



UNIMINUTO

CORPORACIÓN UNIVERSITARIA MINUTO DE DIOS

**AUTOMATIZACIÓN DEL MÓDULO CONTROL DEL
MANTENIMIENTO PREVENTIVO Y CORRECTIVO DE EQUIPOS
DE CÓMPUTO EN *COMPUTER HKS***

Proyecto presentado como requisito
de grado para optar al título de
Tecnólogo en Informática

LAURA CAROLINA CASTELLANOS CABEZAS

MARTHA LILIANA DIAZ MENDIGAÑA

Profesor Guía: CARLOS CHARRY MORA

Soacha, Cundinamarca 2010

CORPORACION UNIVERSITARIA MINUTO DE DIOS

FACULTAD

INGENIERIA



**AUTOMATIZACIÓN DEL MÓDULO CONTROL DEL
MANTENIMIENTO PREVENTIVO Y CORRECTIVO DE EQUIPOS
DE CÓMPUTO EN *COMPUTER HKS***

Proyecto presentado como requisito
de grado para optar al título de
Tecnólogo en Informática

LAURA CAROLINA CASTELLANOS CABEZAS

MARTHA LILIANA DIAZ MENDIGAÑA

Profesor Guía: CARLOS CHARRY MORA

Soacha, Cundinamarca 2010

Nota de aceptación

Jurado

Bogotá, 12 de mayo de 2010

AGRADECIMIENTOS

Damos gracias en primer lugar y más que a nadie a Dios por permitirnos continuar y por guiar el camino para lograr culminar el trabajo de forma exitosa.

Deseamos expresar de igual forma nuestra gratitud a todas las personas que de una u otra forma contribuyeron a la realización de este proyecto.

A nuestros padres por su apoyo indispensable a lo largo de toda la carrera y especialmente en el momento de realización del presente trabajo.

Agradecemos también al profesor Carlos Charry por su invaluable colaboración pero sobre todo por ser un guía para lograr sacar este proyecto adelante.

DEDICATORIA

*Agradezco a Dios por la esperanza que me mueve y el amor que me da felicidad.
A mis padres, César y Martha por su amor, comprensión y paciencia.
A mi hermana por su apoyo.
A mi familia, por sus ánimos.
A mis amigos, por sus buenos deseos.*

Martha Lilitiana Díaz

*A Dios, que me permitió la luz para prepararme y cumplir la misión que me encomendó.
A mis padres Héctor y Blanca, por su amor y apoyo incondicional.
A mi esposo Germán, por su adorable compañía en mi afán por alcanzar mi sueño.*

Laura Castellanos

TABLA DE CONTENIDO

	Pág.
1. INTRODUCCIÓN	8
2. PLANTEAMIENTO DEL PROBLEMA	10
3. ANÁLISIS DE VARIABLES	12
3.1 OUTSOURCING	12
3.2 VENTAJAS DE LA CONSTRUCCIÓN DE CDH SOFTWARE	13
4. OBJETIVOS	14
4.1 GENERAL	14
4.2 OBJETIVOS ESPECIFICOS	14
5. JUSTIFICACIÓN	16
6. MARCO TEÓRICO	19
7. ANTECEDENTES	20
8. MARCO CONCEPTUAL	22
8.1 MODELO EN ESPIRAL	22
8.2 BASE DE DATOS DISTRIBUIDA	23
8.3 PROTOTIPO	24
8.4 DISEÑO ORIENTADO A OBJETOS	24
9. ANALISIS Y DISEÑO	26

9.1 DESCRIPCIÓN DEL SISTEMA ACTUAL	26
9.1.1 DIAGRAMA DE CASOS DE USO	28
9.1.2 DIAGRAMA DE SECUENCIA	32
9.2 SISTEMA PROPUESTO	34
9.2.1 DIAGRAMA DE CASOS DE USO	34
9.2.2 DIAGRAMA DE SECUENCIA	39
9.2.3 REQUERIMIENTOS	40
9.2.3.1 REQUERIMIENTOS DE HARDWARE	40
9.2.4 MODELADO DE CLASES	43
9.2.5 DICCIONARIO DE DATOS	50
10. CONCLUSIONES	76
11. REFERENCIAS BIBLIOGRAFICAS	77

1. INTRODUCCIÓN

El CDH SOFTWARE (Computer Doctor helper) es una aplicación computacional, creada con el fin de implantar en COMPUTER HKS, una empresa dedicada a la prestación del servicio de mantenimiento preventivo y correctivo así como el ensamble de equipos de cómputo, una herramienta automatizada para la manipulación de información en lo concerniente al seguimiento de los procesos internos relacionados con el mantenimiento de equipos, ensamble, atención a clientes, seguimiento a la labor de técnicos, entre otros servicios. Así mismo, brindar a la gerencia informes veraces, puntuales y oportunos, que le ayuden en la toma de decisiones y, por consiguiente, a su gestión.

Este programa brinda a la empresa la oportunidad crecer, de posicionarse en el mercado y aumentar su competitividad, ya que implica la implementación de nuevas tecnologías sin las cuales COMPUTER HKS podría quedar fuera de competencia con el tiempo, siendo considerada obsoleta.

El presente proyecto nace de la necesidad que se ha identificado en el sistema informático actual que impera en la empresa COMPUTER HKS, de contar con una herramienta computacional automatizada, que esté acorde con el desarrollo de la tecnología imperante en el medio y que demandan hoy día las empresas del sector. Máxime,

cuando el servicio que ofrece la empresa está relacionado con los elementos que componen los sistemas informáticos computarizados.

El programa no maneja procesos contables ni de inventarios, en su primera versión, pero queda abierto para que en el futuro se le implementen interfaces contables que permitan manejar rubros y de esta manera generen los presupuestos contables de la empresa. Así mismo, manejar stock mínimos de inventarios.

Las múltiples falencias que presenta el sistema manual actual en la empresa COMPUTER HKS, incentivaron el desarrollo de este programa en su primera versión (prototipo) como una propuesta de solución tecnológica que facilite el procesamiento de información.

2. PLANTEAMIENTO DEL PROBLEMA

La empresa *Computer HKS*, cuenta con un sistema manual que impide el manejo adecuado de la información; lo anterior, ha hecho que no sea posible contar con mecanismo que permita actualizar, y manipular con exactitud todos los datos que fluyen dentro de su sistema de informático, volviéndose de esta manera vulnerable, y haciendo que su información no sea exacta, completa, oportuna y veraz.

Debido a lo expresado anteriormente, se genera un mal aprovechamiento del tiempo, ya que por no ser su sistema informático organizado, frecuentemente, la aglomeración de los equipos represados, no permite que sean entregados en las fechas establecidas al cliente.

Al no contar con un sistema automatizado y organizado de la información, en dicha empresa se dificulta llevar una estadística de control de los equipos revisados, e impide realizar una toma de decisiones de manera eficiente. Otro problema frecuente se presenta cuando en la entrega de los equipos, se debe buscar sus datos en un archivador, por lo que aumenta el tiempo perdido tanto para la persona encargada de hacer su entrega como las personas que se disponen a recibirla; fomentando así, que se malgaste el tiempo por parte del funcionario e impidiéndole que éste no pueda

realizar otras labores dentro de la empresa, como la de recibir otros equipos, o disponerse a realizar los respectivos mantenimientos o el de realizar la solicitud y compra de las partes que son necesarias para el mantenimiento preventivo y correctivo de los equipos.

3. ANÁLISIS DE VARIABLES

3.1 OUTSOURCING

La compañía contratante, o comprador, en este caso COMPUTER HKS se beneficiará de una relación de Outsourcing ya que logrará en términos generales, una "Funcionalidad mayor" a la que tenía internamente con "Costos Inferiores" en la mayoría de los casos, en virtud de la economía de escala que obtienen las compañías contratadas.

Como en todo proceso existen aspectos negativos que forman parte integral del mismo. El Outsourcing no queda exento de esta realidad.

Se decide construir el programa antes de obtenerlo por Outsourcing. Ya que no permite que la empresa pueda tomar decisiones sobre el software, puesto que por ser arrendatario no se le pueden realizar cambios, ni actualizaciones tecnológicas y de esta manera, no permitirán el crecimiento de la empresa en los ámbitos que en ella requiera. Otro riesgo en que recaen las empresas que adquieren contratos con Outsourcing, es que su información es manipulada por terceras personas que no están

comprometidas y no manejan la confidencialidad de sus datos puesto que estos pueden ser utilizados por otras empresas.

3.2 VENTAJAS DE LA CONSTRUCCIÓN DE CDH SOFTWARE (Computer Doctor Helper)

- CDH software (Computer Doctor Helper), es un programa hecho a la medida, pero abierto a cambios tecnológicos innovadores para la empresa Computer HKS, ya que les permite realizar cambios constantes, y al ser una empresa dedicada precisamente al área de tecnología, necesita actualizarse con frecuencia de manera directa.
- Permite total confidencialidad ya que los datos son tratados directamente en la empresa Computer HKS, porque al contar con una aplicación propia, los datos almacenados son totalmente restringidos, y solo pueden acceder a ellos, las personas autorizadas por la empresa.
- Aumenta los beneficios para la empresa ya que la ubica tecnológicamente en el mercado, a partir de la automatización se genera más estabilidad y confiabilidad, que mejora la visión, tanto de la empresa, como de los clientes que solicitan sus servicios.

4. OBJETIVOS

4.1 GENERAL

Automatizar el módulo de mantenimiento de equipos de cómputo preventivo y correctivo de la empresa *Computer HKS*, que permita llevar un control óptimo de todos los procesos que se generan en lo concerniente a la atención de los clientes.

4.2 OBJETIVOS ESPECIFICOS

- Mantener un seguimiento de control de cada equipo dependiendo de las revisiones técnicas que se le realicen, mediante un módulo que permita agilizar cada uno de los procesos, a partir de consultas creadas en MYSQL que optimicen el tratamiento de los datos consultados.
- Agilizar el proceso de registro y reparación de los equipos mediante la sistematización de una base de datos que permita identificar “equipo/problema” y la respectiva solución.
- Ampliar el nivel de competitividad de la empresa y posicionarla en el mercado frente a las demás empresas del sector,

mediante la aplicación de nuevas tecnologías, que permitan a la empresa ampliar su mercado.

5. JUSTIFICACIÓN

Este proyecto es de gran beneficio para el crecimiento y desarrollo de la empresa COMPUTER HKS, ya que para el momento de la implementación del módulo CDH (Computer Doctor Helper) la empresa solo contaba con un sistema de registro de base de datos manual lo que disminuye de forma notable su competitividad. Teniendo en cuenta que en este momento todas las grandes empresas cuentan con programas adecuados a sus necesidades que facilitan la manipulación de información, el no contar con un software para la organización de los datos la pone en riesgo no ser competitiva y por consiguiente de salir del mercado fácilmente. En un mundo en el que hoy por hoy la tecnología avanza rápidamente, es necesario adaptarse a los cambios para mantenerse en el mercado.

Como fue planteado anteriormente, todas estas debilidades ponen a la empresa en gran desventaja puesto que el sistema con el que se cuenta para el registro de datos es aún obsoleto y no parece ser la más sencilla. De acuerdo con la información brindada por los propietarios, al realizar un registro manual de todos los equipos, no era posible llevar un control de cuántos computadores tenían a cargo o habían sido reparados a lo largo de un determinado período de tiempo. Debido a esto la contabilidad de la empresa se hace un poco más complicada puesto que al realizar los balances finales,

con el sistema que manejan, deben contar la bitácora de cada computador para tener acceso al historial de equipos que han recibido mantenimiento.

De la observación de todas estas dificultades surgió la idea de diseñar el programa que le brindará a la empresa Computer HKS la oportunidad de posicionarse con más fuerza en el mercado al ponerse acorde con la vanguardia tecnológica.

Al tener una base de datos sistematizada le brinda a la empresa la oportunidad de llevar un control claro y preciso de todos los equipos que son reparados aún después de transcurrir algunos meses, pues con el sistema actual, que consiste en almacenar las bitácoras de los equipos a cargo en un archivador sólo durante el tiempo de duración de la garantía. Esto impedía que una vez finalizada la garantía fuese posible acceder a la información del mantenimiento que se ha realizado.

La implementación de un programa computacional automatizado como el propuesto en el presente proyecto, es posible ya que a pesar de que la empresa COMPUTER HKS no cuenta con una base de datos sistematizada tiene una clientela bastante amplia que se ha mantenido gracias a la calidad del trabajo que se realiza. Esto, traducido en términos monetarios, obviamente representa mejores ingresos y la posibilidad, en este caso, de invertir en el módulo que definitivamente facilite el control de la información que hasta ahora ha sido incierta.

Es inevitable mencionar que la implementación de una aplicación de esta dimensión que en resumidas cuentas le permite una mejor manipulación de la información de manera sencilla; a su vez, le permite el control del rendimiento de la compañía y ampliar su cobertura trayendo con ello su consabido posicionamiento en el mercado; implica una inversión significativa tanto en el momento de la instalación como en las ocasiones en las que el módulo requiera ser actualizado. Por ello es necesario crear conciencia de que para lograr ese nivel de competitividad la compañía debe estar dispuesta a dar ese paso y realizar la inversión que evidentemente al observar los resultados, es más que justificada.

6. MARCO TEÓRICO

Para comenzar, es importante mencionar las bases teóricas consultadas en el momento de diseñar el módulo. La aplicación realizada es desarrollada mediante la aplicación del modelo en espiral tal como lo plantea Roger Pressman (2002) en su libro *Ingeniería del Software Un enfoque práctico*. Según lo expresado por este autor el modelo en espiral consta de 6 regiones que son: comunicación con el cliente, planificación, análisis de riesgos, ingeniería, construcción y acción, y evaluación del cliente. Se considera que éste trabajo sigue estos lineamientos puesto que fueron los elementos anteriormente planteados lo utilizados para el diseño del módulo.

Por otra parte, el programa diseñado contiene una base de datos distribuida que, según Date (2001), lo conforman datos transparentes y administrados por un servidor que opera diferentes máquinas conectadas por una variedad de redes de comunicación. Además permite manejar varios sistemas operativos. Esta es la forma en la que la aplicación que se presenta en este proyecto fue diseñado.

7. ANTECEDENTES

En el momento de la realización de este proyecto, se consultaron dos modelos de mantenimiento utilizando computadores como su plataforma de funcionamiento. Estos diseños tienen en común con el *CDH Software* el hecho de que el control total se ejerce desde un computador utilizando un software diseñado para tal fin.

A diferencia de este proyecto, los diseños de software encontrados no se dedican al registro de computadores en mantenimiento. Uno de ellos, llamado “MP Software” se dedica al registro y almacenamiento en una base de datos de diferentes equipos entre los cuales no se encuentran computadores. Este programa permite al usuario realizar un seguimiento del estado de sus máquinas. En este caso, el programa está más orientado hacia el mantenimiento de vehículos y se encuentra posicionado en un mercado diferente. Se dedica al mantenimiento preventivo, inventario y control de herramientas. Este software está orientado al mercado latinoamericano en general. A continuación se ofrece una breve explicación acerca del funcionamiento de este módulo.

MP Software

Es un software profesional para control y administración del mantenimiento que permite mantener toda la información del departamento de mantenimiento documentada y organizada.

Día a día, el MP informa sobre los trabajos de mantenimiento que se deben realizar y una vez que se realizan, el MP reprograma la fecha próxima para cuando deban volver a realizarse, ajustando automáticamente los calendarios de mantenimiento.

Este programa puede ser usado en cualquier lugar en donde haya equipos, maquinaria e instalaciones sujetas a mantenimiento como Industrias, Constructoras, Hoteles, Hospitales, Flotillas, Empresas de servicios, etc¹.

¹ Disponible en: [http:// www.mpsoftware.com.mx/home.html](http://www.mpsoftware.com.mx/home.html)

8. MARCO CONCEPTUAL

A continuación se presentan los conceptos, más detallados de los lineamientos que sirvieron como base para la creación y estructuración de este proyecto.

8.1 Modelo en espiral

Este modelo fue planteado por Roger Pressman (2002) y contiene 6 regiones de tareas que se describen a continuación:

- Comunicación con el cliente: Las tareas requeridas para establecer comunicación entre el desarrollador y el cliente.
- Planificación: Las tareas requeridas para definir recursos, el tiempo y otra información relacionadas con el proyecto.
- Análisis de riesgos: Las tareas requeridas para evaluar riesgos técnicos y de gestión.
- Ingeniería: las tareas requeridas para construir una o mas representaciones de la aplicación.

- Construcción y acción: las tareas requeridas para construir, probar, instalar y proporcionar soporte al usuario.
- Evaluación del cliente: las tareas requeridas para obtener la reacción del cliente según la evaluación de las representaciones del software creadas durante la etapa de ingeniería e implementada durante la etapa de instalación.

Cada una de estas regiones está compuesta por un conjunto de tareas del trabajo, llamado *conjunto de tareas*, que se adaptan a las características del proyecto que va a emprenderse.

8.2 Base de datos distribuida

La base de datos distribuida, según Date (2001) es en sí un tema extenso llevada a su conclusión lógica, el soporte total a la base de datos distribuida implica que una sola aplicación debe ser capaz de operar de manera transparente sobre los datos que están dispersos a través de una variedad de base de datos diferentes, las cuales son administradas por una variedad de DBMS distintos, operan en varias máquinas distintas, son manejadas por varios sistemas operativos diferentes y están conectadas por una variedad de redes de comunicación distintas; aquí, “de manera transparente” significa

que la aplicación operada desde el punto de vista lógico como si los datos fueran manejados por un solo DBMS y una sola máquina.

8.3 Prototipo

De acuerdo a la definición dada por James A. Seen (1992) es un sistema que funciona -no solo una idea en el papel- desarrollado con la finalidad de probar ideas y suposiciones relacionadas con el nuevo sistema. Al igual que cualquier sistema basado en computadora, está constituido por software que acepta entrada, realiza cálculos, produce información ya sea impresa o presentada en pantalla, o que lleva a cabo otras actividades significativas. Es la primera versión o iteración, de un sistema de información; es el modelo original.

Los usuarios evalúan el diseño y la información generada por el sistema. Lo anterior solo puede hacerse con efectividad si los datos utilizados, al igual que las situaciones son reales. Por otra parte deben esperarse cambios a medida que el sistema sea utilizado.

8.4 Diseño orientado a objetos

Este diseño, según Roger Pressman (2002) consiste básicamente en la realización de software de forma específica partiendo del Análisis Orientado a Objetos. El Diseño Orientado a Objetos estaría

compuesto por cuatro etapas diferentes, estas son: 1) el diseño de subsistemas que implementan las funciones principales de sistema y que es considerada la etapa fundamental de todo el diseño; 2) la etapa de clases durante la cual se especifica de manera global la arquitectura de los objetos así como la jerarquía de clases que se requiere para implementar un sistema; 3) la etapa de mensajes que muestra la forma en que la colaboración entre objetos debe ser realizada y; 4) la etapa de responsabilidades que identifica las operaciones y los atributos característicos de cada clase.

9. ANÁLISIS Y DISEÑO

9.1 DESCRIPCIÓN DEL SISTEMA ACTUAL

En la actualidad, la empresa COMPUTER HKS realiza varios pasos, todos de forma manual, para llevar a cabo el control de la información relacionada con el mantenimiento de los equipos que tienen a cargo. Los pasos que se llevan a cabo para cada nuevo ingreso o cliente son los siguientes:

1. Diligenciamiento de folleto al ingresar el computador con la siguiente información
 - 1.1 Información básica acerca del cliente como nombre, teléfono, fecha de ingreso del equipo, fecha de salida y técnico encargado del mantenimiento.
 - 1.2. El técnico realiza la revisión del equipo para determinar su estado y lo reporta de forma escrita en el mismo folleto
 - 1.3. Se genera el diagnóstico basándose en el problema encontrado y cuál sería la respectiva solución. Se hacen las anotaciones pertinentes de forma escrita.

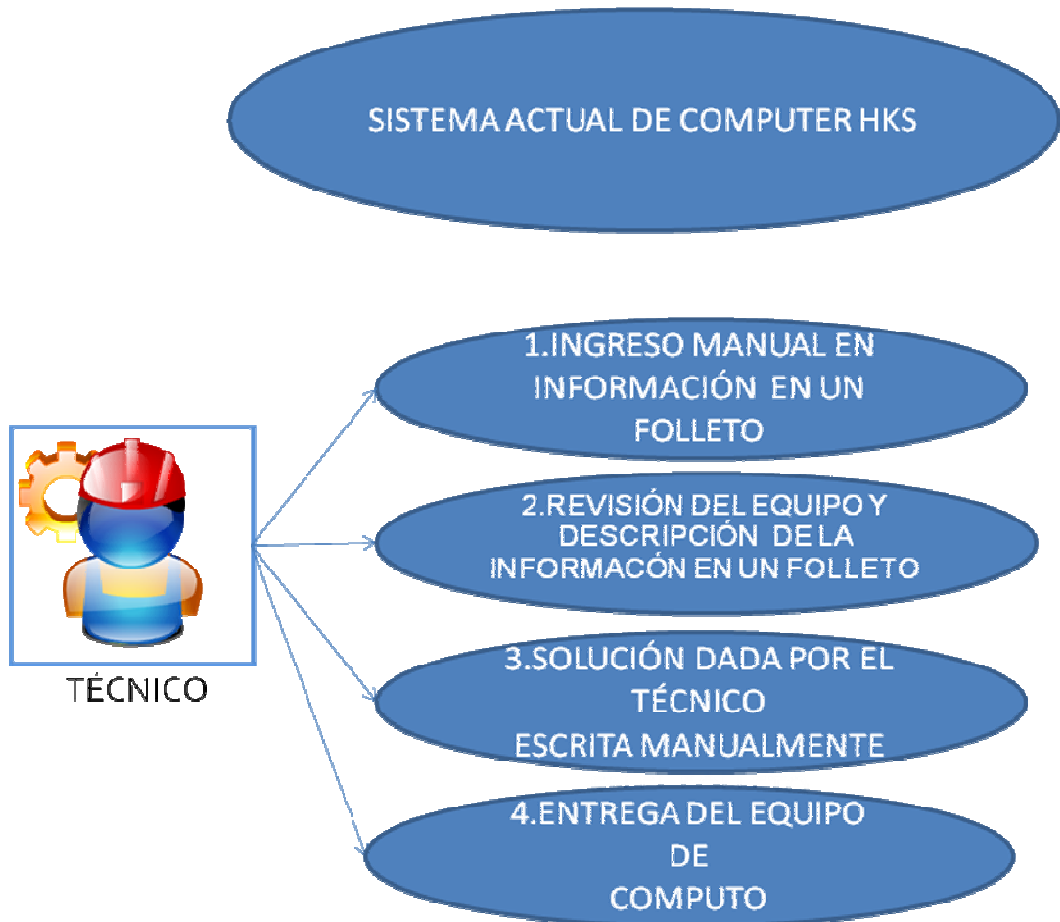
2. Se hace entrega del equipo al cliente funcionando completamente quien una vez verifica que su equipo se encuentra funcionando realiza el pago por el mantenimiento realizado.

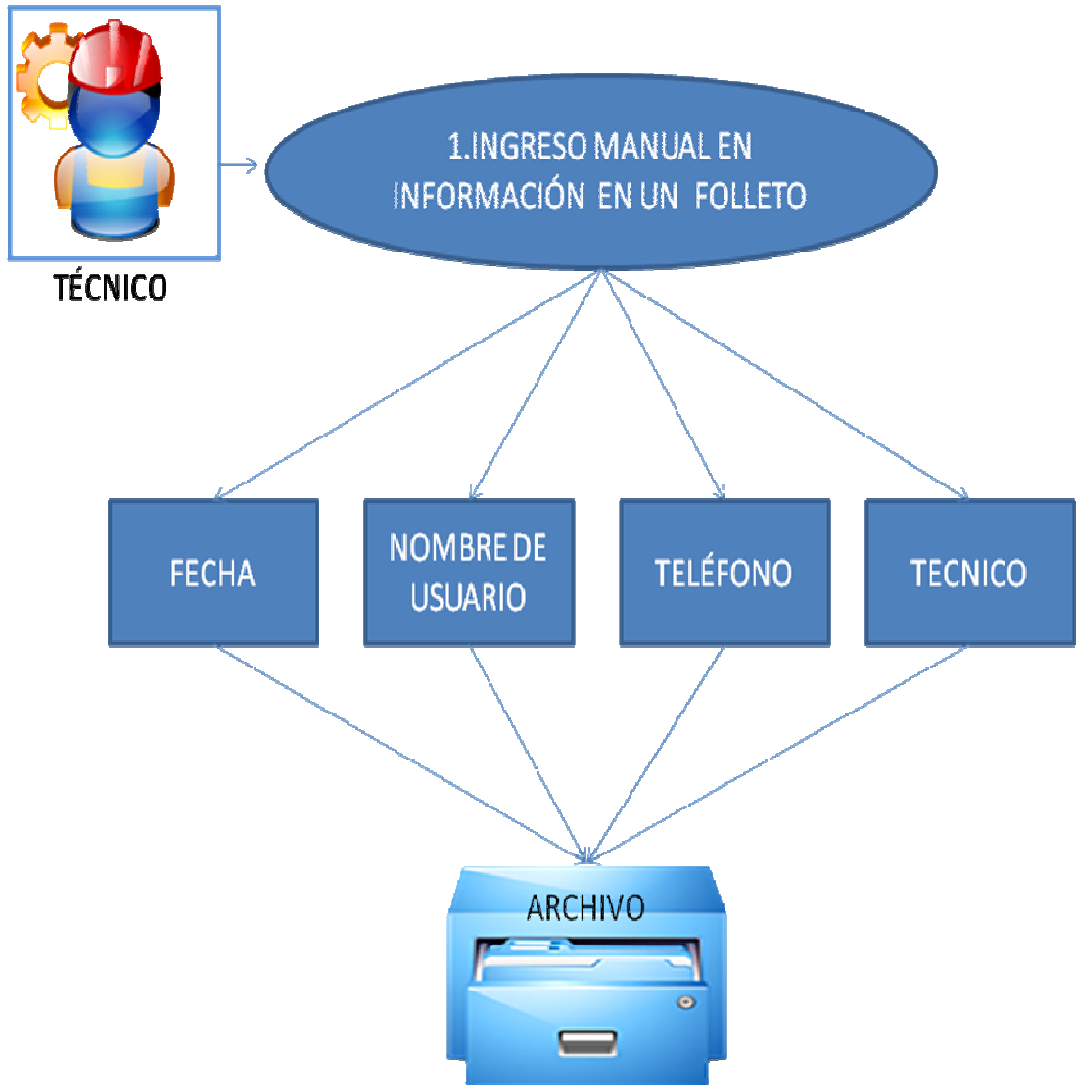
5. El folleto es almacenado en un archivador y permanece allí el mismo tiempo de validez de la garantía.

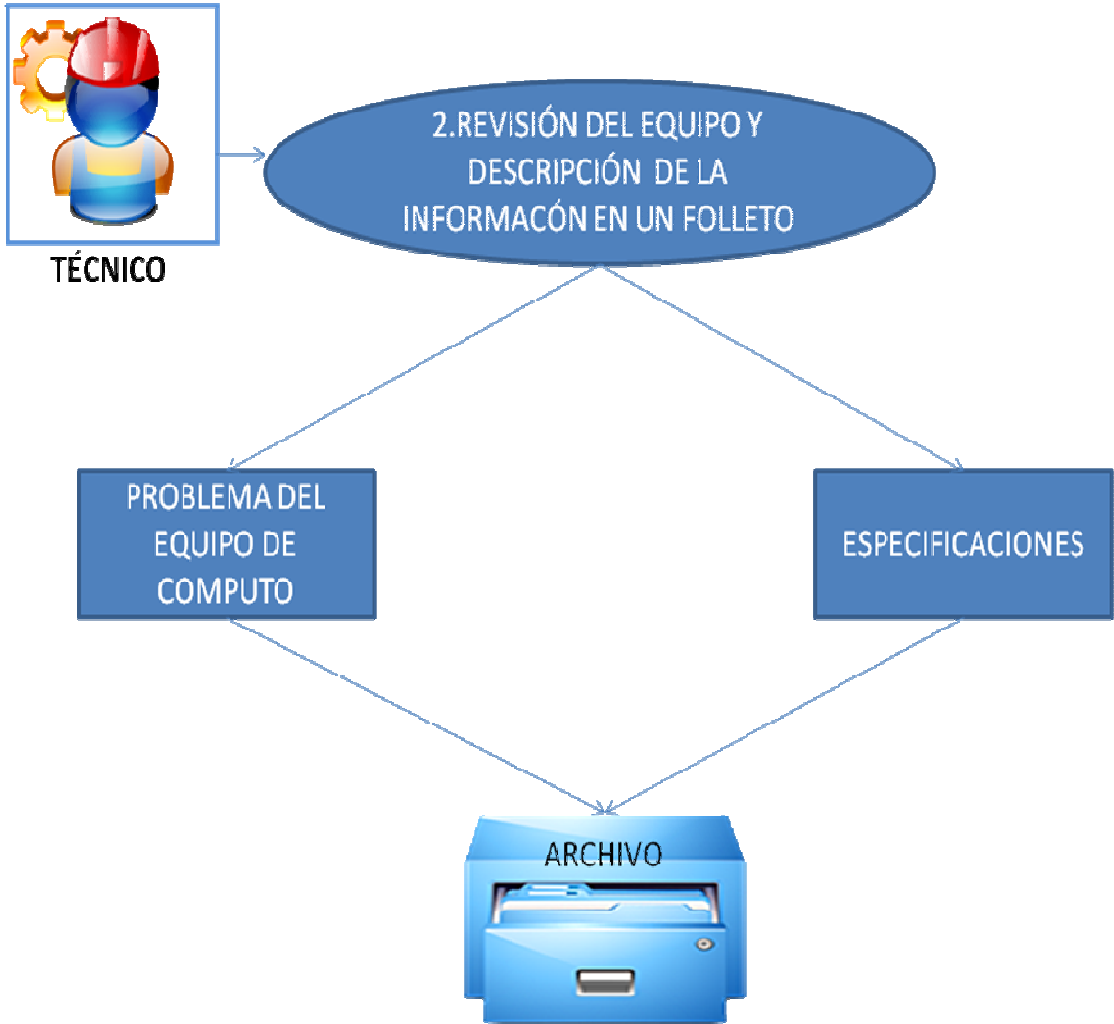
6. Una vez cumplida la garantía los folletos se desechan perdiendo cualquier posibilidad de acceder a información acerca de los mantenimientos realizados.

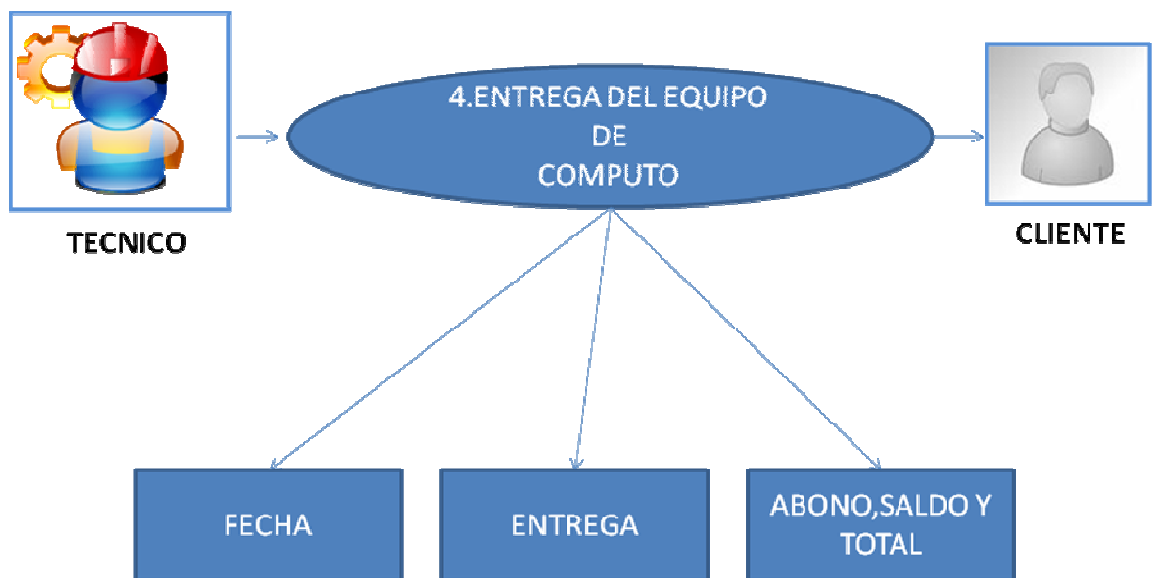
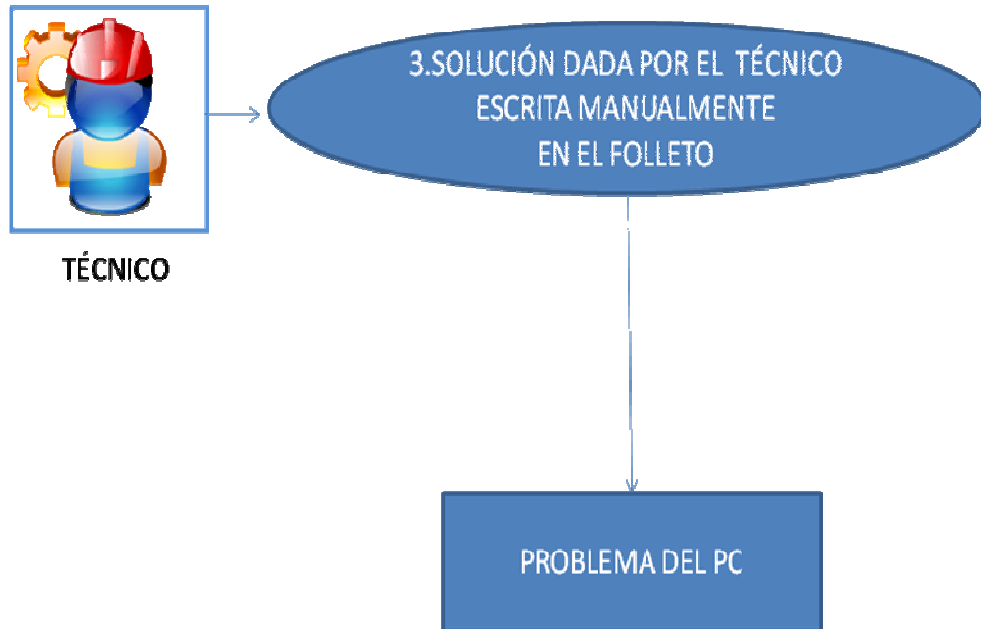
De esta forma los recursos que utilizan actualmente son la papelería para realizar los folletos y el archivador.

9.1.1 DIAGRAMAS DE CASOS DE USO

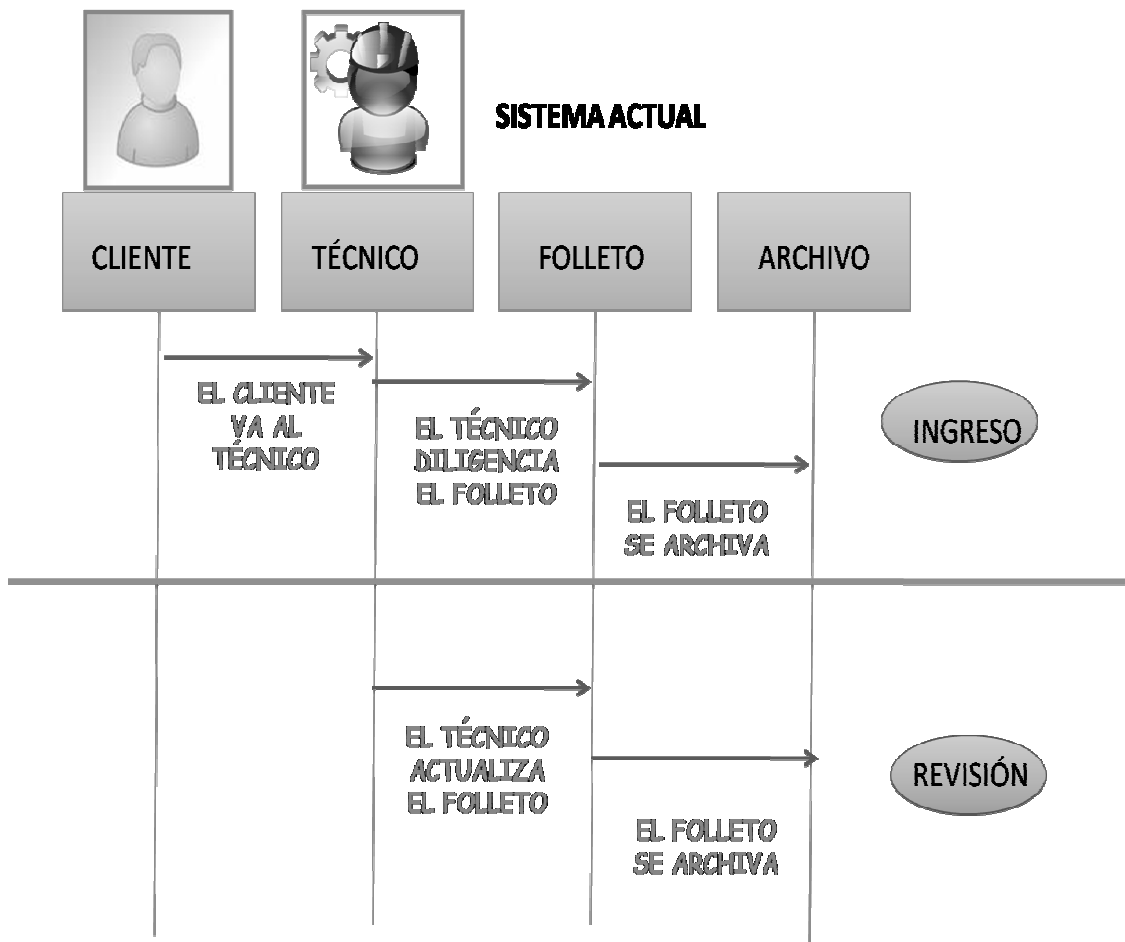


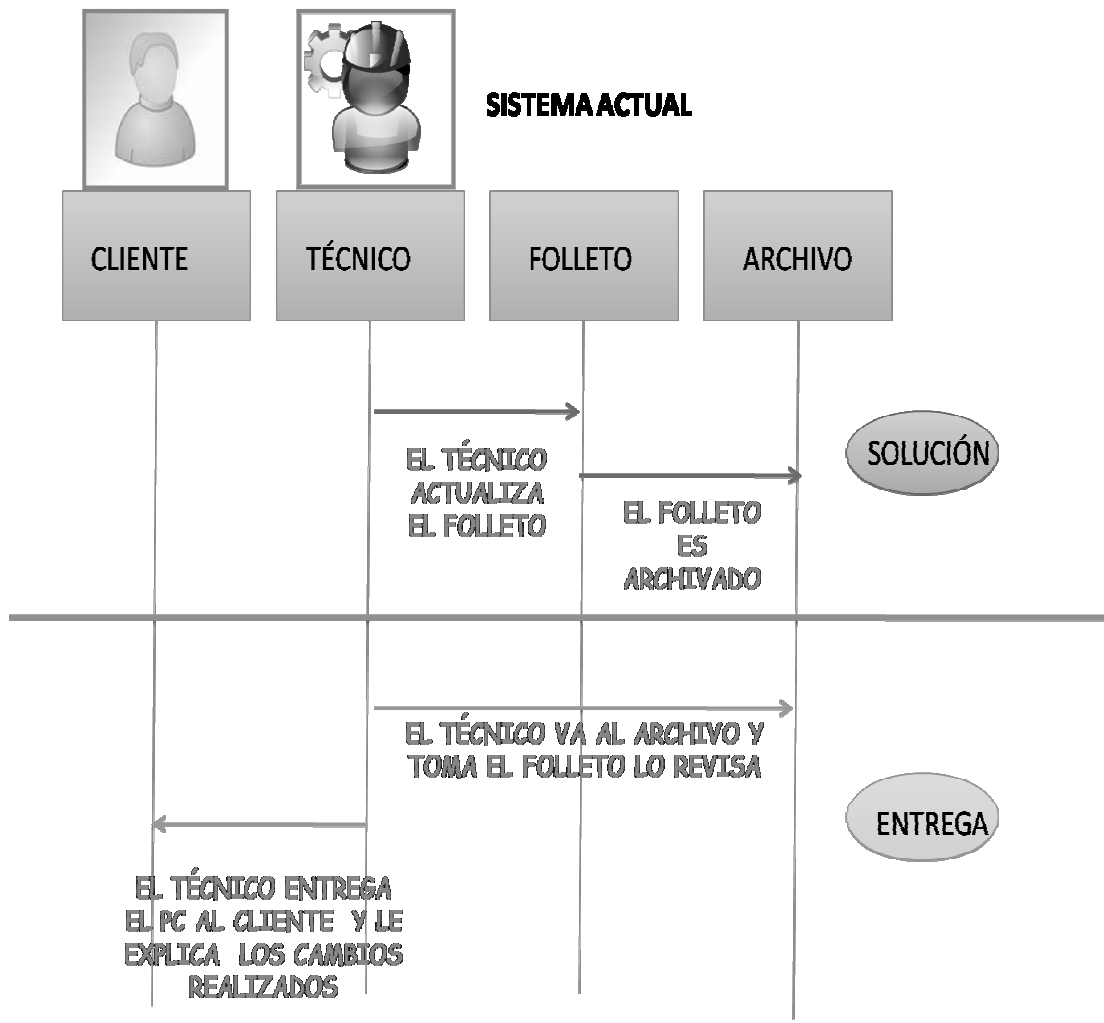






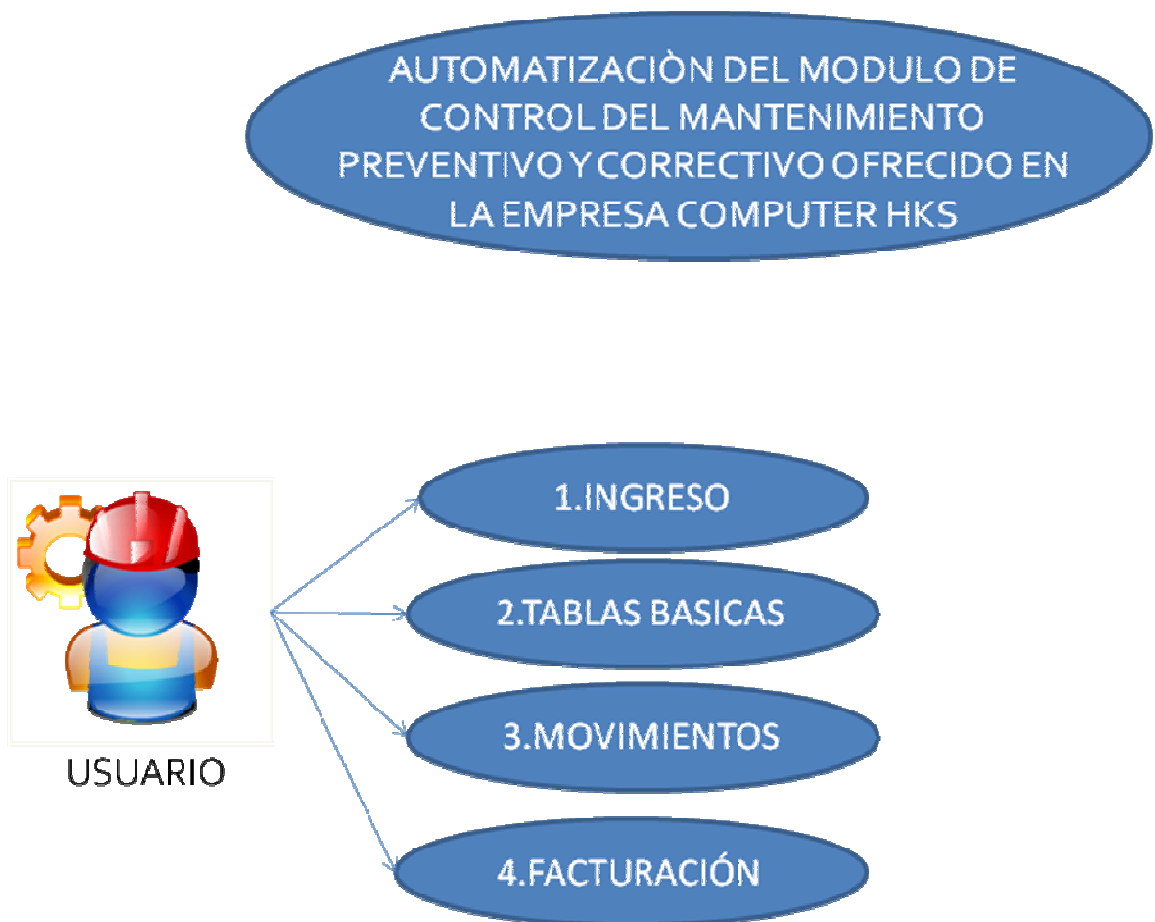
9.1.2 DIAGRAMA DE SECUENCIA

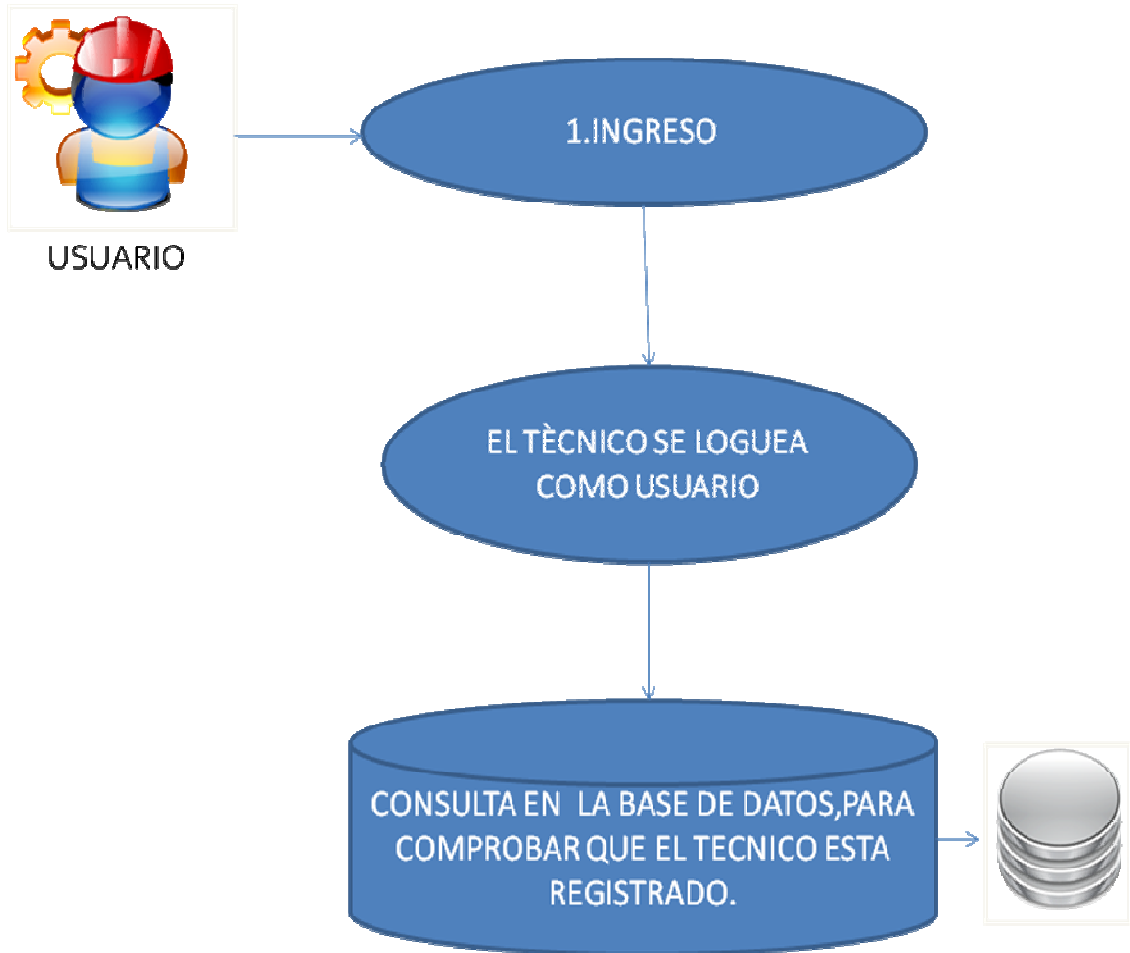


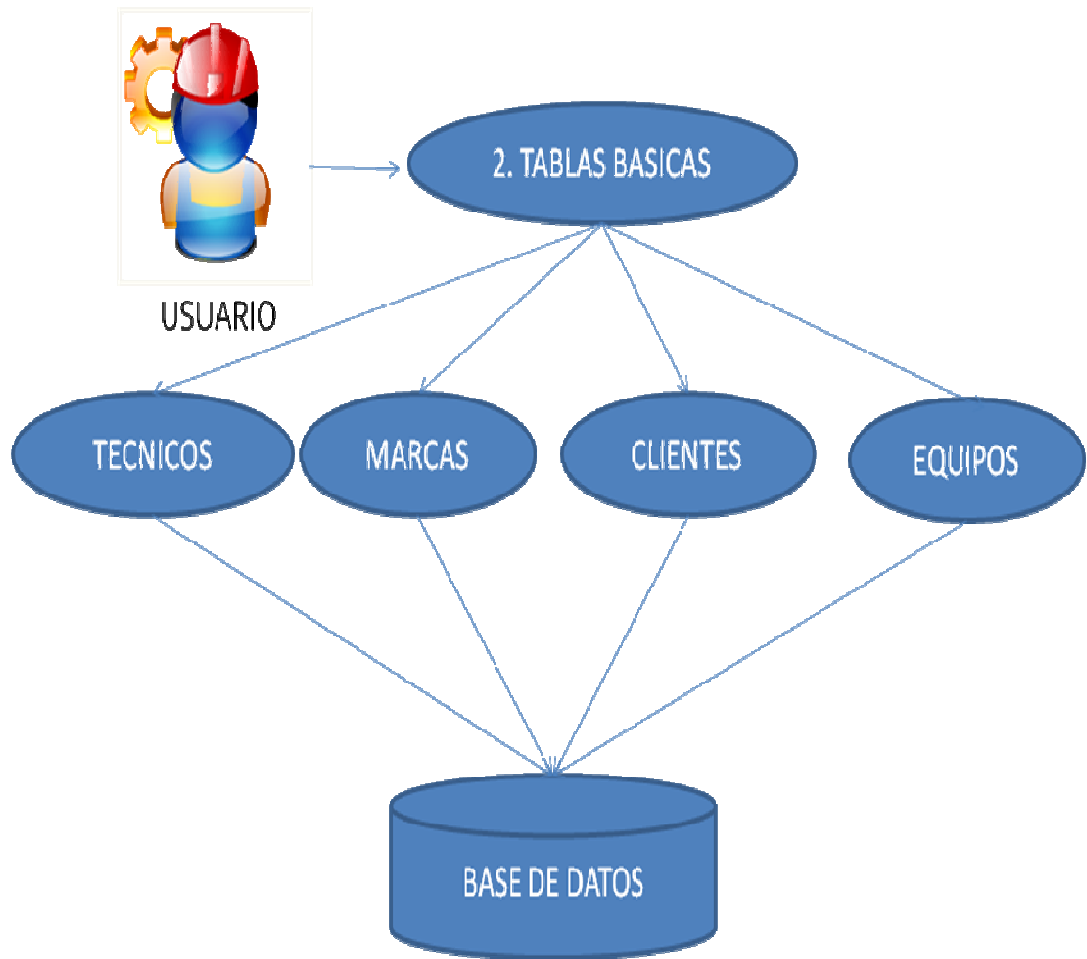


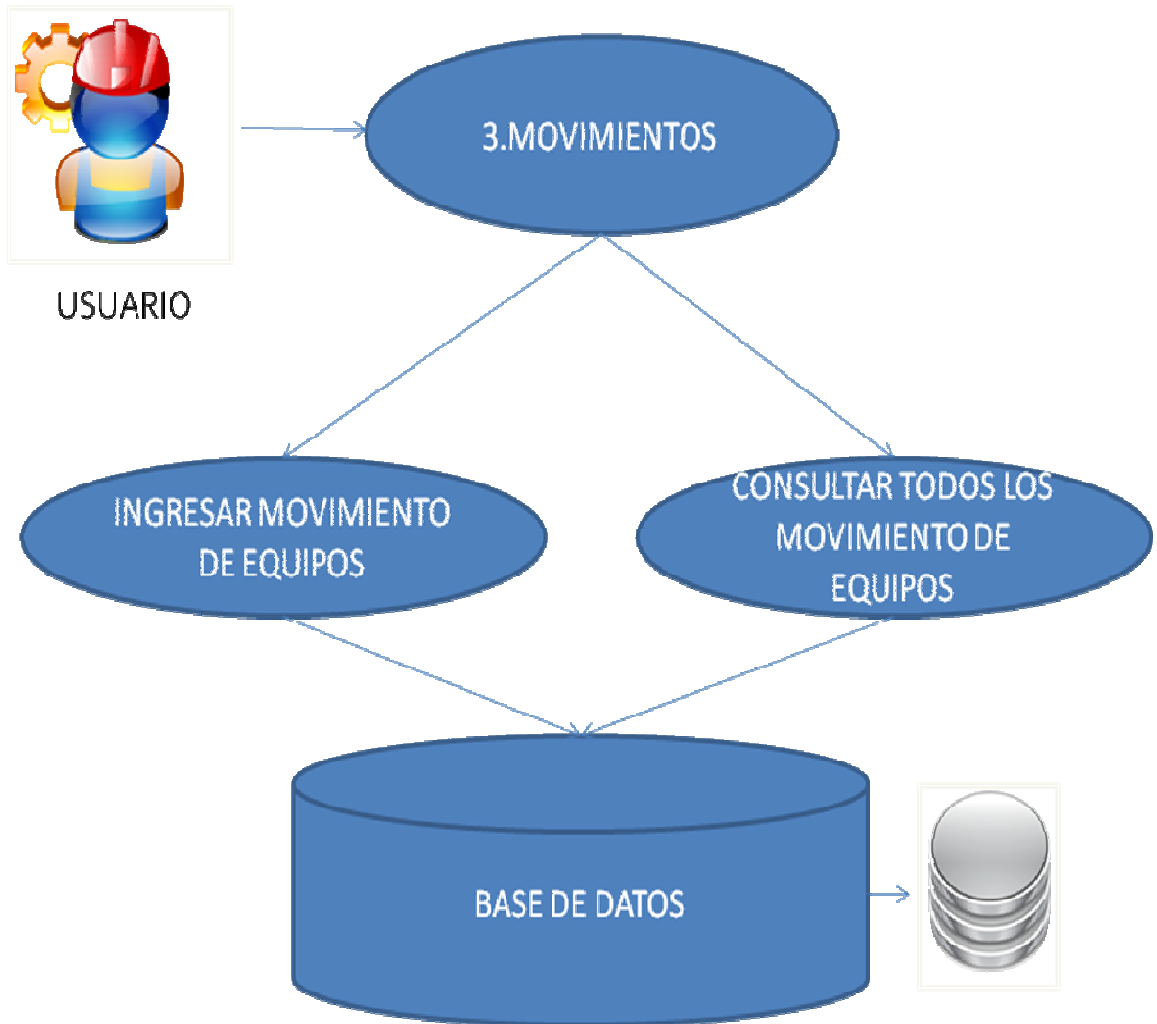
9.2 SISTEMA PROPUESTO

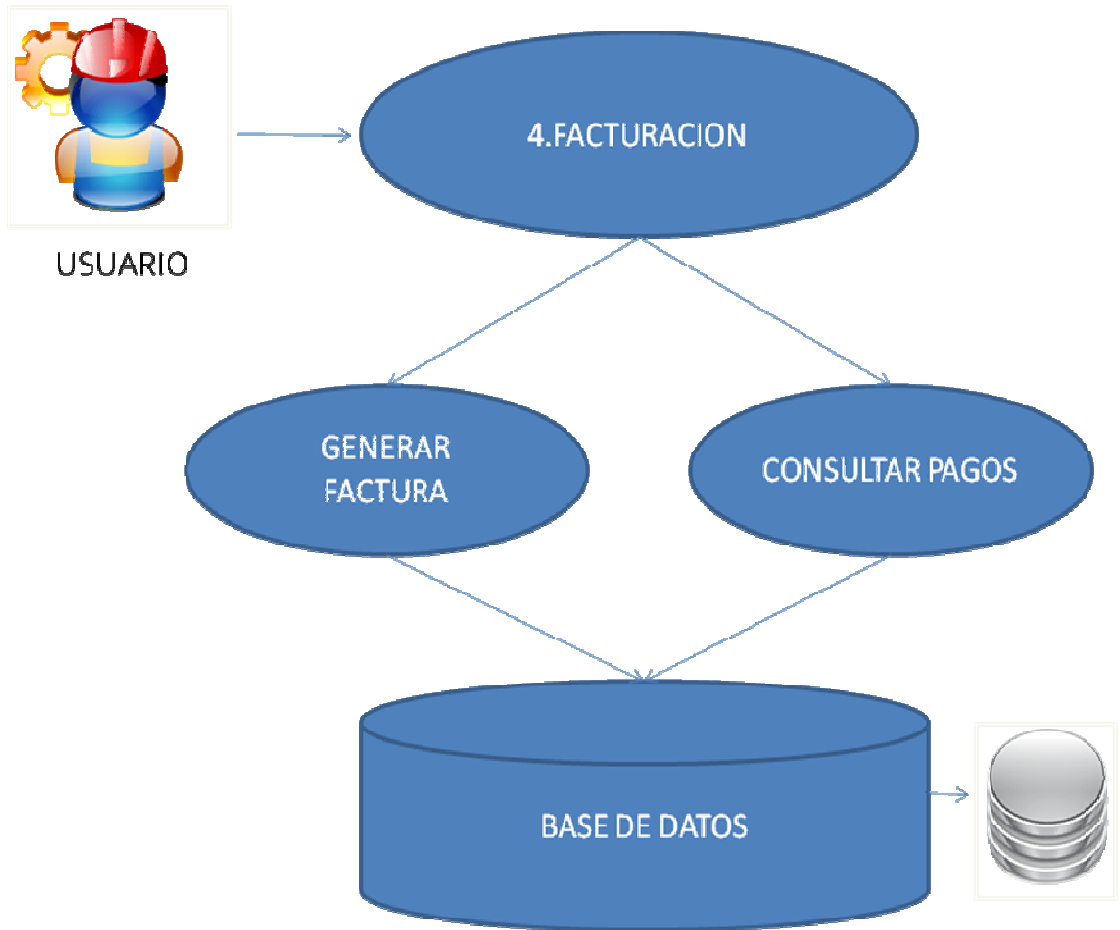
9.2.1 DIAGRAMA DE CASOS DE USO



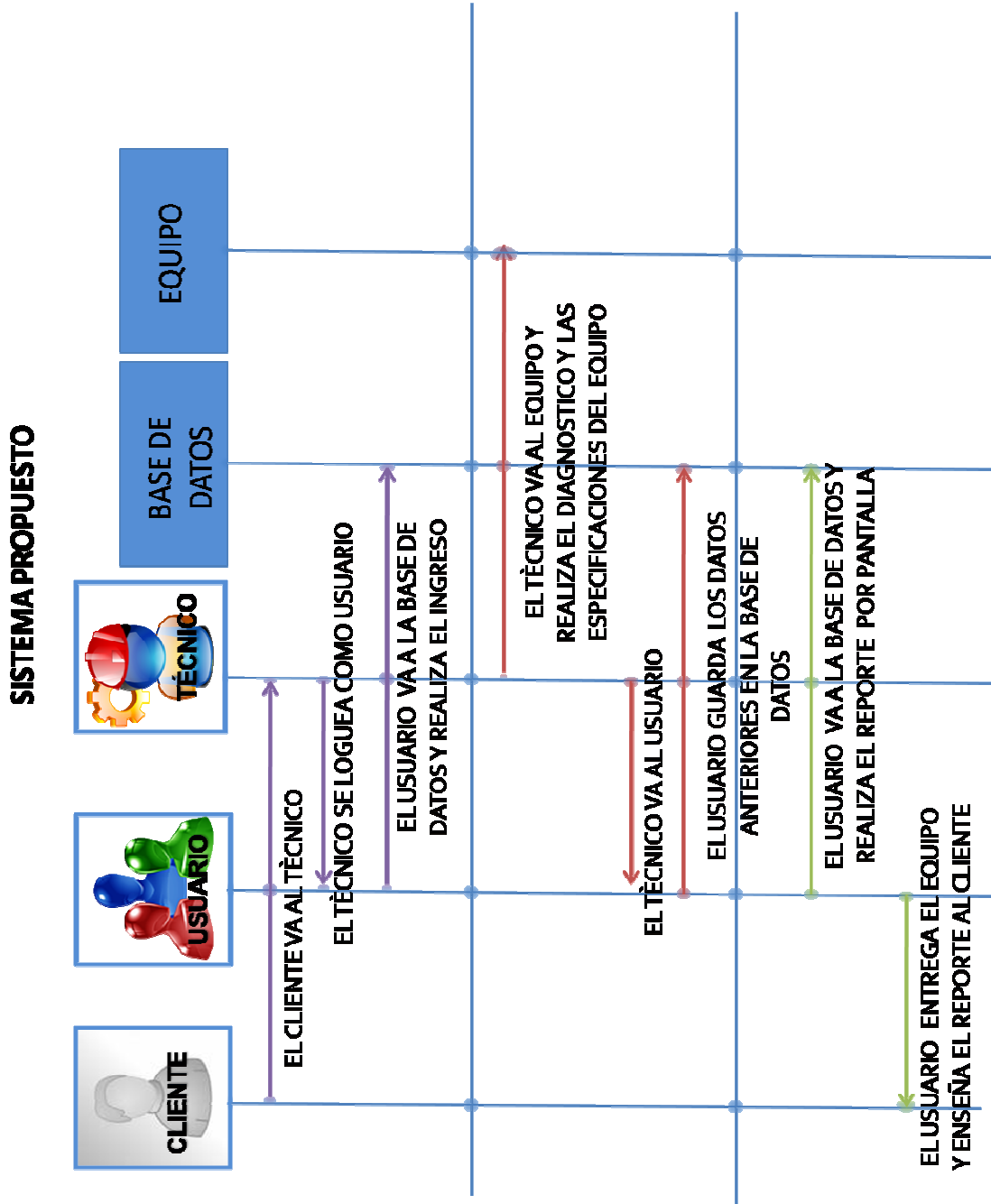








9.2.2 DIAGRAMA DE SECUENCIA



9.2.3 REQUERIMIENTOS

9.2.3.1 REQUERIMIENTOS DE HARDWARE:

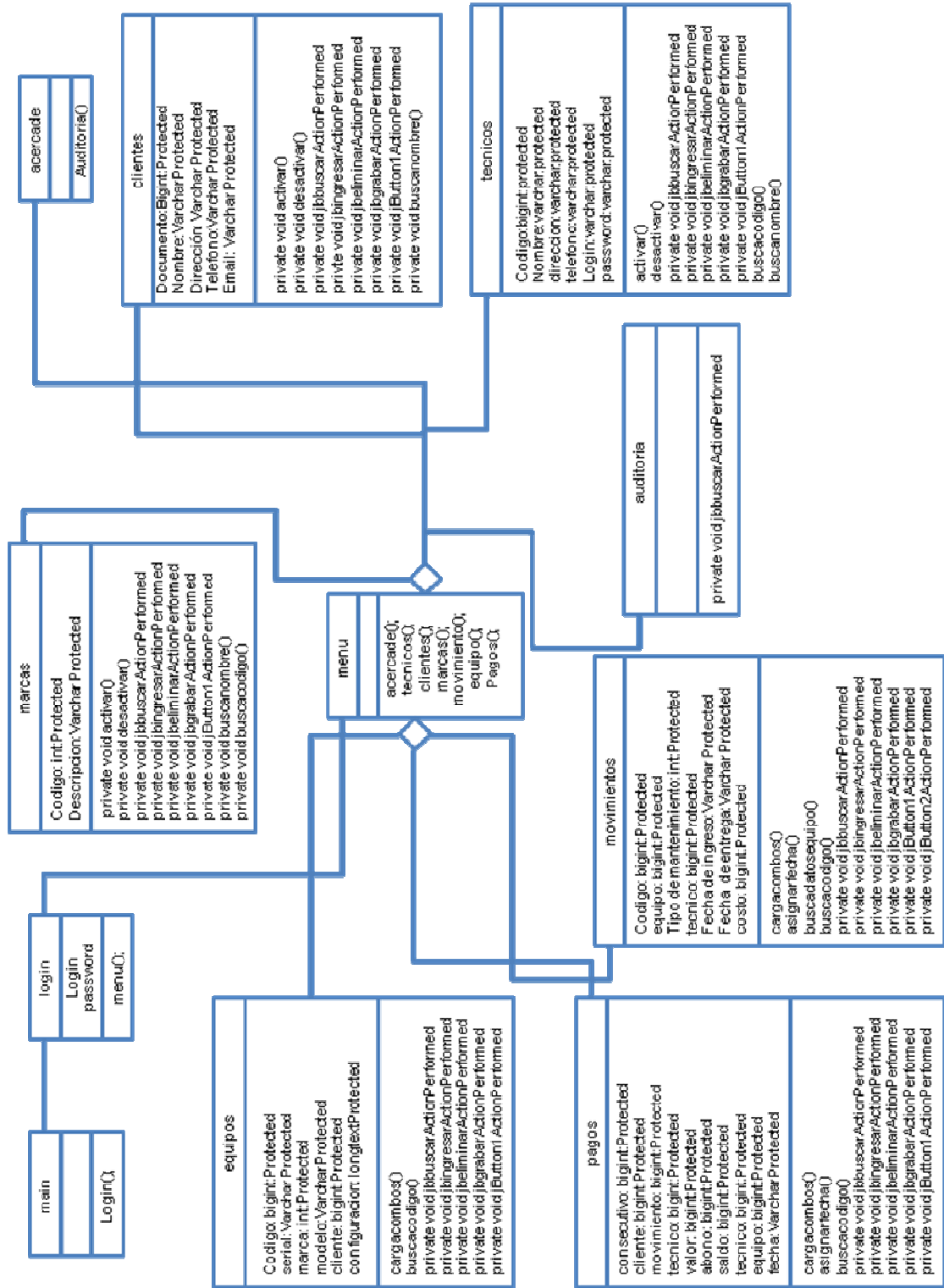
Sistemas operativos compatibles

Configuraciones de hardware mínimas

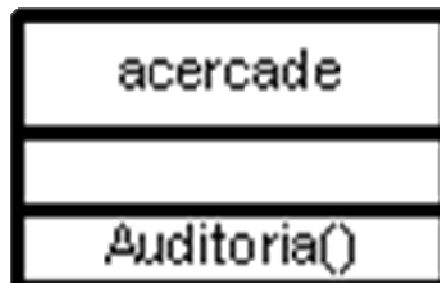
- Microsoft Windows XP Professional SP3:
 - Procesador: Intel Pentium III o equivalente a 800 MHz
 - Memoria: 512 MB
 - Espacio de disco: 750 MB de espacio libre en el disco
- Microsoft Windows Vista SP1:
 - Procesador: Intel Pentium III o equivalente a 800 MHz
 - Memoria: 512 MB
 - Espacio de disco: 750 MB de espacio libre en el disco
- Microsoft Windows 7:
 - Procesador: Intel Pentium III o equivalente a 800 MHz
 - Memoria: 512 MB
 - Espacio de disco: 750 MB de espacio libre en el disco
- Ubuntu 9.04:
 - Procesador: Intel Pentium III o equivalente a 800 MHz
 - Memoria: 512 MB
 - Espacio de disco: 650 MB de espacio libre en el disco
- Solaris OS versión 10 (SPARC):
 - Procesador: UltraSPARC II a 450 MHz
 - Memoria: 512 MB
 - Espacio de disco: 650 MB de espacio libre en el disco
- Solaris OS versión 10 (x86/x64 Platform Edition):
 - Procesador: AMD Opteron serie 1200 a 1,8 GHz
 - Memoria: 512 MB
 - Espacio de disco: 650 MB de espacio libre en el disco
- Macintosh OS X 10.5 Intel:
 - Procesador: Intel Dual Core
 - Memoria: 512 MB
 - Espacio de disco: 650 MB de espacio libre en el disco
- Macintosh OS X 10.5 PPC:
 - Procesador: PowerPC G4
 - Memoria: 512 MB
 - Espacio de disco: 650 MB de espacio libre en el disco

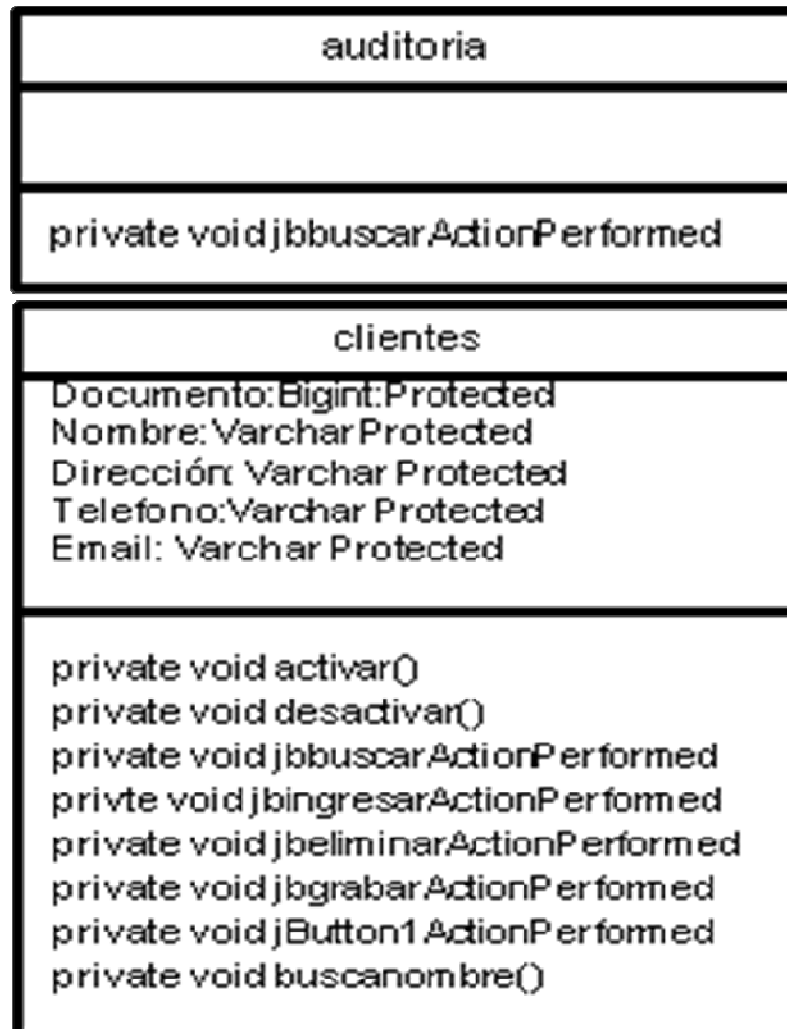
Configuraciones de software recomendadas

- Microsoft Windows XP Professional SP3:
 - Procesador: Intel Pentium IV o equivalente a 2,6 GHz
 - Memoria: 2 GB
 - Espacio de disco: 1 GB de espacio libre en el disco
- Microsoft Windows Vista SP1:
 - Procesador: Intel Pentium IV o equivalente a 2,6 GHz
 - Memoria: 2 GB
 - Espacio de disco: 1 GB de espacio libre en el disco
- Microsoft Windows 7:
 - Procesador: Intel Pentium IV o equivalente a 2,6 GHz
 - Memoria: 2 GB
 - Espacio de disco: 1 GB de espacio libre en el disco
- Ubuntu 9.04:
 - Procesador: Intel Pentium IV o equivalente a 2,6 GHz
 - Memoria: 2 GB
 - Espacio de disco: 850 MB de espacio libre en el disco
- Solaris OS versión 10 (SPARC):
 - Procesador: UltraSPARC IIIi a 1 GHz
 - Memoria: 2 GB
 - Espacio de disco: 850 MB de espacio libre en el disco
- Solaris OS versión 10 (x86/x64 Platform Edition):
 - Procesador: AMD Opteron serie 1200 a 2,8 GHz
 - Memoria: 2 GB
 - Espacio de disco: 850 MB de espacio libre en el disco
- OpenSolaris 2009.06 (edición para plataformas x86/x64):
 - Procesador: AMD Opteron serie 1200 a 2,8 GHz
 - Memoria: 2 GB
 - Espacio de disco: 650 MB de espacio libre en el disco
- Macintosh OS X 10.5 Intel:
 - Procesador: Intel Dual Core
 - Memoria: 2 GB
 - Espacio en disco: 850 MB de espacio libre en el disco
- Macintosh OS X 10.5 PPC:
 - Procesador: PowerPC G5
 - Memoria: 2 GB
 - Espacio en disco: 850 MB de espacio libre en el disco



9.2.4 MODELADO DE CLASES





tecnicos
Codigo: bigint:protected Nombre: varchar:protected direccion: varchar:protected telefono: varchar:protected Login: varchar:protected password: varchar:protected
activar() desactivar() private void jbbuscarActionPerformed private void jbingresarActionPerformed private void jbeliminarActionPerformed private void jbgrabarActionPerformed private void jButton1ActionPerformed buscacodigo() buscanombre()

marcas
Codigo: int:Protected Descripcion: Varchar Protected
private void activar() private void desactivar() private void jbbuscarActionPerformed private void jbingresarActionPerformed private void jbeliminarActionPerformed private void jbgrabarActionPerformed private void jButton1ActionPerformed private void buscanombre() private void buscacodigo()

movimientos

Codigo: bigint:Protected
equipo: bigint:Protected
Tipo de mantenimiento: int:Protected
tecnico: bigint:Protected
Fecha de ingreso: Varchar Protected
Fecha de entrega: Varchar Protected
costo: bigint:Protected

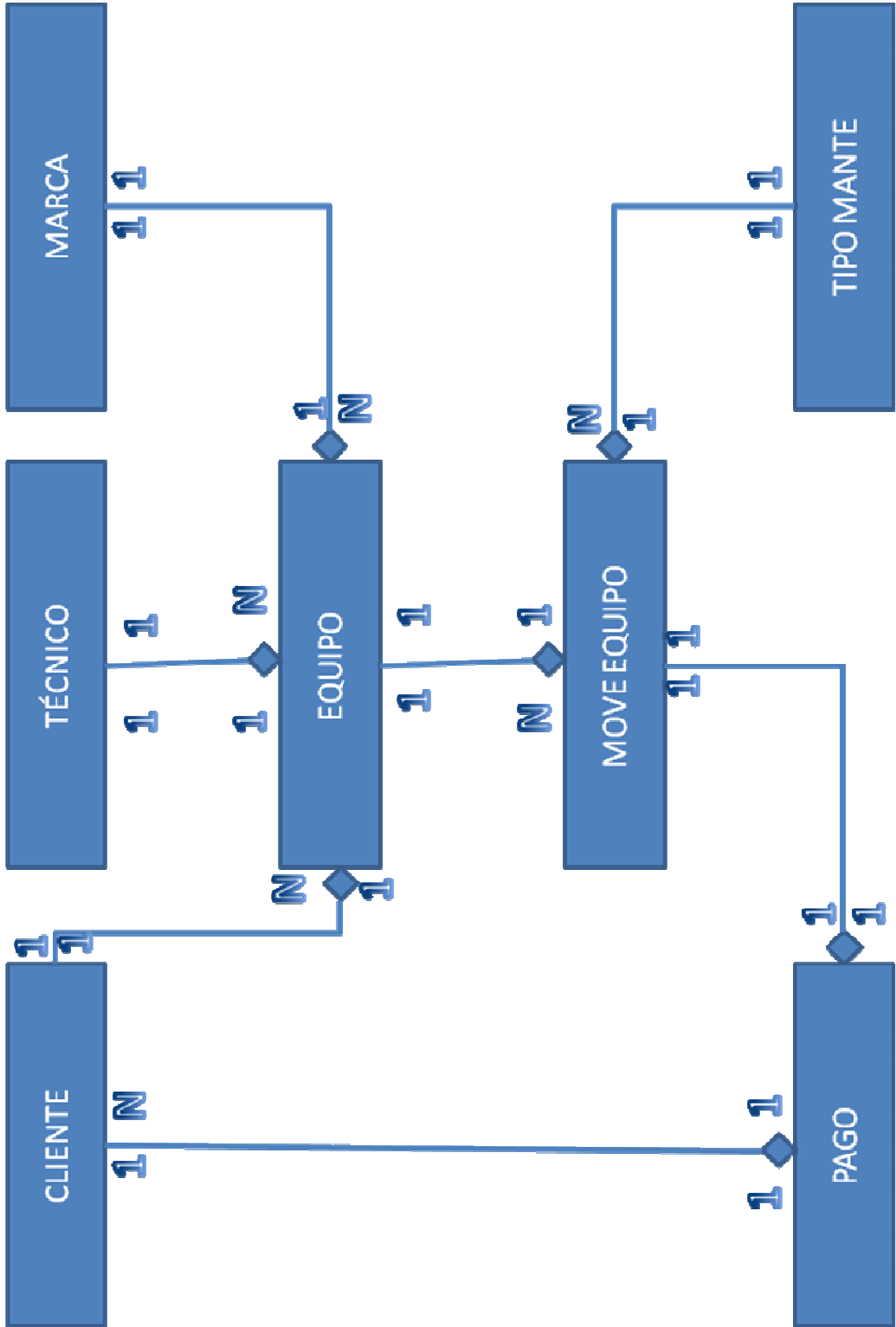
cargacombos()
asignarfecha()
buscadatosequipo()
buscacodigo()
private void jButtonBuscarActionPerformed
private void jButtonIngresarActionPerformed
private void jButtonEliminarActionPerformed
private void jButtonGrabarActionPerformed
private void jButton1ActionPerformed
private void jButton2ActionPerformed

equipos

Codigo: bigint:Protected
serial: Varchar Protected
marca: int:Protected
modelo: Varchar Protected
cliente: bigint:Protected
configuracion: longtextProtected

cargacombos()
buscacadigo()
private void jbbuscarActionPerformed
private void jbingresarActionPerformed
private void jbeliminarActionPerformed
private void jbgrabarActionPerformed
private void jButton1ActionPerformed

pagos
consecutivo: bigint:Protected cliente: bigint:Protected movimiento: bigint:Protected tecnico: bigint:Protected valor: bigint:Protected abono: bigint:Protected saldo: bigint:Protected tecnico: bigint:Protected equipo: bigint:Protected fecha: VarcharProtected
cargacombos() asignarfecha() buscacadigo() private void jbbuscar.ActionPerformed private void jbingresar.ActionPerformed private void jbeliminar.ActionPerformed private void jbgrabar.ActionPerformed private void jButton1.ActionPerformed



9.2.5 DICCIONARIO DE DATOS

1. TABLA CLIENTE

NOMBRE: cliente							
DESC: tabla							
CAMPO	TIPO DE DATO	LONG	CLAVE	UNICO	OBLIGATORIEDAD	INDEXADO	DESC
Cli_doc	Numérico	20	PK	SI	SI	SI	Campo que almacena datos
Cli_nom	Texto	50	——	NO	SI	SI	Campo que almacena datos
Cli_dir	Texto	50	——	NO	SI	NO	Campo que almacena datos
Cli_tel	Texto	20	——	NO	SI	NO	Campo que almacena datos
Cli_email	Texto	60	——	SI	SI	NO	Campo que almacena datos

2. TABLA MARCA

NOMBRE: marca							
DESC: tabla							
CAMPO	TIPO DE DATO	LONG	CLAVE	UNICO	OBLIGATORIEDAD	INDEXADO	DESC
mar_cod	Numérico	11	PK	SI	SI	SI	Campo que almacena datos
Mar_des	Texto	30	——	NO	SI	NO	Campo que almacena datos

3. TABLA TÉCNICO*

NOMBRE: técnico							
DESC: tabla							
CAMPO	TIPO DE DATO	LONG	CLAVE	UNICO	OBLIGATORIEDAD	INDEXADO	DESC
tec_cod	Numérico	20	PK	SI	SI	SI	Campo que almacena datos
tec_nom	Texto	50	——	NO	SI	SI	Campo que almacena datos
tec_dir	Texto	50	——	NO	SI	NO	Campo que almacena datos
tec_tel	Texto	20	——	NO	SI	NO	Campo que almacena datos
tec_login	Texto	10	——	SI	SI	NO	Campo que almacena datos
tec_pass	Texto	10	——	SI	SI	NO	Campo que almacena datos
tec_login	Texto	10	——	SI	SI	NO	Campo que almacena datos

4. TABLA TIPO DE MANTENIMIENTO

NOMBRE: tipomante							
DESC: tabla							
CAMPO	TIPO DE DATO	LONG	CLAVE	UNICO	OBLIGATORIEDAD	INDEXADO	DESC
tec_cod	Númerico	11	PK	SI	SI	SI	Campo que almacena datos
tpm_des	Texto	30	——	NO	SI	NO	Campo que almacena datos

5. TABLA EQUIPO

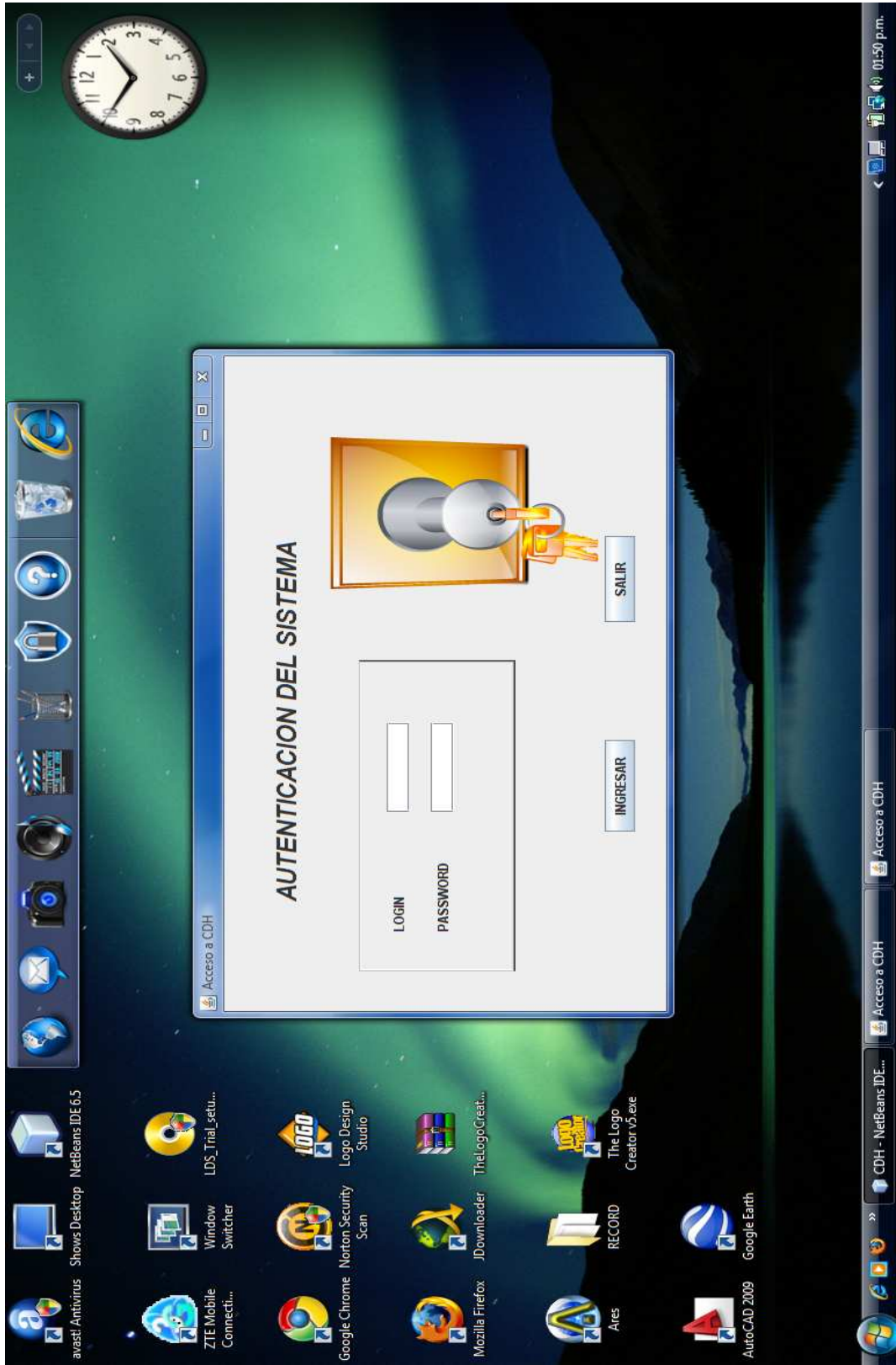
NOMBRE: equipo							
DESC: tabla							
CAMPO	TIPO DE DATO	LONG	CLAVE	UNICO	OBLIGATORIEDAD	INDEXADO	DESC
equi_cod	Númerico	20	PK	SI	SI	SI	Campo que almacena datos
equi_serial	Texto	50	——	SI	SI	SI	Campo que almacena datos
equi_marca	Númerico	11	FK	NO	SI	SI	Campo que almacena datos
equi_modelo	Texto	20	——	NO	SI	NO	Campo que almacena datos
equi_conf	Texto		——	SI	SI	NO	Campo que almacena datos

6. TABLA MOVIMIENTO EQUIPO

NOMBRE: movequipo							
DESC: tabla							
CAMPO	TIPO DE DATO	LONG	CLAVE	UNICO	OBLIGATORIEDAD	INDEXADO	DESC
moe_cod	Numérico	20	PK	SI	SI	SI	Campo que almacena datos
Moe equip	Numérico	20	——	NO	SI	NO	Campo que almacena datos
Moe_tpm	Numérico	11	——	NO	SI	SI	Campo que almacena datos
Moe_tec	Numérico	20	——	NO	SI	SI	Campo que almacena datos
Moe_fecing	Texto	10	——	NO	SI	NO	Campo que almacena datos
Moe_fecfin	Texto	10	——	NO	SI	NO	Campo que almacena datos
Moe_costo	Numérico	20	——	NO	SI	NO	Campo que almacena datos
Moe_desc	Texto		——	NO	SI	NO	Campo que almacena datos
Moe_dialogo	Texto		——	NO	SI	NO	Campo que almacena datos

7. TABLA PAGO

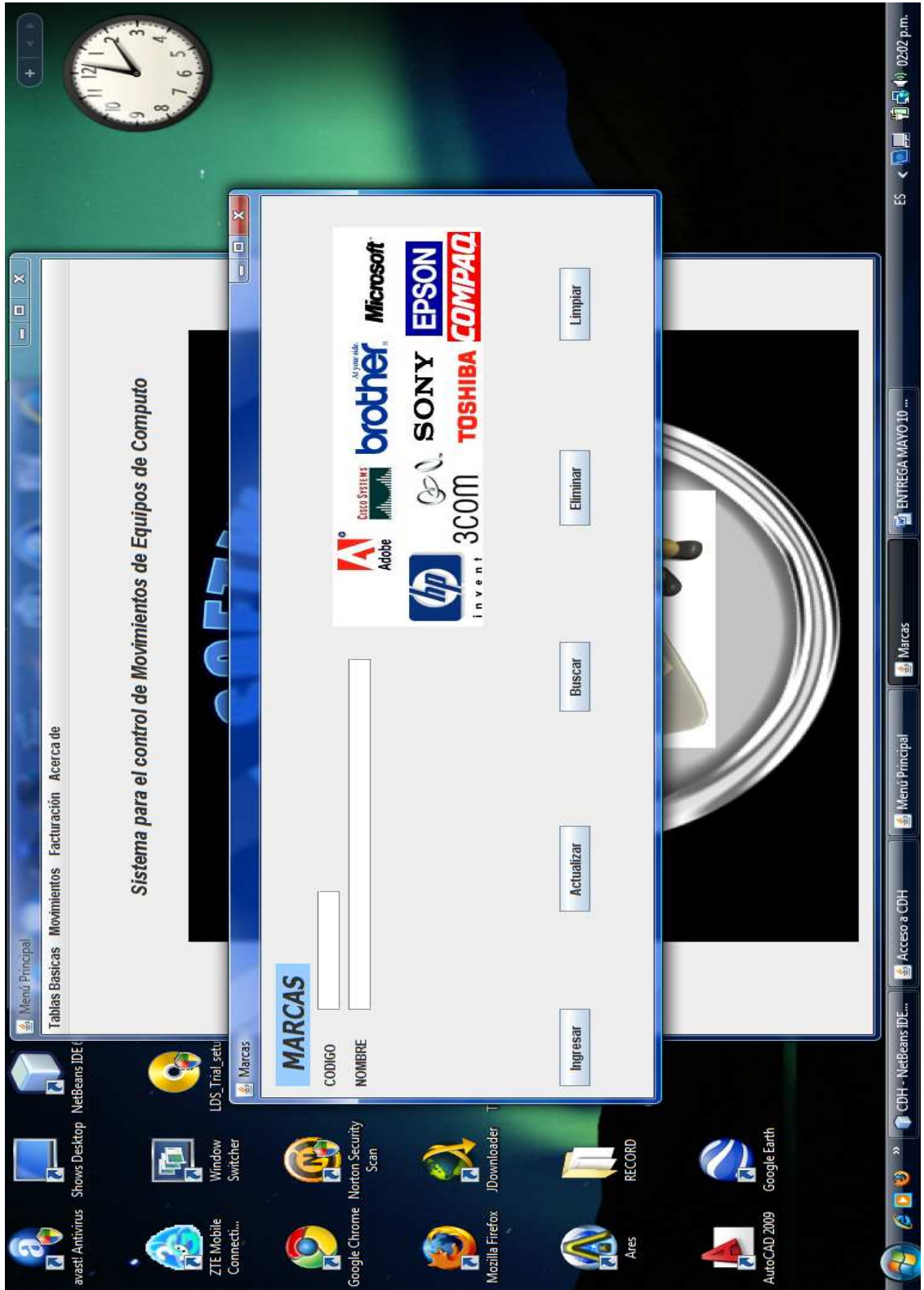
NOMBRE: pago							
DESC: tabla							
CAMPO	TIPO DE DATO	LONG	CLAVE	UNICO	OBLIGATORIEDAD	INDEXADO	DESC
Pago_conse	Numérico	20	PK	SI	SI	SI	Campo que almacena datos
Pago_cli	Numérico	20	——	NO	SI	SI	Campo que almacena datos
Pago_mov	Numérico	20	——	NO	SI	NO	Campo que almacena datos
Pago_valor	Numérico	20	——	NO	SI	NO	Campo que almacena datos
Pago_abono	Numérico	20	——	NO	SI	NO	Campo que almacena datos
Pago_saldo	Numérico	20	——	NO	SI	NO	Campo que almacena datos
Pago_tecnico	Numérico	20	——	NO	SI	NO	Campo que almacena datos
Pago_equipo	Numérico	20	——	NO	SI	NO	Campo que almacena datos
Pago_fecha	texto	20	——	NO	SI	NO	Campo que almacena datos



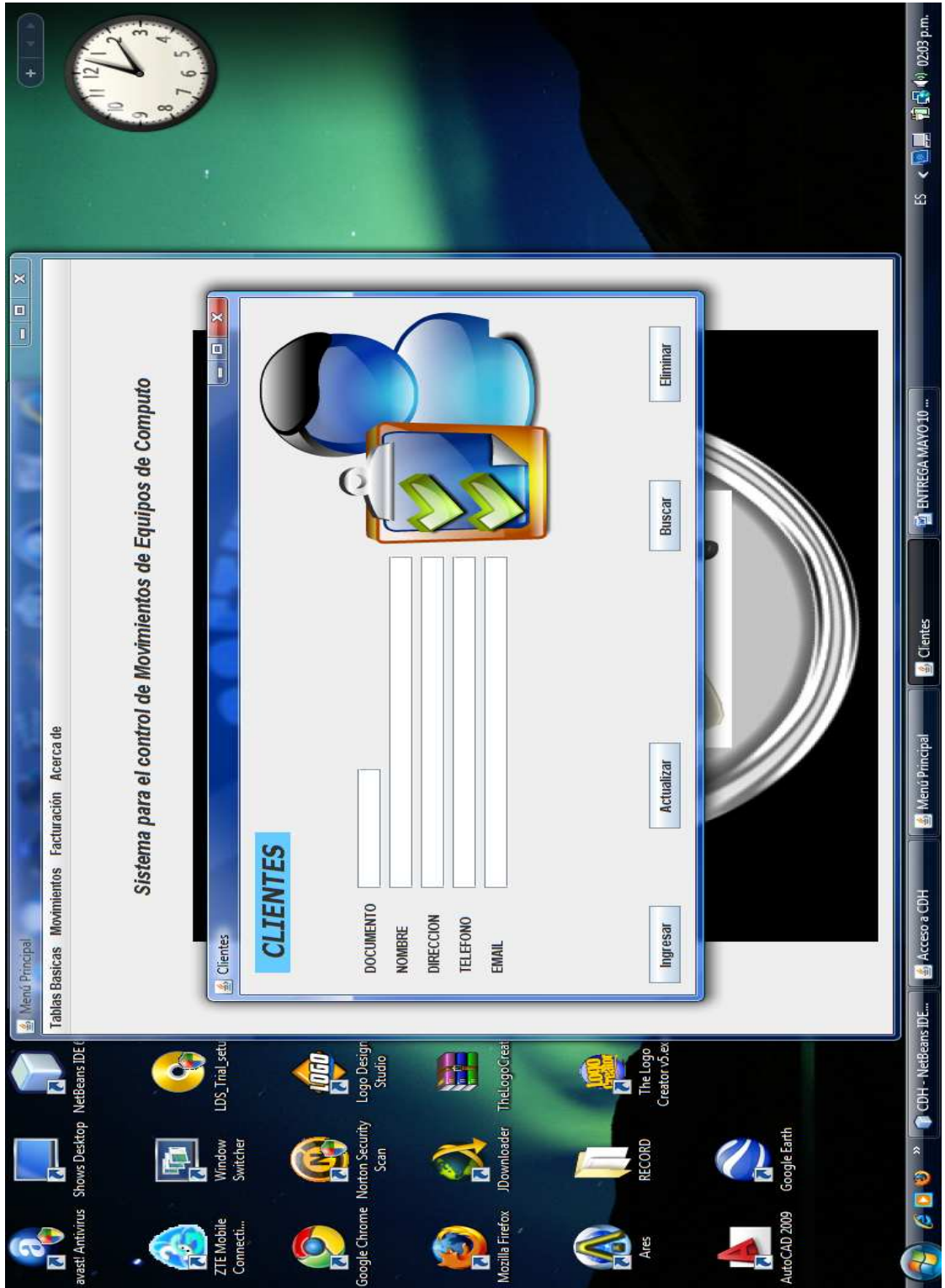








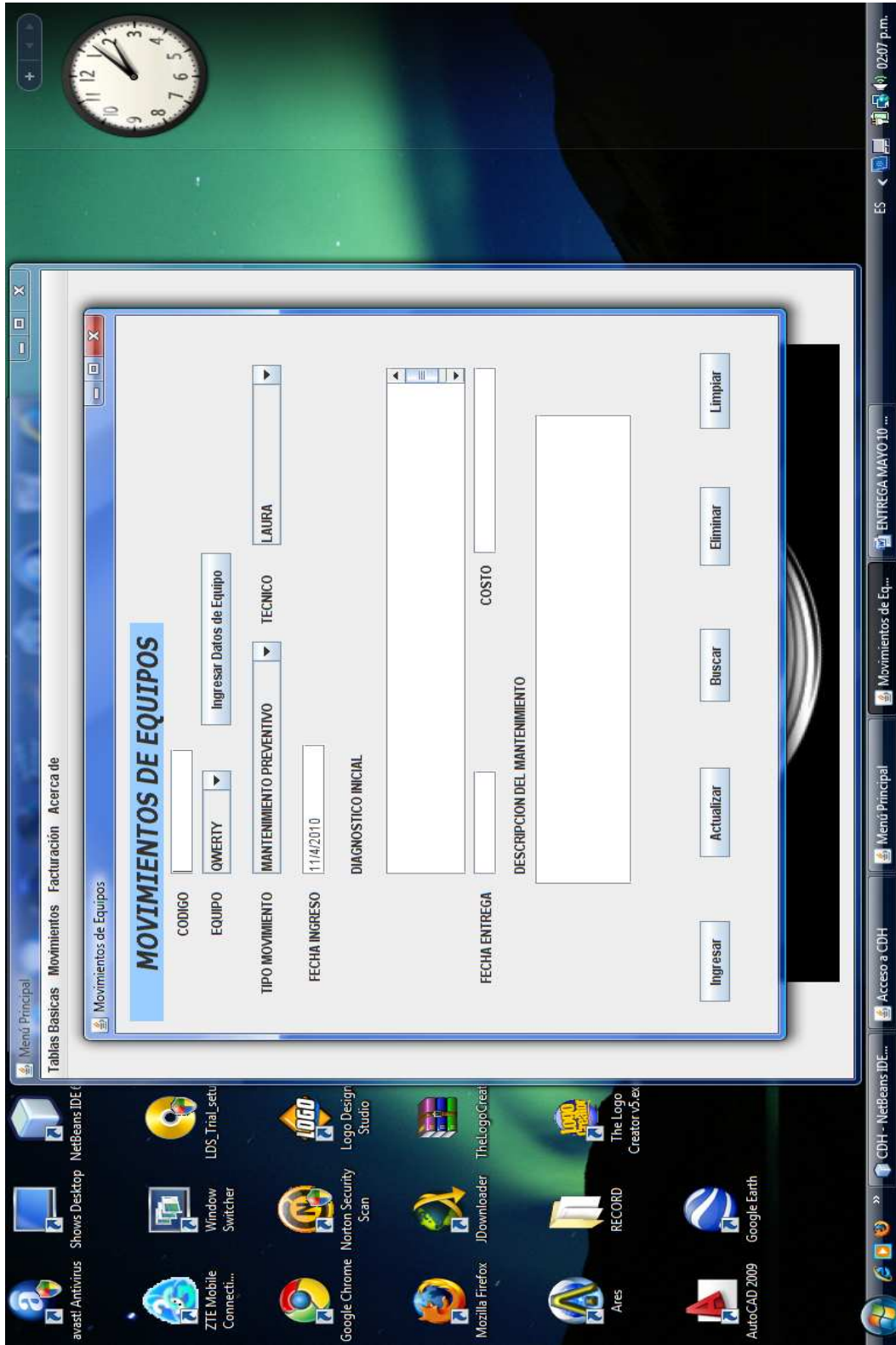






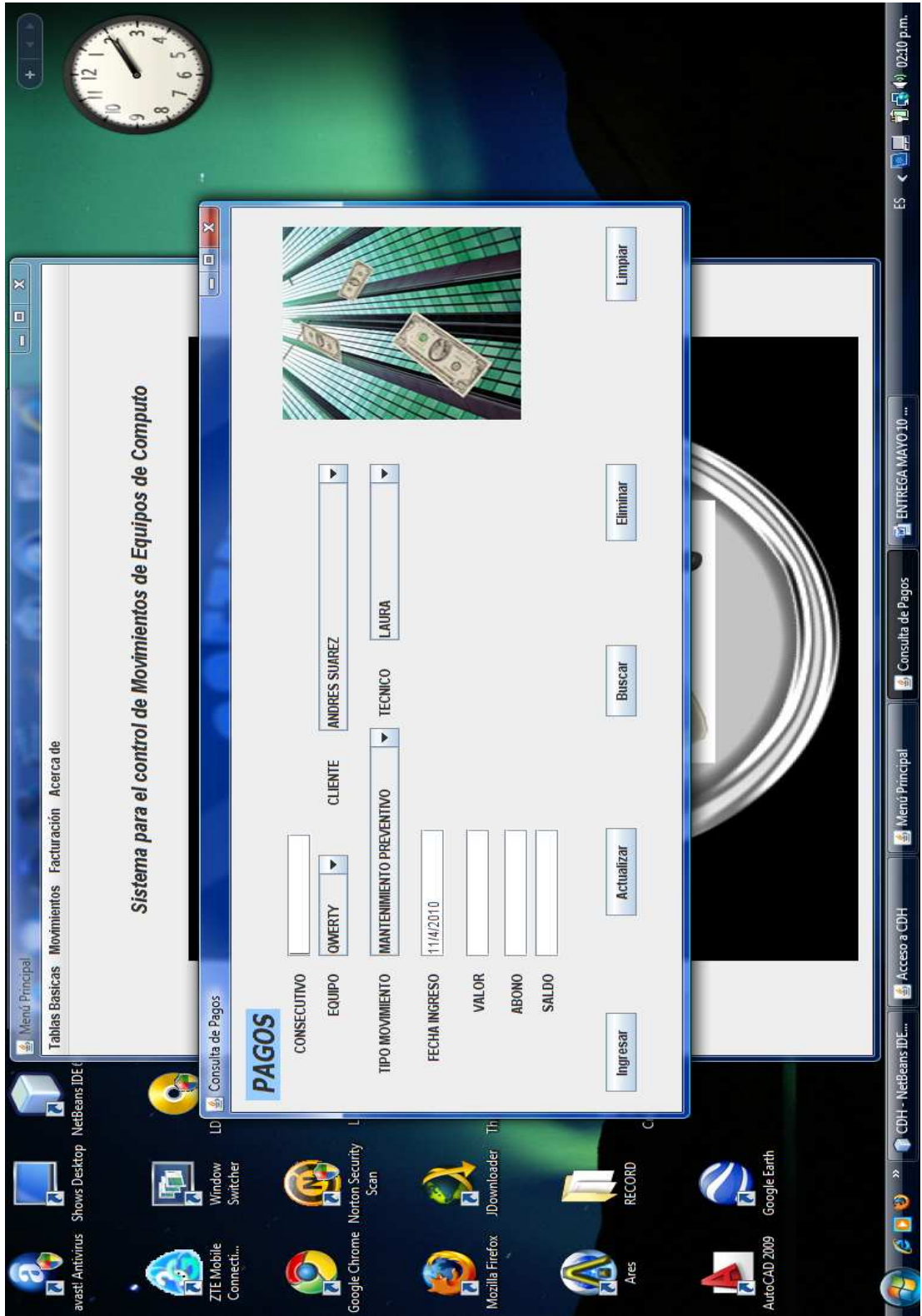




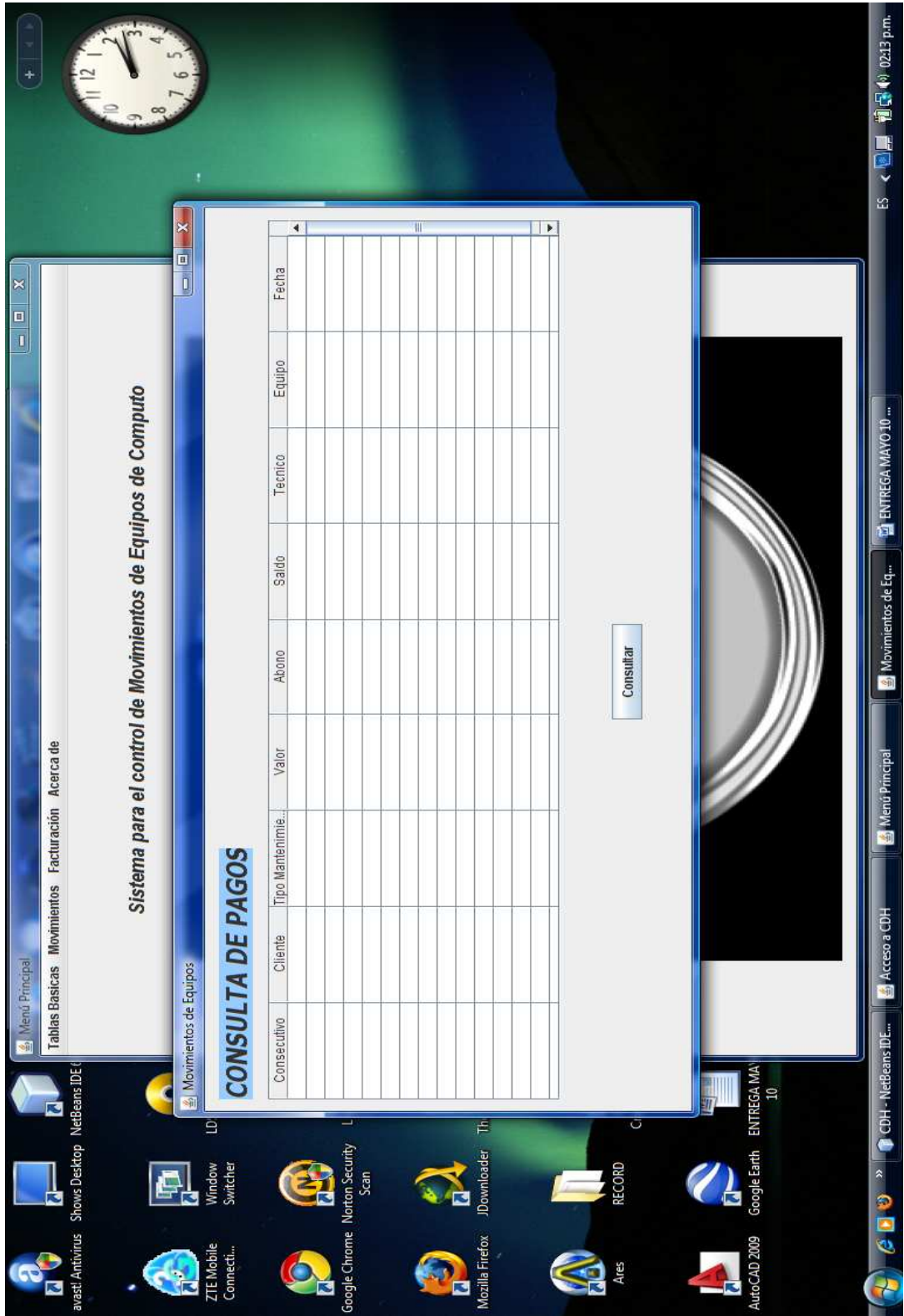




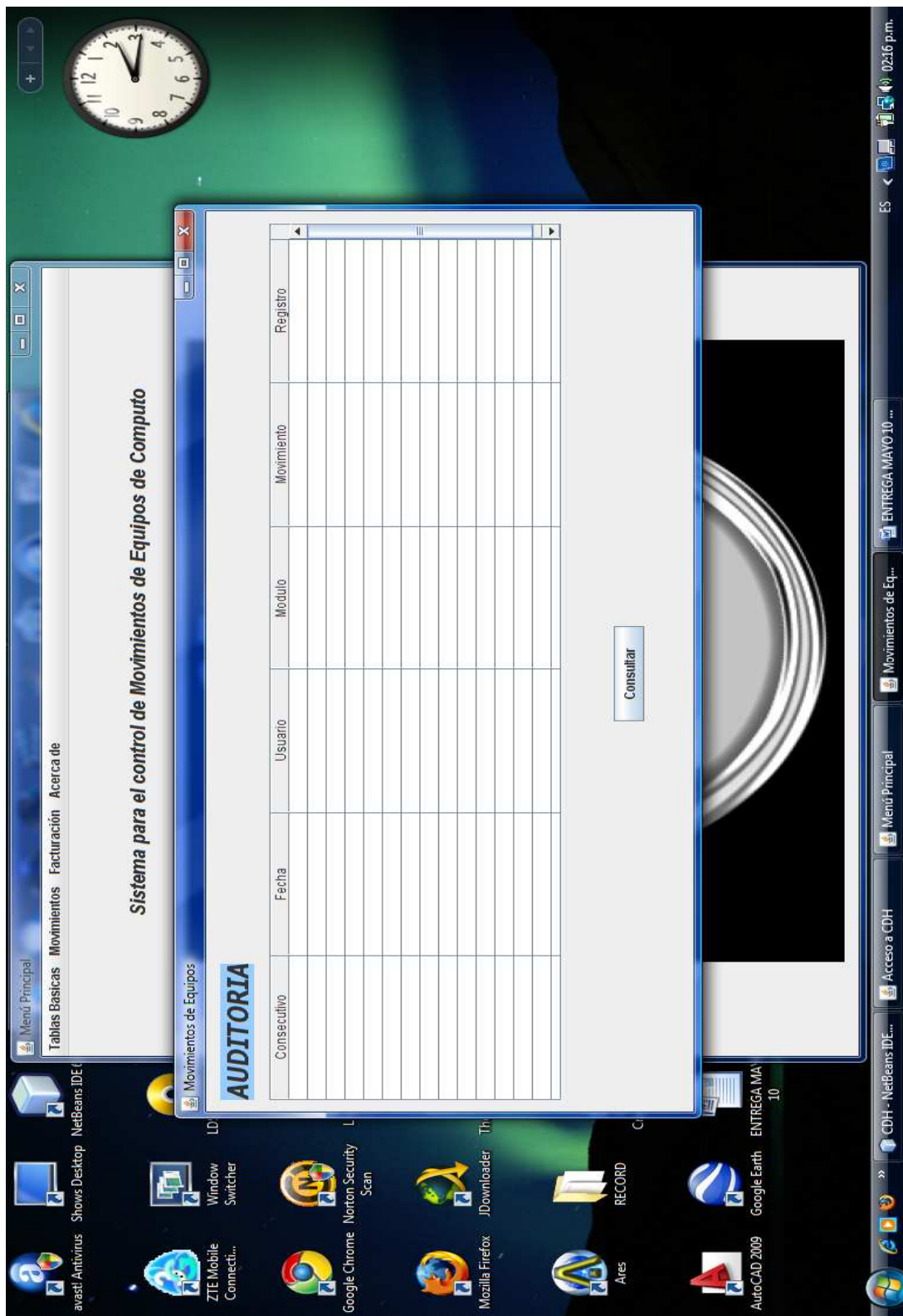














10. CONCLUSIONES

Con la elaboración del presente proyecto, se da solución a uno de los principales problemas que afectan los procesos que se llevan a cabo en la empresa *COMPUTER HKS*, como es la atención a clientes de una manera eficiente y el manejo de la información de manera oportuna, segura y veraz.

La implantación y puesta en marcha del software CDH SOFTWARE (Computer Doctor Helper) soluciona las deficiencias detectadas en el sistema actual, convirtiéndose de esta manera en un instrumento de apoyo a la gerencia en la toma de decisiones.

La automatización del sistema informático, hace que *COMPUTER HKS* entre en condiciones favorables, frente a otras empresas del mismo orden, en mercados que en estos momentos son muy competidos.

El desarrollo del presente proyecto le ha permitido a sus autoras poner en práctica los conocimientos adquiridos en el transcurso de la carrera.

11. REFERENCIAS BIBLIOGRÁFICAS

Date, C.J. Introducción a los sistemas de bases de datos. Capítulo 2. "Arquitectura de los sistemas de bases de datos". Pág. 54. Editorial Prentice Hall. Séptima Edición. 2001.

Pressman, Roger. Ingeniería del Software, un Enfoque Práctico. Pág. 24-25. Editorial Mc Graw Hill. Quinta Edición. 2002

Senn, James. Análisis y diseño de sistemas de información. Pág 43. Editorial Mc Graw Hill. Segunda edición. 1992