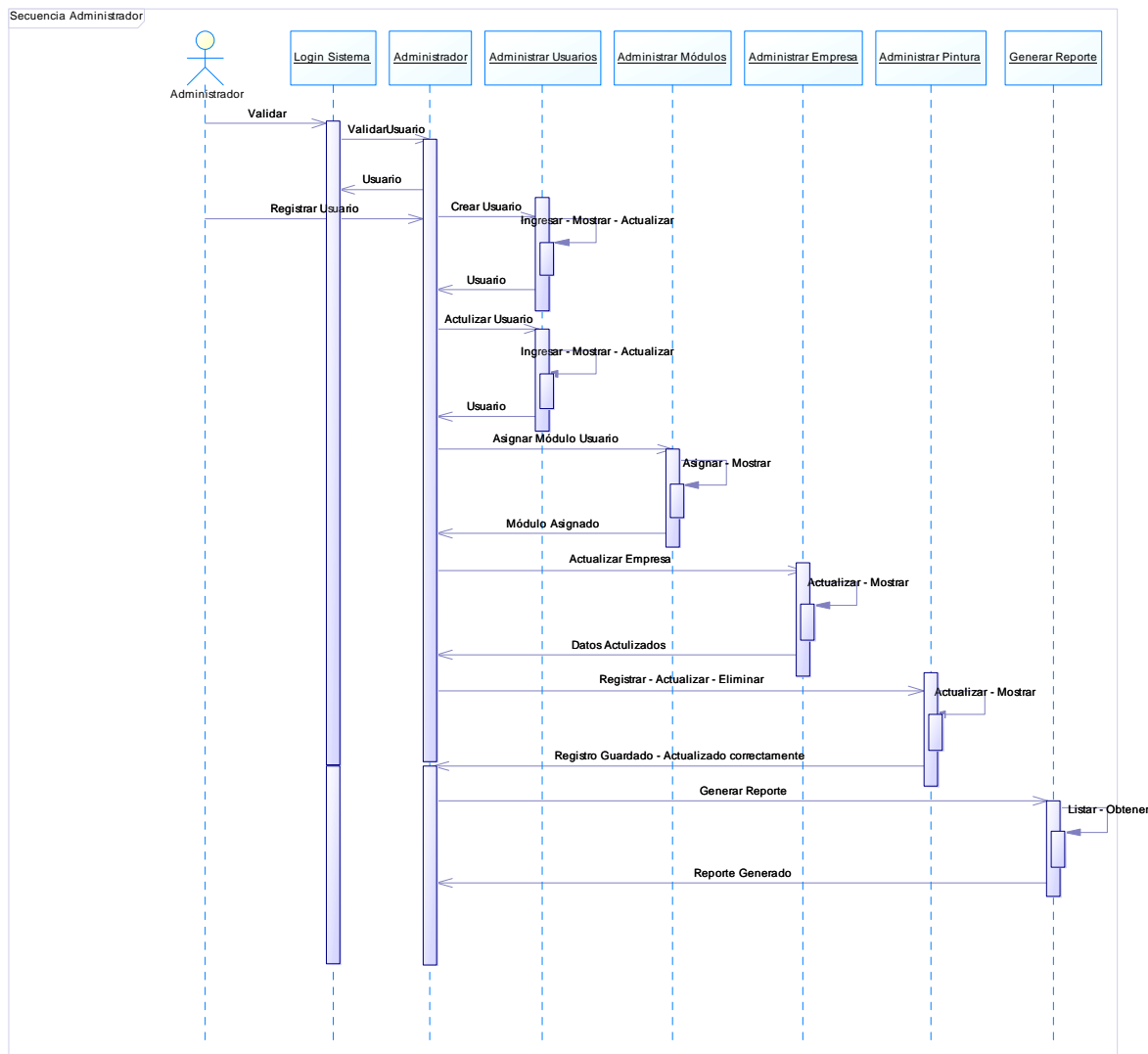
- **El tiempo.-** El diagrama representa al tiempo en dirección vertical, un mensaje que este en la parte superior ocurrirá antes que uno que este

---

<sup>41</sup> Vargas Y, Díaz Mabel mayo 2009, Diagrama de Secuencia, <http://exposicinds.blogspot.com/>

cerca en la parte inferior. Es por esta razón que el Diagrama de Secuencia tiene dos dimensiones: la dimensión horizontal es la disposición de los objetos mientras que la dimensión vertical muestra el paso del tiempo.

- **Recursividad.-** Es la operación cuando un objeto invoca a sí mismo.



**Diagrama de Secuencia**

#### 3.6.2.2.4 Diagrama de Colaboración

Muestra la implementación de una operación. La comunicación muestra los parámetros y las variables locales de la operación, así como asociaciones más permanentes. Cuando se implementa el comportamiento, la secuencia de los

mensajes corresponde a la estructura de llamadas anidadas y el paso de señales del programa<sup>42</sup>.

### Elementos del Diagrama de Colaboración:

- **Rol de la Clase.-** Describe cómo se comporta una clase pero no enlista sus atributos.
- **Rol de Asociaciones.-** Describe el comportamiento de una asociación en un caso en particular, para esto se usa líneas simples con estereotipos.
- **Mensajes.-** El diagrama de Colaboración no cuenta con una forma explícita e denotar el tiempo es por ello que se numera a los mensajes en orden de ejecución

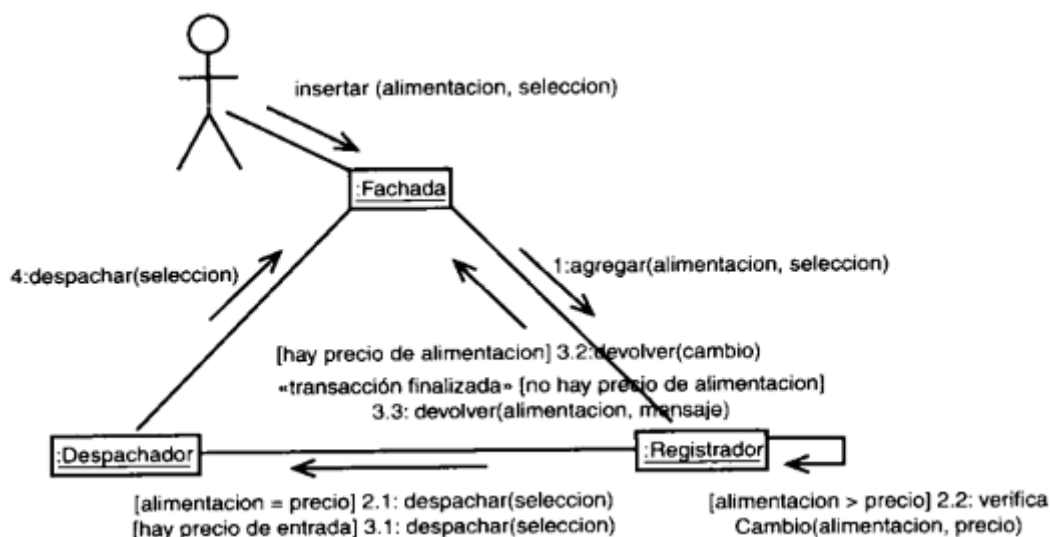


Diagrama de Colaboración<sup>43</sup>

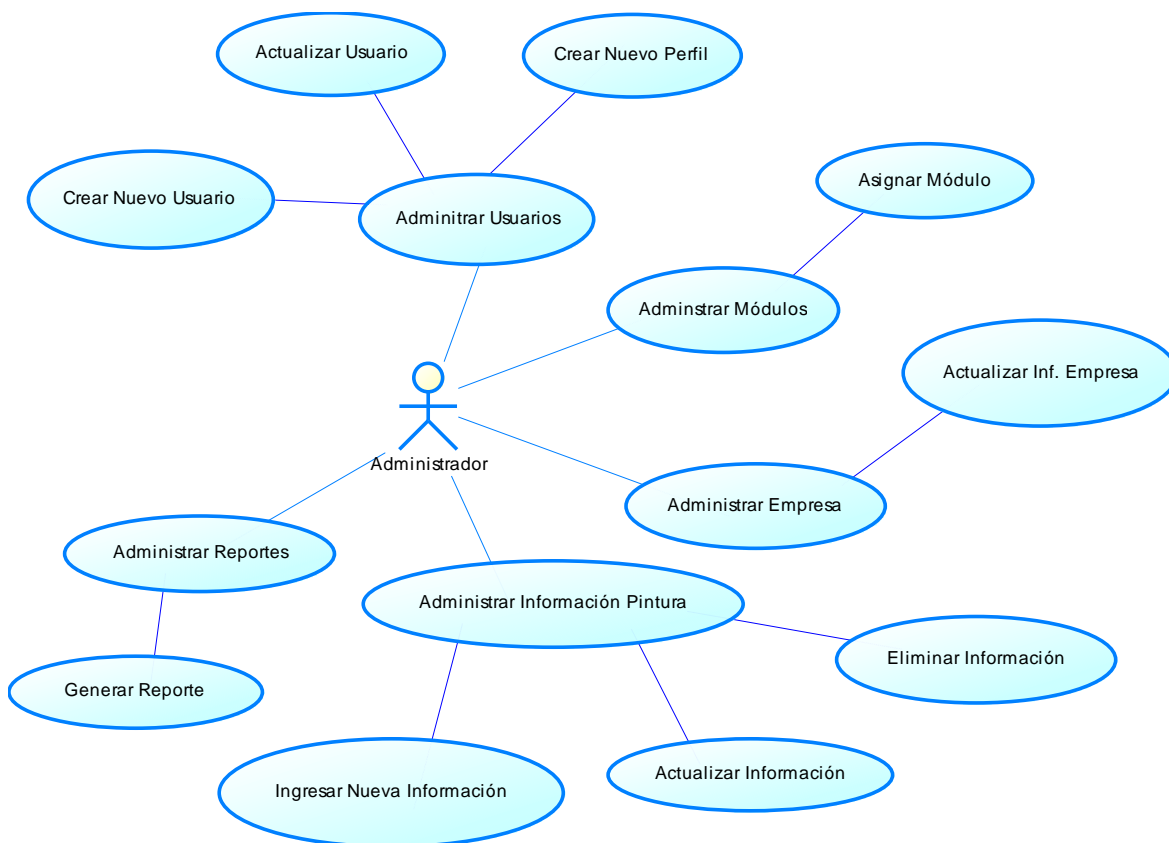
#### 3.6.2.2.5 Diagrama de Casos de Uso

Permite documentar el comportamiento del sistema desde el punto de vista del usuario, es decir determinan los requisitos los requisitos funcionales del sistema.

<sup>42</sup> Abril 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_colaboraci%C3%B3n](http://es.wikipedia.org/wiki/Diagrama_de_colaboraci%C3%B3n)

<sup>43</sup> Mayo 2013, <http://ppancardo.webs.com/LSD/UML%20diagrama%20de%20colaboraciones.pdf>

Una de las ventajas más importantes radica en la facilidad para interpretarlo, es por ello que este diagrama es útil para poder comunicarse con el cliente.



**Diagrama de Casos de Uso**

#### **Elementos Diagrama Casos de Uso:**

- **Autores.-** Representan el tipo de usuario involucrados en los procesos que realiza el sistema, también puede ser otro sistema u unidades organizativas de la empresa, es decir un actor es un rol que alguien juega en el sistema.
- **Caso de Uso.-** Es la tarea a realizarse con el apoyo del Sistema, son representados mediante un óvulo.

**Relaciones:**

- **Asociación.-** Es un tipo de relación que indica la invocación desde un caso de uso a otra operación, es representada con una flecha simple.



- **Dependencia o Instanciación.-** Es una forma particular de relación entre clases en donde una clase depende de otra, es representada con una flecha punteada.



- **Generalización.-** Cumplen doble función dependiendo de su estereotipo, estos pueden ser de uso <<uses>> y herencia <<extends>>.

## 3.7 CONCLUSIONES Y RECOMENDACIONES

### 3.7.1 CONCLUSIONES

- Las herramientas de desarrollo brindan mayores facilidades al momento de la implementación del software.
- La utilización del software libre reduce el costo de desarrollo y mantenimiento.
- Las seguridades de los usuarios son definidas con claridad al inicio del estudio y posteriormente en el desarrollo.
- Los riesgos del proyecto son identificados al inicio del proyecto.

### 3.7.2 RECOMENDACIONES

- Se recomienda obtener un respaldo de la información al menos tres veces por semana.

- Para realizar una actualización del sistema obtener un respaldo de todo el paquete y sus principales componentes. Principalmente un backup de la Base de Datos.
- El usuario de administrador deberá ser la persona encargada quien realice la creación de nuevos usuarios.
- El manual de usuario deberá ser manejado por el administrador del sistema ya que es la persona encargada de manejo del sistema.
- Validar los permisos que se brindó a cada uno de los usuarios del sistema.
- Es necesario implementar un módulo de reportes gráficos, donde se pueda apreciar el desempeño de la planta.

## BIBLIOGRAFÍA

- <sup>1</sup> Cota A. 1994 "Ingeniería de Software". Soluciones Avanzadas. Julio de 1994. pp. 5-13.
- <sup>2</sup> Lewis G. 1994. "What is Software Engineering?" DataPro (4015). Feb 1994. pp. 1-10.
- <sup>3</sup> <http://www.monografias.com/trabajos5/inso/inso.shtml#estra>
- <sup>4</sup> Jacobson, I. 1998. "Applying UML in The Unified Process" Presentación. Rational Software. Presentación disponible en <http://www.rational.com/uml> como UMLconf.zip
- <sup>5</sup> [http://www.suronline.net/nuevo\\_sitio/beneficios-funcionamiento-aplicaciones-web.asp](http://www.suronline.net/nuevo_sitio/beneficios-funcionamiento-aplicaciones-web.asp)
- <sup>6</sup> <http://unidad6aplicacionesweb1.blogspot.com/2012/10/estructura-de-las-aplicaciones-web.html>
- <sup>7</sup> [http://luisf.wikispot.org/Arquitectura\\_web](http://luisf.wikispot.org/Arquitectura_web), 2007
- <sup>8</sup> Agosto 2009,  
<http://www.consumoteca.com/telecomunicaciones/internet/servidor-web/>
- <sup>9</sup> <http://www.mastermagazine.info/termino/6055.php>
- <sup>10</sup> <http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node21.html>

- <sup>11</sup> [http://es.wikipedia.org/wiki/Dise%C3%B1o\\_web#Etapas Septiembre, 2013](http://es.wikipedia.org/wiki/Dise%C3%B1o_web#Etapas_Septiembre,2013)
- <sup>12</sup> Julio 2013, [http://es.wikipedia.org/wiki/Adobe\\_Dreamweaver](http://es.wikipedia.org/wiki/Adobe_Dreamweaver)
- <sup>13</sup> <http://www.mitecnologico.com/Main/HerramientasFrontEnd>
- <sup>14</sup> Digital Learning, Que hace un servidor Web como Apache, diciembre 2011, <http://www.digitallearning.es/blog/apache-servidor-web-configuracion-apache2-conf/>
- <sup>15</sup> Desarrolloweb.com, Arquitectura en Módulos Apache, enero 2013, <http://www.desarrolloweb.com/articulos/1112.php>
- <sup>16</sup> <http://dspace.espace.edu.ec/bitstream/123456789/96/1/18T00372.pdf>  
Pag39
- <sup>17</sup> <http://www.desarrolloweb.com/articulos/392.php>
- <sup>18</sup> [http://www.blogdellamas.vacau.com/curso\\_php5/lecciones/lecc01\\_php5\\_introduccion.html](http://www.blogdellamas.vacau.com/curso_php5/lecciones/lecc01_php5_introduccion.html)
- <sup>19</sup> <http://dspace.espace.edu.ec/bitstream/123456789/96/1/18T00372.pdf>  
Pag 21
- <sup>20</sup>  
[http://administraciondesistemas.pbworks.com/f/Manual\\_PHP5\\_Basico.pdf](http://administraciondesistemas.pbworks.com/f/Manual_PHP5_Basico.pdf)  
Pag.14
- <sup>22</sup> Abril de 2011,  
<http://klarimartinezbenjumea.blogspot.com/2011/04/ventajas-y-desventajas.html>
- <sup>23</sup> Junio 2011, <http://www.ibm.com/developerworks/ssa/web/library/wa-html5fundamentals/>
- <sup>24</sup> Julio 2013, <http://www.maestrosdelweb.com/editorial/%C2%BFque-es-javascript/>
- <sup>25</sup> Septiembre 2013, <http://es.wikipedia.org/wiki/JQuery>
- <sup>26</sup> noviembre de  
2011, [http://unksexplora.blogspot.com/2011\\_11\\_01\\_archive.html](http://unksexplora.blogspot.com/2011_11_01_archive.html)
- <sup>27</sup> abril 2013, [http://es.wikipedia.org/wiki/Desarrollo\\_en\\_espiral](http://es.wikipedia.org/wiki/Desarrollo_en_espiral)
- <sup>28</sup> <http://es.scribd.com/doc/31440864/Metodologia-RUP>



- <sup>29</sup> Diciembre 2010,  
<http://fabianbermeop.blogspot.com/2010/12/metodologia-rup-desarrollo-de-software.html>
- <sup>30</sup> Octubre 2013,  
[http://es.wikipedia.org/wiki/Lenguaje\\_Unificado\\_de\\_Modelado](http://es.wikipedia.org/wiki/Lenguaje_Unificado_de_Modelado)
- <sup>31</sup> <http://profesores.fi-b.unam.mx/carlos/aydoo/uml.html>
- <sup>32</sup> Ingeniería del software, Escrito por Benet Campderrich Falgueras, Pag38
- <sup>33</sup>  
[http://www.sparxsystems.com.ar/resources/tutorial/uml2\\_deploymentdiagram.html](http://www.sparxsystems.com.ar/resources/tutorial/uml2_deploymentdiagram.html)
- <sup>34</sup> [http://www.ecured.cu/index.php/Diagrama\\_de\\_despliegue](http://www.ecured.cu/index.php/Diagrama_de_despliegue)
- <sup>35</sup> <http://docs.kde.org/stable/es/kdesdk/umbrello/uml-elements.html>
- <sup>36</sup> marzo 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_objetos](http://es.wikipedia.org/wiki/Diagrama_de_objetos)
- <sup>37</sup> marzo 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_componentes](http://es.wikipedia.org/wiki/Diagrama_de_componentes)
- <sup>38</sup> <http://es.scribd.com/doc/30208081/MODELO-DINAMICO>
- <sup>39</sup> Octubre 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_flujo](http://es.wikipedia.org/wiki/Diagrama_de_flujo)
- <sup>40</sup> Septiembre 2013, [http://es.wikipedia.org/wiki/Diagrama\\_de\\_secuencia](http://es.wikipedia.org/wiki/Diagrama_de_secuencia)
- <sup>41</sup> Vargas Y, Díaz Mabel mayo 2009, Diagrama de Secuencia,  
<http://exposicinds.blogspot.com/>
- <sup>42</sup> Abril 2013,  
[http://es.wikipedia.org/wiki/Diagrama\\_de\\_colaboraci%C3%B3n](http://es.wikipedia.org/wiki/Diagrama_de_colaboraci%C3%B3n)

## MANUAL TÉCNICO

1.1	ANÁLISIS DE REQUERIMIENTOS.....	67
1.1.1	IDENTIFICACIÓN DE ACTORES .....	67
1.2	DISEÑO .....	67
1.2.1	DIAGRAMA ADMINISTRADOR .....	67
1.2.2	DIAGRAMA DE USUARIO .....	68
1.2.3	DIAGRAMA DE OPERADOR .....	68
1.2.4	DICCIONARIO DE CASOS DE USO .....	69
1.2.5	MÓDELO FÍSICO .....	72
1.2.6	MÓDELO LÓGICO.....	73
1.2.7	DIAGRAMA DE CLASES.....	74
1.3	DIAGRAMA DE SECUENCIAS.....	75
1.3.1	DIAGRAMA ADMINISTRADOR .....	75
1.3.2	DIAGRAMA USUARIO .....	76
1.4	DIAGRAMA DE ACTIVIDADES .....	77
1.4.1	Diagrama Registro de Downtime .....	77
1.4.2	Diagrama Registro de Producción.....	77
1.4.3	Diagrama Registro de Unidades Rezagadas.....	78
1.5	construcción .....	78
1.5.1	BACK END .....	78
1.5.2	BACK END .....	¡Error! Marcador no definido.
1.6	PRUEBAS .....	82
1.6.1	ESTRATEGIA DE PRUEBAS .....	82

# MANUAL TÉCNICO

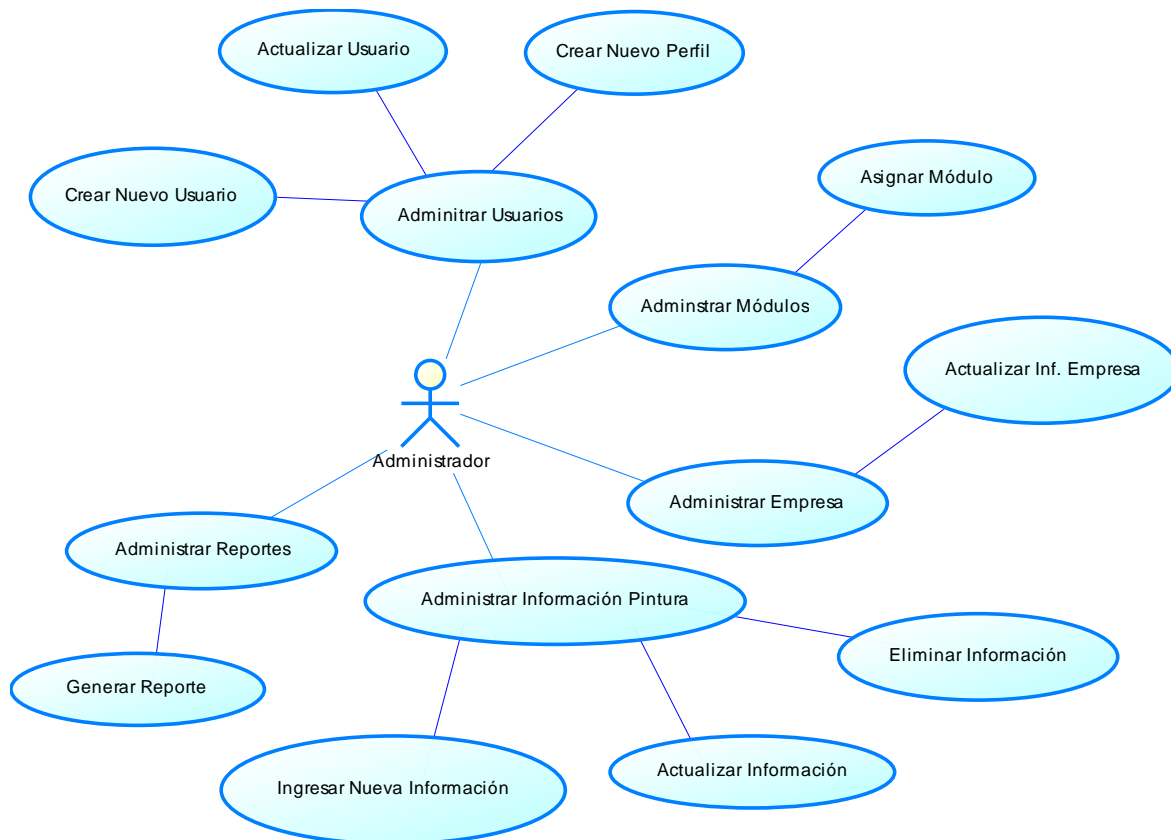
## 3.8 ANÁLISIS DE REQUERIMIENTOS

### 3.8.1 IDENTIFICACIÓN DE ACTORES

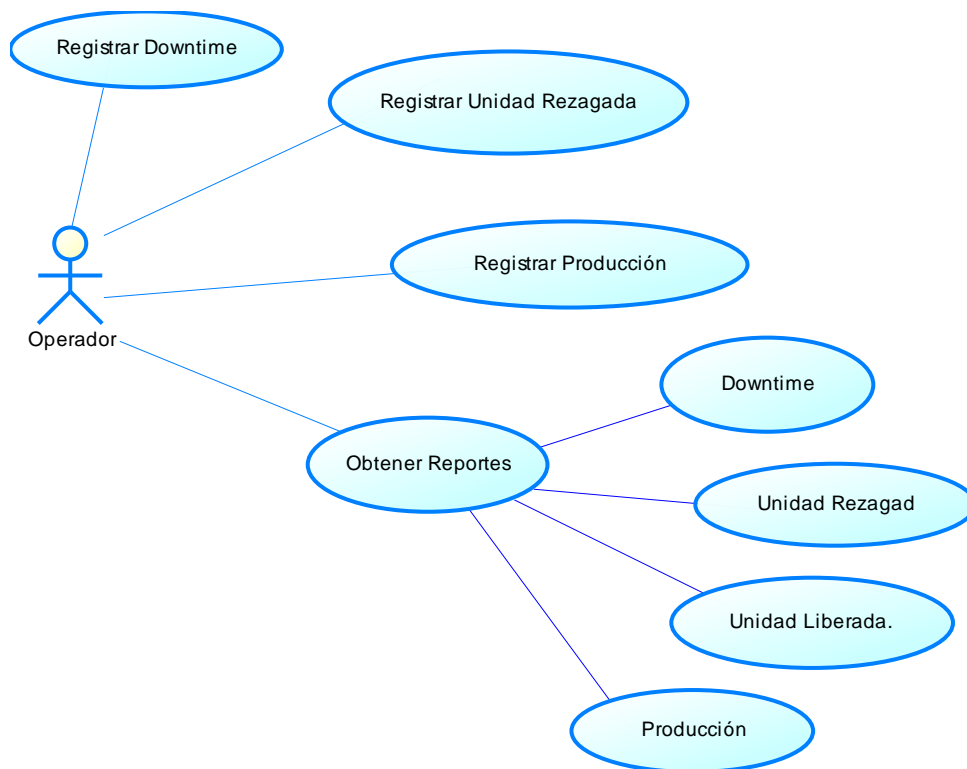
Actores	Actividades
Administrador	Administrar Usuarios Administrar Módulos Administrar Empresa Administrar Información Pintura Administrar y Visualizar Reportes
Operador	Registrar Información (Producción, Downtime, Unidades rezagadas) Obtener Reportes Generales
Invitado	Obtener Reporte General

## 3.9 DISEÑO

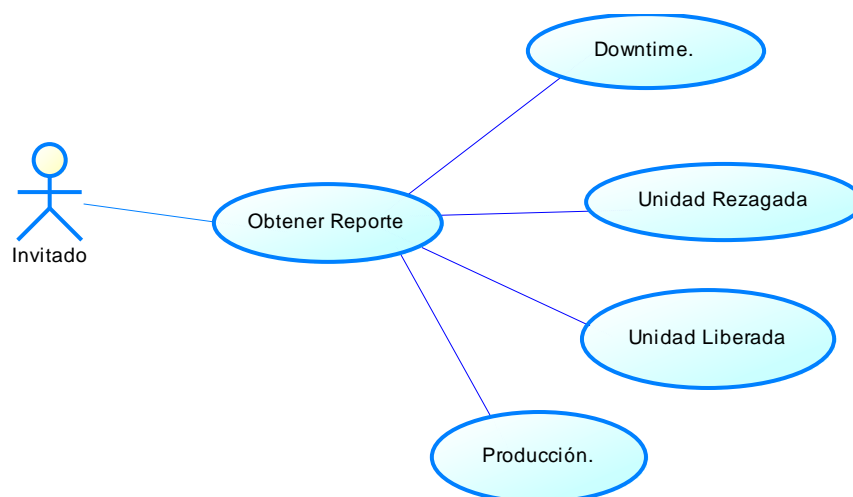
### 3.9.1 DIAGRAMA ADMINISTRADOR DEL SISTEMA



### 3.9.2 DIAGRAMA DEL USUARIO OPERADOR



### 3.9.3 DIAGRAMA DEL USUARIO INVITADO



### 3.9.4 DICCIONARIO DE CASOS DE USO

#### ADMINISTRACIÓN DE USUARIOS

Nombre	Acción	Reacción
Registrar usuario nuevo	Registrar usuario que va a realizar el registro de datos.	Usuario nuevo registrado
Actualizar datos del usuario	Modificar datos el usuario existente.	Datos actualizados.
Crear un nuevo perfil	Crear un nuevo perfil dentro del sistema para un usuario.	Perfil nuevo creado
Actualizar datos del nuevo perfil	Modificar datos del nuevo perfil.	Datos actualizados.
Asignar permisos a los usuarios	Conceder permisos a los diferentes módulos del sistema.	Módulo asignado.

#### ADMINISTRACIÓN DE MÓDULOS

Nombre	Acción	Reacción
Módulos Sistema	Asignar el módulo al que un usuario puede acceder con su perfil.	Módulo asignado.

#### ADMINISTRACIÓN DE EMPRESA

Nombre	Acción	Reacción
Actualizar empresa	Actualizar la información interna de la empresa.	Datos Actualizados.

--	--	--

### ADMINISTRAR INFORMACIÓN PINTURA

Nombre	Acción	Reacción
Registrar Información	Registrar la información necesaria para cada una de las áreas de la planta pintura.	Datos Registrados.
Actualizar Información	Actualizar información de la planta pintura si fuera el caso.	Datos Actualizados.
Eliminar Información	Realizar una limpieza de información de la base de datos que se encuentra obsoleta.	Datos Eliminados.

### DOWNTIME

Nombre	Acción	Reacción
Registrar Downtime	Registrar datos de Downtime de la planta pintura.	Datos registrados.

### UNIDADES REZAGADAS

Nombre	Acción	Reacción
Registrar Unidad rezagada	Registrar datos de Unidades Rezagadas de la planta pintura.	Datos Registrados.

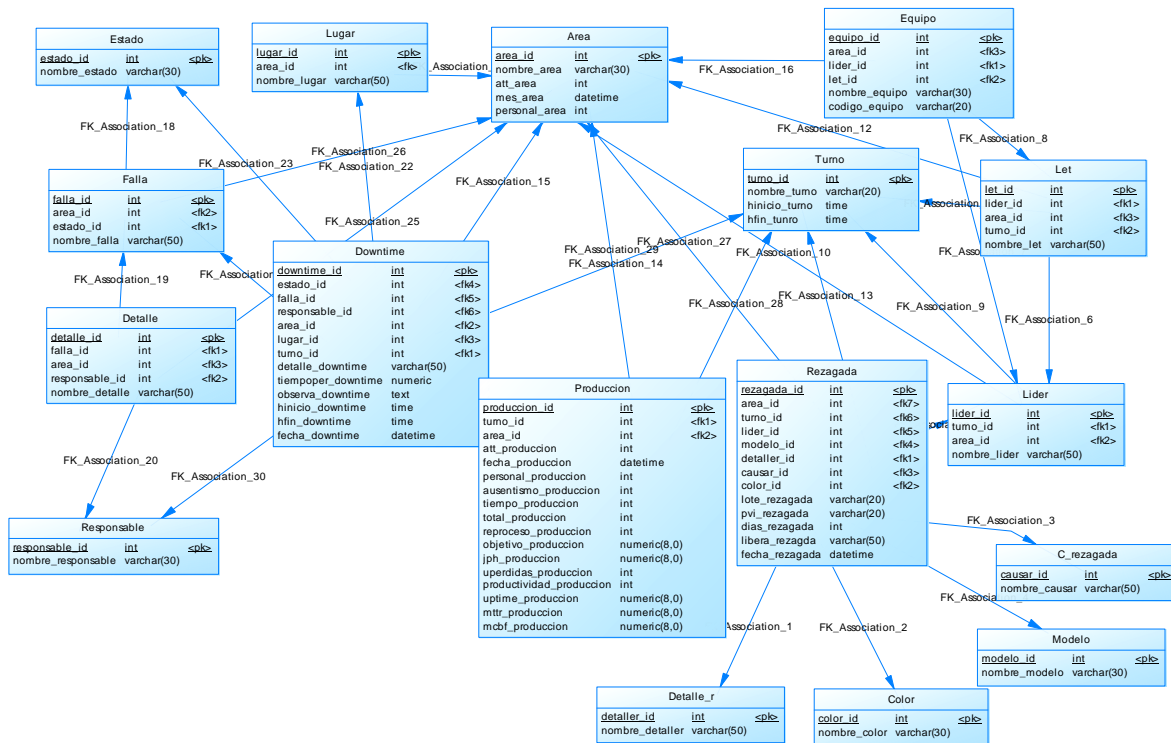
**PRODUCCIÓN**

Nombre	Acción	Reacción
Registrar Producción	Registrar datos de producción de la planta pintura.	Datos Registrados.

**CONSULTAS Y REPORTE**

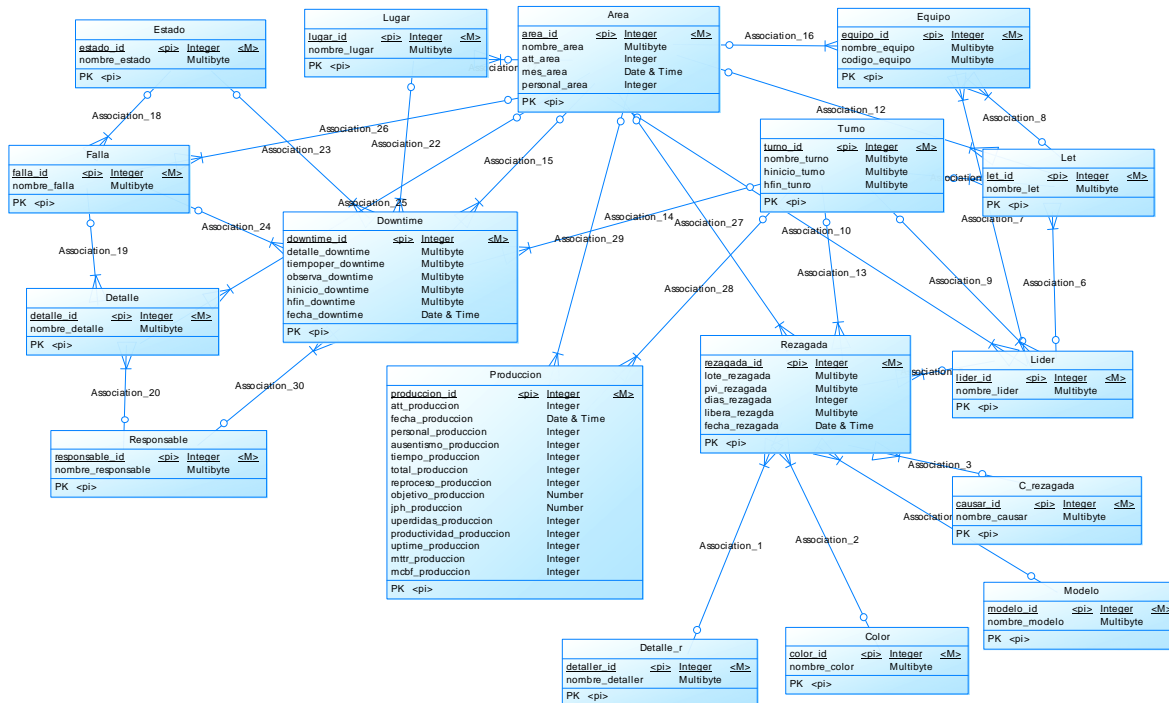
Nombre	Acción	Reacción
Consulta por criterio	Consulta de downtime, unidades rezagadas y producción, de acuerdo a criterios solicitados por el usuario.	Mostrar consulta, Descargar Reporte.

### 3.9.5 MÓDELO FÍSICO

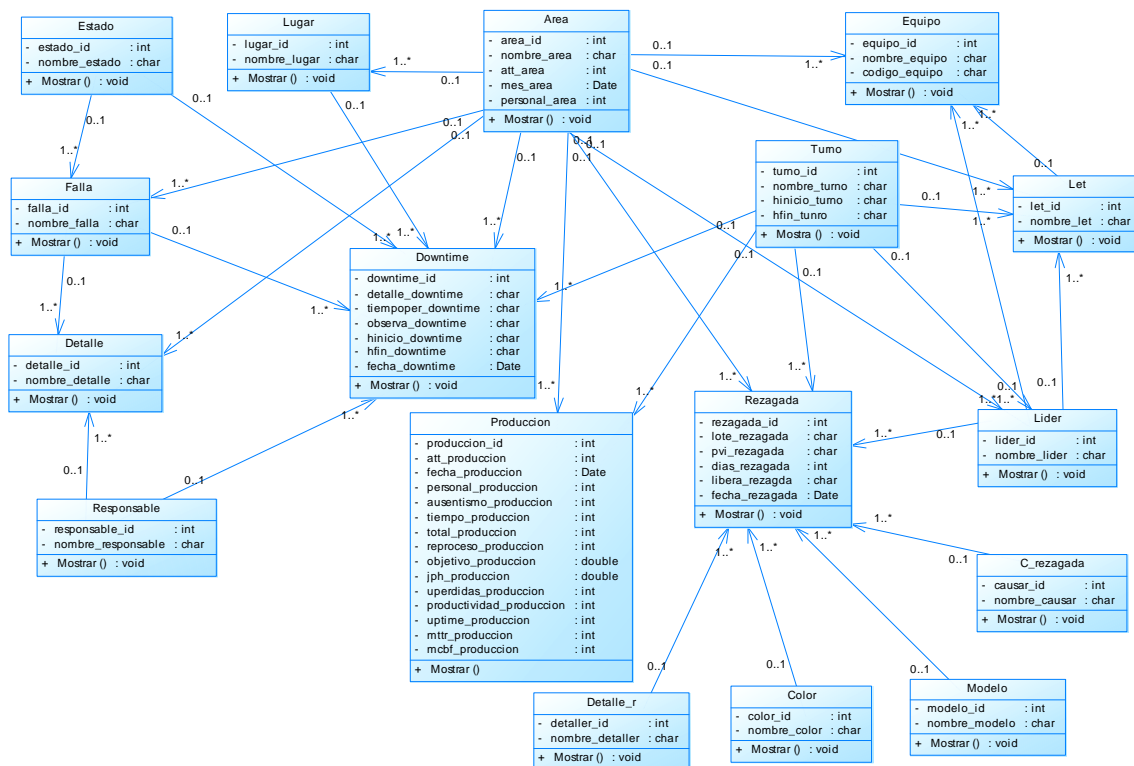




### 3.9.6 MÓDELO LÓGICO

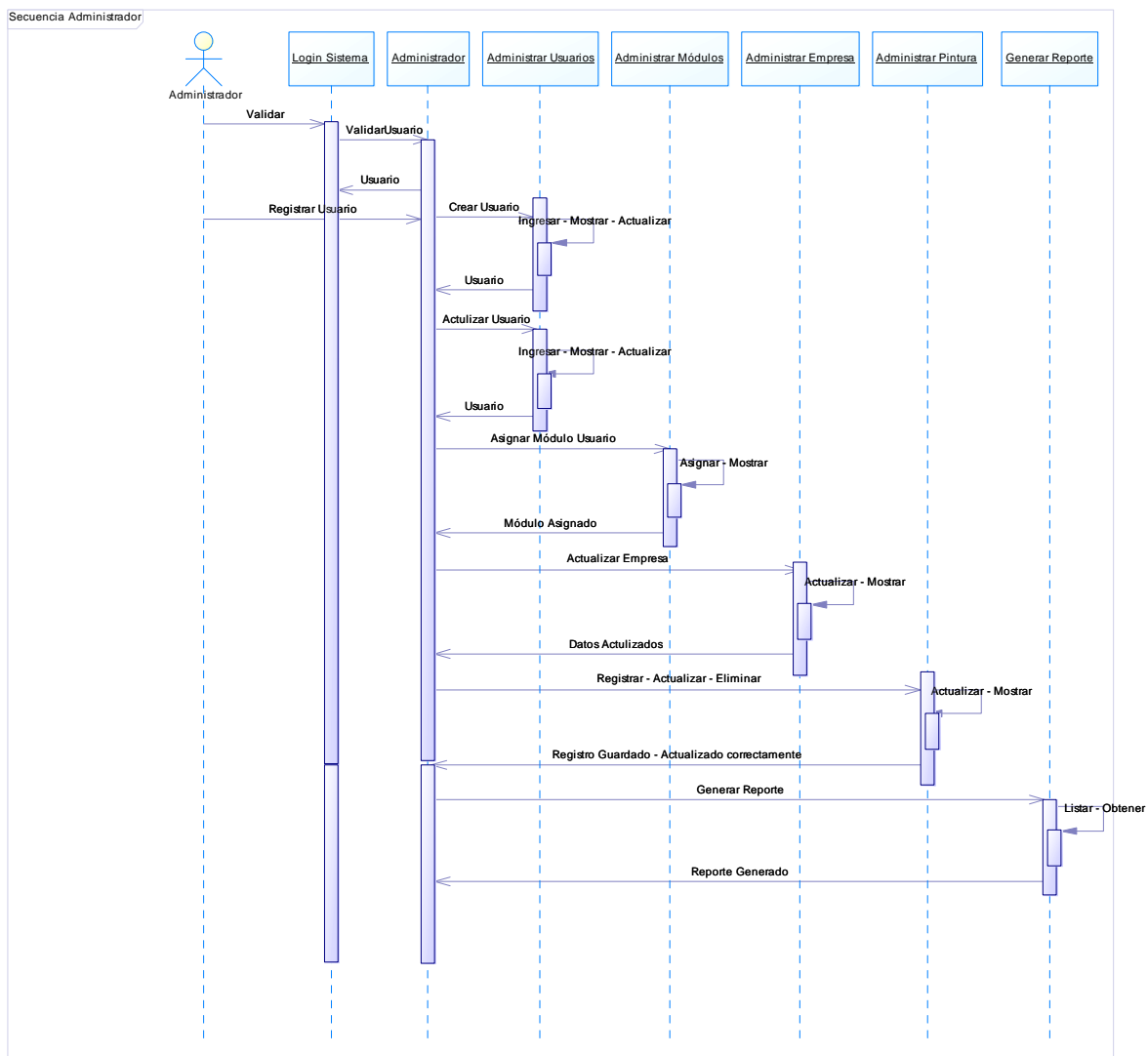


### 3.9.7 DIAGRAMA DE CLASES

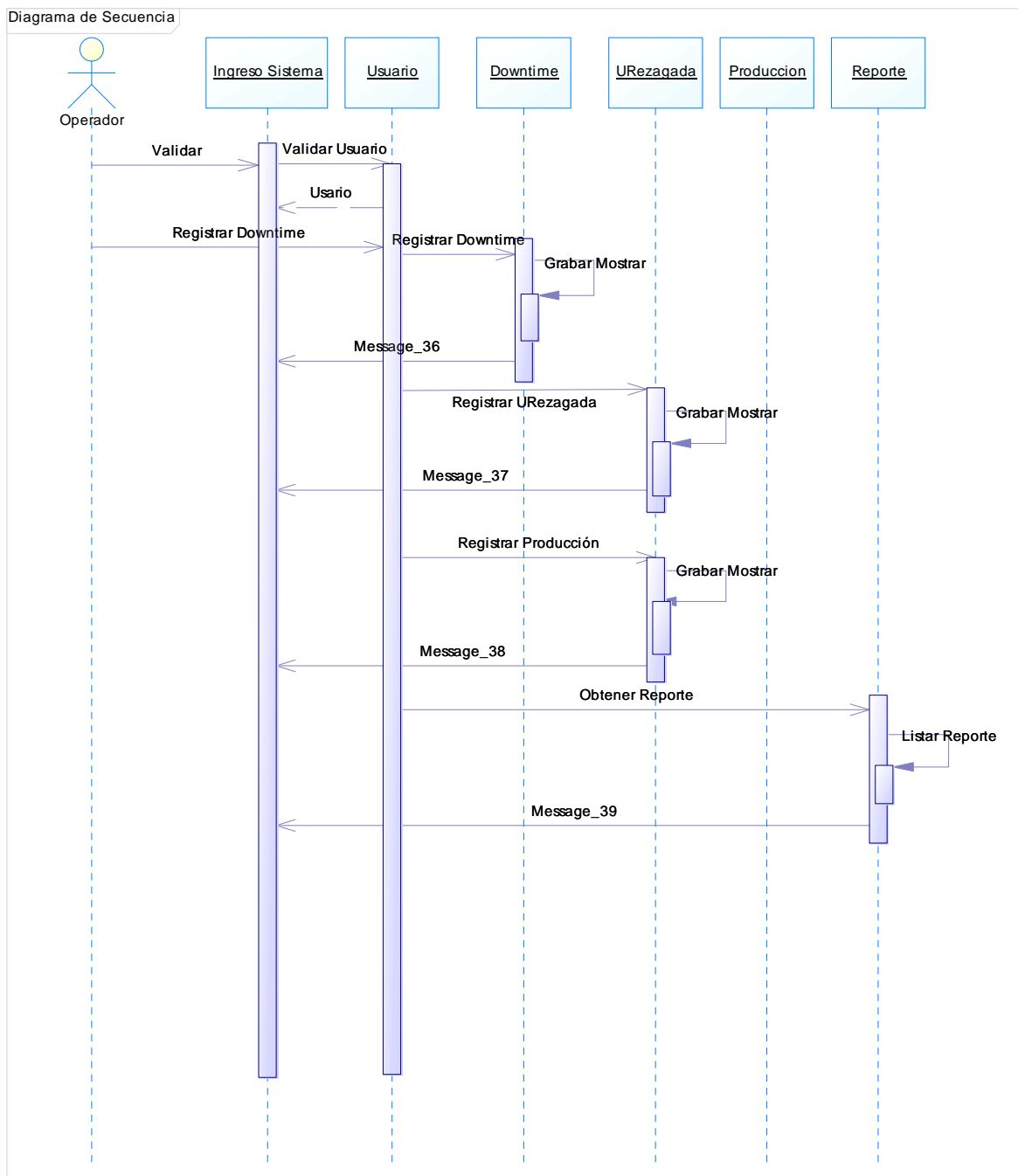


### 3.10 DIAGRAMA DE SECUENCIAS

#### 3.10.1 DIAGRAMA ADMINISTRADOR DEL SISTEMA

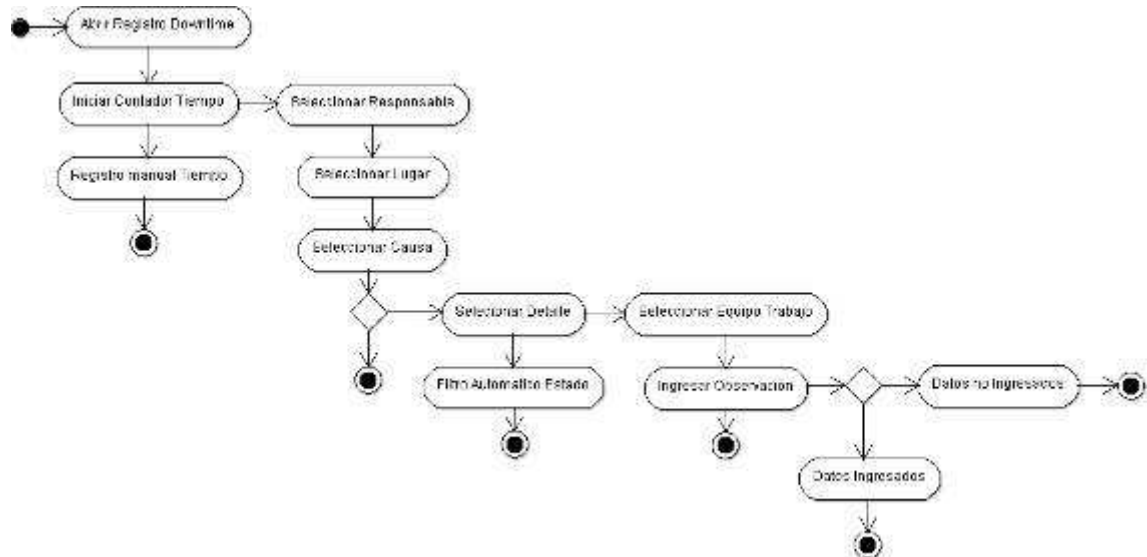


### 3.10.2 DIAGRAMA DE USUARIO OPERADOR

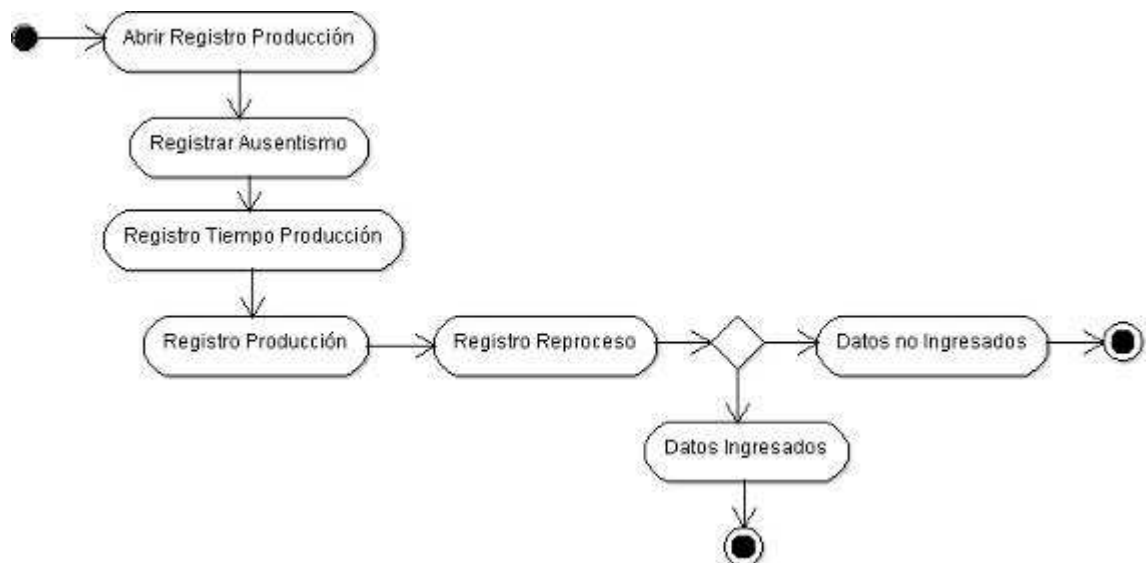


### 3.11 DIAGRAMA DE ACTIVIDADES

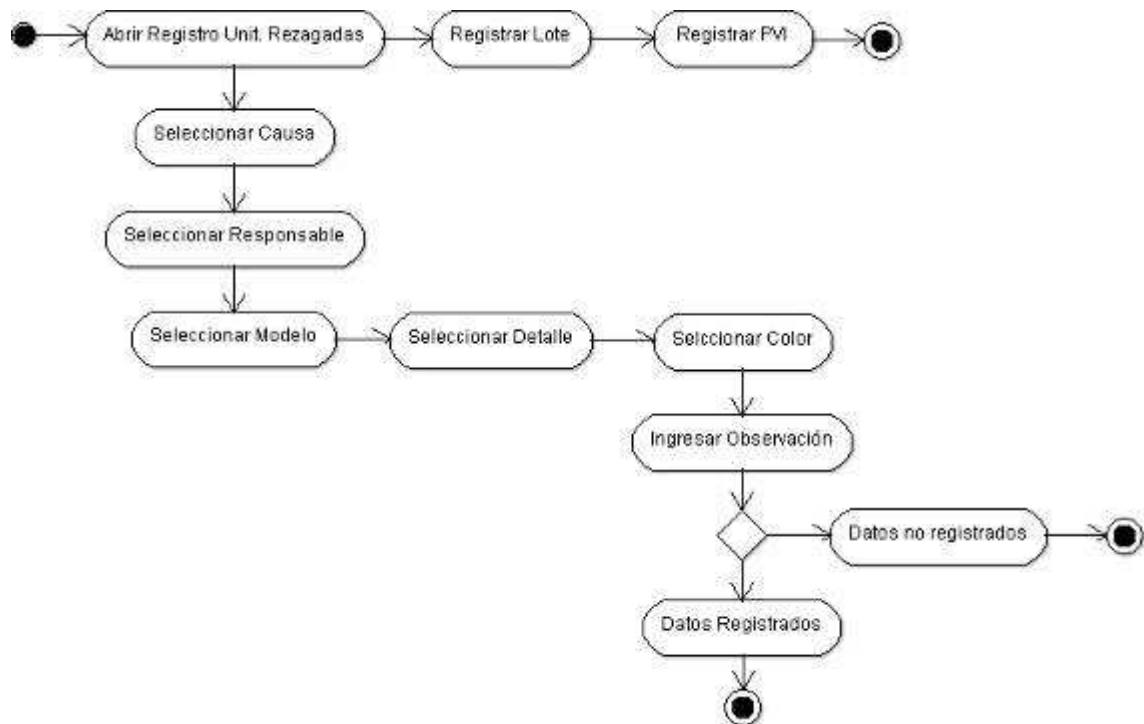
#### 3.11.1 DIAGRAMA DE REGISTRO DE DOWNTIME



#### 3.11.2 DIAGRAMA DE REGISTRO DE PRODUCCIÓN



### 3.11.3 DIAGRAMA DE REGISTRO DE UNIDADES REZAGADAS



## 3.12 COSTRUCCIÓN

### 3.12.1 BACK END

Conexión a la Base de Datos.

```

class Connection {
    private $host = DB_HOST;
    private $user = DB_USER;
    private $password = DB_PASSWORD;
    private $database = DB_NAME;
    var $connection;
    var $connectionMessage;
    function __construct()
    {
        if ($this->connection = $this->connect()) {
            $this->connectionMessage = "Conected";
            return $this->select_db();
        } else {
            return $this->connectionMessage;
        }
    }
}
function connect()
{
    $link = mysql_connect($this->host, $this->user, $this->password);

```

```

        if ($link) {
            return $link;
        } else {
            $this->connectionMessage = "We are updating our server, please try
later";
            return false;
        }
    }
    function select_db()
    {
        if (mysql_select_db($this->database)) {
            $this->connectionMessage = "Connected to Database";
        } else {
            $this->connectionMessage = "We are updating our database, please
try later";
        }
        return $this->connection;
    }
    function disconnect()
    {
        mysql_close();
        return true;
    }
    function __destruct() {
        return $this->disconnect();
    }
}

```

## Class Menú

```

class Menu extends TableBaseClass {
    private $id;
    private $modulo_id;
    private $menu_id;
    private $pagina;
    private $tipo;
    private $nombre;
    private $descripcion;
    private $codigo;
    private $orden;
    public $arrOpciones = array();
    private $arrCategory;
    public function __construct()
    {
        $this->tableName = "menu";
        $this->idField = "id";
        return parent::__construct();
    }
    public function viewByLevel($parentId=0, $level=0, $moduloId=0, $rolId=1) {
        $level++;
        $sql_tree = "SELECT id, nombre, codigo
                    FROM $this->tableName
                    WHERE menu_id = $parentId";
        if($moduloId) {
            $sql_tree .= " AND modulo_id = $moduloId";
        }
        if($rolId != 1) {

```

```

$clsDatRolMenu = new RolMenu();
$clsDatRolMenu->whereClause = "rol_id = $rolId";
$arrRolesMenu = $clsDatRolMenu->search();
if(is_array($arrRolesMenu)) {
    foreach($arrRolesMenu as $rolMenu) {
        $arrMenusActivos[] = $rolMenu["menu_id"];
    }
    $menusActivos = implode(", " , $arrMenusActivos);
}
if($menusActivos) {
    $sql_tree .= " AND id IN ($menusActivos)";
} else {
    $sql_tree .= "AND 0";
}
}
$res_tree = mysql_query($sql_tree, $this->connection);
while ($row = mysql_fetch_assoc($res_tree)) {
    $nextIndex = count($this->arrOpciones);
    $this->arrOpciones[$nextIndex] = $row;
    $this->arrOpciones[$nextIndex]["level"] = $level;
    $this->viewByLevel($row["id"], $level, $moduloId, $rolId);
}
return $this->arrOpciones;
}
function tree($parentId=0, $level=0)
{
    $sql_tree = "SELECT id, nombre, pagina, modulo_id, codigo
                FROM $this->tableName
                WHERE menu_id = $parentId";
    $res_tree = mysql_query($sql_tree, $this->connection) or
die(mysql_error());
    while ($row = mysql_fetch_assoc($res_tree)) {
        $nextId = count($this->arrOpciones) + 1;
        $this->arrOpciones[$nextId] = $row;
        $this->arrOpciones[$nextId]["level"] = $level;
        $this->arrOpciones[$nextId]["opciones"] = $this->subTree($row["id"],
$level);
    }
    return $this->arrOpciones;
}
function subTree($parentId=0, $level = 0)
{
    $level++;
    $sql_tree = "SELECT id, nombre, pagina, modulo_id, codigo
                FROM $this->tableName
                WHERE menu_id = $parentId";
    $res_tree = mysql_query($sql_tree, $this->connection) or
die(mysql_error());
    $i=1;
    while ($row = mysql_fetch_assoc($res_tree)) {
        $arr_categories[$i] = $row;
        $arr_categories[$i]["level"] = $level;
        $arr_categories[$i]["opciones"] = $this->subTree($row['id'],
$level);
        $i++;
    }
    return $arr_categories;
}
}

```



```
}

```

### Script de Base de Datos

```
CREATE TABLE IF NOT EXISTS `area` (
  `area_id` int(11) NOT NULL AUTO_INCREMENT,
  `nombre_area` varchar(50) NOT NULL,
  `att_area` int(11) NOT NULL,
  `mes_area` varchar(50) NOT NULL,
  `personas_area` int(11) NOT NULL,
  `user1` date DEFAULT NULL,
  `user2` int(11) DEFAULT NULL,
  `user3` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`area_id`),
  KEY `AI_area_id` (`area_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=13 ;
```

```
CREATE TABLE IF NOT EXISTS `responsable` (
  `responsable_id` int(11) NOT NULL AUTO_INCREMENT,
  `nombre_responsable` varchar(50) NOT NULL,
  `user1` int(11) DEFAULT NULL,
  `user2` date DEFAULT NULL,
  `user3` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`responsable_id`),
  KEY `AI_responsable_id` (`responsable_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=12 ;
```

```
CREATE TABLE IF NOT EXISTS `downtime` (
  `downtime_id` int(11) NOT NULL AUTO_INCREMENT,
  `detalle_downtime` varchar(50) NOT NULL,
  `tiempoperd_downtime` float NOT NULL,
  `observa_downtime` text NOT NULL,
  `inicio_downtime` varchar(50) NOT NULL,
  `hfin_downtime` varchar(50) NOT NULL,
  `fecha_downtime` date NOT NULL,
  `user1` int(11) DEFAULT NULL,
  `user2` date DEFAULT NULL,
  `user3` varchar(50) DEFAULT NULL,
  `turno_id` int(11) NOT NULL,
  `area_id` int(11) NOT NULL,
  `responsable_id` int(11) NOT NULL,
  `lugar_id` int(11) NOT NULL,
  `falla_id` int(11) NOT NULL,
  `estado_id` int(11) NOT NULL,
  `equipo_id` int(11) NOT NULL,
  PRIMARY KEY (`downtime_id`),
```

```

KEY `AI_downtime_id` (`downtime_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1213 ;

```

```

CREATE TABLE IF NOT EXISTS `rezagada` (
  `rezagada_id` int(11) NOT NULL AUTO_INCREMENT,
  `lote_rezagada` varchar(50) NOT NULL,
  `pvi_rezagada` varchar(50) NOT NULL,
  `dias_rezagada` int(11) DEFAULT NULL,
  `responsable_id` int(11) NOT NULL,
  `fecha_rezagada` date NOT NULL,
  `status_id` int(11) NOT NULL,
  `user1` int(11) DEFAULT NULL,
  `user2` date DEFAULT NULL,
  `user3` varchar(50) DEFAULT NULL,
  `modelo_id` int(11) NOT NULL,
  `area_id` int(11) NOT NULL,
  `turno_id` int(11) NOT NULL,
  `color_id` int(11) NOT NULL,
  `detaller_id` int(11) NOT NULL,
  `causar_id` int(11) NOT NULL,
  `lider_id` int(11) NOT NULL,
  `observa_rezagada` varchar(100) NOT NULL,
  `estado_rezagada` int(11) NOT NULL,
  PRIMARY KEY (`rezagada_id`),
  KEY `AI_rezagada_id` (`rezagada_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

```

### 3.13 PRUEBAS

#### 3.13.1 ESTRATEGIA DE PRUEBAS

**Funcionalidad.-** El sistema cumple con los requisitos solicitados, permite administrar la información de la planta pintura la cual es necesaria para el registro información en cada una de las áreas, así como también permite registrar la pérdida de tiempo (Downtime), registro de producción y registro de unidades rezagadas.

**Integridad.-** La integridad esta descrito por el correcto funcionamiento de los diferentes módulos que tiene el sistema, los cuales se describirán a continuación:

- Administración General.- Registro de usuario, perfiles y actualización de datos.
- Seguridades.- Ingreso de nuevos usuarios, perfiles.
- Editar información.- Actualizar la información del usuario administrador e información de la empresa.
- Administración Pintura.- Registro de información de la planta pintura para el registro de datos desde las diferentes áreas.
- Downtime.- Registro de tiempo perdido en cada una de las áreas.
- Producción.- Registro de producción de la planta pintura.
- Unidades rezagadas.- Registro de unidades con defectos las mismas que son apartadas de la línea de producción.
- Reportes.- Obtener un reporte de los datos registrados por los operadores en la línea de producción.

**Sobrecarga.**- El funcionamiento adecuado de la Base de Datos la misma que es la parte fundamental del sistema, cumple con las reglas establecidas por los usuarios.

**Tensión.**- El sistema mantiene un correcto funcionamiento de las diferentes terminales.

**Ergonomía.**- Los diferentes módulos del sistema funcionan correctamente de forma unitaria y en conjunto. Es decir globalmente el sistema cumple con las especificaciones del usuario.

#### **Creación de Nuevos Usuarios:**

Pres-requisitos:

- Para validar que los datos se guarden de acuerdo al requerimiento se realizó la carga de tres usuarios que se puede observar a continuación. La persona encargada de realizar la creación de nuevos usuarios es el administrador del sistema.

Nombre / Apellidos / Email	Estado
Jorge Quinchusa jorge.quinchusa@gmail.com	Activo
Carlos Mesa cmesa@gmail.com	Activo
Invitado invitado@gmail.com	Activo

### Creación de Nuevos Perfiles:

- Se creó nuevos perfiles, para la asignación a los nuevos usuarios del sistema, el usuario que puede realizar el requerimiento es el Administrador del sistema.

Nombre	Descripción
Administrador	Registra peres, producción, unidades rezagadas en el sistema
Invitado	Generar reporte

### Asignación de Módulos:

- Los siguientes roles fueron creados previamente por el Administrador del sistema, los mismos que serán asignados a los nuevos usuarios.



### Actualización de Usuarios:

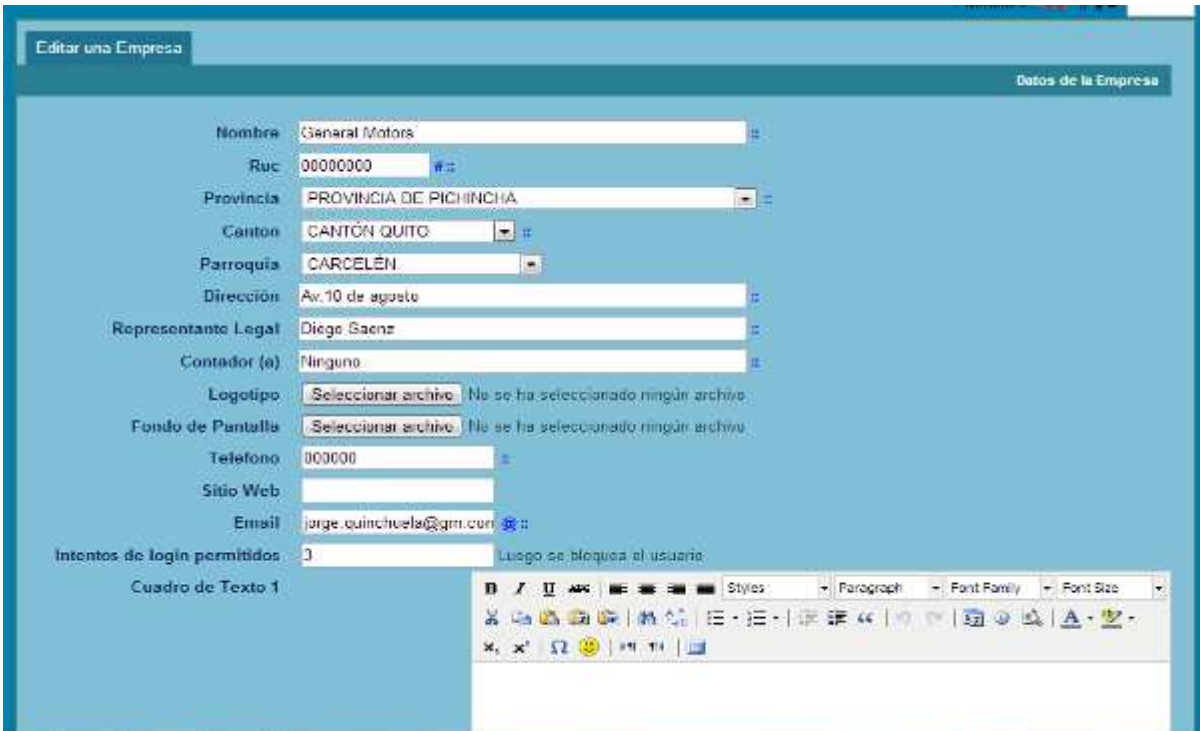
- Se realizó una prueba de actualización de información para un usuario.

The screenshot shows the 'Crear un Usuario' form with the following fields:

- Area: -Seleccione una-
- Nombre:
- Apellidos:
- Cédula:
- Estado Civil:
- Genero:
- Instrucción:
- Fecha Nacimiento:
- Email:
- Password:
- Provincia: -Seleccione una provincia-
- Canton: -Seleccione un Cantón-
- Parroquia: -Seleccione una Parroquia-
- Dirección:
- Teléfono:
- Extensión:
- Fotografía:  (No se ha seleccionado ningún archivo)
- Perfil:  (Opciones: Operador, Invitado, Invitado)

### Actualización de Empresa:

- Prueba de actualización de datos de la empresa, la información sera modificada unicamente por el administrador del sistema.



The screenshot shows a web application interface for editing company data. The page has a blue header with the title "Editar una Empresa" on the left and "Datos de la Empresa" on the right. The main content area contains a form with the following fields:

- Nombre:** General Motors
- Ruc:** 00000000
- Provincia:** PROVINCIA DE PICHINCHA
- Canton:** CANTÓN QUITO
- Parroquia:** CARCELÉN
- Dirección:** Av. 10 de agosto
- Representante Legal:** Diego Saenz
- Contador (a):** Ninguno
- Logotipo:** Selecionar archivo. No se ha seleccionado ningún archivo.
- Fondo de Pantalla:** Selecionar archivo. No se ha seleccionado ningún archivo.
- Telefono:** 000000
- Sitio Web:**
- Email:** jorge.quinchuela@gm.com
- Intentos de login permitidos:** 3. Luego se bloquea el usuario.

At the bottom of the form, there is a text box labeled "Cuadro de Texto 1" and a rich text editor toolbar with various icons for text formatting and alignment.

## Reportes

- Prueba donde se puede observar la forma en que se presentan los reportes del sistema, previamente la información fue registrada por los usuarios operadores en la línea de producción de los diferentes puntos de control.

Reporte Downtime Area - Fecha[Inicio - Fin] - Turno

Area: CLPO

Turno: 1 TURNO

Fecha[Inicio]: 2014-01-31

Fecha[Fin]: 2014-01-31

Exportar

Obtener reporte

Mostrar: 10 registros

Buscar:

ID	Area	Turno	Problema	Responsable	Fecha	Causa	Tipo	Estado	Primer	Segundo	Tercero	Inicio	Fin	Operario
283	CLPO	1 TURNO	18	PRODUCCION	DESCARGA	LINIA LLENA	FPO DESCARGA	BLOCKEADO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
284	CLPO	1 TURNO	1803	PRODUCCION	DESCARGA	LINIA LLENA	FPO DESCARGA	BLOCKEADO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
332	CLPO	1 TURNO	784	PROCESOS	LUBRICA	LUBRICA DE UNIDADES	DESALEBRADA	NINGUNO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
333	CLPO	1 TURNO	807	PRODUCCION	CARGA	LINIA LLENA	FPO DESCARGA	BLOCKEADO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
338	CLPO	1 TURNO	1067	PRODUCCION	DESCARGA	LINIA LLENA	FPO DESCARGA	BLOCKEADO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
348	CLPO	1 TURNO	75	PRODUCCION	CARGA	SECA HORNO 01	SECA ELECTRICA	NINGUNO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
355	CLPO	1 TURNO	890	MANTENIMIENTO	CARGA	SECA HORNO 01	TRABAJANDO	NINGUNO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
365	CLPO	1 TURNO	1002	MANTENIMIENTO	CARGA	SECA HORNO 01	SECA ELECTRICA	NINGUNO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
373	CLPO	1 TURNO	813	MANTENIMIENTO	CARGA	SECA TRANSFERA CARGA	SECA ELECTRICA	NINGUNO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31
380	CLPO	1 TURNO	82	MANTENIMIENTO	OTROS	OPERAR EL BLO	SECA DE COLOCACION	NINGUNO	NINGUNO	NINGUNO	NINGUNO	01/31/14	01/31/14	2014-01-31

Showing 1 to 10 of 30 entries

Anterior 1 2 3 Siguiente Ultimo

## MANUAL DE USUARIO

### Diseño de la Página Principal

El diseño de la Página Principal es una interfaz amigable, para que el usuario pueda encontrar las cosas con facilidad.





## Módulo de Administración de General de la Planta de Pintura GM

El módulo de administración general permite manejar:

- Asignar módulos a perfiles creados.
- Controlar las seguridades del sistema
- Editar información de usuario administrador
- Editar información de empresa



Para crear seguridades y asignar módulos del sistema es necesario haber completado los siguientes pasos:

1. Crear de un Perfil
2. Crear un nuevo usuario

### Crear un Nuevo Perfil:

Iniciar como usuario administrador del sistema, abrir la pestaña de Administración que se encuentra en la parte izquierda, seguido hacer click en la pestaña de Seguridades / Perfiles / Crear ó Listar





Ingresamos la información solicitada:

- Nombre.- Es el nombre con el cual se maneja dentro del sistema.
- Descripción.- Es una pequeña descripción del nuevo perfil a crear.

Para realizar la verificación de la información ingresada ir a la pestaña de Seguridades / Perfiles / Listar.

ID	Nombre	Descripcion	Editar	Eliminar
1	Super Administrador	Es el super administrador del Sistema		
2	Administrador de Empresa	Es la persona que administra una empresa y tiene todos los permisos con esa empresa		
3	Operador	Registra parax, producción, unidades recuperadas en el sistema.		
4	Analista	Generar reporte		
5	Proveedor	Proveedor de materiales		

Se puede observar los diferentes perfiles creados dentro del sistema, en la ventana es posible Editar la información ingresada  ó Eliminar uno de los perfiles . Es importante recordar que al eliminar un perfil es necesario previamente eliminar los usuarios asignados al mismo. Para eso ir a la pestaña de Seguridades / Usuarios / Listar, buscar el usuario a eliminar.

### Crear un Nuevo Usuario:

Para crear un nuevo usuario es necesario haber completado el anterior paso, seguir los siguientes pasos:

1. Revisar la información a ingresar
2. Conocer a que area pertenece
3. Verificar el perfil asignar al nuevo usuario.

Modulo Seguridades Editar Usuario Editar Empresa Cerrar Sesión

Pantalla # 114 / a #

Crear un Usuario

Datos del Usuario

Área --Seleccione una--

Nombres

Apellidos

Cédula

Estado Civil Soltero

Genero Masculino

Instrucción Ninguna

Fecha Nacimiento

Email

Password

Provincia --Seleccione una provincia--

Canton --Seleccione un Cantón--

Parroquia --Seleccione una Parroquia--



Dirección

Telefono

Extensión

Fotografía Seleccionar archivo Ningún archivo seleccionado

Perfil Invitado

Para verificar la información ingresada ir a la pestaña de Seguridades / Usuarios / Listar. En la ventana es posible Editar la información ingresada  ó Eliminar el usuario .

Modulo Seguridades Editar Usuario Editar Empresa Cerrar Sesión





















Pantalla # 115 / a #

Listado de Usuarios Atras

Registros Actuales

Mostrar 10 registros

Buscar

#	Nombre	Apellidos	Email	Estado	Editar	Eliminar
3	Carlos	Ortiz	carlos@pr.com	Activo		
4	Invitado	Invitado	invitado@pr.com	Activo		
22	JORGE	QUINCHULEA	jorge.quinchulea@pr.com	Activo		
23	Elio	Elio	elio	Activo		
24	Edroto	Edroto	edroto	Activo		
25	Franco	Franco	franco	Activo		
26	Pitner	Pitner	pitner	Activo		
27	Sesla 20	Sesla 20	sesla20	Activo		
28	Sesla 60	Sesla 60	sesla60	Activo		
29	Patorico	Patorico	patorico	Activo		

Showing 1 to 10 of 11 entries

Primero Anterior 1 2 Siguiente Ultimo

## Asignación de Módulos a Perfiles

Permite controlar el acceso a los diferentes módulos que el sistema posee.

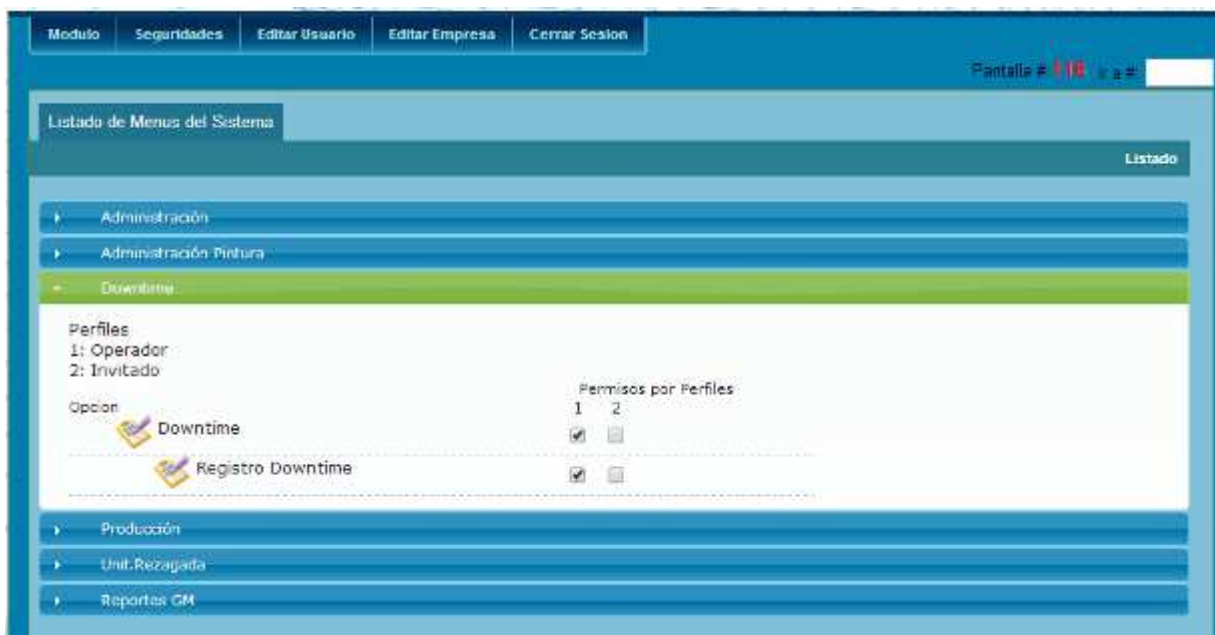
Pestaña de Administración.- Permite crear un menu semejante al perfil de administrador general, seguir los siguientes pasos:

1. Verificar el el usuario este creado.
2. Ingresar en otro navegador con la sesión del usuario creado.
3. Asignar los permisos respectivos.



Asignación de módulos a los perfiles creados con anterioridad.

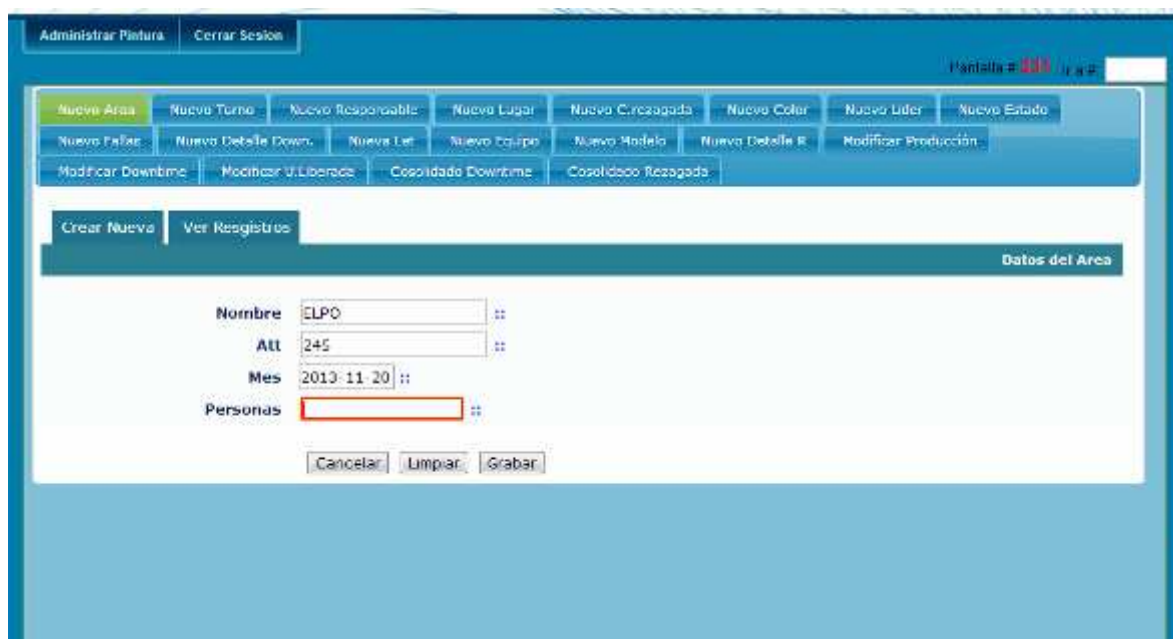
Para asignar un módulo hay que marcar el recuadro así como se muestra en la siguiente ventana. Todos los usuarios asignados al perfil pueden tener acceso al módulo.



### Módulo de Administración de información de la Planta de Pintura GM

Muestra la información interna de la planta de pintura la misma que se utilizara en los diferentes módulos.


Antes de realizar el registro es necesario conocer los datos a guardar, ya que esto se refleja directamente en los módulos que utiliza el usuario operador.



**Nota:** El usuario que puede realizar el registro de nueva información es el Administrador del sistema. Tomar en cuenta que todos los campos son obligatorios para la carga de nueva información.

### Actualizar datos ingresado:

Para realizar la modificación de información se debe seguir los siguientes pasos:

- Click sobre la pestaña Ver Registros.
- Muestra el listado de información que se encuentra guardada.
- Click sobre el icono  .
- Realizar la modificaciones necesarias y guardar los cambios.



The screenshot shows the 'REGISTROS ACTUALES GM' interface. At the top, there are buttons for 'Administrar Pastura' and 'Cerrar Sesión'. The main title is 'REGISTROS ACTUALES GM'. Below the title, there are tabs for 'Listado de Areas' and 'Atras'. A search bar is present with the text 'Mostrar 10 registros' and a 'Buscar' field. The table below has columns for 'Id', 'Nombre', 'Att', 'Mes', and 'Personas'. Each row has an 'Editar' icon (a pencil) and a 'Eliminar' icon (a red circle with a minus sign).

Id	Nombre	Att	Mes	Personas
1	ELPO	270	NOVIEMBRE	20
2	FINESSE	280	NOVIEMBRE	0
3	ESMALTE	210	NOVIEMBRE	21
4	HONOR	280	NOVIEMBRE	76
5	PLASTICOS	210	NOVIEMBRE	10
6	STATUS 00	160	NOVIEMBRE	4
7	STATUS 00	180	NOVIEMBRE	0

Showing 1 to 7 of 7 entries. Page: 1 | Anterior | Siguiente | Último

Ventana donde se realiza la modificación de información.



The screenshot shows the 'MODIFICAR REGISTROS GM' interface. At the top, there are buttons for 'Administrar Pastura' and 'Cerrar Sesión'. The main title is 'MODIFICAR REGISTROS GM'. Below the title, there is a tab for 'Ver Registros'. A search bar is present with the text 'Datos de Area'. The form below has fields for 'Nombre', 'Att', 'Mes', and 'Personas'. At the bottom, there are 'Cancelar' and 'Grabar' buttons.


Nombre: FINESSE  
 Att: 280  
 Mes: NOVIEMBRE  
 Personas: 16

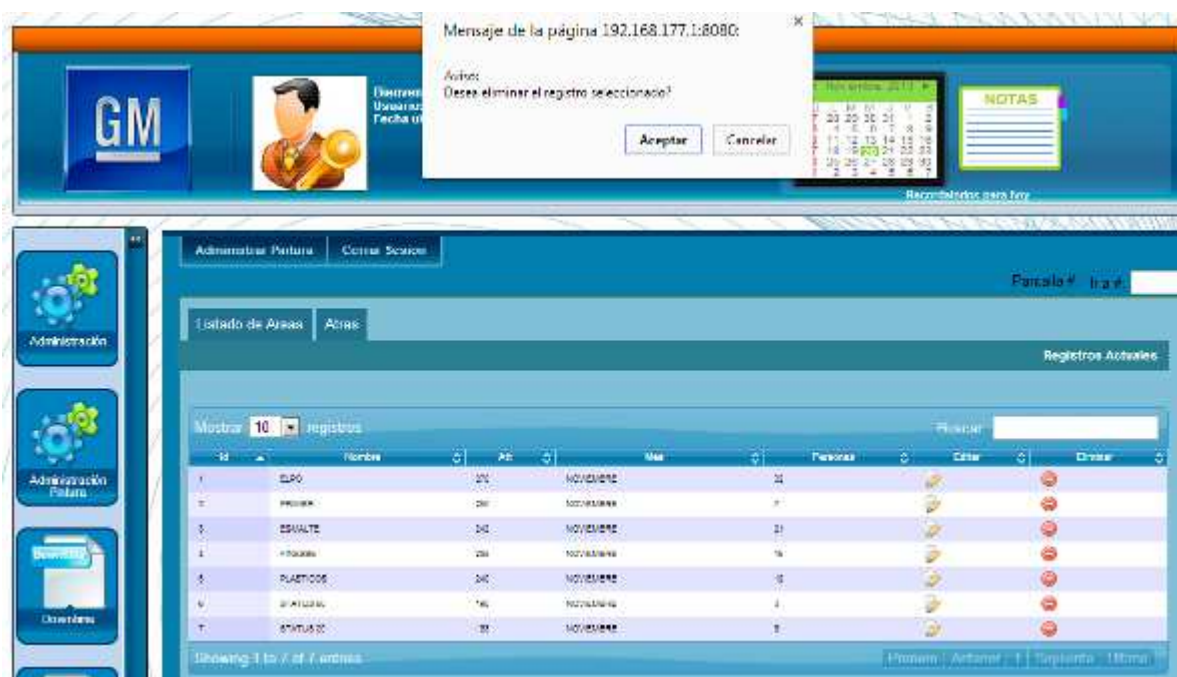
Cancelar Grabar

## Eliminar Información ingresada:

Tomar en cuenta que una vez que se elimina la información no es posible recuperar la misma.

Pasos para el borrado de información.


- Listar los datos guardados.
- Click sobre el icono , muestra un mensaje de confirmación, para borrar el registro.



## Actualizar registros de producción

La información que se modificara fue registrada desde la línea de producción.

Seguir los siguientes pasos:

- Abrir el módulo de **Administración Pintura**.
- Abrir la pestaña **Modificar Downtime**.
- Ingresar el **ID** del registro a modificar.
- Click sobre el boton **Buscar**.
- Click sobre el icono  .

- Validar la información que se va a cambiar.
- Realizar la modificación de información y guardar los cambios.

Datos obtenidos de la consulta realizada.



## Consolidado Downtime

Esta pestaña muestra información que fue registrada desde la línea de producción de los puntos de control de cada uno de los turnos. Esta ventana es solo de visualización.

## Consolidado de Unidades Rezagadas

En esta pestaña se puede observar la unidades rezagas existentes en cada una de la áreas. Las unidades que se muestran son de los diferentes turnos y de diferentes fechas si existieran. Esta ventana es solo de visualización.

Administración de Pintura Cerrar Sesión

Panel de 303 4 de 4

Nuevo Área Nuevo Turno Nuevo Responsable Nuevo Lote Nuevo Color Nuevo Causa Nuevo Detalle Nuevo Modelo Nuevo Responsable  
 Nuevo Fallos Nuevo Detalle Down. Nueva Lot. Nuevo Equipo Nuevo Modelo Nuevo Detalle R. Modificar Producción  
 Modificar Downtime Modificar U.Liberada Copiloteado Downtime Copiloteado Rezagada

### U. REZAGADAS ELPO

Mostrar 10 registros

Id	Área	Turno	Lote	Pvt	Color	Causa	Detalle	Modelo	Responsable
2	ELPO	2 TURNO	810H-AD	10499025	ROJO	DESAFORNO SOBRECARGADO	1. 201	GRAN APATA	MANTENIMIENTO

Mostrando 1 de 1 de 1 unidades

### U. REZAGADAS PRIMER

Mostrar 10 registros

Id	Área	Turno	Lote	Pvt	Color	Causa	Detalle	Modelo	Responsable
No existen registros									

Mostrando 0 de 0 de 0 unidades

### Módulo de Downtime

Permite el registro problemas que generan tiempo perdido en cada una de las áreas en los diferentes turnos dentro de la planta de pintura GM. Como paso previo al registro de downtime es necesario ingresar la información que se utilizara en los diferentes puntos de control, esto se lo realizara en el módulo de administración.

### Descripción de campos:

Ventana que utiliza el usuario para el registro de tiempo perdido dentro de la planta de pintura GM.

- **Turno (automático):** indica el turno actual en el cual se registrara la información, el usuario no puede cambiar esta opción.
- **Área (automático):** indica el área para el registro.
- **Fecha (automático):** indica la fecha actual del sistema.
- **H. Fin (automático):** indica la hora del sistema.
- **H. Inicio:** registra la hora a la que inicio un problema en la línea de producción. El campo se llena al momento de iniciar el cronometro de tiempo.
- **Responsable:** indica el responsable o causante de la pérdida de tiempo.
- **Lugar:** indica el lugar donde se registró el problema.
- **Causa:** registra la causa de la pérdida de tiempo.
- **Observación:** el usuario puede ingresar texto, realizando una observación.
- **Detalle (dependiente):** el campo depende de la selección de Causa, contiene un detalle del campo dependiente.
- **Equipo:** registra el código de equipo de trabajo, si fuese el causante de la pérdida de tiempo.

- **Estado (dependiente):** el campo depende de la selección de Detalle, indica si la causa produjo la detención de la línea de producción, lo que genera pérdida de unidades.
- **Botón Iniciar:** inicia el cronometro para la contabilización de tiempo perdido.
- **Botón Detener:** detiene el conteo de tiempo, el mismo que cambia de fondo al momento de iniciar el contador de tiempo.
- **T. Segundos:** indica el tiempo transcurrido en segundos.
- **T. Minutos:** indica el tiempo transcurrido en minutos el cual se registra en la base de datos.
- **Botón Limpiar:** regresa los campos a su estado original para iniciar un nuevo registro de datos.
- **Botón Actualizar:** actualiza la lista de registros.
- **Botón Guardar:** guarda la información en la base de datos.

**Nota:** la información utilizada para el registro de tiempo perdido es ingresado por el usuario administrador quien es el encargado de mantener la información actualizada en la base de datos.

### **Módulo de Producción**

Permite el registro de información de producción de cada uno de las áreas y en los diferentes turnos de la planta de pintura GM. Para el buen funcionamiento del módulo es necesario ingresar la información solicitada en módulo de Administración Pintura.

### **Descripción de campos:**

Ventana que utiliza el usuario para el registro de producción de la planta pintura.



- **Turno (automático):** indica el turno actual en el cual se registrara la información, el usuario no puede cambiar esta opción.
- **Área (automático):** indica el área para el registro.
- **Fecha (automático):** indica la fecha actual del sistema.
- **Tiempo (automático):** indica la hora del sistema.
- **Personas:** indica el número de personas por cada una de las áreas.
- **Ausentismo:** el valor por defecto es cero, pero se puede cambiar dependiendo al número de personas que esta ausentes.
- **Tiempo Producción:** registra el tiempo total de producción por cada turno.
- **Producción:** registra el número real de unidades producidas.
- **Objetivo (automático):** campo calculado.
- **JPH (automático):** indica las unidades por hora.
- **U. Perdidas (automático):** calcula las unidades perdidas.
- **Productividad (automático):** campo calculado el mismo que indica la productividad de la planta.
- **Uptime(%):** calcula el tiempo efectivo para la producción.
- **MTTR y MCBF:** campo procesado utilizando batch.

- **Botón Limpiar:** regresa los campos a su estado original para iniciar un nuevo registro de datos.
- **Botón Actualizar:** actualiza la lista de registros.
- **Botón Guardar:** guarda la información en la base de datos.

### Módulo de Unidades Rezagadas

Permite el registro de unidades con problemas que son causados por mal manejo de herramientas mecánicas o error humano. Lo que se desea es controlar y llevar un control del problema más frecuente en la planta.

### Descripción de campos:

Ventana utilizada por el usuario para el registro de unidades rezagadas de la planta pintura.

The screenshot shows a web application interface for recording delayed units in the GM paint plant. The interface is titled 'UNIDADES REZAGADAS PINTURA GM'. It features a header with the GM logo, a user profile section, and a calendar. The main form includes the following fields and controls:

- Turno:** 2 Turno (dropdown)
- Area:** ELPO (dropdown)
- Líder:** LIZANDRO JATVA (dropdown)
- Fecha:** 2013-11-12 (text input)
- Dias:** 0 (text input)
- Lote:** (text input)
- Pvi:** (text input)
- Causa:** -Seleccione una causa- (dropdown)
- Responsable:** -Seleccione una- (dropdown)
- Modelo:** -Seleccione una- (dropdown)
- Color:** -Seleccione una- (dropdown)
- Detalle:** -Seleccione una causa- (dropdown)
- Observación:** (text area)

At the bottom of the form, there are three buttons: 'Limpiar', 'Actualizar', and 'Grabar'. Below the form is a search bar and a table with columns for filtering: 'Id', 'Lider', 'Area', 'Turno', 'Lote', 'Pvi', 'Causa', 'Detalle', 'Modelo', and 'Responsable'. The table currently shows 'No existen registros'.

- **Turno (automático):** indica el turno actual en el cual se registrara la información, el usuario no puede cambiar esta opción.
- **Área (automático):** indica el área para el registro.

- **Líder (automático):** nombre de la persona responsable de cada área.
- **Fecha (automático):** indica la fecha actual del sistema.
- **Días (automático):** el valor por defecto es cero, con este valor se controlara el tiempo que la unidad permanece rezagada.
- **Lote:** registra el nombre o número de lote al que pertenece la unidad.
- **PVI:** registra el número con el cual la unidad está registrada en el lote de producción.
- **Causa:** registra la causa que esta previamente estandarizada.
- **Observación:** el usuario puede ingresar una justificación.
- **Responsable:** registra la persona o área responsable de que la unidad tenga algún tipo de defecto.
- **Modelo:** la lista de modelos se encuentra estandarizada en la base de datos.
- **Detalle:** registra un detalle para brindar mayor información del por qué la unidad se encuentra rezagada.
- **Color:** registra el color de la unidad.
- **Botón Limpiar:** regresa los campos a su estado original para iniciar un nuevo registro de datos.
- **Botón Actualizar:** actualiza la lista de registros.
- **Botón Guardar:** guarda la información en la base de datos.

### **Liberar Unidades Rezagadas:**

Las unidades registradas como rezagadas pueden ser liberadas, esto significa que el defecto en la unidad fue corregido. Seguir los siguientes pasos:

- Click sobre el Link que se encuentra en color azul liberar.

La siguiente ventana que se muestra es la siguiente:

- Es importante ingresar la persona (Líder) quien realizo la gestión para que la unidad sea liberada y siga el proceso normal en la línea de producción.
- Realizar la revisión de la información ingresada antes de liberar la unidad.



**Módulo de Reportes de Pintura.-** Permite la obtener información que fue guardada desde la línea de producción en cada una de las áreas y desde los diferentes puntos de control de la planta.

### Descripción de campos:

La siguiente ventana muestra los diferentes tipos de reportes que se pueden obtener de la información que es registrada en la línea de producción, estos reportes fueron solicitados por los usuarios del sistema.



Ejemplo de reporte.

Reporte: [Detalle Area](#) Fecha[Desde: Hasta]: [Inicio](#)

Area: **ELPD**

Turno: **1 TURNO**

Fecha[Desde]: **2013-10-01** [m. 2013/10/20]

Fecha[Hasta]: **2013-10-15** [m. 2013/10/20]

[Exportar](#) [Obtener reporte](#)

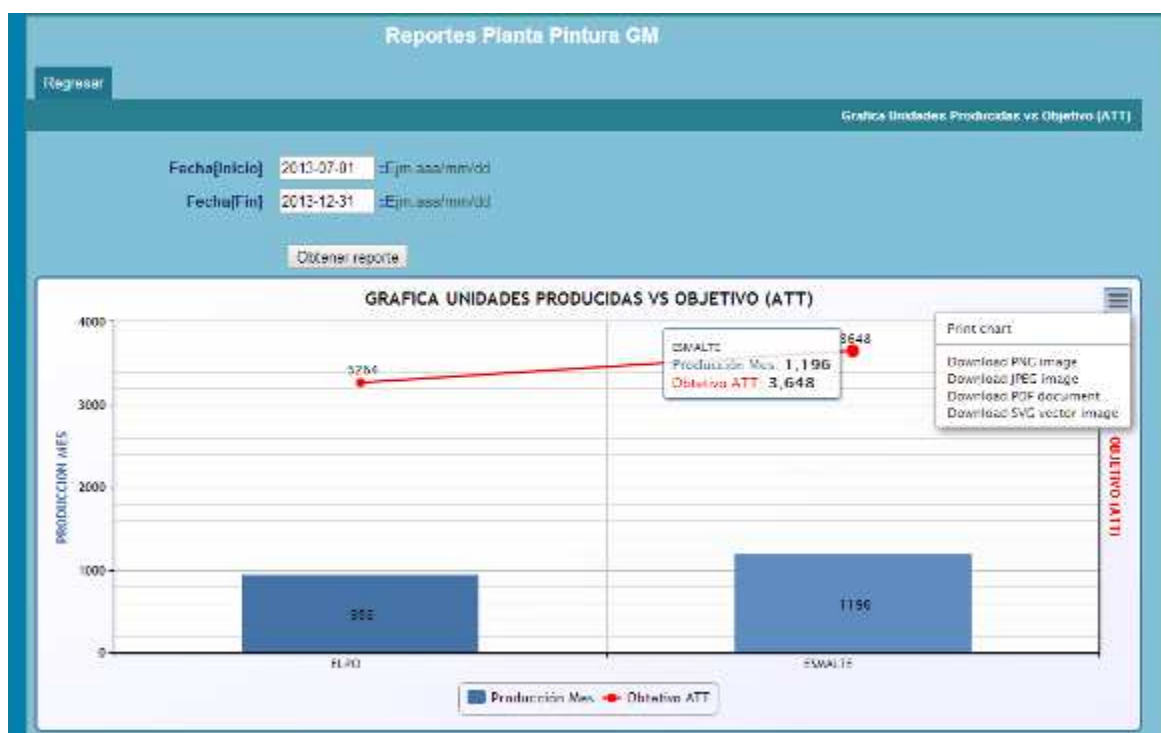
Mostrar **10** registros Buscar

#	Area	Terc	Fecha	Responsable	Logr	Causa	Detalle	Estado	Estado	Cover	Habla	PLP	Nota
818	ELPD	1 TURNO	8/8	MANTENIMIENTO	CARGA	CARGA DE BATERIAS	FALLA DE CONEXION	NO LOGRO	NO LOGRO	AREA CARGA DE BATERIAS	33.33%	33.33%	2013-10-01
817	ELPD	1 TURNO	8	MANTENIMIENTO	CARGA	LINEA LUNA	FALTA DE CARGA	NO LOGRO	NO LOGRO	LINEA LUNA	33.33%	33.33%	2013-10-01
816	ELPD	1 TURNO	8/23	MANTENIMIENTO	CARGA	CARGA DE BATERIAS	FALTA DE CARGA	NO LOGRO	NO LOGRO	AREA CARGA DE BATERIAS	33.33%	33.33%	2013-10-01
815	ELPD	1 TURNO	8/23	PRODUCCION	DESGARRE	DESGARRE EN LA LUNA	FALTA DE CARGA	NO LOGRO	NO LOGRO	DESGARRE EN LA LUNA	33.33%	33.33%	2013-10-01
814	ELPD	1 TURNO	21/23	WPS	CARGA	WPS	FALTA DE CARGA	NO LOGRO	NO LOGRO	WPS	33.33%	33.33%	2013-10-01
813	ELPD	1 TURNO	20	MANTENIMIENTO	CARGA	CARGA DE BATERIAS	FALTA DE CARGA	NO LOGRO	NO LOGRO	AREA CARGA DE BATERIAS	33.33%	33.33%	2013-10-01
812	ELPD	1 TURNO	16	MANTENIMIENTO	CARGA	ESTADO DE BATERIAS	FALTA DE CARGA	NO LOGRO	NO LOGRO	ESTADO DE BATERIAS	33.33%	33.33%	2013-10-01
811	ELPD	1 TURNO	16/27	MANTENIMIENTO	CARGA	CARGA DE BATERIAS	FALTA DE CARGA	NO LOGRO	NO LOGRO	AREA CARGA DE BATERIAS	33.33%	33.33%	2013-10-01
810	ELPD	1 TURNO	4/27	MANTENIMIENTO	CARGA	CARGA DE BATERIAS	FALTA DE CARGA	NO LOGRO	NO LOGRO	AREA CARGA DE BATERIAS	33.33%	33.33%	2013-10-01
809	ELPD	1 TURNO	8	MANTENIMIENTO	CARGA	CARGA DE BATERIAS	FALTA DE CARGA	NO LOGRO	NO LOGRO	AREA CARGA DE BATERIAS	33.33%	33.33%	2013-10-01

Showing 1 to 10 of 52 rows Primero Anterior 1 2 3 4 5 Siguiente Ultimo

## Reportes Graficos

Para la obtener un reporte grafico es necesario ingresar la producción diaria de cada una de las areas, de lo contrario no es posible generar el reporte.



Es importante mencionar que la información que se obtiene en los reportes es registrada por las personas que utilizan el sistema dentro de la planta de pintura GM en los diferentes puntos de control.