

“VSM”

VISUAL SUNDAY MANAGER

SISTEMA ADMINISTRADOR Y MANEJADOR DE INFORMACION  
INMOBILIARIA

ANGIE PAOLA RODRÍGUEZ RAMÍREZ

CORPORACION UNIVERSITARIA MINUTO DE DIOS

FACULTAD DE INGENIERÍA

DEPARTAMENTO DE TECNOLOGÍA EN INFORMÁTICA

BOGOTÁ, D.C

I - 2009

“VSM”

VISUAL SUNDAY MANAGER

SISTEMA ADMINISTRADOR Y MANEJADOR DE INFORMACION  
INMOBILIARIA

ANGIE PAOLA RODRÍGUEZ RAMÍREZ

Trabajo de grado para optar al título de Tecnólogo en Informática

ASESOR DE PROYECTO

LUIS FERNANDO MEDINA CARDONA

CORPORACION UNIVERSITARIA MINUTO DE DIOS

FACULTAD DE INGENIERÍA

DEPARTAMENTO DE TECNOLOGÍA EN INFORMÁTICA

BOGOTÁ, D.C

I - 2009

NOTA DE ACEPTACIÓN

---

---

---

---

---

---

---

**Firma del presidente del Jurado**

---

**Firma del Jurado**

---

**Firma del Jurado**

Bogotá D.C 18 de Enero de 2009

Dedicado a mis padres Martha Ramírez y Néstor Rodríguez quienes fueron las principales personas que creyeron en mí y me apoyaron todo el tiempo, también quiero dedicarle mi proyecto a mi hermana Ivón quien fue la persona que me motivó a continuar pese a las dificultades y fue un gran ejemplo para mí. Finalmente, con especial cariño a mi tío Edgar Rodríguez quien fue la persona que me brindó la oportunidad de comenzar un nuevo ciclo en mi vida.

## Agradecimientos

Quiero expresar mis más grandes agradecimientos a todos mis compañeros y compañeras de la carrera, que como yo también tienen sueños y metas que cumplir y que estuvieron a mi lado a lo largo de nuestro proceso de formación en la Corporación. A mi familia que siempre creyó en mí y que me apoyó incondicionalmente y a todas las personas cercanas que en algún momento dado en mi carrera me brindaron su ayuda.

## TABLA DE CONTENIDO

INTRODUCCION	16
1. PRESENTACION DEL PROBLEMA	18
1.1 TEMA	18
1.2 TITULO DEL PROYECTO	18
1.3 PLANTEAMIENTO Y DESCRIPCIÓN DEL PROBLEMA	19
1.4 ALCANCE Y JUSTIFICACIÓN	21
1.5 OBJETIVOS	22
1.5.1 Objetivo general	22
1.5.2 Objetivos específicos	23
1.6 METODOLOGÍA DE LA INVESTIGACIÓN	23
1.7 LÍNEA DE INVESTIGACIÓN	24
2. MARCO REFERENCIAL	25
2.1 ESTADO DEL ARTE	25
2.1.1 CFINMO – Software de gestión inmobiliaria:	25
2.1.2. Software de gestión inmobiliaria de GS Consultora	26
2.2 FUNDAMENTOS TEÓRICOS	30
2.2.1 Entorno de desarrollo	30
2.2.2 Lenguajes de programación	32
2.2.3 Frameworks	37
2.2.4 Servidor	40
2.2.5 Entorno de desarrollo y construcción de la base de datos	41
2.2.6 Plataforma de desarrollo	43
3. INGENIERIA DEL PROYECTO	44
3.1 METODOLOGÍA Y MODELO DE DESARROLLO	44

4. ANÁLISIS Y DISEÑO	46
4.1 DEFINICIÓN Y DESCRIPCIÓN DEL SISTEMA ACTUAL	46
4.2 DEFINICIÓN DEL SISTEMA PROPUESTO	46
4.2.1 Requerimientos del sistema	48
4.2.2 Requerimientos funcionales	48
4.2.3 Requerimientos no funcionales	49
4.3 DISEÑO DEL SISTEMA PROPUESTO	50
4.3.1 Diccionario de datos	50
4.3.2 Diagrama Entidad Relación	58
4.4 DESCRIPCIÓN DETALLADA DE CADA PROCESO	58
5. DESARROLLO	74
5.1 ESPECIFICACIONES TÉCNICAS	74
5.2 SOFTWARE	74
5.3 HARDWARE	74
5.4 CRONOGRAMA	75
5.5 RECURSOS	75
6. GLOSARIO	76
7. CONCLUSIONES	79
8. SUGERENCIAS Y RECOMENDACIONES	80
9. BIBLIOGRAFIA	81
ANEXOS	82
MANUAL DEL SISTEMA	83
MANUAL DE USUARIO	99

## LISTA DE TABLAS

Tabla 1 Diccionario de datos Clientes	51
Tabla 2 Diccionario de datos Empleado	52
Tabla 3 Diccionario de datos Inmueble	53
Tabla 4 Diccionario de datos Tipo	54
Tabla 5 Diccionario de datos Estado	54
Tabla 6 Diccionario de datos Estrato	55
Tabla 7 Diccionario de datos Barrio	55
Tabla 8 Diccionario de datos Inmueble por empleado	56
Tabla 9 Diccionario de datos Pertenece	56
Tabla 10 Diccionario de datos Arriendo	57
Tabla 11 Diccionario de datos Venta	57



## LISTA DE FIGURAS

Figura 1	Árbol AWT	39
Figura 2	Modelo Desarrollo en Espiral – Ciclo de vida del Software	45
Figura 3	Diagrama Entidad Relación	58
Figura 4	Caso de uso loguearse.	59
Figura 5	Caso de uso Gestionar Asesor	60
Figura 6	Caso de uso Gestionar Inmueble	62
Figura 7	Caso de uso Gestionar Clientes	63
Figura 8	Caso de uso Gestionar Usuarios	65
Figura 9	Caso de uso Gestionar Mercantiles	66
Figura 10	Caso de uso acciones empleado	68
Figura 11	Caso de uso Crear usuario	69
Figura 12	Caso de uso adicionar inmueble	71
Figura 13	Caso de uso Cerrar negocio	72
Figura 14	Diagrama de procesos	73
Figura 15	Cronograma	75
Figura 16	Instalar Xampp	83
Figura 17	Seleccionar Ubicación	84
Figura 18	Identificar los servicios	85
Figura 19	Seleccionar servicios	85
Figura 20	Instalar	86
Figura 21	Instalando	86
Figura 22	Finalizando instalación	87
Figura 23	Instalación completa	87
Figura 24	Verificando servicios	88

Figura 25 Verificando la correcta instalación	89
Figura 26 Nombrar la base de datos	90
Figura 27 Opción importar	90
Figura 28 Examinar base de datos	91
Figura 29 Ubicación base de datos	92
Figura 30 Creación base de datos - Continuar	92
Figura 31 Base de datos creada	93
Figura 32 Pantalla de presentación Apache TomCat	94
Figura 33 Aceptar términos de la licencia	94
Figura 34 Componentes	95
Figura 35 Ubicación	95
Figura 36 Puerto	96
Figura 37 Ubicación máquina virtual de Java	97
Figura 38 Instalando componentes	97
Figura 39 Instalación completa	98
Figura 40 Instalación completa	98
Figura 41 Pantalla Inicio De Sesión	100
Figura 42 Menú Principal	101
Figura 43 Módulo Asesores	102
Figura 44 Mensaje Asesor Registrado	102
Figura 45 Listado de Asesores	103
Figura 46 Detalle Asesores	103
Figura 47 Listado Asesores – Selección para Borrar	104
Figura 48 Listado Asesores – Selección para Modificar	105
Figura 49 Consulta Asesor – Modificando Campos	105

Figura 50 Pantalla Inicial Usando el Botón Limpiar	106
Figura 51 Ingresar Inmueble	107
Figura 52 Pantalla para Ingresar Foto del bien	108
Figura 53 Pantalla para Ingresar Foto del bien	109
Figura 54 Otras Características del Inmueble	109
Figura 55 Otras Características del Inmueble	110
Figura 56 Buscar Inmueble	111
Figura 57 Reporte de Consulta del Inmueble	111
Figura 58 detalles del inmueble	112
Figura 59 Opción de Reporte	112
Figura 60 Reporte para Imprimir	113
Figura 61 Selección del Inmueble – Para Eliminar	114
Figura 62 Selección del Inmueble – Para Eliminar	115
Figura 63 Verificar Borrado de Información	115
Figura 64 Botón Modificar Inmuebles	116
Figura 65 Módulo Clientes	117
Figura 66 Selección de lista de Clientes	118
Figura 67 Reporte de Clientes	119
Figura 68 Modificar Cliente	119
Figura 69 Crear Nueva Cuenta de Usuario	120
Figura 70 Confirmación Nueva Cuenta de Usuario	120
Figura 71 Listado Cuentas de Usuario	121
Figura 72 Eliminar Cuentas de Usuario	121
Figura 73 Seleccionar Usuario - Modificación Usuario	122
Figura 74 Módulo Mercantiles	123

Figura 75 Nuevo Cliente	124
Figura 76 Adicionar Asesor	124
Figura 77 Consulta de Negocios Cerrados	125
Figura 78 Reporte Clientes - Comprador - Asesor	126
Figura 79 Selección Modificar Negocio	127
Figura 80 Modificar Negocio	127
Figura 81 Eliminar Negocio	128

## RESUMEN

Actualmente diferentes empresas utilizan una gran variedad de herramientas informáticas y aplicativos de software que son bastante útiles para la realización eficiente de tareas, estos sistemas son muy beneficiosos para las compañías ya que una de sus principales ventajas es proporcionar agilidad en el tiempo de duración de cualquier labor, ya que realizada manualmente tardaría casi el doble de tiempo.

No todas las empresas inmobiliarias, poseen los medios necesarios para contar con tecnología que se ajuste a sus necesidades, por consiguiente manejan los métodos tradicionales para realizar sus labores, haciendo que se queden un paso más atrás en comparación con las demás empresas y que las labores de sus empleados sean demoradas y tediosas en cuanto a la consecución de información y consultas de los diferentes registros que a diario se generan en este tipo de empresas.

“**VSM**” Visual Sunday Manager es un sistema administrador y manejador de información inmobiliaria, diseñado para agilizar las labores comerciales de una empresa dedicada a la asesoría de venta y arriendo de inmuebles. Este sistema contiene una base de datos en la que se podrá registrar toda la información de los dueños de los inmuebles, de los vendedores o asesores, de los clientes compradores y de los bienes inmuebles, adicional a esto, el software brinda la posibilidad de adjuntar imágenes correspondientes a las casas, apartamentos o locales que se van a arrendar o a ofrecer para la venta.

Este sistema de información dispone de una opción para generar reportes de los inmuebles registrados, de los clientes y de los registros de ventas generadas, estos reportes son bastante útiles para revisar cómo van las ventas y evaluar si la empresa va por buen camino o si por el contrario es necesario realizar ajustes que mejoren la forma de trabajo. La consecución de información es bastante práctica y fácil y ahorra bastante tiempo en cuanto a la obtención de resultados por parte del trabajador.

“**VSM**” es un sistema bastante seguro en cuanto al manejo de información de la inmobiliaria, ya que no todos los empleados tienen acceso al aplicativo; este sistema está diseñado para que cada empleado ingrese con su usuario y contraseña, pero no todos tienen los mismos permisos de acceso a la base de datos. Maneja una interfaz agradable para el usuario y es bastante práctico y cómodo de utilizar.

## ABSTRACT

Nowadays, different companies use a great variety of computer tools and software applications which are quite useful enough for the efficient accomplishment of tasks. These systems are very beneficial for the companies because one of its main advantages is to provide agility in the time of duration of any labor, since the manual realization of them would take the double of time.

Not all real estate companies have the means to have technology that meets their needs, thus they manage traditional methods to perform their tasks, making themselves stay a backward step in comparison with other companies, and the work of its employees to be delayed and boring as to achieve information and consultation of different records that are generated daily in these companies.

**"VSM"** Sunday Visual Manager is an administrator and manager system of real estate information, which is designed to streamline the work of a business consulting company dedicated to selling and leasing property. This system contains a database in which all information from owners of properties, vendors, consultants, customers and buyers of real state. In addition, the software offers the possibility to attach pictures of houses, apartments or premises to be leased or offered for sale.

This information system has an option to generate reports of the properties registered, customers and generated sales of records, these reports are quite useful to review how the sales and assess whether the company is on track or whether contrast necessary adjustments that will improve the way they work. The pursuit of information is very practical and very easy and saves time in obtaining results from the worker.

Not all real estate companies have the means to have technology that meets their needs, thus manage to perform its traditional tasks, making them stay a step backward in comparison with other companies and that the work of its employees to be delayed and boring as to achieve information and consultation of the various records that are generated daily in these companies.

**"VSM"** Sunday Visual Manager is a system administrator and manager of real estate information, designed to streamline the work of a business consulting company dedicated to selling and leasing property. This system contains a database which will record all information from the owners of the properties of

the vendors or consultants, customers and buyers of real estate, in addition to this, the software offers the possibility to attach pictures of houses, apartments or premises to be leased or offered for sale.

This information system has an option to generate reports of the properties registered, customers and generated sales of records, these reports are quite useful to review how the sales and assess whether the company is on track or whether contrast necessary adjustments that will improve the way they work. The pursuit of information is very practical and very easy and saves time in obtaining results from the worker.

"VSM" is a fairly safe in terms of information management of real estate because not all employees have access to the application, this system is designed so that each employee login with your username and password, but not all have the same access permissions to the database. Drive for a user-friendly interface and is very practical and easy to use.

## INTRODUCCION

En el mundo actual el nivel de competencia de las empresas, crece constantemente a una gran velocidad, sin importar el sector o el tipo de servicio que ofrece, por esta razón, los empresarios necesitan adaptarse e ir evolucionando y cambiando a un ritmo acelerado, pero deben tener precaución de no descuidar la satisfacción y el servicio que el cliente se merece, de esta forma no tendrán inconvenientes quedándose atrás con respecto al nivel de la competencia brindando un excelente servicio según su actividad comercial.

El éxito de toda empresa comienza con una adecuada organización en el manejo de la información producto de su actividad, para el caso específico de las organizaciones inmobiliarias, el interés es presentar un software diseñado específicamente para este tipo de compañías.

Se consideran inmuebles todos aquellos bienes que no se pueden trasladar sin ocasionarle daños a los mismos porque forman parte de un terreno, como los son, casas, lotes, apartamentos, fincas, etc. Una inmobiliaria es una empresa que se dedica al asesoramiento de compra y venta de todo tipo de inmuebles o bienes, para estas empresas es obligatorio llevar registros de todos los bienes que tienen destinados para la venta y/o arriendo, precios, ganancias, localidades, estratos, así como los datos de las personas involucradas en el cierre de un negocio, dueños de los inmuebles y clientes compradores. En el momento en el que se tenga toda esa información en una base de datos debidamente organizada y clasificada por módulos, va a ser mucho más fácil para estas empresas manejar procesos y conclusiones correspondientes al funcionamiento y progreso de las mismas.

El sistema de información definido como “**VSM** Visual Sunday Manager”, se elaboró bajo el entorno de desarrollo NetBeans 6.5 (de Sun Microsystems [www.sun.com](http://www.sun.com)), se definió establecer la construcción del aplicativo bajo este



entorno de desarrollo integrado (**IDE**) debido a su amplio repertorio de recursos, los cuales se han destacado por ofrecer diversos esquemas de desarrollo facilitando los procesos de ingeniería y construcción de software, agregando el que es un entorno que está capacitado para trabajar con diferentes plataformas, en este caso se trabaja con java y el **FrameWork** de aplicaciones para ventanas “**Swing**” en conjunto con el apoyo de librerías como el convencional **AWT** con los que se realiza el componente de la interfaz de usuario, cabe afirmar que el manejo de reportes (y algunas consultas especializadas) fueron llevadas a cabo por medio de entornos web (**JEE**) basados en la utilización de recursos como el **HTML** y el **JSP (Java Server Page)** ya que el sistema es considerado como un híbrido entre un aplicativo de ventanas y uno web, y finalmente se utiliza **SQL** con sintaxis de BD Mysql. En cuanto al funcionamiento del módulo de reportes, se trabaja con el servidor TomCat 6.0 de JaKarta.

Este proyecto se desarrolla con el objeto de poner en práctica los conocimientos adquiridos a través del proceso de aprendizaje en la Corporación Universitaria Minuto de Dios, con el fin de optar al título de Tecnólogo en informática.

# 1. PRESENTACION DEL PROBLEMA

## 1.1 TEMA

La necesidad que se desea satisfacer con este software, es canalizar y organizar toda la información que se produce en el proceso, desde que el dueño del inmueble se acerca a colocar en manos de la empresa inmobiliaria la edificación, hasta el retorno del mismo al dueño, incluyendo también los datos de la prestadora de servicio, sus asesores comerciales, compradores y dueños.

Cuando se habla de comercio o mercadeo, siempre se va a relacionar con el hecho de ofrecer al mercado un bien, servicio o producto, en este caso la idea es centrarla en la venta y/o arrendamiento de un bien inmueble, ya sea, casa, lote, apartamento o local.

## 1.2 TITULO DEL PROYECTO

Este aplicativo que consiste en la administración y manejo de información comercial de una empresa inmobiliaria, lleva como título las siglas **VSM** por su significado, Visual Sunday Manager “administrador y manejador de información comercial” debido a que ésta será la esencia de la gestión de su trabajo.

El proyecto toma este nombre porque maneja las siguientes vertientes que es importante destacar:

➤ A nivel comercial

De acuerdo con el componente de requerimientos, la aplicación deberá considerar entre otras cosas, registros mercantiles de ventas, manejo de salarios y comisiones.

➤ Como administrador de recursos

Se hace necesario la existencia de un mecanismo capaz de gestionar (ingresar, consultar, modificar, etc.) información proveniente de los recursos comerciales de la empresa (apartamentos, lotes, bodegas, etc.) y en los cuales radica el éxito de la misma.

➤ Como manejador de recurso humano

El cual está idealizado en el registro del flujo de información propia del grupo de empleados y clientes.

### **1.3 PLANTEAMIENTO Y DESCRIPCIÓN DEL PROBLEMA**

En la actualidad todo tipo de empresa trabaja con mucha información producto de su actividad económica; las inmobiliarias, que, se dedican a la venta y arriendo de bienes o inmuebles, deben saber manejar de forma adecuada toda la información que poseen, de lo contrario, se verá reflejado un caos en toda la organización.

En las pequeñas inmobiliarias existen falencias que ocasionan que la consecución de información sea demasiado complicada e infructuosa, ya que todavía se maneja en carpetas o fólderes físicos. Existe una carpeta llamada inmuebles, que guarda la información únicamente por un código asignado en la inmobiliaria, otra llamada clientes, y una tercera denominada negocio, en la que se guardan los procesos que se generan cuando se logra arrendar o vender uno de los inmuebles registrados. Se resaltan estas tres carpetas porque son las más importantes, pero realmente existen otras que se manejan de esta misma forma. Mantener esta información en documentos físicos hace que la consecución de información cruzada sea difícil de obtener, porque, si se quiere saber qué cliente posee qué inmuebles, se hace necesario, primero,

revisar la información del cliente por apellido en su correspondiente carpeta, en ella, se consulta el código de los inmuebles a su nombre y luego se ubican en la carpeta inmuebles por los correspondientes códigos. Tampoco se puede obtener información de la cantidad de casas para la venta o arriendo que se tienen en total, ya que los bienes inmuebles se registran por código, no por tipo. No hay registros en donde se pueda identificar el nombre del antiguo dueño y del nuevo dueño, porque cuando se cierra un negocio sólo se registra el código del inmueble y se le asignan los datos del nuevo dueño.

Esta forma de llevar los registros y a la información hace que se pierda tiempo valioso en la consecución de datos en cada consulta, ocasionando demoras en los procesos típicos de la empresa dejando así, de ocupar el tiempo en otro tipo de tareas más importantes y de ofrecer un buen servicio a los clientes. El resultado del manejo de información siguiendo el anterior proceso, hace que la empresa pierda eficiencia, ya que con el avance tecnológico y la cantidad de soluciones empresariales que se pueden obtener partiendo de esta particularidad, se expone a que otras compañías ofrezcan los mismos servicios con excelentes resultados en menos tiempo y además les permite ofrecer mejores garantías para los clientes con facilidad para que puedan tener acceso al seguimiento de cada proceso.

Por estas razones, se hace necesario aportar una solución informática efectiva que conlleve a encontrar la forma en la que las inmobiliarias puedan realizar las mismas funciones de una manera eficiente y práctica, dándole libre acceso a que se puedan establecer mas tareas que ayuden al crecimiento de la compañía y a ofrecer un servicio de una mejor calidad a los clientes, quienes son el principal motor para que la empresa pueda avanzar día a día. Debido a esto surge la siguiente pregunta.

¿Cuál es la incidencia del inadecuado manejo de la información a nivel comercial de una empresa inmobiliaria? ¿Qué alternativa se puede implementar para lograr un adecuado proceso de administración?

## 1.4 ALCANCE Y JUSTIFICACIÓN

La aplicación que se va a desarrollar tendrá como funcionalidad principal manejar los procesos que se originan en una inmobiliaria, como: identificar quién fue el vendedor que cerró un negocio, quién es el dueño de un inmueble, observar cuáles inmuebles están disponibles para mostrar al cliente, los datos del cliente que adquirió un inmueble y toda la información acerca del mismo como, ubicación, precio de venta o arriendo, estrato, el área del perímetro en metros cuadrados, tiempo de construcción, imágenes y sus correspondientes reportes, que serán de gran ayuda para poder hacer seguimientos en cuanto al progreso de la compañía y procesos que se realizan en las labores diarias dentro de la empresa. Los reportes serán:

- Reporte de negocio concluido: Este reporte se va a generar por el módulo Mercantiles, una vez haya cerrado un negocio, este reporte mostrará por qué vendedor fue hecho, quién fue el cliente que hizo la compra o arriendo y la fecha en la que fue generado el reporte.
- Reporte de registros de clientes: Aquí se podrá listar todos los clientes de la inmobiliaria. Existen dos tipos de clientes, el comprador, es el interesado en adquirir o arrendar un inmueble y Cliente publicador, quien está interesado en vender o arrendar. Este módulo permitirá ver datos personales de este tipo de clientes y qué inmuebles están relacionados con ellos, ya sea como actuales y nuevos dueños.
- Reporte de detalles del inmueble: En este reporte, se observarán los datos del inmueble y los datos del cliente publicador, éste, será un soporte claro y conciso para el cliente interesado en conseguir un inmueble, ya que, allí podrá identificar todas las características del mismo, incluyendo algunas imágenes correspondientes a él.

En la actualidad, empresas con la misma actividad económica establecen su propia forma y ritmo de trabajo, hecho que implica que haya ciertas diferencias

en la ejecución procesos y funciones, sin embargo, la mayoría coinciden en contar con las herramientas tecnológicas necesarias que conllevan a lograr un excelente nivel de desempeño, pues existen muchas posibilidades que son útiles para agilizar los procesos, que realizados manualmente, tardarían un tiempo adicional. Debido a esto existe mucha competencia laboral y es muy común encontrar empresas que ofrecen servicios de excelente calidad que le permitan al cliente sentirse tranquilo con el producto que decida consumir.

La administración de una inmobiliaria es una tarea que demanda un adecuado manejo de la información generada por su actividad comercial, que permita realizar organizada y eficazmente las consultas de los diferentes procesos que se efectúan a diario, así como, el registro de nuevos datos de clientes, inmuebles y negocios. De esta manera se logrará darle una mejor visión a la empresa realizando una labor más organizada, con mejores resultados a nivel de competencia empresarial. Para poder llegar a todos esos objetivos, es de suma importancia construir un aplicativo manejador de información inmobiliaria que sea fácil de manipular por los funcionarios, pero al mismo tiempo robusto, seguro y confiable que permita la administración de todos los negocios y procesos que se generan diariamente en la empresa; el objetivo es reducir significativamente el tiempo de ejecución de las diferentes tareas que antes tomaban horas o tal vez días para obtener mejores resultados que optimicen el desempeño laboral y garanticen la productividad de la organización.

## **1.5 OBJETIVOS**

### **1.5.1 Objetivo general**

Desarrollar un aplicativo que permita administrar eficazmente la información producida como factor de la actividad comercial, en una empresa dedicada a las asesorías inmobiliarias.

### **1.5.2 Objetivos específicos**

- Llevar un control de información de todos los inmuebles que la empresa ofrece para la venta o arriendo, así como de los que quedan dentro de un negocio.
- Establecer la opción de generar reportes en los cuales se observen individualmente los negocios cerrados por asesor, características del inmueble y toda la información de los clientes involucrados con la compañía.
- Ofrecer seguridad en la información registrada en la base de datos, asignando únicamente al usuario administrador todos los permisos para que pueda realizar cualquier modificación y tenga acceso a todos los procesos que quedan registrados en la aplicación.

### **1.6 METODOLOGÍA DE LA INVESTIGACIÓN**

La metodología de investigación es un gran apoyo para el estudiante, ya que es una guía muy importante en el proceso de adquirir conocimientos y efectuar la elección del mejor método que permita desarrollar un proyecto, en este caso, “VSM”, por esta razón es de suma importancia manejar un desarrollo de software, en el cual se puedan definir unos pasos claros a seguir para lograr todos los objetivos propuestos frente a dicho proyecto.

Para este proyecto se vio conveniente trabajar bajo la metodología de investigación descriptiva<sup>1</sup>, ya que permite conseguir el levantamiento de información a través de la descripción exacta de las actividades y situaciones de los procesos y procedimientos que se realizan en una inmobiliaria para poder tener una mejor visión de la problemática que se va a atacar

---

<sup>1</sup> <http://noemagico.blogia.com/2006/091301-la-investigacion-descriptiva.php>, tomado el día 9-mar-09.

precisamente con el desarrollo de la aplicación, en este caso, la necesidad de construir un aplicativo que permita organizar la información tanto del personal que trabaja en la empresa como de los clientes y de los inmuebles que se suscriban a ella, así como, la captura de los negocios realizados, optimizando de esta manera las tareas diarias del personal y obteniendo mejores resultados para la compañía. Esta metodología de investigación parte del hecho de que hay una realidad desconocida, pero al mismo tiempo es interesante y muy importante para el desarrollador.

## **1.7 LÍNEA DE INVESTIGACIÓN**

La Corporación Universitaria Minuto de Dios ha establecido cuatro líneas de Investigación la cuales se adaptan perfectamente a los diferentes proyectos de investigación realizados en ella. La facultad de Ingeniería está inscrita a la Línea de Investigación número tres:

“Innovaciones tecnológicas y cambio social”.

Esta línea, así, como las demás pertenecientes a la Universidad, se componen de unas sub-líneas; para la línea de investigación nombrada anteriormente, el programa de Tecnología en Informática propone las siguientes sub-líneas, basadas en los perfiles profesionales de los egresados:

- Sub-línea1: Desarrollo de software
- Sub-línea2: Sistemas de información
- Sub-línea3: Redes de computadores
- Sub-línea4: Plataformas

El presente proyecto se identifica con la primera Sub-línea, ya que esta contribuye al análisis, especificación, diseño y desarrollo del sistema “VSM”.



## **2. MARCO REFERENCIAL**

### **2.1 ESTADO DEL ARTE**

En la actualidad existen varios sistemas que brindan un gran apoyo a la realización de tareas en empresas inmobiliarias, por esta razón, se realizó una investigación de los diferentes sistemas existentes con características muy similares a las del proyecto a desarrollar, que también cumplan con suplir las necesidades en cuanto al manejo de información y organización para la empresa inmobiliaria.

#### **2.1.1 CFINMO<sup>2</sup> – Software de gestión inmobiliaria:**

Es un software especialmente creado para resolver la problemática de la gestión comercial inmobiliaria, ya que gestiona bienes destinados a la compra, venta y traspaso, este sistema es como un asistente para realizar tareas, porque es capaz de llevar a cabo el cruce de ofertas con demandas e indicarle al usuario a cuáles productos es rentable revisarle los precios asignados. Este es un programa muy completo que ofrece al cliente una variedad de soluciones y ayudas para hacer más fácil su trabajo.

Este software cuenta con dos módulos opcionales de búsqueda y publicación, que son:

- **Publicación Internet:** Con este módulo, el cliente podrá subir desde internet las publicaciones que tiene destinadas para ofrecer desde la opción de publicación, mediante un proceso automático permitiendo mostrar los datos generales y tres fotografías de cada inmueble. El

---

<sup>2</sup> <http://cfinmo.cisisl.com/>, Software inmobiliario, tomado el día 2-mar-09.

usuario que desee buscar productos desde su web, tiene las opciones de seleccionar la población, zona, tipo de producto, obra nueva, habitaciones, superficie e importe, de esta manera, el cliente encontrará justo lo que desea ver.

- Pantalla Táctil: Como su nombre lo dice, permite realizar búsqueda de inmuebles y publicarlos utilizando una pantalla táctil, haciendo más interesante y atractiva la forma en que se muestra la información tanto para clientes como para el administrador.

Este software tiene las especificaciones básicas para mostrar el producto con toda la información necesaria al cliente, pero se basa más en una presentación superficial de lo que la inmobiliaria necesita, aunque tiene bastantes requerimientos similares a los del proyecto a desarrollar, como búsqueda de inmuebles, con posibilidad de generar informe; existen bastantes diferencias y sobre ellas se puede visualizar el valor agregado que va a tener el proyecto, como, la posibilidad de crear un usuario con las seguridades necesarias para mantener toda la información a salvo, además de ofrecer la posibilidad de generar informes para poder realizar seguimientos en los procesos realizados para la empresa y si se está llevando por buen camino según las cifras arrojadas. El software ofrece varias posibilidades pensadas para la organización de toda la información que se acostumbra a manejar en empresas inmobiliarias.

### **2.1.2. Software de gestión inmobiliaria de GS Consultora<sup>3</sup>**

El sistema de gestión de inmobiliarias de GS Consultora brinda todas las herramientas de control para realizar un trabajo ordenado y eficiente de gestión de las propiedades que están a su cargo. Cada propiedad genera una carpeta debidamente identificada, sobre la cual se realiza la carga y seguimiento de todos los movimientos vinculados a la misma - tanto a nivel contable como

---

<sup>3</sup> <http://www.datahousecompany.com.ar/sistema-gestion-inmobiliarias.html>, tomado el día 9-mar-09

referente a documentación - . Desde el contrato de alquiler hasta el chequeo de la documentación a entregar y recibir, emisión de recibos y comprobantes, con la posibilidad de agrupar inmuebles por propietario, cobros, pagos, comisiones y liquidaciones.

Características:

El sistema le brinda potentes prestaciones a través de una interface de sencillo uso. Entre sus características, se cuenta con:

- Sistema multiusuario, multiempresa y multimonetario
- Manual en línea (*help*) vinculado temáticamente al módulo que esté operando en ese momento.
- Informes previsualizables por pantalla, con opción impresión y exportación a otros formatos (texto, **Excel**, **Word**)
- Acceso restringido por claves a tres niveles, definido por el usuario (administrador, operador y consulta)

El aplicativo cuenta con los siguientes módulos, pero así mismo maneja cuatro versiones para que se ajusten a la necesidad del cliente y de su presupuesto, es decir, la versión más pequeña maneja unos módulos, no todos, llegando así a la versión más grande que es la que contiene el paquete con todas las herramientas completas.

- Módulo de administración de alquileres estándar

Registra el contrato de alquiler; carga los datos del propietario, del inmueble, inquilino, garantes, datos económicos, totalidad de la documentación a recibir / entregar durante el período de duración del contrato, y proyección económica de la totalidad de los alquileres. Además utiliza un módulo de informes para el control de documentación a recibir / recibida / a entregar / entregada, así como vencimientos de las mismas.

- Módulo de administración de alquileres temporarios

Para alquileres acotados a días o semanas. Para la administración de propiedades en lugares turísticos (alquiler por temporada), contratación de oficinas temporales, etc. El módulo incluye una grilla de múltiple visualización que proyecta la disponibilidad de los inmuebles.

- Módulo de gestión de tesorería

Puede definir cajas y monedas diferentes según su uso: fondo fijo de la inmobiliaria, fondos por inmueble o propietario. Realiza transferencias entre monedas, cajas, o aplica movimientos a una carpeta determinada. Además el sistema cuenta con la opción de realizar cierres de caja, que bloquean movimientos con fecha anterior. Una vez "*certificado*" el saldo de caja, se realiza el cierre y no se pueden registrar movimientos de caja con fecha anterior o igual a la del cierre. Realice la liquidación de comisiones, gastos y fondos ingresados por propiedad o grupo de propiedades, en caso de pertenecer a un mismo propietario.

- Módulo de facturación

Lleva la gestión de las cuentas corrientes de clientes, con emisión de documentos (facturas, notas de crédito y débito).

- Módulo de administración de notas

Registra notas en forma independiente para cada entidad: propietarios, propiedades e inquilinos. Si bien a estos *memos* se puede acceder desde otros sitios dentro de la estructura del sistema, también se tiene un acceso directo para facilitar el acceso a usuarios de bajo nivel y limitados permisos.

- Módulo de administración de vencimientos

Los operadores de sistema ahora pueden registrar los distintos vencimientos, e imputarlos a una propiedad o contrato. De este modo, se incluye automáticamente en la liquidación que se le entrega al propietario.

- Módulo de utilitarios

Dentro de esta opción se agrupan funciones que en términos generales están restringidas a usuarios de alto nivel y de permisos amplios. Se le puede adicionar el sub-módulo Biblioteca, que arma un índice de documentos asociados, y a cada uno de ellos se le puede asignar una o más entidades del sistema, con opciones de búsqueda y selección. De este modo, es posible acceder a la apertura automática de esos documentos, como por ejemplo: copia del contrato de alquiler, registro de firmas, fotografías de las propiedades, etc.

- Módulo de gestión de compras

Registra Facturas, Notas de Débito / Crédito de sus proveedores. Asigna los documentos a las propiedades correspondientes. Su registro pasa los datos pertinentes al Libro IVA y al Libro Percepciones. Efectúe los pagos necesarios a través del módulo Tesorería. Con opciones para realizar ajustes de saldos, emisión de informes.

Este software es muy completo, pero va dirigido inicialmente a grandes empresas ya que posee bastantes características que son de gran ayuda para los procesos y movimientos de tipo contable que se pueden requerir en una empresa del sector inmobiliario.

Visual Sunday Manager, es un aplicativo que surge debido a la necesidad que tienen las inmobiliarias de manejar organizadamente su información para así poder obtener buenos resultados a nivel empresarial e ir creciendo gracias a las evaluaciones que se pueden realizar con sólo observar los reportes generados por la aplicación, sin embargo, este es un proyecto pensado para empresas que hasta ahora empiecen a ejercer en este negocio, obviamente buscando mejorar cada vez más para llegar a ofrecer algo mucho más completo e ir adaptándolo a la primera versión del sistema.

## 2.2 FUNDAMENTOS TEÓRICOS

### 2.2.1 Entorno de desarrollo

**NetBeans:** Se refiere a una plataforma para el desarrollo de aplicaciones de escritorio usando Java y a un entorno de desarrollo integrado (IDE) desarrollado usando la Plataforma NetBeans.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados *módulos*. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento, y con cerca de 100 socios en todo el mundo. Sun Microsystems fundó el proyecto de código abierto NetBeans en junio 2000 y continúa siendo el patrocinador principal de los proyectos.

NetBeans comenzó como un proyecto estudiantil en República Checa (originalmente llamado Xelfi), en 1996 bajo la tutoría de la Facultad de Matemáticas y Física en la Universidad de Charles en Praga. La meta era escribir un entorno de desarrollo integrado (IDE) para Java parecida a la de Delphi.

Tiempo después, ellos fueron contactados por Roman Stanek, un empresario que ya había estado relacionado con varias iniciativas en la República Checa. Él estaba buscando una buena idea en que invertir, y encontró en Xelfi una buena oportunidad. Ellos se reunieron, y el negocio surgió.

El plan original era desarrollar unos componentes JavaBeans para redes. Jarda Tulach, quien diseñó la arquitectura básica de la IDE, surgió con la idea de llamarlo NetBeans, con el fin de describir lo que ellos harían. Cuando las especificaciones de los Enterprise JavaBeans salieron, ellos decidieron trabajar con este estándar, ya que no tenía sentido competir con él, sin embargo el nombre de NetBeans se quedó.

En la primavera de 1999, Netbeans DeveloperX2 fue lanzado, soportando Swing. Las mejoras de rendimiento que llegaron con el JDK 1.3, lanzado en otoño de 1999, hicieron a NetBeans una alternativa realmente viable para el desarrollo de herramientas. En el verano de 1999, el equipo trabajó duro para rediseñar a DeveloperX2 en un NetBeans más modular, lo que lo convirtió en la base de NetBeans hoy en día.

Sun adquirió otra compañía de herramientas al mismo tiempo, Forté, y decidió renombrar NetBeans a Forté for Java. El nombre de NetBeans desapareció de vista por un tiempo.

Seis meses después, se tomó la decisión de hacer a NetBeans open source. Mientras que Sun había contribuido considerablemente con líneas de código en varios proyectos de código abierto a través de los años, NetBeans se convirtió en el primer proyecto de código abierto patrocinado por ellos. En Junio del 2000 NetBeans.org fue lanzado.

**IDE:** Un **entorno de desarrollo integrado** o, en inglés, ***Integrated Development Environment*** (***IDE***), es un programa compuesto por un conjunto de herramientas para un programador.

Puede dedicarse en exclusiva a un sólo lenguaje de programación o bien, poder utilizarse para varios.

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica GUI. Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. El lenguaje Visual Basic, por ejemplo, puede ser usado dentro de las aplicaciones de Microsoft Office, lo que hace posible escribir sentencias Visual Basic en forma de macros para Microsoft Word.

Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, Python, Java, C#, Delphi, Visual Basic, etc. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución, en donde se permite utilizar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto, como es el caso de Smalltalk u Objective-C.

Es posible que un mismo IDE pueda funcionar con varios lenguajes de programación. Este es el caso de Eclipse, al que mediante pluggins se le puede añadir soporte de lenguajes adicionales.

## Componentes

- Un editor de texto.
- Un compilador.
- Un intérprete.
- Herramientas de automatización.
- Un depurador.
- Posibilidad de ofrecer un sistema de control de versiones.
- Factibilidad para ayudar en la construcción de interfaces gráficas de usuarios.

### 2.2.2 Lenguajes de programación

- **HTML:** HyperText Markup Language (*Lenguaje de Marcas de Hipertexto*), es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de "etiquetas", rodeadas por corchetes angulares (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un *script* (por ejemplo Javascript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

HTML también es usado para referirse al contenido del tipo de MIME text/html o todavía más ampliamente como un término genérico para el HTML, ya sea en forma descendida del XML (como XHTML 1.0 y posteriores) o en forma descendida directamente de SGML (como HTML 4.01 y anteriores).

Por convención, los archivos de formato HTML usan la extensión .htm o .html.

La primera descripción de HTML disponible públicamente fue un documento llamado *HTML Tags* (Etiquetas HTML), publicado por primera vez en Internet por Tim Berners-Lee en 1991. Describe 22 elementos comprendiendo el diseño inicial y relativamente simple de HTML. Trece de estos elementos todavía existen en HTML 4.



Berners-Lee consideraba a HTML una ampliación de SGML, pero no fue formalmente reconocida como tal hasta la publicación de mediados de 1993, por la IETF, de una primera proposición para una especificación de HTML: el boceto *Hypertext Markup Language* de Berners-Lee y Dan Connolly, el cual incluía una Definición de Tipo de Documento SGML para definir la gramática. El boceto expiró luego de seis meses, pero fue notable por su reconocimiento de la etiqueta propia del navegador Mosaic usada para insertar imágenes sin cambio de línea, reflejando la filosofía del IETF de basar estándares en prototipos con éxito. Similarmente, el boceto competidor de Dave Raggett *HTML+ (Hypertext Markup Format)* (Formato de marcaje de hipertexto), de 1993 tardío, sugería, estandarizar características ya implementadas tales como tablas.

- **SQL: El Lenguaje de consulta estructurado** (Structured Query Language) es un lenguaje declarativo de acceso a bases de datos relacionales que permite especificar diversos tipos de operaciones sobre las mismas. Una de sus características es el manejo del álgebra y el cálculo relacional permitiendo lanzar consultas con el fin de recuperar - de una forma sencilla- información de interés de una base de datos, así como también hacer cambios sobre la misma.

Los orígenes del SQL están ligados a los de las bases de datos relacionales. En 1970 E. F. Codd propone el modelo relacional y asociado a este un sublenguaje de acceso a los datos basado en el cálculo de predicados. Basándose en estas ideas, los laboratorios de IBM definen el lenguaje SEQUEL (Structured English QUery Language) que más tarde sería ampliamente implementado por el SGBD (Sistemas Gestores de Bases de Datos) experimental System R, desarrollado en 1977 también por IBM. Sin embargo, fue Oracle quien lo introdujo por primera vez en 1979 en un programa comercial.

El SEQUEL terminaría siendo el predecesor de SQL, siendo éste una versión evolucionada del primero. El SQL pasa a ser el lenguaje por excelencia de los diversos SGBD relacionales surgidos en los años siguientes y es por fin estandarizado en 1986 por el ANSI, dando lugar a la primera versión estándar de este lenguaje, el "SQL-86" o "SQL1". Al año siguiente este estándar es también adoptado por la ISO.

Sin embargo este primer estándar no cubre todas las necesidades de los desarrolladores e incluye funcionalidades de definición de almacenamiento que se consideraron suprimir. Así que en 1992 se lanza

un nuevo estándar ampliado y revisado del SQL llamado "SQL-92" o "SQL2".

En la actualidad el SQL es el estándar *de facto* de la inmensa mayoría de los SGBD comerciales. Y, aunque la diversidad de añadidos particulares que incluyen las distintas implementaciones comerciales del lenguaje es amplia, el soporte al estándar SQL-92 es general y muy amplio.

- **JAVA:** La tecnología Java se creó como una herramienta de programación para ser usada en un proyecto de set-top-box en una pequeña operación denominada *the Green Project* en Sun Microsystems en el año 1991. El equipo (*Green Team*), compuesto por trece personas y dirigido por James Gosling, trabajó durante 18 meses en Sand Hill Road en Menlo Park en su desarrollo.

El lenguaje se denominó inicialmente *Oak* (por un roble que había fuera de la oficina de Gosling), luego pasó a denominarse *Green* tras descubrir que *Oak* era ya una marca comercial registrada para adaptadores de tarjetas gráficas y finalmente se renombró a *Java*.

El término Java fue acuñado en una cafetería frecuentada por algunos de los miembros del equipo. Pero no está claro si es un acrónimo o no, aunque algunas fuentes señalan que podría tratarse de las iniciales de sus creadores: *James Gosling*, *Arthur Van Hoff*, y *Andy Bechtolsheim*. Otros abogan por el siguiente acrónimo, *Just Another Vague Acronym* ("sólo otro acrónimo ambiguo más"). La hipótesis que más fuerza tiene es la que Java debe su nombre a un tipo de café disponible en la cafetería cercana, de ahí que el icono de java sea una taza de café caliente. Un pequeño signo que da fuerza a esta teoría es que los 4 primeros bytes (el *número mágico*) de los archivos .class que genera el compilador, son en hexadecimal, 0xCAFEBABE. Otros simplemente dicen que el nombre fue sacado al parecer de una lista aleatoria de palabras.

Los objetivos de Gosling eran implementar una máquina virtual y un lenguaje con una estructura y sintaxis similar a C++. Entre junio y julio de 1994, tras una sesión maratoniana de tres días entre John Gage, James Gosling, Joy Naughton, Wayne Rosing y Eric Schmidt, el equipo reorientó la plataforma hacia la Web. Sintieron que la llegada del navegador web Mosaic, propiciaría que Internet se convirtiese en un medio interactivo, como el que pensaban era la televisión por cable.

Naughton creó entonces un prototipo de navegador, WebRunner, que más tarde sería conocido como HotJava.

El 9 de enero del año 1996, Sun fundó el grupo empresarial JavaSoft para que se encargase del desarrollo tecnológico. Dos semanas más tarde la primera versión de Java fue publicada.

La promesa inicial de Gosling era *Write Once, Run Anywhere* (Escríbelo una vez, ejecútalo en cualquier lugar), proporcionando un lenguaje independiente de la plataforma y un entorno de ejecución (la JVM) ligero y gratuito para las plataformas más populares de forma que los binarios (bytecode) de las aplicaciones Java pudiesen ejecutarse en cualquier plataforma.

El entorno de ejecución era relativamente seguro y los principales navegadores web pronto incorporaron la posibilidad de ejecutar applets Java incrustadas en las páginas web.

Java es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de los años 90. El lenguaje en sí mismo toma mucha de su sintaxis de C y C++, pero tiene un modelo de objetos más simple y elimina herramientas de bajo nivel, que suelen inducir a muchos errores, como la manipulación directa de punteros o memoria.

- **JavaServer Pages (JSP)** es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

Esta tecnología es un desarrollo de la compañía Sun Microsystems. La Especificación JSP 1.2 fue la primera que se liberó y en la actualidad está disponible la Especificación JSP 2.1.

Las JSP's permiten la utilización de código Java mediante scripts. Además, es posible utilizar algunas acciones JSP predefinidas mediante etiquetas. Estas etiquetas pueden ser enriquecidas mediante la utilización de Librerías de Etiquetas (TagLibs o Tag Libraries) externas e incluso personalizadas.

JSP puede considerarse como una manera alternativa, y simplificada, de construir servlets. Es por ello que una página JSP puede hacer todo lo que un servlet puede hacer, y viceversa. Cada versión de la

especificación de JSP está fuertemente vinculada a una versión en particular de la especificación de servlets.

El funcionamiento general de la tecnología JSP es que el Servidor de Aplicaciones interpreta el código contenido en la página JSP para construir el código Java del servlet a generar. Este servlet será el que genere el documento (típicamente HTML) que se presentará en la pantalla del Navegador del usuario.

El rendimiento de una página JSP es el mismo que tendría el servidor equivalente, ya que el código es compilado como cualquier otra clase Java. A su vez, la máquina virtual compilará dinámicamente a código de máquina las partes de la aplicación que lo requieran. Esto hace que JSP tenga un buen desempeño y sea más eficiente que otras tecnologías web que ejecutan el código de una manera puramente interpretada.

La principal ventaja de **JSP** frente a otros lenguajes es que el lenguaje Java es un lenguaje de propósito general que excede el mundo web y que es apto para crear clases que manejen lógica de negocio y acceso a datos de una manera prolija. Esto permite separar en niveles las aplicaciones web, dejando la parte encargada de generar el documento HTML en el archivo JSP.

Otra ventaja es que JSP hereda la portabilidad de Java, y es posible ejecutar las aplicaciones en múltiples plataformas sin cambios. Es común incluso que los desarrolladores trabajen en una plataforma y que la aplicación termine siendo ejecutada en otra.

Los servlets y Java Server Pages (JSPs) son dos métodos de creación de páginas web dinámicas en servidor usando el lenguaje Java. En ese sentido son similares a otros métodos o lenguajes tales como el PHP, ASP o los CGIs (common gateway interface), programas que generan páginas web en el servidor. Sin embargo, se diferencian de ellos en otras cosas.

Para empezar, los JSPs y servlets se ejecutan en una máquina virtual Java, lo cual permite que, en principio, se puedan usar en cualquier tipo de ordenador, siempre que exista una máquina virtual Java para él. Cada servlet o JSP, se ejecuta en su propia hebra, es decir, en su propio contexto; pero no se comienza a ejecutar cada vez que recibe una petición, sino que persiste de una petición a la siguiente, de forma que no se pierde tiempo en invocarlo (cargar programa + intérprete). Su persistencia le permite también hacer una serie de cosas de forma más eficiente: conexión a bases de datos y manejo de sesiones, por ejemplo.

Los JSPs son en realidad servlets: un JSP se compila a un programa en Java la primera vez que se invoca, y del programa en Java se crea una clase que se empieza a ejecutar en el servidor como un servlet. La principal diferencia entre los servlets y los JSPs es el enfoque de la programación: un JSP es una página Web con etiquetas especiales y código Java incrustado, mientras que un servlet es un programa Java puro que recibe peticiones y genera a partir de ellas una página web.

### 2.2.3 Frameworks

- **Framework:** En el desarrollo de software, un Framework es una estructura de soporte definida, mediante la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y una metodología de trabajo la cual extiende o utiliza las aplicaciones del dominio. Son diseñados con el intento de facilitar el desarrollo de software, permitiendo a los diseñadores y programadores pasar más tiempo identificando requerimientos de software que tratando con los tediosos detalles de bajo nivel de proveer un sistema funcional.

Dentro de este aspecto, el sistema se basa en el modelo MVC (Controlador => Modelo => Vista) ya que se requiere fragmentar en módulos de programación. Siendo útil contemplar estos aspectos básicos en cuanto a la implementación del sistema en cuestión:

#### **Controlador:**

Con este apartado se controla el acceso (incluso todo) a la aplicación, esto puede ser: archivos, scripts o programas; cualquier tipo de información que permita la interfaz. Así, se puede diversificar el contenido de forma dinámica, y estática (a la vez); pues, sólo se necesita controlar ciertos aspectos (como se ha mencionado antes).

**Modelo:**

Este miembro del controlador maneja las operaciones lógicas, y de manejo de información (previamente enviada por su ancestro) para resultar de una forma explicable, y sin titubeos. Cada miembro debe ser meticulosamente llamado, en su correcto nombre y en principio, con su verdadera naturaleza: el manejo de información, su complementación directa.

**Vista:**

Al final, a este miembro de la familia le corresponde dibujar, o expresar la última forma de los datos: la interfaz gráfica que interactúa con el usuario final del programa (GUI). Después de todo, a este miembro le toca evidenciar la información obtenida hasta hacerla llegar con el controlador. Solo (e inicialmente), se espera demostrar la información.

- La **Abstract Window Toolkit AWT**, (en español “Kit de Herramientas de Ventana Abstracta”) es un kit de herramientas de gráficos, interfaz de usuario, y sistema de ventanas independiente de la plataforma original de Java. AWT es ahora parte de las Java Foundation Classes (JFC) - la API estándar para suministrar una interfaz gráfica de usuario (GUI) para un programa Java.

Cuando Sun Microsystems liberó Java en 1995, AWT suministró solo un nivel de abstracción muy fino sobre la interfaz de usuario nativa subyacente. Por ejemplo, crear una caja de verificación AWT causaría que AWT directamente llame a la subrutina nativa subyacente que cree una caja de verificación. Sin embargo, una caja de verificación en Microsoft Windows no es exactamente lo mismo que una caja de verificación en Mac OS o en los distintos tipos de UNIX. Algunos desarrolladores de aplicaciones prefieren este modelo porque suministra un alto grado de fidelidad al kit de herramientas nativo subyacente y mejor integración con las aplicaciones nativas. En otras palabras, un programa GUI escrito usando AWT parece como una aplicación nativa Microsoft Windows cuando se ejecuta en Windows, pero el mismo programa parece una aplicación nativa Apple Macintosh cuando se ejecuta en un Mac, etc. Sin embargo, algunos desarrolladores de aplicaciones desprecian este modelo porque prefieren que sus aplicaciones se vean exactamente igual en todas las plataformas.

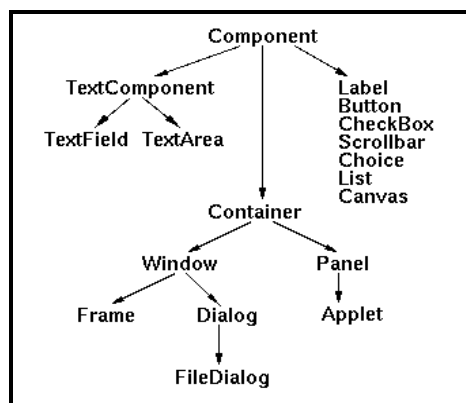
En J2SE 1.2, los widgets de AWT fueron ampliamente superados por aquellos del kit de herramientas Swing. Además de proveer un conjunto

más rico de widgets UI, Swing dibuja sus propios widgets (usando Java 2D para llamar a las subrutinas de bajo nivel en el subsistema de gráficos local) en lugar de confiar en el módulo de interfaz de usuario de alto nivel del sistema operativo. Swing suministra la opción de usar un aspecto nativo o de plataforma cruzada para la aplicación.

AWT continúa suministrando el núcleo del subsistema de eventos GUI y la interfaz entre el sistema de ventanas nativo y la aplicación Java, suministrando la estructura que necesita Swing. También suministra gestores de disposición básicos, un paquete de transferencia de datos para uso con el Bloc de notas y Arrastrar y Soltar, y la interface para los dispositivos de entrada tales como el ratón y el teclado.

Cada uno de los componentes de una ventana en AWT se representa mediante uno o más objetos de la aplicación. Estos objetos pertenecen a las clases que se observan en la siguiente jerarquía de clases para AWT:

Figura 1 Árbol AWT



Fuente: Luis Mateu B.

La clase de los contenedores sirve para crear áreas en la ventana cuyo único fin es colocar otras componentes en su interior.

- **SWING:** El paquete Swing es parte de la JFC (Java Foundation Classes) en la plataforma Java. La JFC provee facilidades para ayudar a la gente a construir GUIs. Swing abarca componentes como botones, tablas, marcos, etc...

Las componentes Swing se identifican porque pertenecen al paquete *javax.swing*.

Swing existe desde la JDK 1.1 (como un agregado). Antes de la existencia de Swing, las interfaces gráficas con el usuario se realizaban

a través de AWT (Abstract Window Toolkit), de quien Swing hereda todo el manejo de eventos. Usualmente, para toda componente AWT existe una componente Swing que la reemplaza, por ejemplo, la clase Button de AWT es reemplazada por la clase JButton de Swing (el nombre de todas las componentes Swing comienza con "J").

Las componentes de Swing utilizan la infraestructura de AWT, incluyendo el modelo de eventos AWT, el cual rige cómo una componente reacciona a eventos tales como, eventos de teclado, mouse, etc... Es por esto, que la mayoría de los programas Swing necesitan importar dos paquetes AWT: *java.awt.\** y *java.awt.event.\**.

- **Java Database Connectivity**, más conocida por sus siglas **JDBC**, es una API que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java, independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede, utilizando el dialecto SQL del modelo de base de datos que se utilice.

El API JDBC se presenta como una colección de interfaces Java y métodos de gestión de manejadores de conexión hacia cada modelo específico de base de datos. Un manejador de conexiones hacia un modelo de base de datos en particular es un conjunto de clases que implementan las interfaces Java y que utilizan los métodos de registro para declarar los tipos de localizadores a base de datos (URL) que pueden manejar. Para utilizar una base de datos particular, el usuario ejecuta su programa junto con la biblioteca de conexión apropiada al modelo de su base de datos, y accede a ella estableciendo una conexión, para ello provee el localizador a la base de datos y los parámetros de conexión específicos. A partir de allí puede realizar con cualquier tipo de tareas con la base de datos a las que tenga permiso: consulta, actualización, creación, modificación y borrado de tablas, ejecución de procedimientos almacenados en la base de datos, etc.

#### 2.2.4 Servidor

- **TOMCAT**: (también llamado **Jakarta Tomcat** o **Apache Tomcat**) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las



especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems.

Tomcat es un servidor web con soporte de servlets y JSPs. Tomcat no es un servidor de aplicaciones, como JBoss o JOnAS. Incluye el compilador Jasper, que compila JSPs convirtiéndolas en servlets. El motor de servlets de Tomcat a menudo se presenta en combinación con el servidor web Apache.

Tomcat puede funcionar como servidor web por sí mismo. En sus inicios existió la percepción de que el uso de Tomcat de forma autónoma era sólo recomendable para entornos de desarrollo y entornos con requisitos mínimos de velocidad y gestión de transacciones. Hoy en día ya no existe esa percepción y Tomcat es usado como servidor web autónomo en entornos con alto nivel de tráfico y alta disponibilidad.

Dado que Tomcat fue escrito en Java, funciona en cualquier sistema operativo que disponga de la máquina virtual Java.

Tomcat es mantenido y desarrollado por miembros de la Apache Software Foundation y voluntarios independientes. Los usuarios disponen de libre acceso a su código fuente y a su forma binaria en los términos establecidos en la *Apache Software Licence*. Las primeras distribuciones de Tomcat fueron las versiones 3.0.x. Las versiones más recientes son las 6.x, que implementan las especificaciones de Servlet 2.5 y de JSP 2.1. A partir de la versión 4.0, Jakarta Tomcat utiliza el contenedor de servlets Catalina.

### 2.2.5 Entorno de desarrollo y construcción de la base de datos

- **MYSQL:** Es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB — desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia

específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual, MySQL es propietario y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius.

SQL (*Lenguaje de Consulta Estructurado*) fue comercializado por primera vez en 1981 por IBM, el cual fue presentado a ANSI y desde ese entonces ha sido considerado como un estándar para las bases de datos relacionales. Desde 1986, el estándar SQL ha aparecido en diferentes versiones como por ejemplo: SQL:92, SQL:99, SQL:2003. MySQL es una idea originaria de la empresa open source MySQL AB establecida inicialmente en Suecia en 1995 y cuyos fundadores son David Axmark, Allan Larsson, y Michael "Monty" Widenius. El objetivo que persigue esta empresa consiste en que **MySQL** cumpla el estándar SQL, pero sin sacrificar velocidad, fiabilidad o usabilidad.

Michael Widenius en la década de los 90 trató de usar *mSQL* para conectar las tablas usando rutinas de bajo nivel ISAM, sin embargo, *mSQL* no era rápido y flexible para sus necesidades. Esto lo conllevó a crear una API SQL denominada **MySQL** para bases de datos muy similar a la de *mSQL* pero más portable.

La procedencia del nombre de MySQL no es clara. Desde hace más de 10 años, las herramientas han mantenido el prefijo My. También, se cree que tiene relación con el nombre de la hija del cofundador Monty Widenius quien se llama *My*.

Por otro lado, el nombre del delfín de MySQL es Sakila y fue seleccionado por los fundadores de **MySQL** AB en el concurso "Name the Dolphin". Este nombre fue enviado por Ambrose Twebaze, un desarrollador de Open source Africano, derivado del idioma SiSwate, el idioma local de Swazilandia y corresponde al nombre de una ciudad en Arusha, Tanzania, cerca de Uganda la ciudad origen de Ambrose.

## 2.2.6 Plataforma de desarrollo

- **Java Platform, Enterprise Edition o Java EE** (anteriormente conocido como Java 2 Platform, Enterprise Edition o J2EE hasta la versión 1.4), es una plataforma de programación—parte de la Plataforma Java—para desarrollar y ejecutar software de aplicaciones en Lenguaje de programación Java con arquitectura de N niveles distribuida, basándose ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones. La plataforma Java EE está definida por una *especificación*. Similar a otras especificaciones del Java Community Process, Java EE es también considerada informalmente como un estándar debido a que los proveedores deben cumplir ciertos requisitos de conformidad para declarar que sus productos son *conformes a Java EE*; estandarizado por The Java Community Process / JCP.

Java EE incluye varias especificaciones de API, tales como JDBC, RMI, e-mail, JMS, Servicios Web, XML, etc y define cómo coordinarlos. Java EE también configura algunas especificaciones únicas para Java EE para componentes. Estas incluyen Enterprise JavaBeans, servlets, portlets (siguiendo la especificación de Portlets Java), JavaServer Pages y varias tecnologías de servicios web. Esto permite al desarrollador crear una Aplicación de Empresa portable entre plataformas y escalable, a la vez que integrable con tecnologías anteriores. Otros beneficios añadidos son, por ejemplo, que el servidor de aplicaciones puede manejar transacciones, la seguridad, escalabilidad, concurrencia y gestión de los componentes desplegados, significando que los desarrolladores pueden concentrarse más en la lógica de negocio de los componentes en lugar de en tareas de mantenimiento de bajo nivel.

La especificación original J2EE fue desarrollada por Sun Microsystems.

Comenzando con J2EE 1.3, la especificación fue desarrollada bajo el Java Community Process. JSR 58 especifica J2EE 1.3 y JSR 151 especifica J2EE 1.4.

El SDK de J2EE 1.3 fue liberado inicialmente como beta en Abril de 2001. La beta del SDK de J2EE 1.4 fue liberada por Sun en Diciembre de 2002.

La especificación Java EE 5 fue desarrollada bajo el JSR 244 y la liberación final fue hecha el 11 de Mayo de 2006.

### 3. INGENIERIA DEL PROYECTO

#### 3.1 METODOLOGÍA Y MODELO DE DESARROLLO

Para el desarrollo del sistema VSM “administrador y manejador de información comercial”, se ha elegido de entre varias metodologías que ofrece la Ingeniería de Software, el modelo desarrollo en espiral<sup>4</sup>, debido a que con este modelo se busca cubrir las características del ciclo de vida clásico, adicionando la creación de prototipos y analizando los posibles riesgos que se puedan generar en el proceso de desarrollo del sistema; este modelo se representa como una espiral, cada ciclo en la espiral representa una fase en el proceso del software y tiene en cuenta los riesgos que puedan existir a la hora de desarrollar, se evalúan, se corrigen y se continúa de una forma segura hasta llegar a un punto en el que el software será aceptado y no haya necesidad de seguir modificándolo con un nuevo ciclo.

El desarrollo en espiral se divide en cuatro actividades:

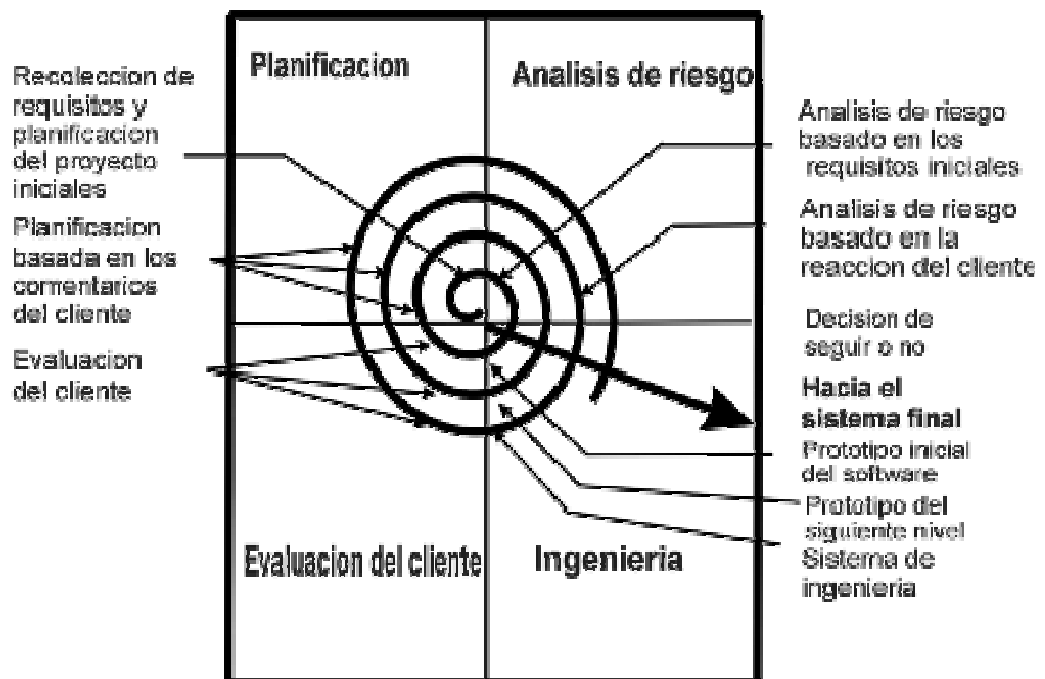
- Definición de objetivos: En esta fase se definen los objetivos específicos y se identifican los riesgos del proyecto, proponiendo estrategias que impidan que esos riesgos interfieran en el proceso.
- Evaluación y Reducción de Riesgos: Se analizan detalladamente los riesgos identificados, se definen pasos para disminuirlos.
- Desarrollo y validación: Se elige un modelo para el desarrollo del sistema.

---

<sup>4</sup> SOMERVILLE, Ian. Ingeniería del Software. Séptima edición. Madrid 2005, Capítulo 4, P. 42.

- **Planificación:** En esta fase, el proyecto es revisado y se toma la decisión de si debe continuar con un nuevo ciclo o si es necesario parar allí, si se decide continuar, se deben desarrollar todos los planes para la siguiente fase del proyecto.

Figura 2 Modelo en Espiral – Ciclo de vida del Software



Fuente: Barry Boehm en 1988

## 4. ANÁLISIS Y DISEÑO

### 4.1 DEFINICIÓN Y DESCRIPCIÓN DEL SISTEMA ACTUAL

Visual Sunday Manager, es una idea basada en las necesidades actuales de este tipo de empresas, pero no está diseñada para una empresa específica, por esta razón no existe un sistema actual definido.

### 4.2 DEFINICIÓN DEL SISTEMA PROPUESTO

Visual Sunday Manager, será un gran apoyo a nivel organizacional en la empresa.

Constará de los siguientes módulos, Asesores, Inmuebles, Clientes, Usuarios y Mercantiles. Cada módulo se realiza bajo el lenguaje java y para construir la interfaz de usuario se utiliza el FrameWork de aplicaciones para ventanas “**Swing**” en conjunto con el apoyo de librerías como AWT. Para el módulo de reportes se trabaja con el lenguaje HTML en combinación con JSP.

- **Módulo Asesores:** este módulo se crea para poder generarle un usuario a cada asesor que ingresa a la compañía, se guardan sus datos personales, la aplicación tiene sus respectivas validaciones o excepciones, para este caso, cuando se solicitan datos numéricos y se escriben datos de tipo carácter, o cuando no se respeta la estructura para ingresar una dirección de correo electrónico el sistema va a mostrar unos avisos en los que se le indique al usuario que está haciendo algo mal.

- **Módulo Inmuebles:** este módulo es muy importante porque aquí se va a registrar la información de los inmuebles, al momento de ingresar la información, se solicitan también los datos del cliente dueño, esta va a ser la relación en la que se puede identificar qué inmueble pertenece a qué cliente y se van a poder diferenciar los dos tipos de clientes que maneja la empresa, que son, el cliente comprador y cliente publicador. Este módulo ofrece la opción de ingresar imágenes del inmueble, esta información es de tipo opcional, si el cliente no quiere adjuntar imágenes, la aplicación sigue sin ningún inconveniente. El módulo inmuebles, como todos los que componen la aplicación tiene sus respectivas validaciones de datos y de información duplicada.
- **Módulo Clientes:** aquí se podrá realizar únicamente consultas de los clientes registrados en la base de datos del sistema, también tiene la opción de hacer cambios en los registros de los clientes, pero solamente los puede realizar un administrador. Este módulo genera reportes de los clientes en general o por separado los publicadores de los compradores.
- **Módulo Usuarios:** este es bastante básico y aquí se pueden crear las nuevas cuentas de usuario, se podrán eliminar y modificar (solamente usuario administrador) y también se podrán consultar, con su respectiva fecha de creación o modificación y los permisos otorgados.
- **Módulo Mercantiles:** aquí se centra toda la atención del aplicativo, ya que aquí es donde quedan registrados los negocios que se concretan, es decir, cuando el asesor logre vender o arrendar un bien, se selecciona el inmueble disponible por su código y se registran los datos del nuevo cliente así como también debe hacerse la elección del asesor involucrado en el negocio. Si no se elige el asesor, automáticamente el sistema va a lanzar una excepción en la que le diga al usuario que es necesario elegir uno. Este módulo genera reportes de los inmuebles que ya han sido ocupados, en donde mostrará la información de la fecha en la que se cerró el negocio y los datos de las personas involucradas, como lo son, el cliente publicador, el asesor y el cliente comprador.

En todos los módulos explicados anteriormente, se pueden hacer consultas, modificaciones y eliminaciones de información pero estas dos últimas opciones sólo las puede realizar un usuario administrador.

#### **4.2.1 Requerimientos del sistema**

Los requerimientos son las características que debe tener un software para poder soportar y/o ejecutar una aplicación, estos requerimientos deben ser especificados por escrito y deben ser cumplidos como cualquier contrato entre dos partes, deben ser posibles de probar o verificar para determinar si se pueden cumplir o no y deben ser bastantes claros especificando qué es lo que debe hacer el sistema.

#### **4.2.2 Requerimientos funcionales**

El aplicativo “**VSM**” Visual Sunday Manager, está basado en los siguientes requerimientos funcionales:

- El aplicativo maneja un único usuario administrador que es quien posee todos los permisos para realizar cualquier cambio en el sistema de información, por seguridad éste es el único que podrá eliminar cualquier tipo de información. Partiendo de este usuario se podrán crear los demás quienes básicamente tendrán los permisos de consulta y generación de reportes y de agregar información.
- Debe tener un módulo de asesores, en el cual se deberá ingresar toda la información de los empleados que van a ofrecer los inmuebles que los clientes “dueños” hayan publicado, es necesario guardar esta información porque de esta manera, se tendrá un control acerca de



cuántos asesores trabajan actualmente con la compañía y cuál es el sueldo base y su porcentaje de comisión, así como cuántos negocios realiza al mes, para poder medir su desempeño en la empresa.

- La esencia del negocio, consiste en la venta o arriendo de inmuebles, por esta razón uno de los requerimientos más importantes del software es almacenar y capturar todos los datos relacionados con un inmueble; la aplicación debe contar con las opciones con las que se pueda identificar qué inmueble se está publicando, su ubicación, estrato, barrio y demás características, para tener una mejor idea del inmueble, la aplicación genera un reporte muy completo con toda esta información y maneja la opción de incluir fotografías, para mostrarle al cliente interesado un historial bastante claro del bien en el que esté interesado.
- Cuando el asesor logre cerrar un negocio, este se podrá registrar en el sistema y se podrá consultar en un reporte la fecha en la que el negocio se realizó, el nombre y los datos principales del cliente que compró o arrendó, los datos del cliente que publicó ese inmueble y finalmente el nombre del asesor. Este tipo de reportes son muy consultados por la empresa, para poder sacar conclusiones de cómo se está moviendo el negocio, de esta manera podrá tomar las respectivas decisiones para mejorar la empresa cada día más.
- La aplicación debe generar un reporte de los clientes involucrados con la compañía, tanto de clientes publicadores, como clientes compradores o que buscan un inmueble en arriendo.

#### **4.2.3 Requerimientos no funcionales**

- El software debe adaptarse fácilmente y debe tener las mismas funcionalidades sin excepciones en cualquier empresa Inmobiliaria.

- El software debe ofrecer una presentación clara y fácil de manejar para el usuario o la persona que lo esté manejando, dándole un ambiente de trabajo familiar e intuitivo que facilite su adopción y productividad.
- Puede ser utilizado en cualquier momento, es decir, este software debe estar disponible y funcionar perfectamente en cualquier instante en el que sea necesario consultar cualquier tipo de información almacenada en la base de datos.
- El equipo en el que se va a ejecutar el aplicativo debe tener instalado el servidor de Xampp-win32-1.7.0 que contiene entre otros componentes los de la base de datos MySQL y del servidor Web Apache, que son los más importantes y que se van a necesitar en este caso.
- El servidor de Apache TomCat-6.0.18 debe estar instalado y se debe verificar que esté encendido, para poder observar vía web los reportes que se requieran con la opción de imprimirlos.

### **4.3 DISEÑO DEL SISTEMA PROPUESTO**

#### **4.3.1 Diccionario de datos**

**CLIENTES:** Esta tabla guarda toda la información de los clientes dueños de inmuebles y compradores o interesados en tomar arrendado un bien.

Tabla 1 Diccionario de datos Clientes

<b>Campo</b>	<b>Descripción</b>	<b>Tipo de Dato</b>	<b>Longitud</b>	<b>Valor</b>	<b>Llave Primaria</b>	<b>Referencia</b>
Cedula_cliente	Identificación del cliente	Int	10		X	
Nombres	Nombres completos del cliente	Varchar	10			
Apellidos	Apellidos completos del cliente	Varchar	10			
Dirección	Dirección casa	Varchar	15			
Teléfono	Número telefónico de la casa del cliente	Int	7			
E_mail	Dirección de correo	Varchar	20			
Celular	Código reserva	Int	10			

Fuente: Angie Paola Rodríguez

**EMPLEADO:** Esta tabla guarda toda la información personal de los empleados.

Tabla 2 Diccionario de datos Empleado

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cédula_empleado	Identificación del empleado	Int	10		X	
Nombres	Nombres completos del empleado	Varchar	10			
Apellidos	Apellidos completos del empleado	Varchar	10			
Dirección	Dirección casa	Varchar	15			
Teléfono	Número telefónico de la casa del empleado	Int	7			
Porcentaje	Porcentaje de comisión	Float	10			
Salario	Código reserva	Int	10			
E_mail	Dirección de correo	Varchar	15			

Fuente: Angie Paola Rodríguez

**INMUEBLE:** En esta tabla van a registrarse los datos principales y características descriptivas de un inmueble.

Tabla 3 Diccionario de datos Inmueble

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_inmueble	Código de identificación del inmueble	Int	6		X	
Area	Area del perímetro en metros	Int	10			
N_habitaciones	Cantidad de habitaciones	Int	2			
N_baños	Cantidad de baños	Int	2			
N_garajes	Cantidad de garajes	Int	2			
Foto	Ruta del archivo PNG ó GIF	Varchar	10			
Cod_tipo	Código del tipo de inmueble	Int	10			Tipo
Cod_estrato	Código del estrato	Int	10			Estrato
Cod_estado	Código del estado (libre - ocupado)	Int	10			Estado
Cod_barrio	Código del barrio	Int	10			Barrio

Fuente: Angie Paola Rodríguez

**TIPO:** Esta tabla describe el tipo de inmueble: Apartamento, Casa, Bodega, Local Comercial, Oficina o Lote.

Tabla 4 Diccionario de datos Tipo

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_tipo	Código del tipo de inmueble	Int	10		X	
Nombre	Nombre del tipo de inmueble	Varchar	10			

Fuente: Angie Paola Rodríguez

**ESTADO:** En esta tabla se podrá identificar si el inmueble está disponible o ya ha sido ocupado.

Tabla 5 Diccionario de datos Estado

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_estado	Código del estado del inmueble	Int	10		X	
Nombre	Nombre del estado del inmueble	Varchar	10			

Fuente: Angie Paola Rodríguez

**ESTRATO:** Como su nombre lo indica, en esta tabla se identificará el estrato de la propiedad.

Tabla 6 Diccionario de datos Estrato

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_estrato	Código del estrato del inmueble	Int	10		X	
Nombre	Nombre del estrato del inmueble	Varchar	10			

Fuente: Angie Paola Rodríguez

**BARRIO:** Se registrará el sector o barrio en donde está ubicado el inmueble.

Tabla 7 Diccionario de datos Barrio

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_barrio	Código del barrio del inmueble	Int	10		X	
Nombre	Nombre del barrio en donde está ubicado el inmueble	Varchar	10			

Fuente: Angie Paola Rodríguez

**INMUEBLE\_X\_EMPLEADO:** En esta tabla se puede identificar el inmueble que ha sido asignado a un empleado.

Tabla 8 Diccionario de datos Inmueble por empleado

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_inm_emp	Código del inmueble asignado al empleado	Int	10		X	
Cédula_empleado	Cédula de identificación del empleado	Int	10			Empleado
Cod_inmueble	Código del inmueble	Int	10			Inmueble

Fuente: Angie Paola Rodríguez

**PERTENECE:** Tabla que identifica a qué cliente le pertenece un inmueble.

Tabla 9 Diccionario de datos Pertenece

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_pertenencia	Código de pertenencia	Int	10		X	
Cod_inmueble	Código del inmueble que le pertenece al cliente	Int	10			Inmueble
Cédula_cliente	Cédula del cliente	Int	10			Cliente

Fuente: Angie Paola Rodríguez



**ARRIENDO:** Esta tabla muestra qué inmuebles han sido arrendados a qué clientes y por qué empleado.

Tabla 10 Diccionario de datos Arriendo

Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_arriendo	Código arriendo	Int	10		X	
Cédula_empleado	Cédula del empleado que cerró el negocio	Int	10			Empleado
Cédula_cliente	Cédula del cliente que arrendó	Int	10			Cliente
Cod_inmueble	Código del inmueble arrendado	Int	10			Inmueble
Precio_arriendo	Precio del arriendo	Int	10			
Fecha_arriendo	Fecha del arriendo	Date	10			

Fuente: Angie Paola Rodríguez

**VENTA:** Esta tabla muestra qué inmuebles han sido vendidos a qué clientes y por qué vendedor.

Tabla 11 Diccionario de datos Venta

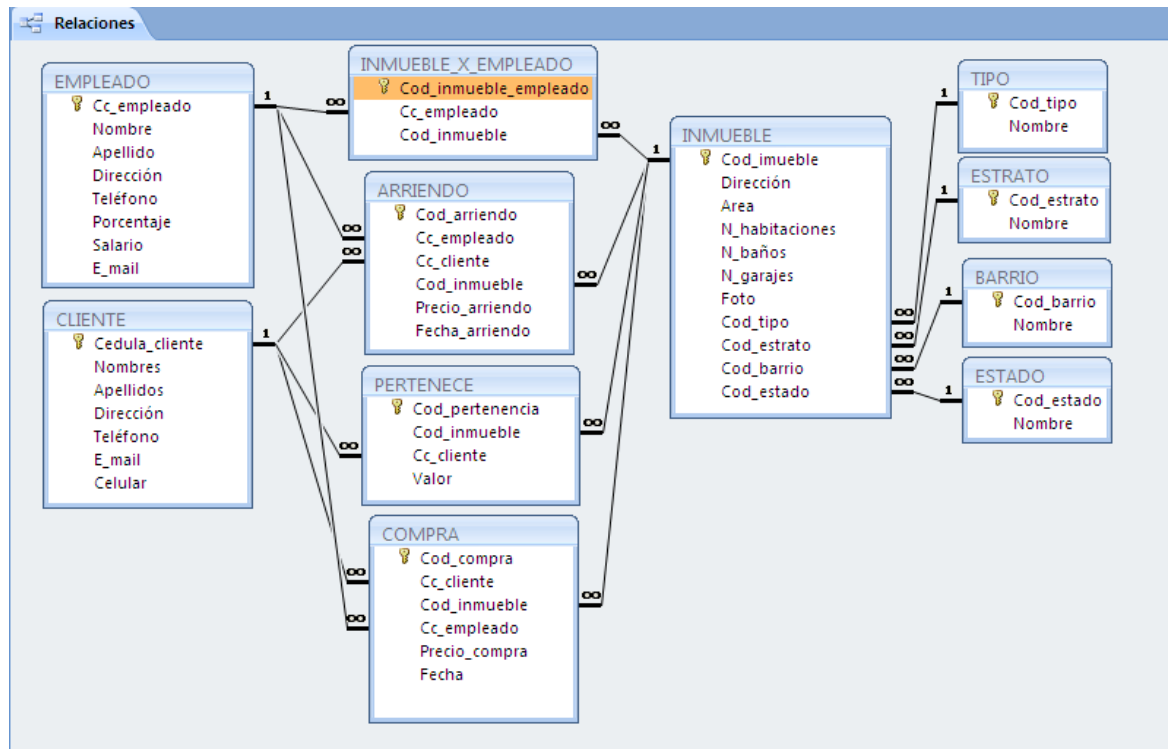
Campo	Descripción	Tipo de Dato	Longitud	Valor	Llave Primaria	Referencia
Cod_venta	Código venta	Int	10		X	
Cédula_empleado	Cédula del empleado que cerró el negocio	Int	10			Empleado
Cédula_cliente	Cédula del cliente que compró	Int	10			Cliente
Cod_inmueble	Código del inmueble arrendado	Int	10			Inmueble
Precio_venta	Precio de la venta	Int	10			
Fecha_venta	Fecha de la venta	Date	10			

Fuente: Angie Paola Rodríguez

### 4.3.2 Diagrama Entidad Relación

A continuación se muestra el Diagrama Entidad Relación del sistema con la que se construyó la base de datos.

Figura 3 Diagrama Entidad Relación



Fuente: Angie Paola Rodríguez

### 4.4 DESCRIPCIÓN DETALLADA DE CADA PROCESO

Para explicar de una manera más clara en qué consisten los requerimientos de la aplicación y cómo va a interactuar el usuario con ella en la etapa final del desarrollo, a continuación se ilustran y se describen los casos de uso para cada requisito. Cabe anotar que cada caso de uso posteriormente va a ser de gran apoyo para obtener un resultado favorable en la construcción del sistema.

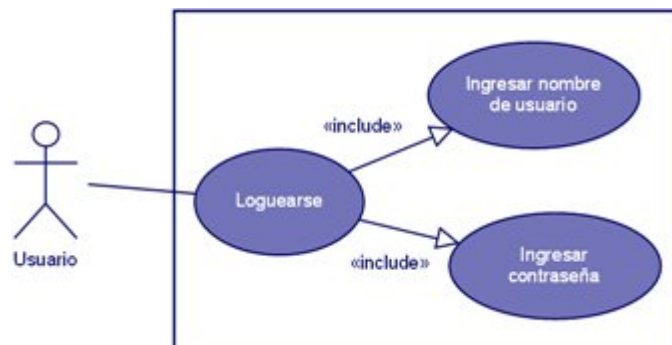
**ID Caso de Uso: CU001**

**Versión: 1.0**

**Nombre del caso de Uso:** Loguearse

**Detalle del Caso de Uso:** Después de asignarle los respectivos usuarios al empleado, este debe ingresar al aplicativo ingresando su nombre de usuario y su respectiva contraseña.

Figura 4 Caso de uso loguearse.



Fuente: Angie Paola Rodríguez

**Secuencia Normal:**

1. Iniciar aplicativo
2. Ingresar usuario
3. Ingresar contraseña
4. Pulsar botón aceptar

**Flujo Alternativo:**

1. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.

**Precondiciones:**

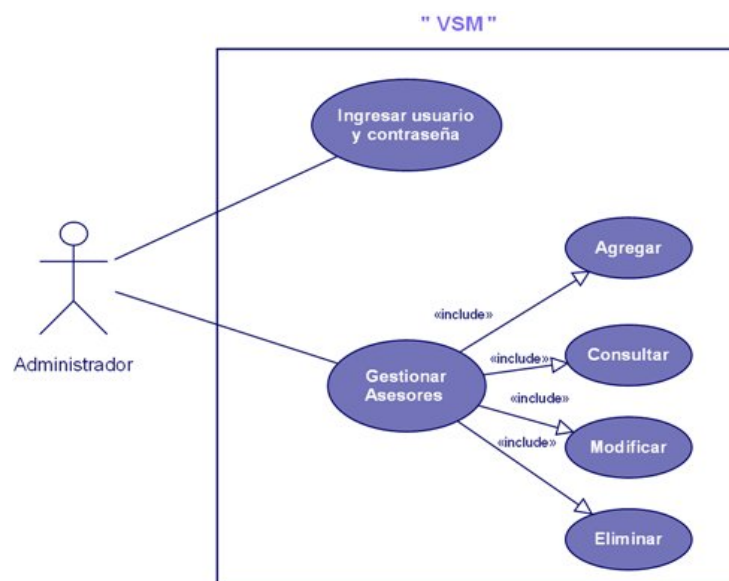
- Debe haber energía
- Deben estar instalados todos los componentes a nivel del sistema para poder iniciar la aplicación

**Post condiciones:** Según los permisos del usuario, puede empezar a trabajar.

**Nombre del caso de Uso:** Gestionar Asesor

**Detalle del Caso de Uso:** El usuario administrador es quien tiene todos los permisos habilitados para realizar cualquier tipo de tarea en el en el sistema, en el siguiente caso de uso se representan las opciones disponibles en el módulo asesores.

Figura 5 Caso de uso Gestionar Asesor



Fuente: Angie Paola Rodríguez

**Secuencia Normal:**

1. Iniciar aplicativo
2. Ingresar usuario administrador
3. Ingresar respectiva contraseña
4. Ingresar al módulo asesores
5. Agregar nuevo asesor
6. Consultar asesores
7. Modificar asesores
8. Eliminar asesores

**Flujo Alternativo:**

1. Debe ingresar como administrador
2. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.
3. Todas las opciones están habilitados para este usuario.

**Precondiciones:**

- Debe haber energía
- Deben estar instalados todos los componentes a nivel del sistema para poder iniciar la aplicación

**Post condiciones:** El administrador puede empezar a trabajar sin ninguna restricción ya que posee todos los permisos.

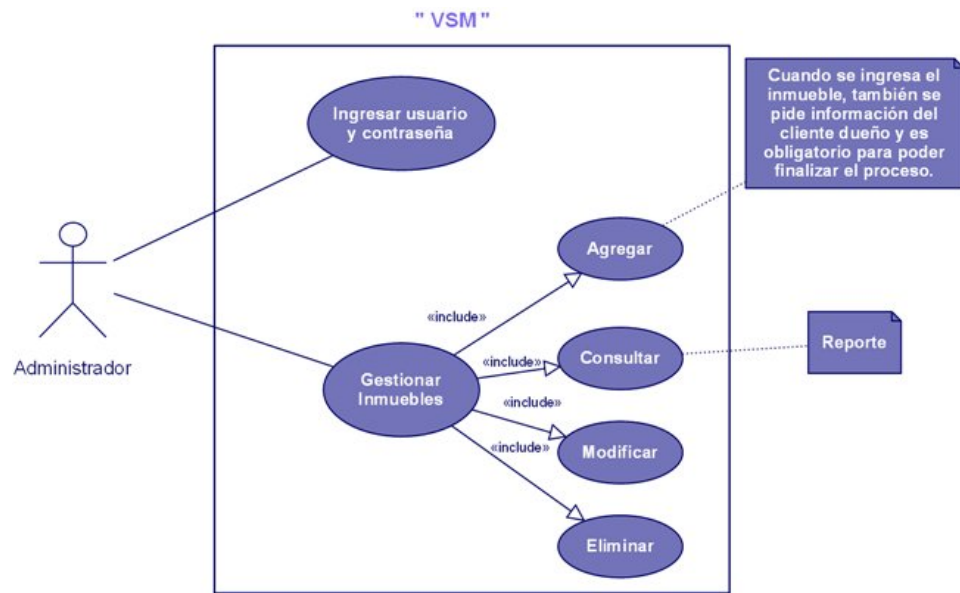
**ID Caso de Uso: CU003**

**Versión: 1.0**

**Nombre del caso de Uso:** Gestionar Inmueble

**Detalle del Caso de Uso:** Se representan las opciones disponibles que el usuario administrador tiene en el módulo Inmuebles. Cuando se ingresa información del nuevo inmueble, también se solicitan los datos del cliente publicador, de esta forma se ingresa la información correspondiente al cliente; para terminar correctamente la operación se debe llenar toda la información solicitada por el sistema.

Figura 6 Caso de uso Gestionar Inmueble



Fuente: Angie Paola Rodríguez

### Secuencia Normal:

1. Iniciar aplicativo
2. Ingresar usuario administrador
3. Ingresar respectiva contraseña
4. Ingresar al módulo inmuebles
5. Agregar nuevo inmueble
6. Consultar inmuebles
7. Modificar inmuebles
8. Eliminar inmuebles

### Flujo Alternativo:

1. Debe ingresar como administrador
2. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.
3. Todas las opciones están habilitados para este usuario.

### Precondiciones:

- Debe haber energía
- Deben estar instalados todos los componentes a nivel del sistema para poder iniciar la aplicación

**Post condiciones:** El administrador puede empezar a trabajar sin ninguna restricción ya que posee todos los permisos.

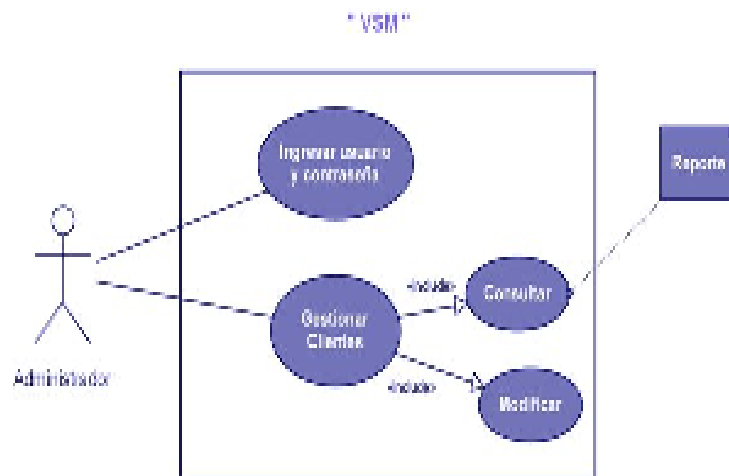
**ID Caso de Uso: CU004**

**Versión: 1.0**

**Nombre del caso de Uso:** Gestionar Clientes

**Detalle del Caso de Uso:** Representación de las funciones que puede realizar el usuario administrador en cuanto al módulo de Clientes.

Figura 7 Caso de uso Gestionar Clientes



Fuente: Angie Paola Rodríguez

**Secuencia Normal:**

1. Iniciar aplicativo
2. Ingresar usuario administrador
3. Ingresar respectiva contraseña
4. Ingresar al módulo clientes
5. Consultar clientes

## 6. Modificar clientes

### **Flujo Alternativo:**

1. Debe ingresar como administrador
2. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.

### **Precondiciones:**

- Debe haber energía
- Deben estar instalados todos los componentes a nivel del sistema para poder iniciar la aplicación

**Post condiciones:** El administrador puede empezar a trabajar sin ninguna restricción ya que posee todos los permisos.

**ID Caso de Uso:** CU005

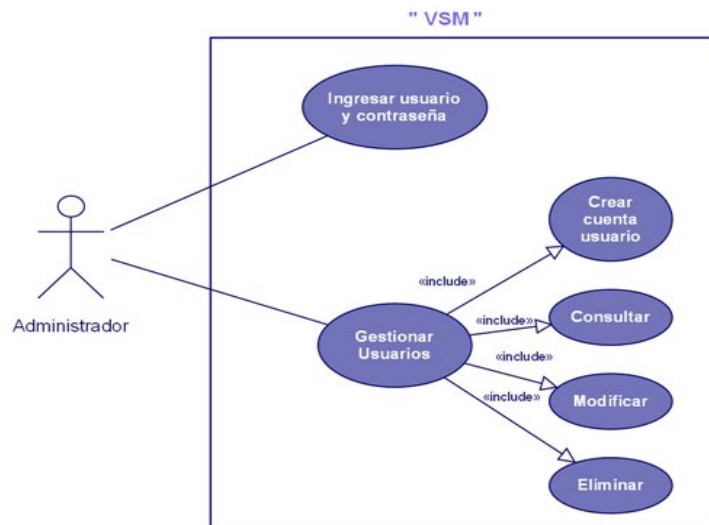
**Versión:** 1.0

**Nombre del caso de Uso:** Gestionar Usuarios

**Detalle del Caso de Uso:** Representación de las opciones que se encuentran disponibles en el módulo Usuarios para el usuario administrador.



Figura 8 Caso de uso Gestionar Usuarios



Fuente: Angie Paola Rodríguez

### Secuencia Normal:

1. Iniciar aplicativo
2. Ingresar usuario administrador
3. Ingresar respectiva contraseña
4. Ingresar al módulo usuarios
5. Agregar nuevo usuario
6. Consultar usuario
7. Modificar usuario
8. Eliminar usuario

### Flujo Alternativo:

1. Debe ingresar como administrador
2. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.
3. Todas las opciones están habilitados para este usuario.

### Precondiciones:

- Debe haber energía
- Deben estar instalados todos los componentes a nivel del sistema para poder iniciar la aplicación

**Post condiciones:** El administrador puede empezar a trabajar sin ninguna restricción ya que posee todos los permisos.

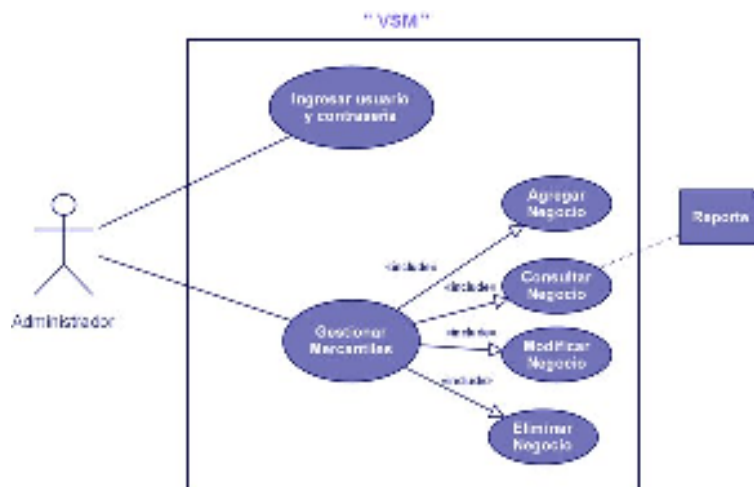
**ID Caso de Uso: CU006**

**Versión: 1.0**

**Nombre del caso de Uso:** Gestionar Mercantiles

**Detalle del Caso de Uso:** Funciones que el usuario administrador puede realizar en el módulo Mercantiles.

Figura 9 Caso de uso Gestionar Mercantiles



Fuente: Angie Paola Rodríguez

**Secuencia Normal:**

1. Iniciar aplicativo
2. Ingresar usuario administrador
3. Ingresar respectiva contraseña
4. Ingresar al módulo mercantiles
5. Agregar negocio

6. Consultar negocio
7. Modificar negocio
8. Eliminar negocio

**Flujo Alternativo:**

1. Debe ingresar como administrador
2. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.
3. Todas las opciones están habilitadas para este usuario.

**Precondiciones:**

- Debe haber energía
- Deben estar instalados todos los componentes a nivel del sistema para poder iniciar la aplicación

**Post condiciones:** El administrador puede empezar a trabajar sin ninguna restricción ya que posee todos los permisos.

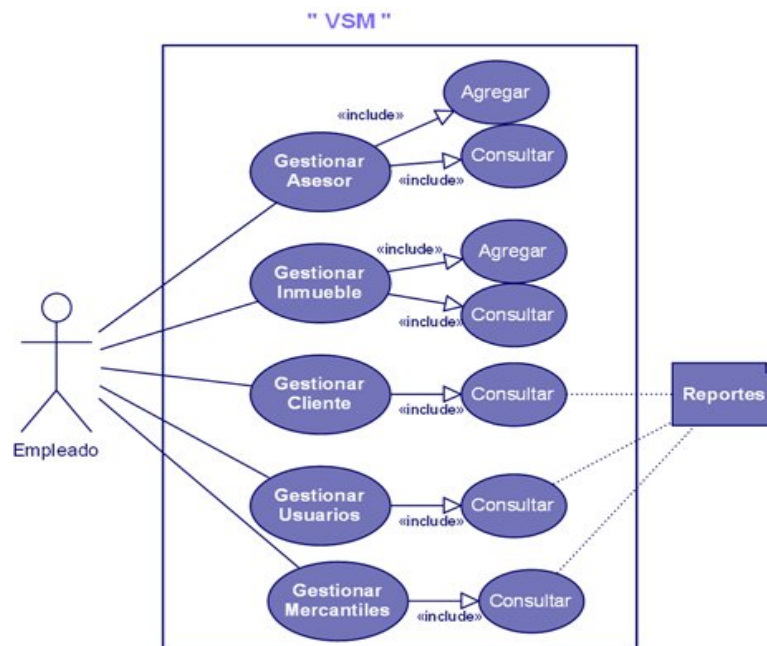
**ID Caso de Uso:** CU007

**Versión:** 1.0

**Nombre del caso de Uso:** Acciones Empleado

**Detalle del Caso de Uso:** El usuario empleado no posee todos los permisos, en este caso este usuario puede consultar información de la base de datos en todos los módulos y solamente puede agregar información en asesores e inmuebles.

Figura 10 Caso de uso acciones empleado



Fuente: Angie Paola Rodríguez

### Secuencia Normal:

1. Iniciar aplicativo
2. Ingresar usuario empleado
3. Ingresar respectiva contraseña
4. Ingresar al modulo Asesor
5. Agregar asesor
6. Consultar asesor
7. Ingresar al módulo Inmueble
8. Agregar inmueble
9. Consultar Inmueble
10. Ingresar al módulo Cliente
11. Consultar cliente
12. Ingresar al módulo Usuarios
13. Consultar usuarios
14. Ingresar al módulo mercantiles
15. Consultar negocios

### Flujo Alterno:

1. Debe ingresar como usuario empleado
2. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar y los debe digitar correctamente.
3. Puede trabajar en todos los módulos

### Precondiciones:

- Debe haber energía
- Deben estar instalados todos los componentes a nivel del sistema para poder iniciar la aplicación

**Post condiciones:** El empleado realiza su trabajo pero lo único que no puede hacer, es eliminar registros o información.

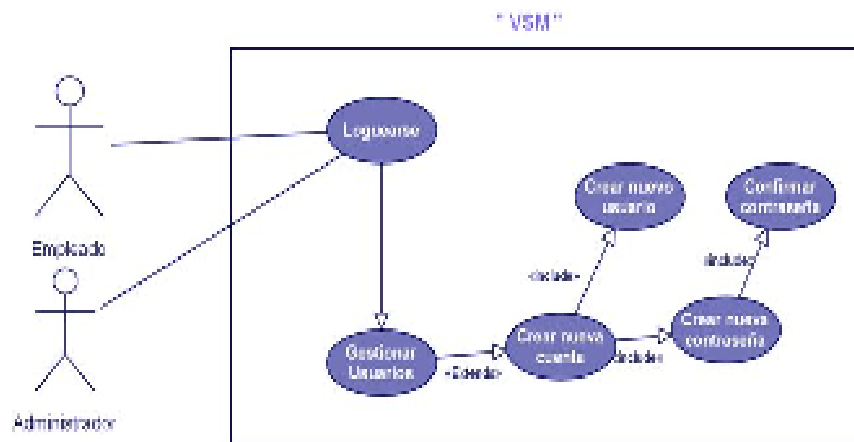
**ID Caso de Uso: CU008**

**Versión: 1.0**

**Nombre del caso de Uso:** Crear nuevo usuario

**Detalle del Caso de Uso:** Ingresando por el módulo de usuarios, crear nueva cuenta, registrar la nueva contraseña y confirmar.

Figura 11 Caso de uso Crear usuario



Fuente: Angie Paola Rodríguez

**Secuencia Normal:**

1. Iniciar aplicativo
2. Ingresar usuario del empleado o del administrador
3. Ingresar contraseña
4. Ir al módulo de usuarios
5. Ir a la opción crear nueva cuenta
6. Crear usuario
7. Crear contraseña
8. Confirmar contraseña
9. Finalizar operación - Aceptar

**Flujo Alternativo:**

1. Debe estar instalado el aplicativo en el PC donde se intenta acceder.
2. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.

**Precondiciones:**

- Debe haber energía

**Post condiciones:** Se ha creado una nueva cuenta de usuario

**ID Caso de Uso:** CU009

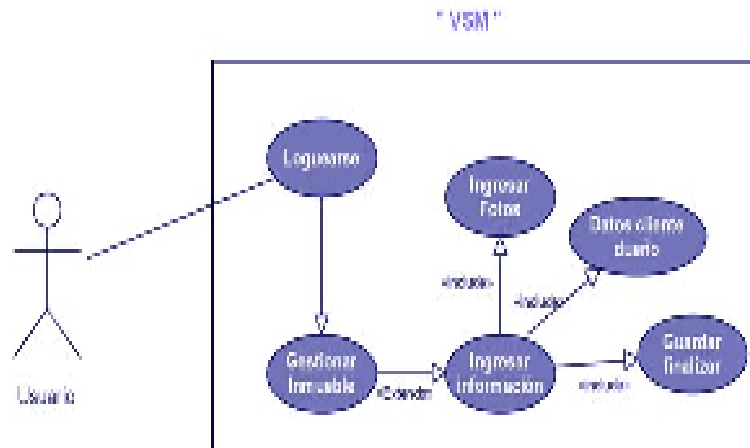
**Versión:** 1.0

**Nombre del caso de Uso:** Adicionar Inmueble

**Detalle del Caso de Uso:** Esta operación la puede realizar tanto el usuario administrador como el usuario creado para los empleados, consiste en ingresar toda la información requerida por el sistema de un inmueble, el sistema solicitará la ubicación, el estrato, el barrio, dirección, precio para arriendo o venta, según el caso, el tipo de negocio, tiempo de construido, habitaciones, baños, garajes, tipo de gas, tipo de comedor y algunas otras descripciones manuales que el cliente quiera agregar para dar a conocer mejor su negocio.

Adicional a esto, también se van a registrar los datos del cliente publicador para poder visualizar en los reportes qué dueño publicó qué inmueble.

Figura 12 Caso de uso adicionar inmueble



Fuente: Angie Paola Rodríguez

### Secuencia Normal:

1. Iniciar aplicativo
2. Ingresar usuario
3. Ingresar contraseña
4. Ingresar al Módulo Inmuebles
5. Registrar información del inmueble
6. Subir fotos (opcional)
7. Registrar datos del cliente publicador o dueño (Obligatorio)
8. Finalizar operación – Guardar.

### Flujo Alternativo:

1. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.

### Precondiciones:

- Debe haber energía

**Post condiciones:** El usuario ha ingresado correctamente un inmueble nuevo.

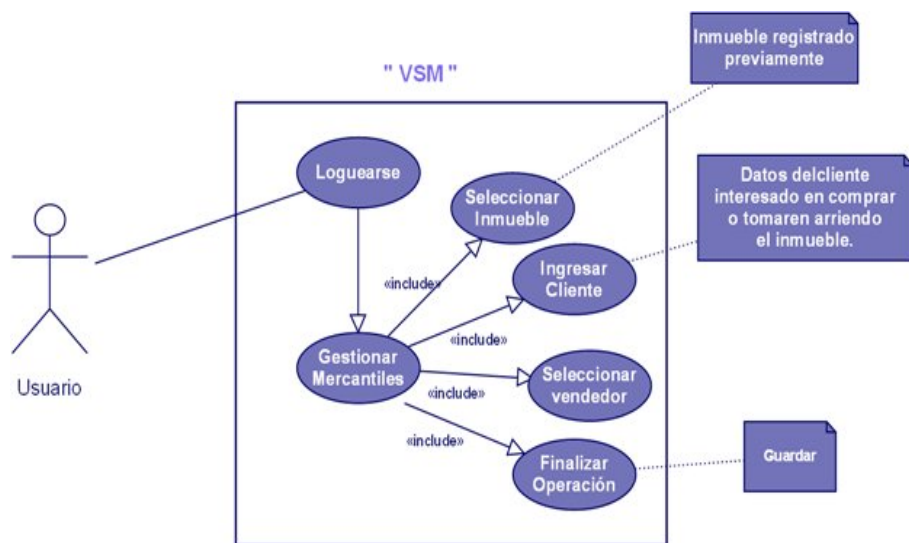
**ID Caso de Uso:** CU0010

**Versión:** 1.0

**Nombre del caso de Uso:** Cerrar negocio

**Detalle del Caso de Uso:** En el módulo mercantiles, cuando se va a cerrar un negocio, hay que elegir un inmueble publicado en el sistema, este se escoge por código, adicional a esto se ingresan los datos personales del nuevo cliente comprador o quién tomó por arriendo el inmueble y se finaliza agregando el empleado que logró realizar el negocio.

Figura 13 Caso de uso Cerrar negocio



Fuente: Angie Paola Rodríguez

### Secuencia Normal:

1. Iniciar aplicativo
2. Ingresar usuario
3. Ingresar contraseña
4. Ingresar al Módulo Mercantiles
5. Seleccionar inmueble



6. Ingresar datos del cliente interesado en comprar o tomar en arriendo el inmueble.
7. Seleccionar datos del asesor involucrado en el negocio
8. Finalizar operación – Guardar.

**Flujo Alterno:**

1. El usuario y la contraseña deben existir en la base de datos para que el usuario pueda ingresar.

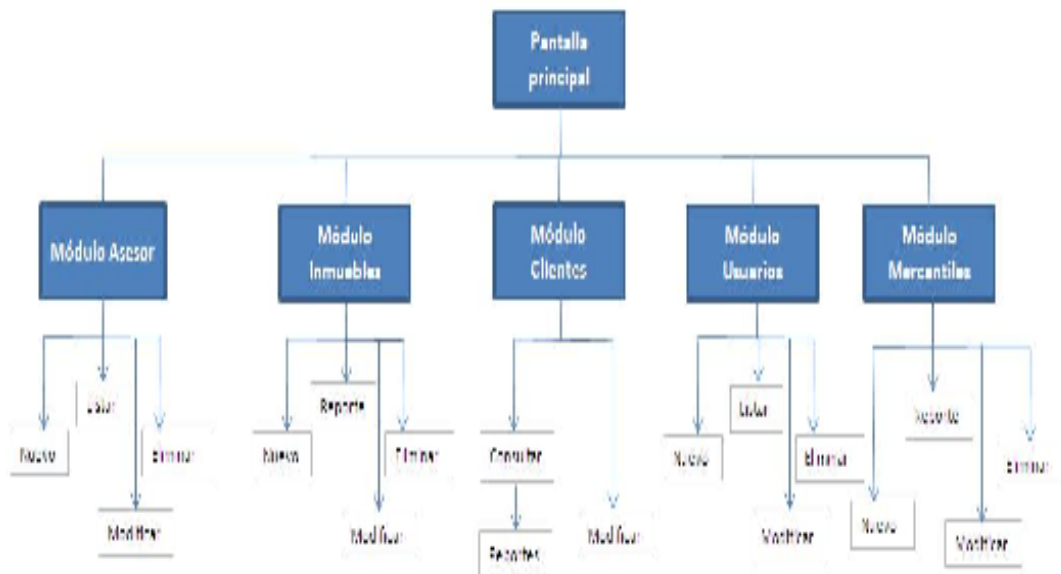
**Precondiciones:**

- Debe haber energía

**Post condiciones:** El usuario ha cerrado un nuevo negocio.

**4.1. DIAGRAMA DE CADA PROCESO**

Figura 14 Diagrama de procesos



Fuente: Angie Paola Rodríguez

## 5. DESARROLLO

### 5.1 ESPECIFICACIONES TÉCNICAS

“**VSM**” Visual Sunday Manager es un aplicativo muy sencillo y fácil de usar ya que las características que posee para su correcta instalación son entendibles para cualquier usuario.

### 5.2 SOFTWARE

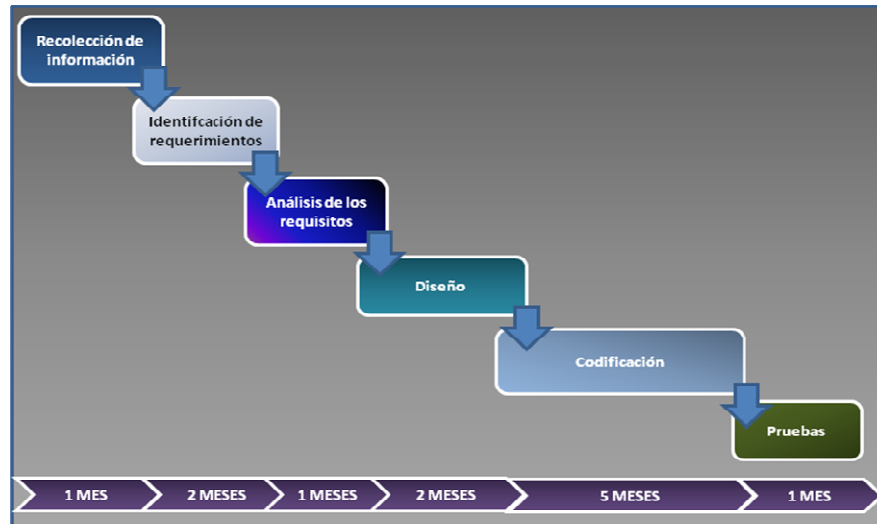
- Windows XP
- Servidor TomCat
- Xampp Win32 1.7
  - MySql
  - phpMyAdmin
- Internet explorer 6 o Firefox (Recomendado).

### 5.3 HARDWARE

- Computadores compatibles para Windows XP
- Ram 512 Mb
- Procesador Pentium II o equivalentes

## 5.4 CRONOGRAMA

Figura 15 Cronograma



Fuente: Angie Paola Rodríguez

## 5.5 RECURSOS

### ➤ Humanos

- Angie Paola Rodríguez Gerente
- Luis Fernando Medina Asesor elaboración proyecto
- Luis Eduardo Pérez Asesor en el desarrollo del proyecto
- Sonya Díaz Asesora en la creación de la Base de Datos

### ➤ Hardware

- Memoria RAM 512
- Disco duro de 20 GB
- Procesador 3 GHz

## 6. GLOSARIO

**API:** una interfaz de programación de aplicaciones o API (del inglés **Application Programming Interface**) es el conjunto de funciones y procedimientos (o métodos, si se refiere a programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

**Applets:** son mini-programas confeccionados en lenguaje Java que se cargan en el ordenador que se conecta ofreciendo unas prestaciones al programa navegador.

**AWT:** (Abstract Window Toolkit). Juego de herramientas de ventana abstracta. Biblioteca de procedimientos de gráficos usada con el lenguaje Java para manipular ventanas en una pantalla por mapa de bits. En diferentes momentos, los diseñadores han expandido el AWT a Alternative Window Toolkit (juego de herramientas alternativas de ventana) y Applet Widget Toolkit (juego de herramientas de adminículo de applet).

**BASES DE DATOS:** (DataBase). Conjunto de datos relacionados que se almacenan de forma que se pueda acceder a ellos de manera sencilla, con la posibilidad de relacionarlos, ordenarlos en base a diferentes criterios, etc. Las bases de datos son uno de los grupos de aplicaciones de productividad personal más extendidos.

**HARDWARE:** se denomina así al conjunto de componentes físicos dentro de la informática (un teclado, una placa, por ej.).

**HTML:** Hypertext Markup Language. Al redactar un escrito en un procesador de textos normal, lo que se ve en pantalla no es lo que realmente se graba en el disco. Si se comprobase el trasfondo de lo escrito, aparecerían caracteres por todas partes ilegibles, pero que sirven al procesador para crear la apariencia de lo que se ve. Cuando se habla de páginas Web ocurre más o menos igual. Tras la apariencia de ésta misma página se esconde la realidad de lo que hay escrito, su color, fondo, tipo de letra, etc. Esto se consigue a través del lenguaje HTML, u otros.

**IDE:** integrated Device Electronic también conocido como interface de bus AT y fue la primera que se utilizó para discos duros en microordenadores y tiene capacidad para 2 unidades de almacenamiento, con un máximo en discos

duros de 528 megas. Fue originario de la empresa Conner y evoluciona en la norma comúnmente utilizado en la actualidad, que es EIDE.

**JAVA:** lenguaje de programación derivado del C++, desarrollado por Sun Microsystems, partiendo del OAK en C++, aunque simplificado en principio y adaptado para el funcionamiento en entornos de IntraNet-Internet.

**JSP:** (Java Server Page) Página de Servidor Java. Se refiere a un tipo especial de páginas HTML, en las cuales se insertan pequeños programas que corren sobre Internet (comúnmente denominados scripts), se procesan en línea para finalmente desplegar un resultado final al usuario en forma de HTML. Por lo general dichos programas hacen consultas a bases de datos y dependiendo del resultado que se despliegue será la información que se muestre a cada usuario de manera individual. Los archivos de este tipo llevan la extensión ".jsp".

**MYSQL:** Es la base de datos de fuentes abiertas probablemente más utilizada en los entornos Unix - Linux. Carece de la complejidad de las grandes bases de datos, casi es suficiente con conocer el lenguaje SQL, y es lo más utilizado, por muchas y buenas razones, para usar en Internet compaginando especialmente con PHP y Perl.

**SERVLETS:** aplicación sin interfaz gráfica que se ejecuta en un servidor de Internet, procesando información HTML previamente recogida por un navegador.

**SOFTWARE:** todos los componentes informáticos de carácter no físico, sino lógico (se denomina también logical), como pueden ser Sistemas Operativos, programas dedicados a la gestión, de diseño, etc.

**SOFTWARE LIBRE:** es la denominación del software que brinda libertad a los usuarios sobre su producto adquirido y por tanto, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente.

**SQL:** Structured Query Language. Lenguaje o sentencias para el manejo y consulta de bases de datos.

**SUN MICROSYSTEM INC:** una de las firmas norteamericanas más importantes en la fabricación y comercialización de estaciones de trabajo, entre otros productos.

**SWING:** es una biblioteca gráfica para Java. Incluye widgets para interfaz gráfica de usuario tales como cajas de texto, botones, desplegados y tablas.

**TOMCAT:** (también llamado Jakarta Tomcat o Apache Tomcat) funciona como un contenedor de servlets desarrollado bajo el proyecto Jakarta en la Apache Software Foundation. Tomcat implementa las especificaciones de los servlets y de JavaServer Pages (JSP) de Sun Microsystems.

**URL:** Uniform Resource Locator. Se conoce por este nombre a las direcciones dentro de Internet, normalmente, aunque no necesariamente, refiriéndose a páginas Web. En este caso se distinguen por iniciarse con http:// No obstante es una simplificación para el usuario el referenciarlas de esta forma, en realidad son secuencias de números que se dirigen de forma inequívoca a una dirección. Esto se conoce como DNS.

## **7. CONCLUSIONES**

Se ha desarrollado un software que sirve de apoyo a las actividades que se realizan en una inmobiliaria, garantizando una mejor organización en el desarrollo de los procesos cotidianos en la empresa, ofreciendo una interfaz de usuario amigable y muy fácil de manejar para cualquier usuario.

Se garantiza seguridad en la información almacenada en el sistema, ya que no todos los empleados tendrán acceso al sistema y sólo un administrador tendrá todos los permisos para agregar, modificar y eliminar datos y registros de empleados, clientes e inmuebles.

Los reportes generados por la aplicación son de gran ayuda para la compañía en cuanto a las evaluaciones de seguimiento y rendimiento tanto a nivel empresarial como a nivel de trabajo de cada uno de los asesores, ya que estos reportes muestran el progreso de los negocios generados y el desempeño de cada empleado.

La aplicación se seguirá mejorando y buscando optimizarla con nuevas versiones, por esta razón se garantiza añadirle mejores funcionalidades puesto que la aplicación cuenta con las características necesarias para poder agregarle nuevas tareas.

## 8. SUGERENCIAS Y RECOMENDACIONES

Verificar que la carpeta temporales que viene en el CD de instalación, quede guardada en la partición C del equipo en el que se instaló la aplicación, debido a que existen procesos importantes que necesitan obtenerse desde esa ruta.

Una vez esté instalado el servidor Apache-TomCat-6.0.18, ingresar a la siguiente ruta:

C:\Archivos de programa\Apache Software Foundation\Tomcat 6.0\webapps

Dentro de esta ubicación pegar la carpeta "VSM" (proyecto web), que viene dentro de la carpeta temporales en el CD de instalación de la aplicación, este procedimiento permite visualizar los reportes se generan desde el sistema.

Activar el servidor Apache Tomcat:

Inicio, Todos los programas, Apache TomCat 6.0, Configure Tomcat y pulsar el botón Start.



## 9. BIBLIOGRAFIA

SOMMERVILLE, Ian. Ingeniería del Software, Séptima edición, Pearson Educacion, S.A

Metro Cuadrado.com, Finca Raiz Metro a Metro

<http://www.metrocuadrado.com/servlet/co.com.m2.servlet.MostrarHome>

Wikipedia enciclopedia libre (On Line)

Página web. <http://es.wikipedia.org>

Prototipado

<http://www.sidar.org/recur/desdi/traduc/es/visitable/tecnicas/Prototyping.htm>  
(22/04/2009)

Comunicaciones y sistemas informáticos

<http://cfinmo.cisisl.com/>

Datahouse Company, Software y soluciones informáticas

<http://www.datahousecompany.com.ar/sistema-gestion-inmobiliarias.html>

# ANEXOS

## MANUAL DEL SISTEMA

En el CD vienen todos los instaladores necesarios para poder hacer uso de “VSM”

Inicialmente, se debe crear la base de datos MySQL de la aplicación y activar el servidor Apache, para esto es necesario ejecutar el instalador de Xampp-win32-1.7.0-installer que viene en el CD dentro de la carpeta instaladores.

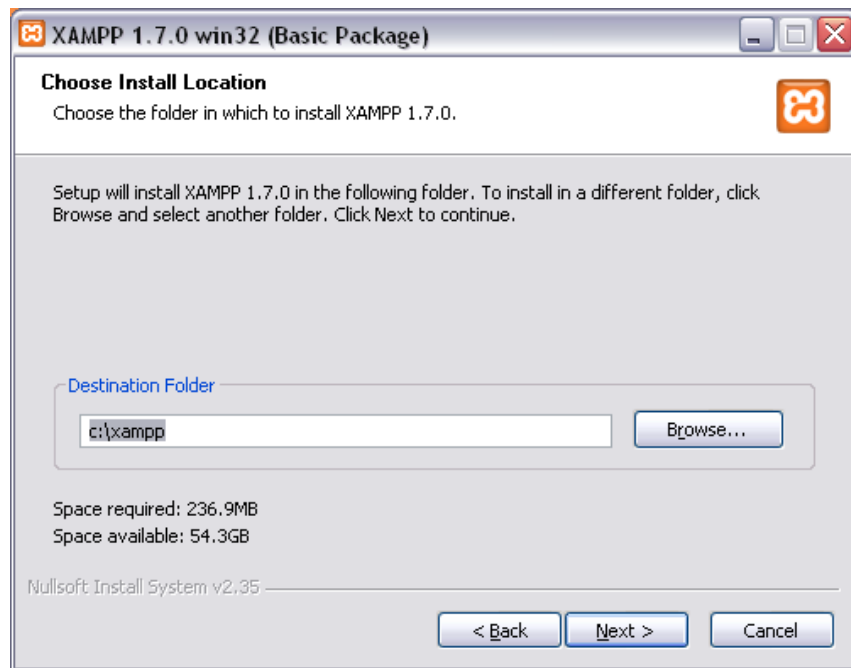
Doble clic en el icono de Xampp

Figura 16 Instalar Xampp



Seleccionar la ubicación, preferiblemente en la unidad C

Figura 17 Seleccionar Ubicación



Seleccionar los servicios necesarios para poder crear la base de datos y generar los reportes.

- ✓ Install Apache as service
- ✓ Install MySQL as service

Figura 18 Identificar los servicios

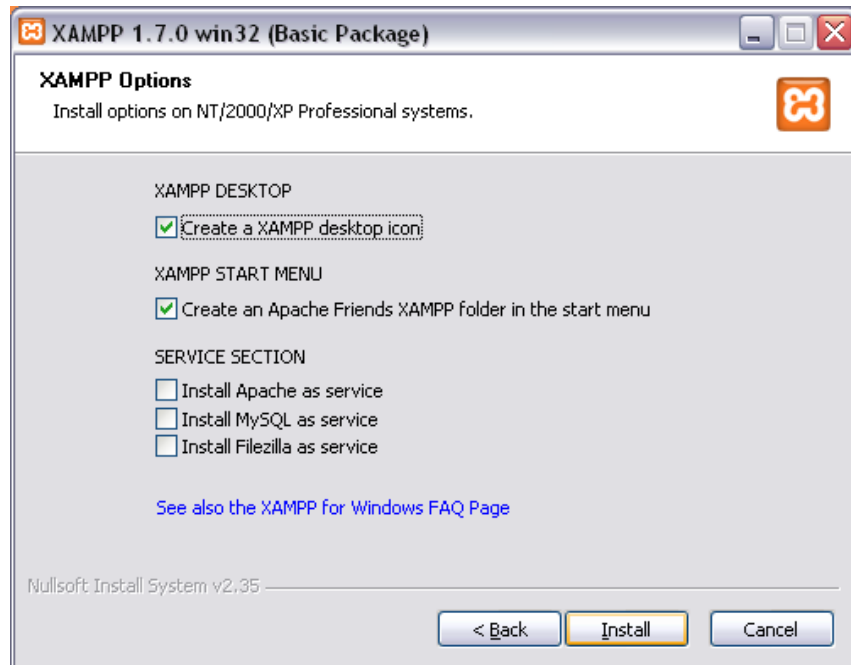
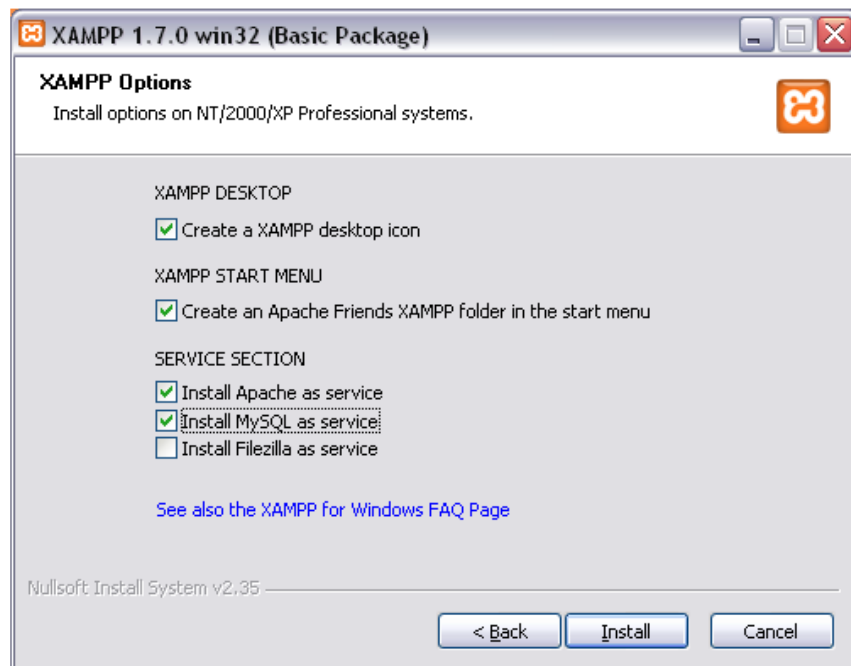


Figura 19 Seleccionar servicios



## INSTALACION

Figura 20 Instalar

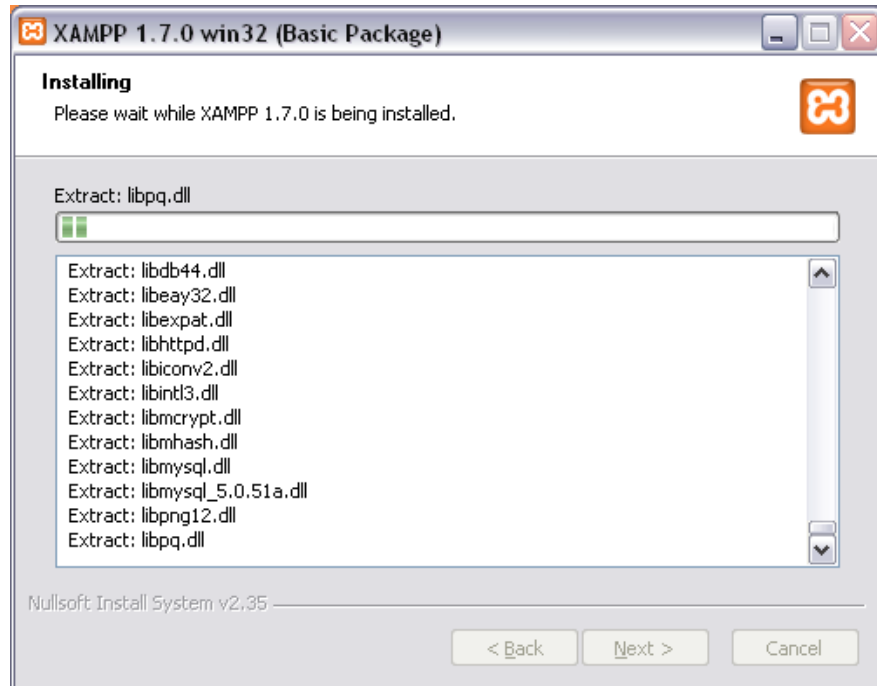


Figura 21 Instalando



Figura 22 Finalizando instalación

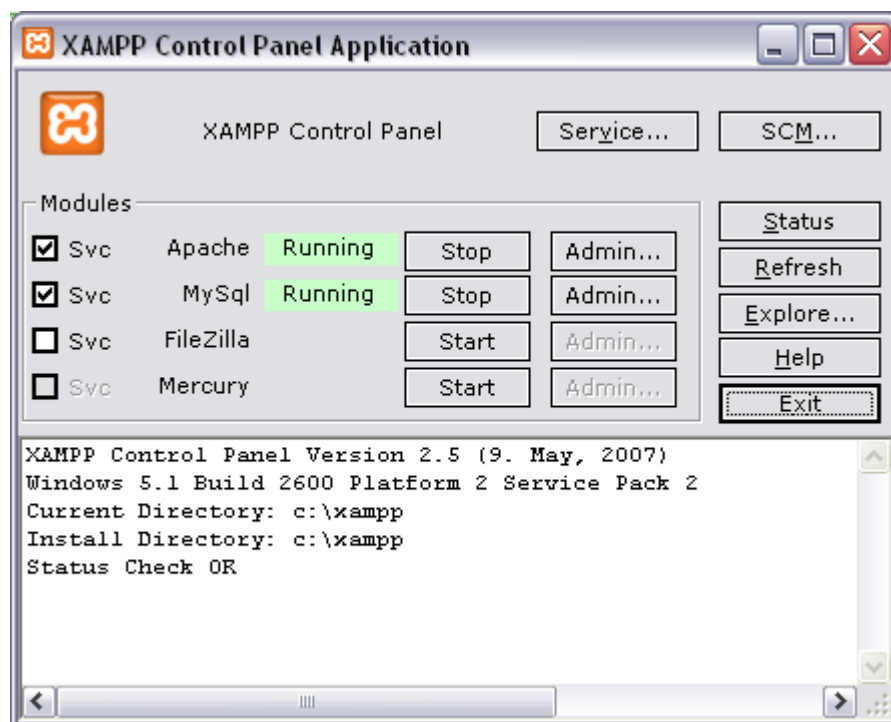


Figura 23 Instalación completa



Cuando la instalación se completa, automáticamente se muestra la siguiente ventana en donde se debe verificar que los servicios activados estén corriendo normalmente.

Figura 24 Verificando servicios



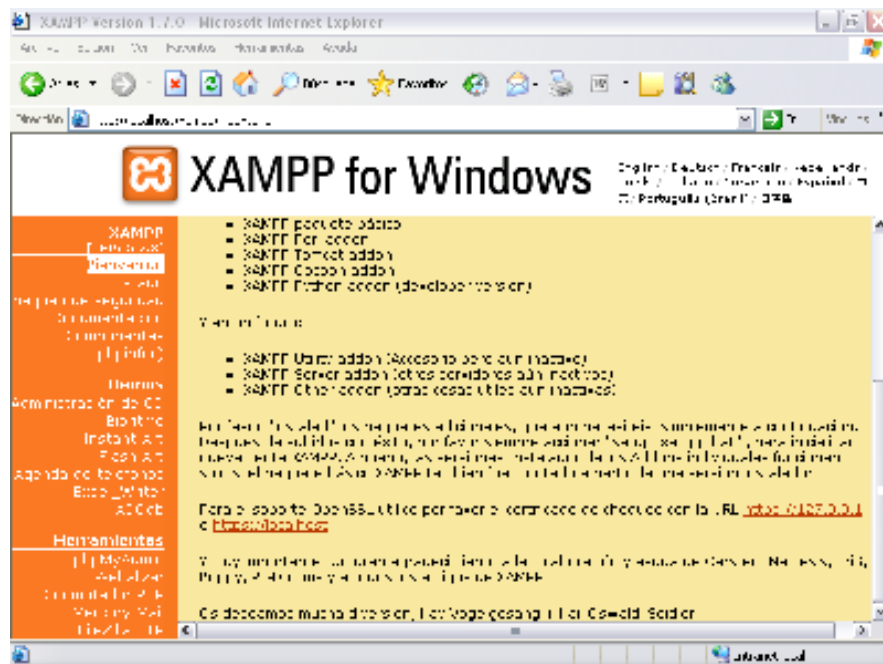


Una vez instalado el Xampp con los servicios seleccionados, para verificar que esté correctamente instalado, se debe abrir el explorador y se digita la siguiente instrucción:

### Localhost

Se selecciona el idioma y luego, en la parte izquierda de la pantalla, en herramientas seleccionar el link phpMyAdmin, para poder crear la base de datos de la aplicación, por consiguiente se muestra la siguiente pantalla.

Figura 25 Verificando la correcta instalación

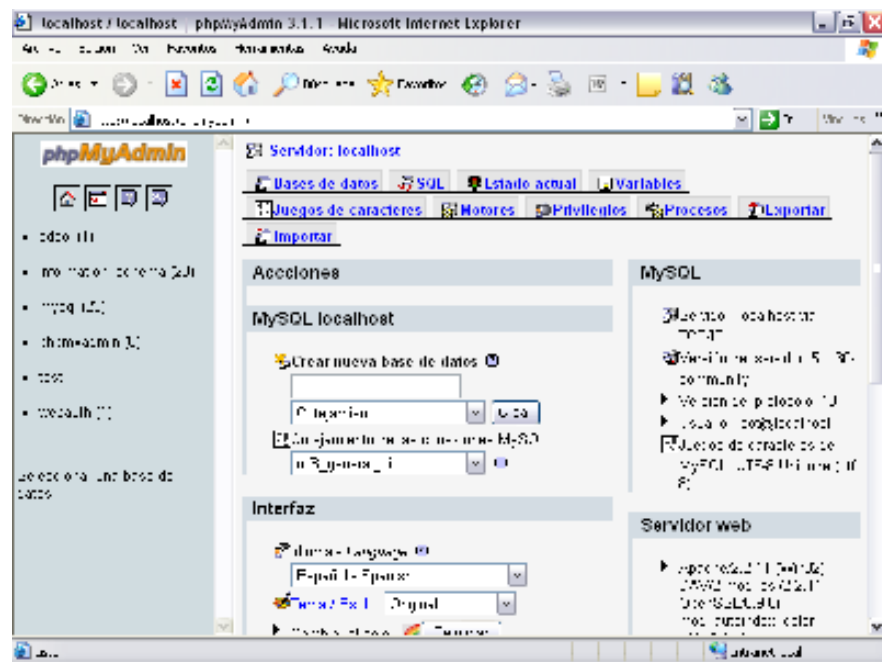


Después de ingresar por phpMyAdmin, se muestra la opción de la creación del nombre de la base de datos. En el campo habilitado para crearlo se debe escribir textualmente:

### Sundaymanager,

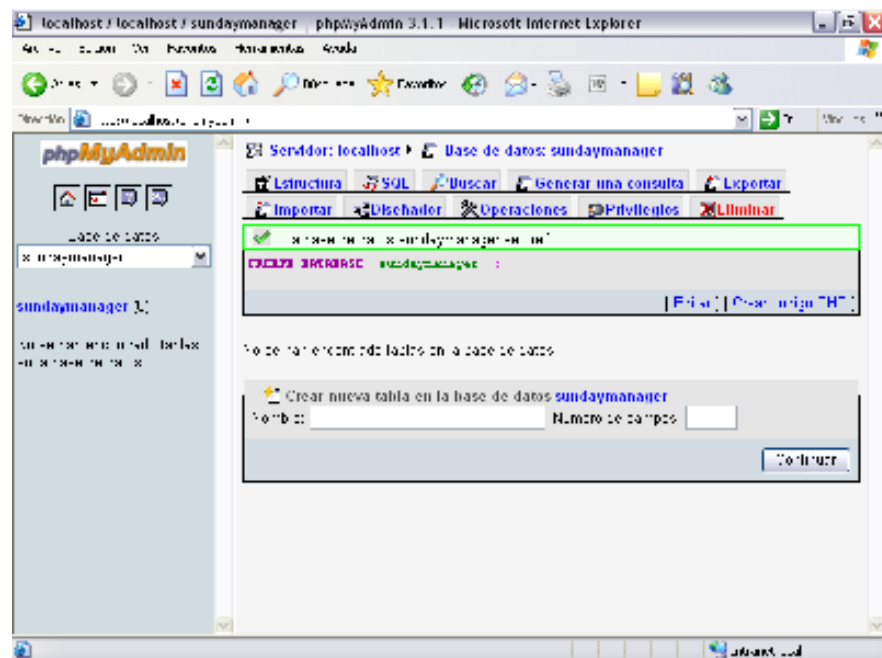
Debido a que la base de datos, se va a exportar desde la carpeta de instalación del aplicativo, por esta razón el nombre debe coincidir. Posteriormente se pulsa el botón crear.

Figura 26 Nombrar la base de datos



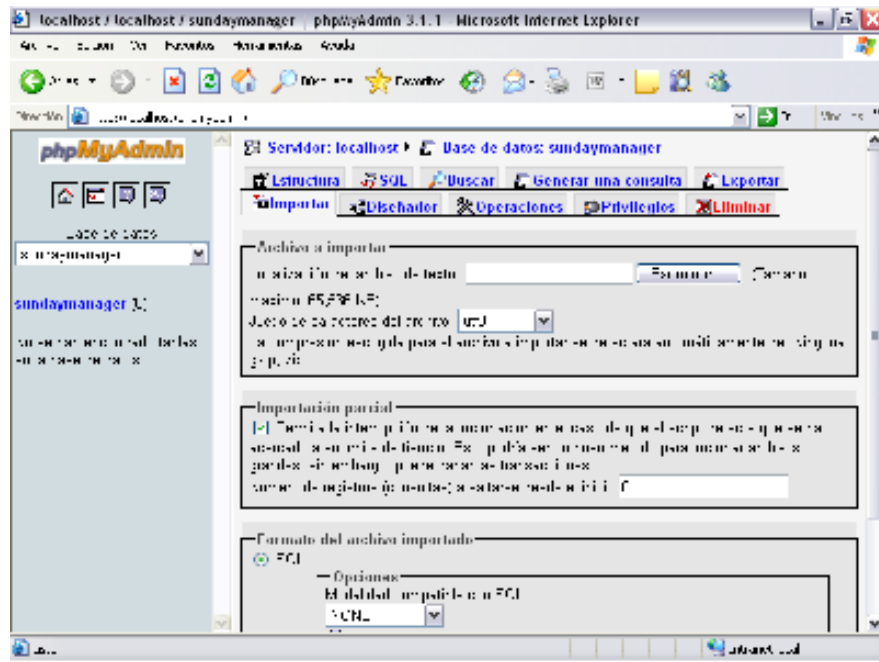
Luego se ingresa por la opción importar, para poder extraer la base de datos.

Figura 27 Opción importar



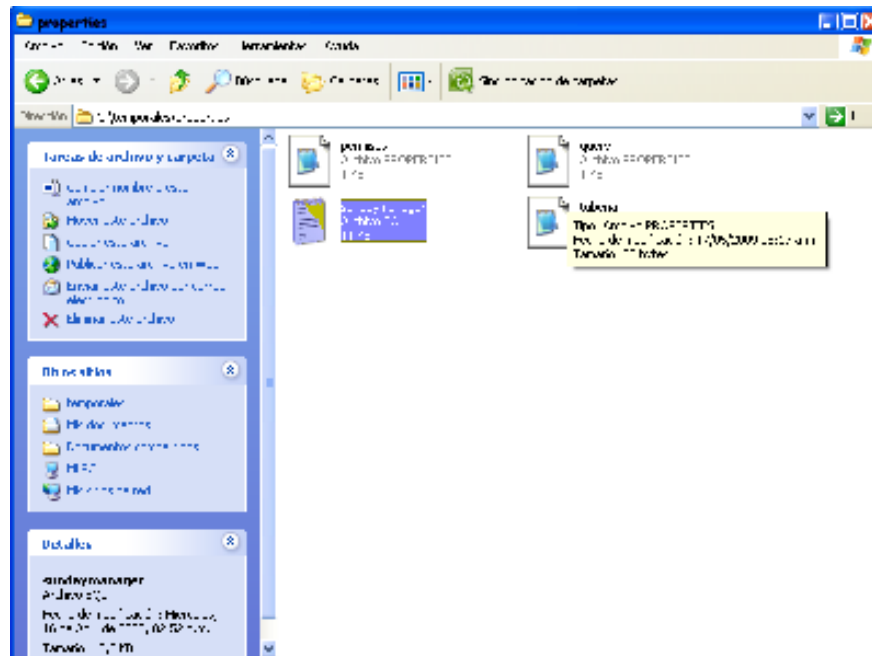
Pulsar el botón examinar, para seleccionar de la carpeta que se ha guardado en C la base de datos y subirla.

Figura 28 Examinar base de datos



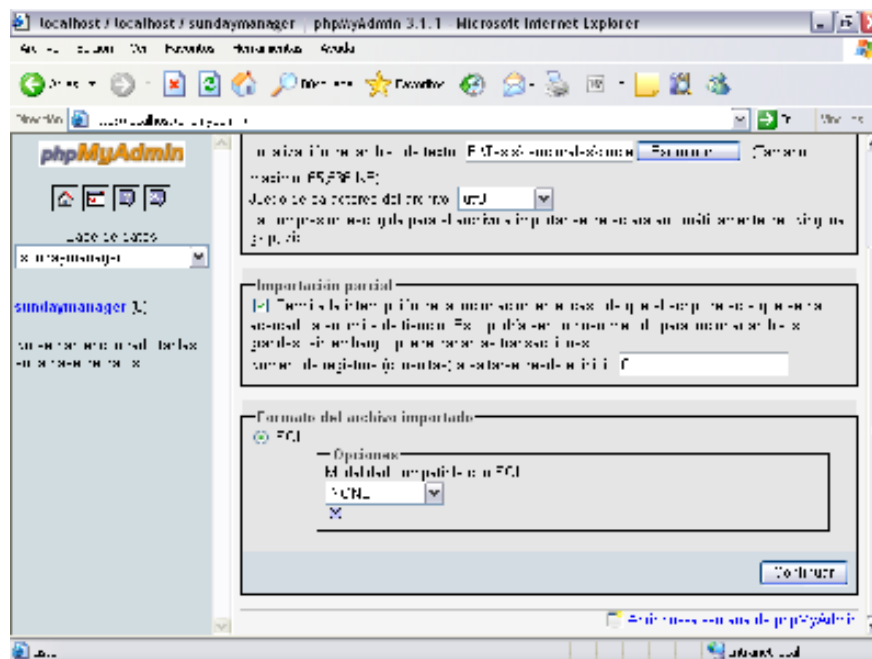
La base de datos se encuentra en la carpeta C:\temporales\properties y sundaymanager como un archivo de block de notas.

Figura 29 Ubicación base de datos



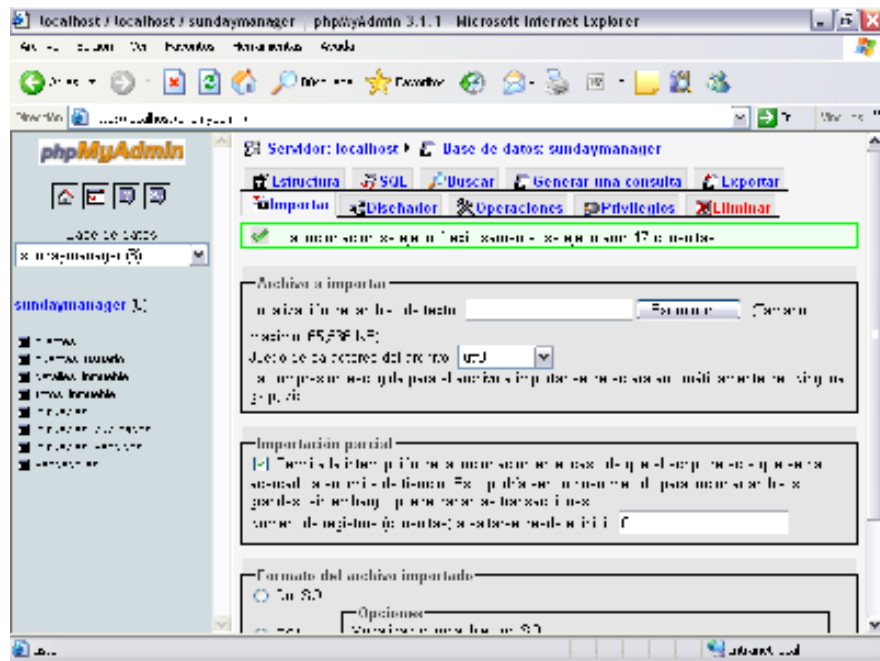
Una vez identificado el archivo que contiene la base de datos, este se selecciona y se importa. Pulsar el botón continuar.

Figura 30 Creación base de datos - Continuar



En la parte izquierda de la pantalla se puede observar que se crearon las tablas necesarias para poder guardar toda la información.

Figura 31 Base de datos creada



Una vez se haya creado la base de datos, se procede a la instalación del servidor Apache-TomCat-6.0.18 que también viene en el CD de instalación dentro de la carpeta instaladores.

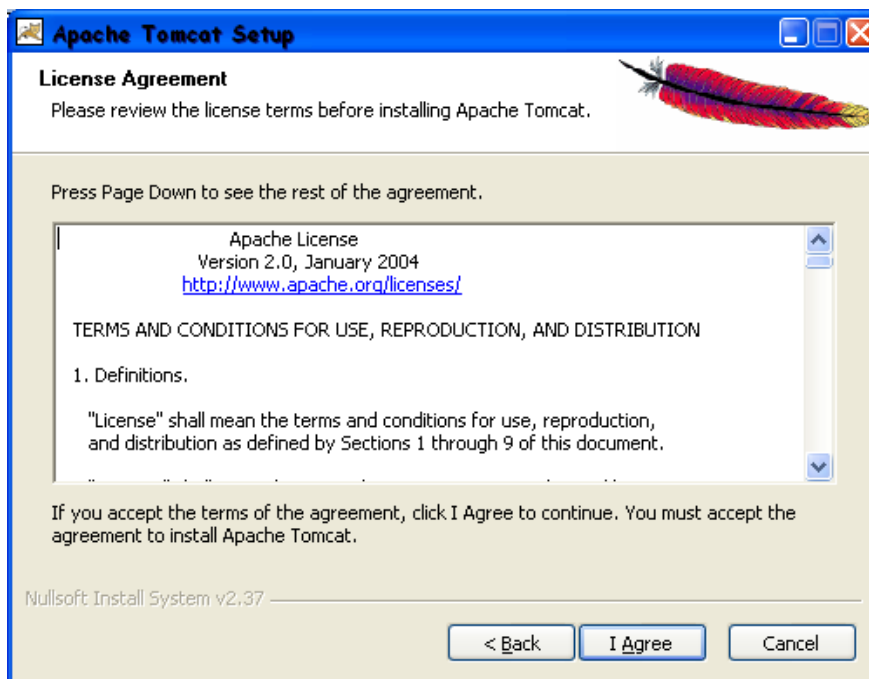
Se inicia haciendo doble clic en el icono del instalador de apache y se continúa con el proceso normal de la instalación, ya que es bastante intuitivo.

Figura 32 Pantalla de presentación Apache TomCat



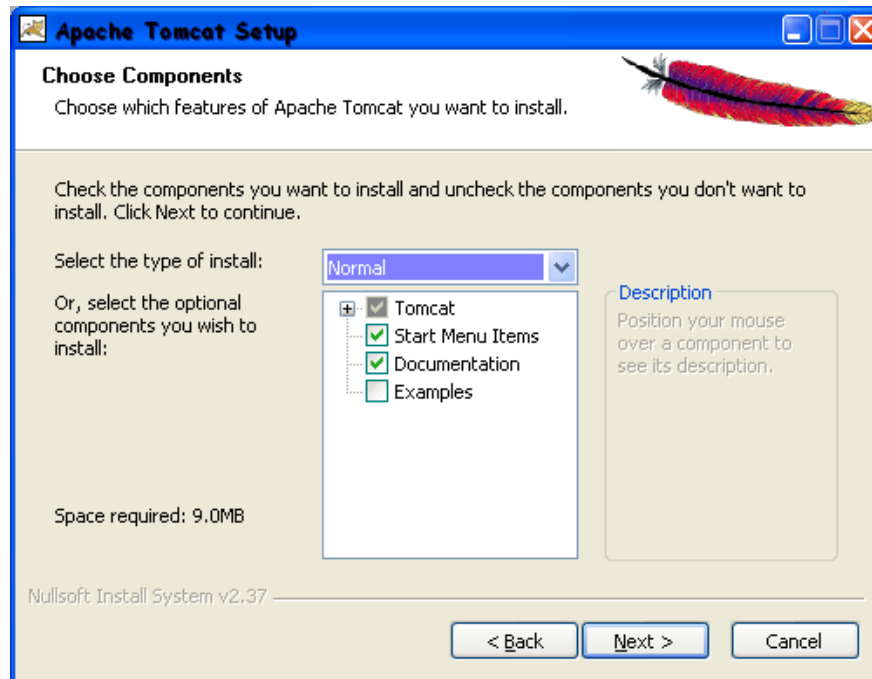
Aceptar los términos de la licencia para poder continuar con el proceso.

Figura 33 Aceptar términos de la licencia



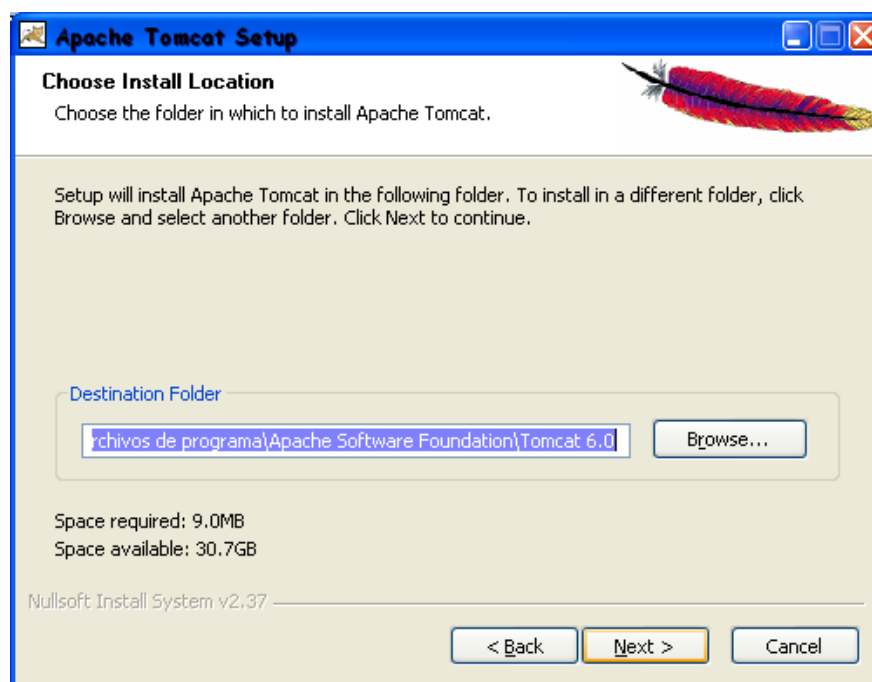
Los componentes están seleccionados por defecto, solamente se debe pulsar el botón siguiente.

Figura 34 Componentes



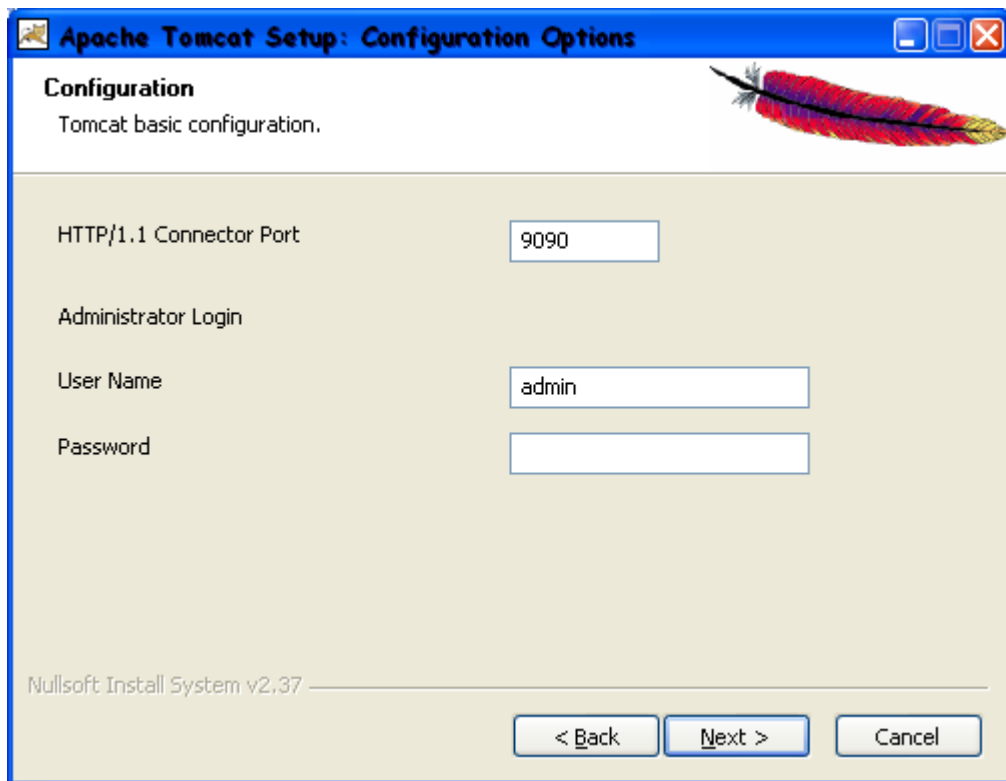
Elegir la ubicación de la instalación de Apache TomCat, preferiblemente en C

Figura 35 Ubicación



Esta parte muestra el puerto con el que se va a conectar con el servidor, en este caso se utilizará el puerto **9090**, es importante verificar que el servidor haya quedado corriendo por este puerto, para que se visualicen correctamente los reportes generados por la aplicación, aquí no es necesario escribir nada, simplemente clic en siguiente.

Figura 36 Puerto



Seleccionar ubicación para la instalación de la máquina virtual de Java



Figura 37 Ubicación máquina virtual de Java

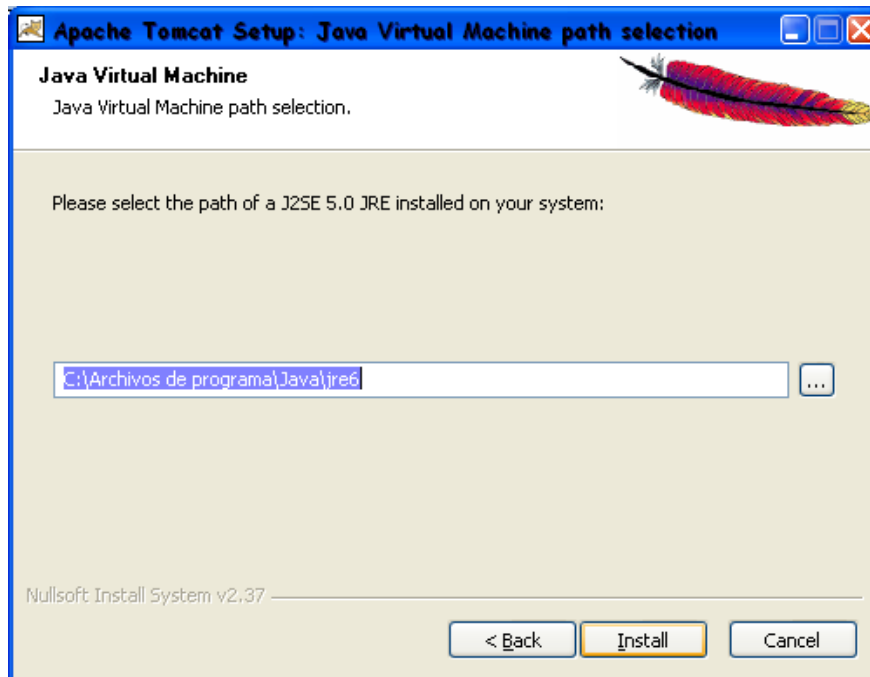
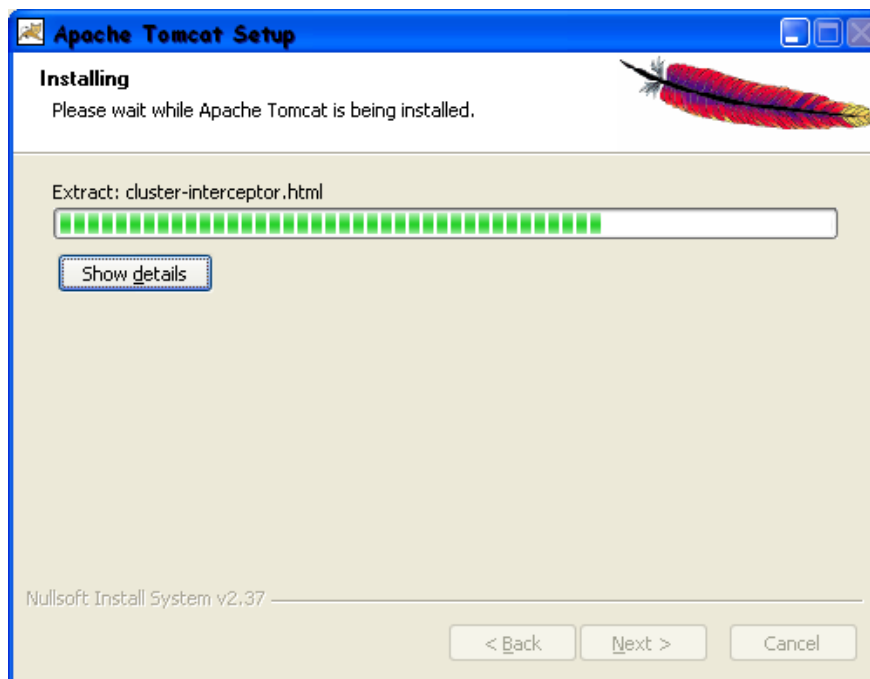


Figura 38 Instalando componentes



Instalación finalizada.

Figura 39 Instalación completa

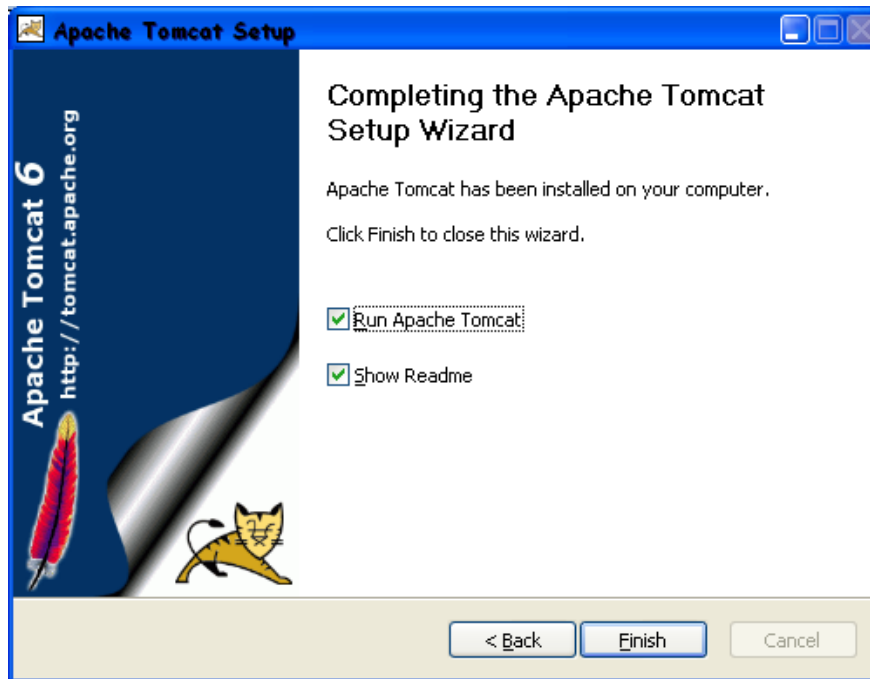
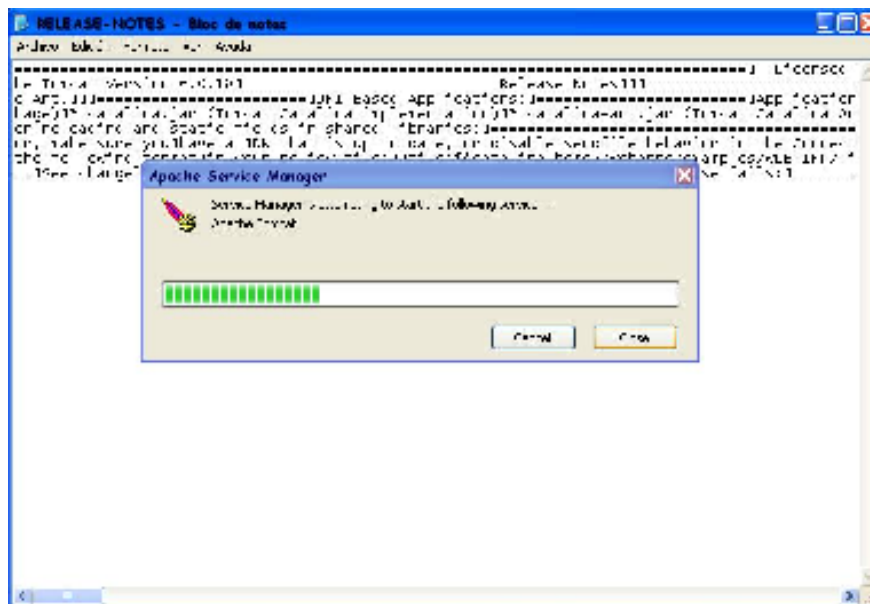


Figura 40 Instalación completa



MANUAL DE USUARIO  
"VSM"  
VISUAL SUNDAY MANAGER  
SISTEMA ADMINISTRADOR Y MANEJADOR DE INFORMACION  
INMOBILIARIA

ANGIE PAOLA RODRÍGUEZ RAMÍREZ

CORPORACION UNIVERSITARIA MINUTO DE DIOS  
FACULTAD DE INGENIERÍA  
DEPARTAMENTO DE TECNOLOGÍA EN INFORMÁTICA  
BOGOTÁ, D.C

I - 2009

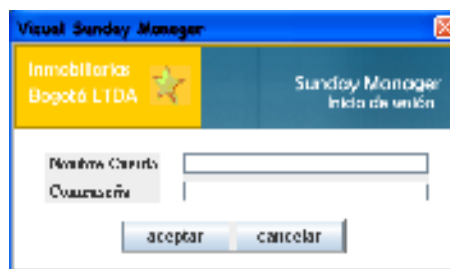
**“VSM”**  
**Visual Sunday Manager**  
**Administrador y manejador de información comercial**

El sistema “VSM” es una aplicación que administra y maneja toda la información comercial a nivel inmobiliario; este sistema lleva un control de los clientes vinculados con la compañía, los bienes que se quieren vender o arrendar, con sus respectivas características, los asesores comerciales y finalmente de los negocios que se logren concretar.

**1. INGRESO AL APLICATIVO**

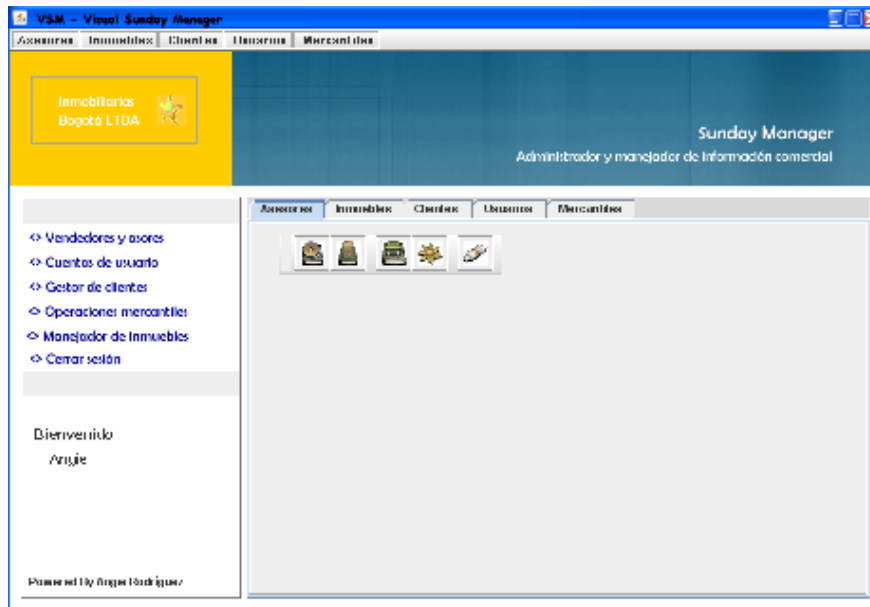
Cuando se ejecuta la aplicación, se le va a solicitar al usuario que ingrese el nombre de la cuenta y su respectiva contraseña.

**Figura 41 Pantalla Inicio De Sesión**



Una vez el usuario haya ingresado correctamente los datos solicitados por la aplicación, aparecerá una pantalla contenedora del menú principal en donde se visualizarán todos los módulos a los que se tendrá acceso, según los permisos de la cuenta, al dar clic en cualquiera de los módulos habilitados se desplegarán las diferentes opciones en las que se quiera trabajar.

Figura 42 Menú Principal

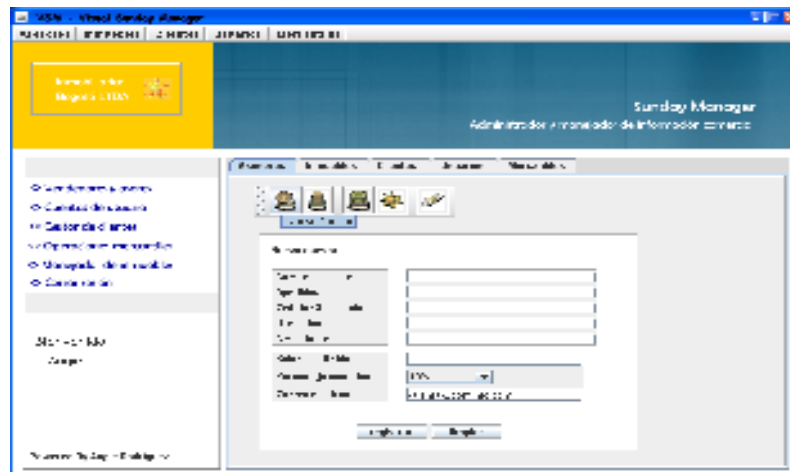


Se ha ingresado al aplicativo con el usuario administrador, por esta razón tendrá todos los permisos habilitados. El usuario para los empleados no podrá eliminar ni modificar información de ningún tipo en ninguna de las fases del sistema, sólo podrá consultar, agregar nuevos clientes, inmuebles, registrar datos cuando se cierre algún negocio e imprimir reportes.

## 2. MÓDULO ASESORES

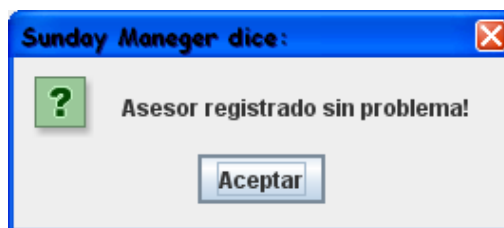
Éste módulo posee cinco características, la primera es agregar un nuevo asesor y se refiere al primer ícono en la pantalla.

Figura 43 Módulo Asesores



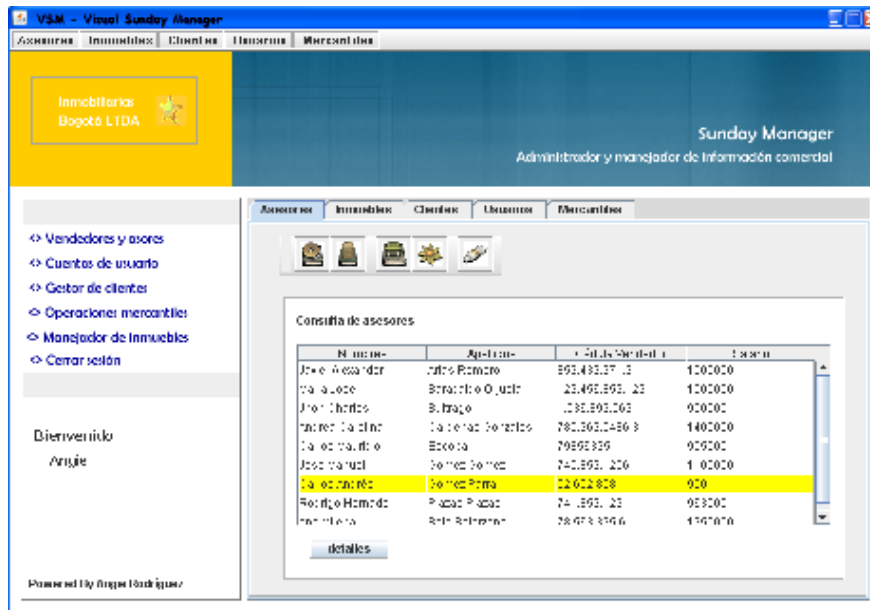
Después de haber llenado correctamente todos los campos, la aplicación mostrará el mensaje vendedor registrado sin problema.

Figura 44 Mensaje Asesor Registrado



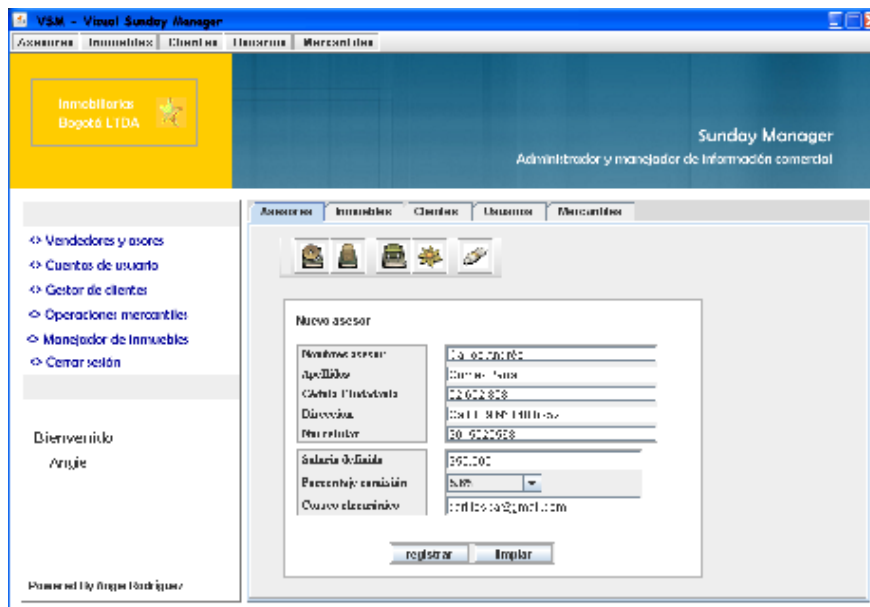
El segundo ícono muestra el listado de todos los asesores que pertenecen a la compañía.

Figura 45 Listado de Asesores



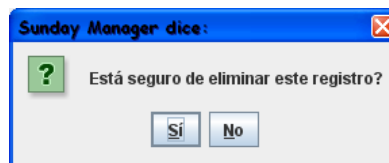
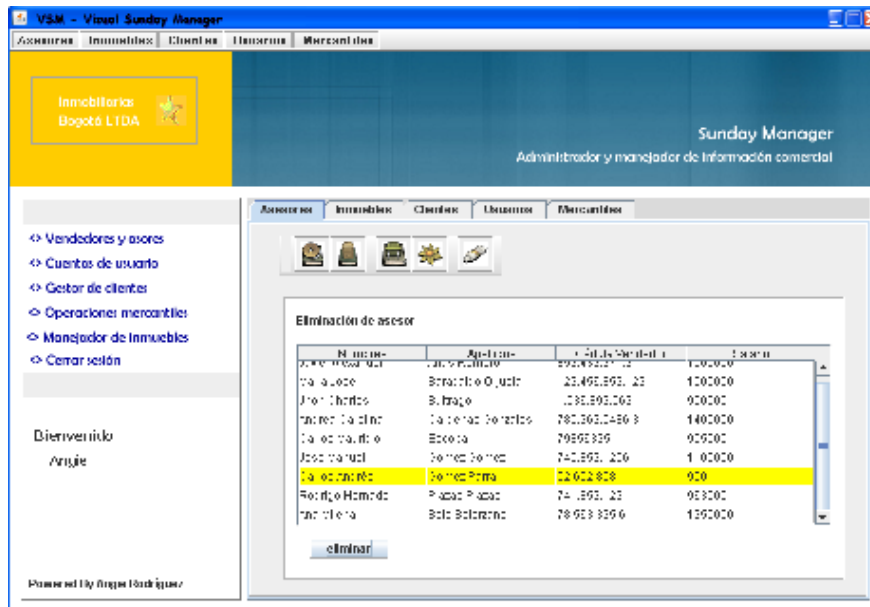
Se selecciona el asesor del cual se desea conocer más información y se pulsa el botón detalles, a continuación se mostrará la siguiente pantalla.

Figura 46 Detalle Asesores



El usuario administrador podrá eliminar la información que no quiera tener más en la aplicación, en este caso eliminar asesor.

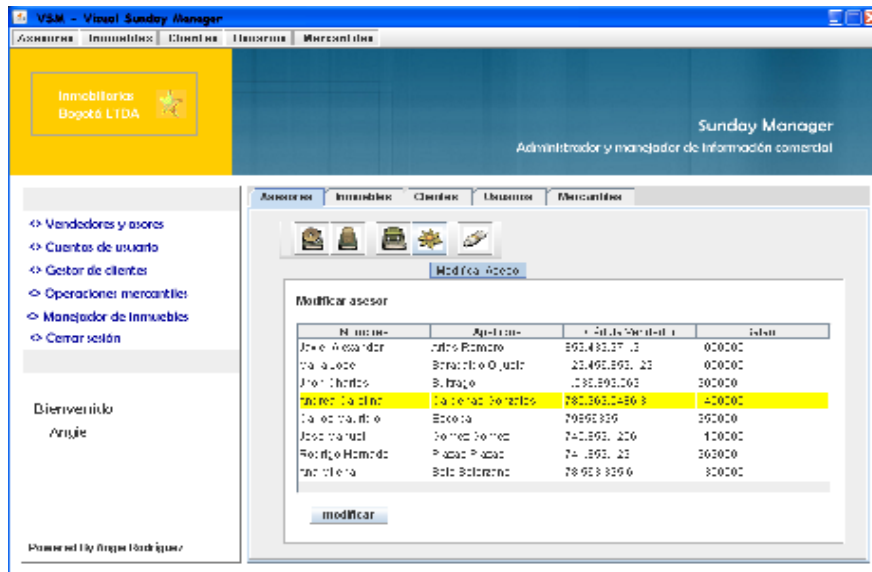
Figura 47 Listado Asesores – Selección para Borrar



Cuarto ícono, modificar asesor, al dar clic se desplegará la lista de todos los asesores que se encuentren en la base de datos, allí se selecciona el nombre del asesor que se quiere modificar y a continuación un clic en modificar.

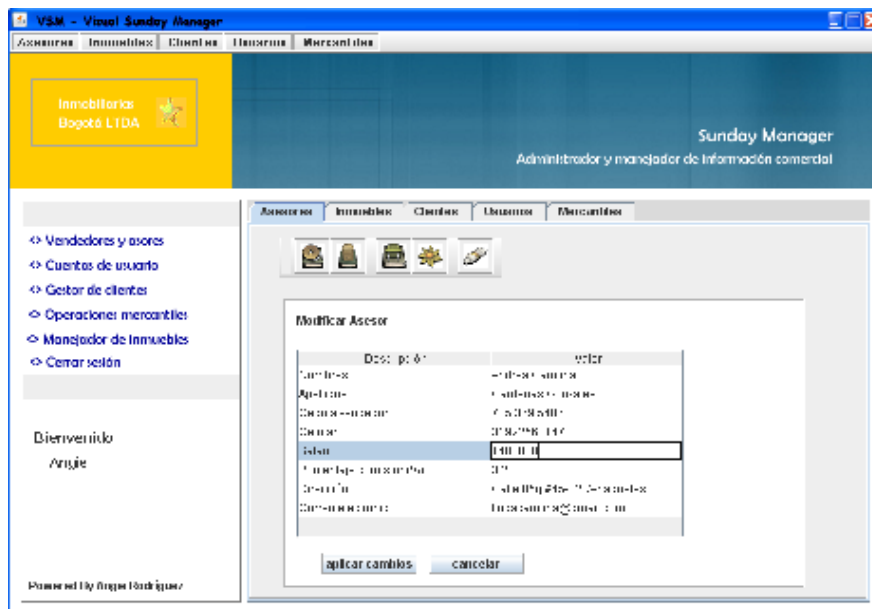


Figura 48 Listado Asesores – Selección para Modificar



Una vez se haya seleccionado el asesor a modificar, se podrán realizar los cambios necesarios y se finaliza con el botón aplicar cambios.

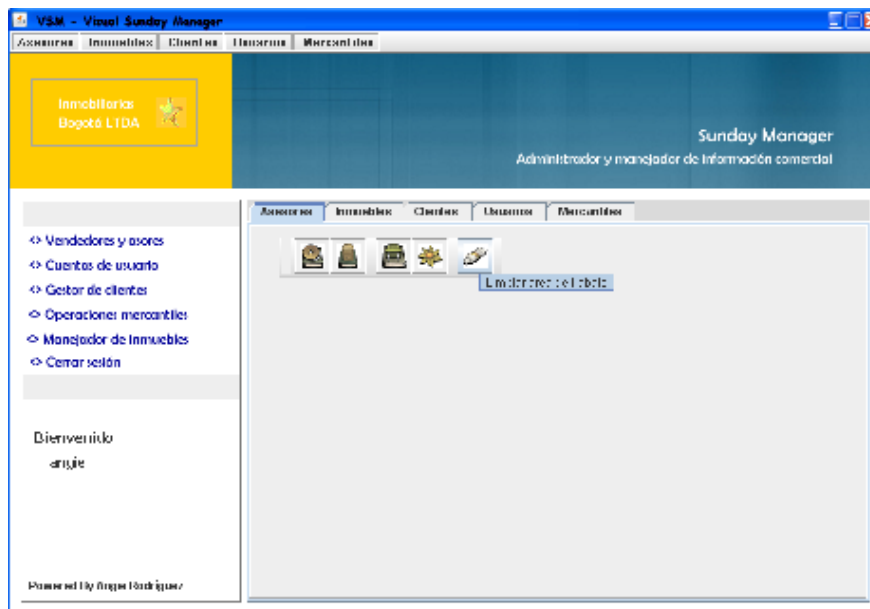
Figura 49 Consulta Asesor – Modificando Campos



El último icono hace referencia a limpiar el área que se está consultando, este icono se encuentra en todos los módulos que se van a explicar más adelante,

cuando se pulsa ese botón, simplemente deja la pantalla en blanco como se puede denotar en la siguiente gráfica.

Figura 50 Pantalla Inicial Usando el Botón Limpiar

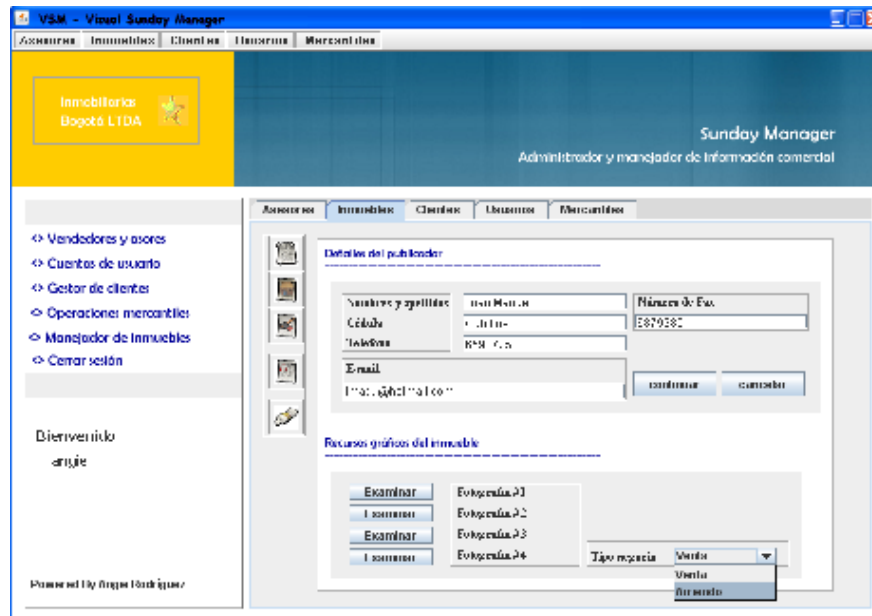


#### MÓDULO INMUEBLES

Ingresando por este módulo, se podrá ver la primera opción que es ingresar inmueble, allí se le solicita al usuario que ingrese inicialmente los datos del cliente publicador o dueño del bien, hay que diligenciar todos los campos para que el sistema permita seguir ingresando la información propia del inmueble.

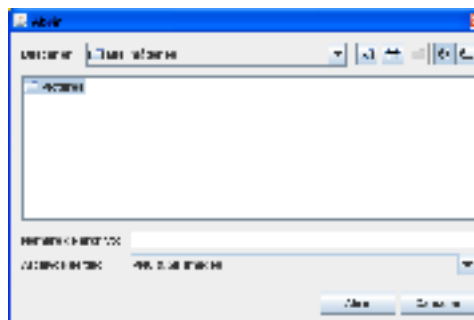
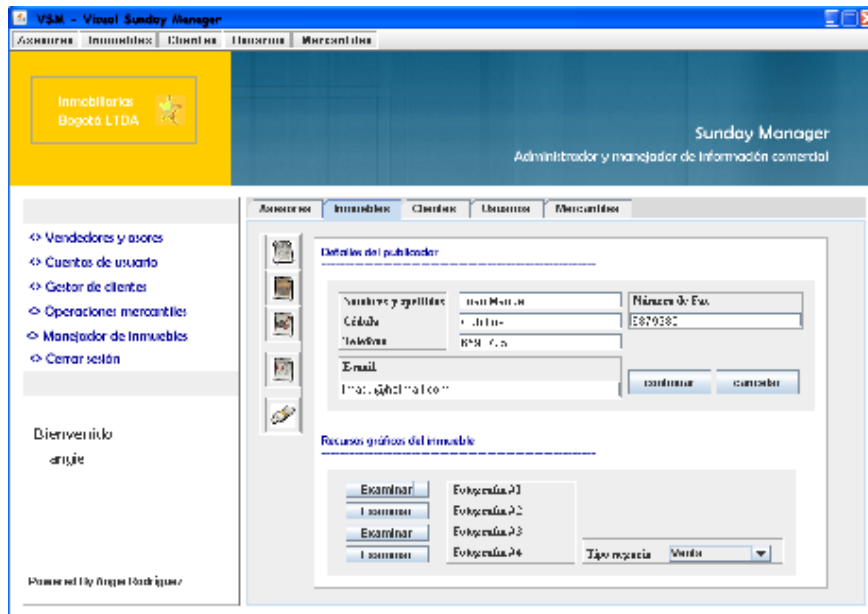
Una vez se hayan registrado los datos del cliente, se puede especificar el tipo de negocio, es decir, si es para la venta o es para rentar.

Figura 51 Ingresar Inmueble



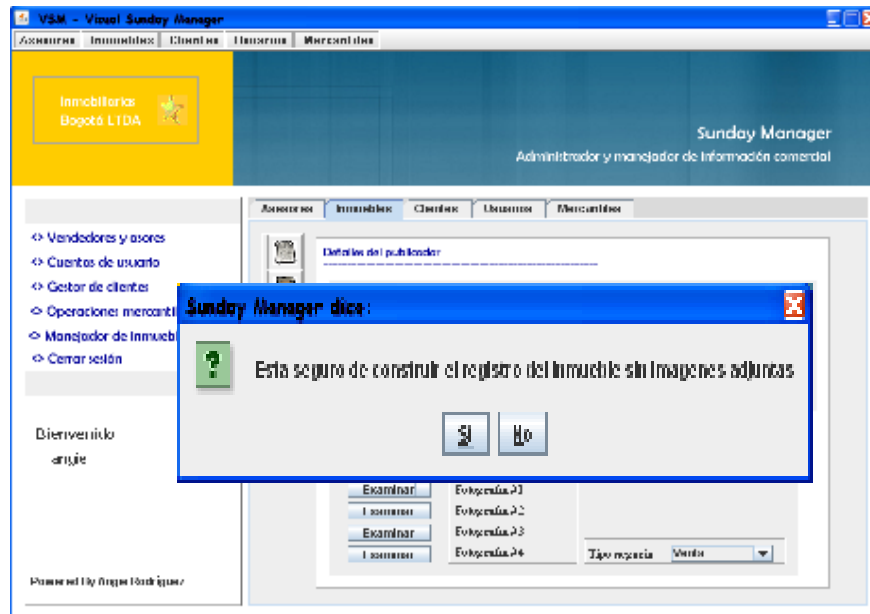
Del lado izquierdo del módulo Inmuebles se puede observar la opción que ofrece la aplicación de agregar imágenes para obtener una mejor visión del inmueble en el que el cliente está interesado. Se hace clic en el cuadro de diálogo examinar y el usuario podrá seleccionar desde su PC la imagen que desee, pero ésta debe ser de tipo PNG y/o GIF.

Figura 52 Pantalla para Ingresar Foto del bien



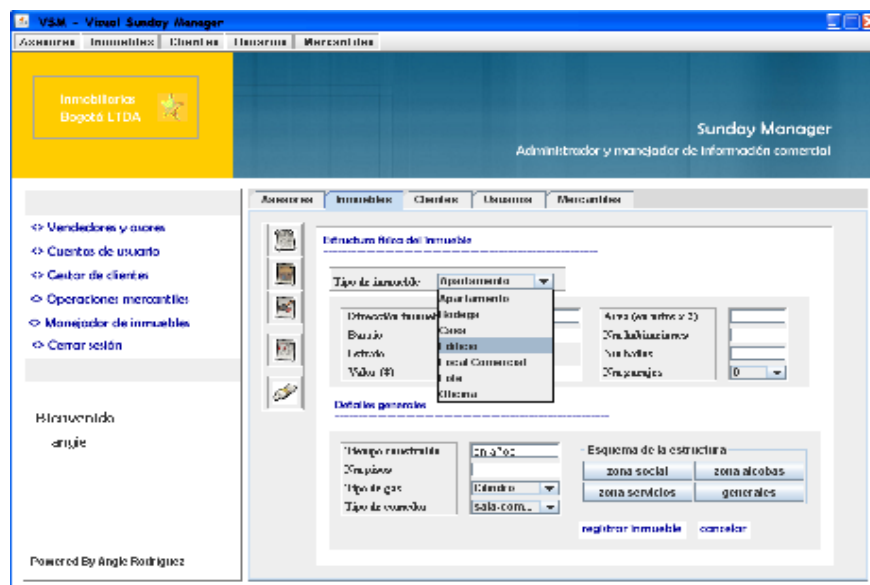
Después de haber subido las fotos al aplicativo, se debe pulsar el botón continuar, si el usuario no quiere ingresar fotos el aplicativo pregunta si desea continuar sin imágenes adjuntas, el usuario elige la opción sí, y la aplicación continúa sin ningún inconveniente.

Figura 53 Pantalla para Ingresar Foto del bien



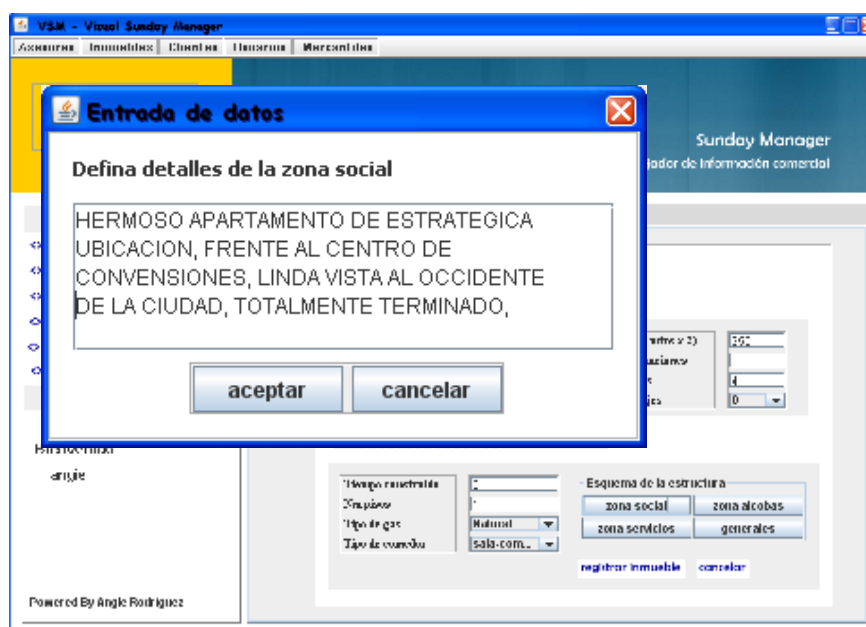
En la siguiente pantalla se solicitan los datos y características propias del inmueble; se selecciona el tipo de inmueble y se continúa diligenciando toda la información que el sistema pide.

Figura 54 Otras Características del Inmueble



En la parte inferior derecha, en el recuadro de Esquema de la estructura, hay cuatro opciones para que el usuario escriba libremente las descripciones que el cliente publicador quiera darle al inmueble, en este ejemplo se ingresa en el botón “zona social” automáticamente se despliega la casilla de texto, como se denota en la siguiente imagen. Una vez se haya ingresado la información de las demás opciones se pulsa la opción “registrar inmueble” para finalizar la operación.

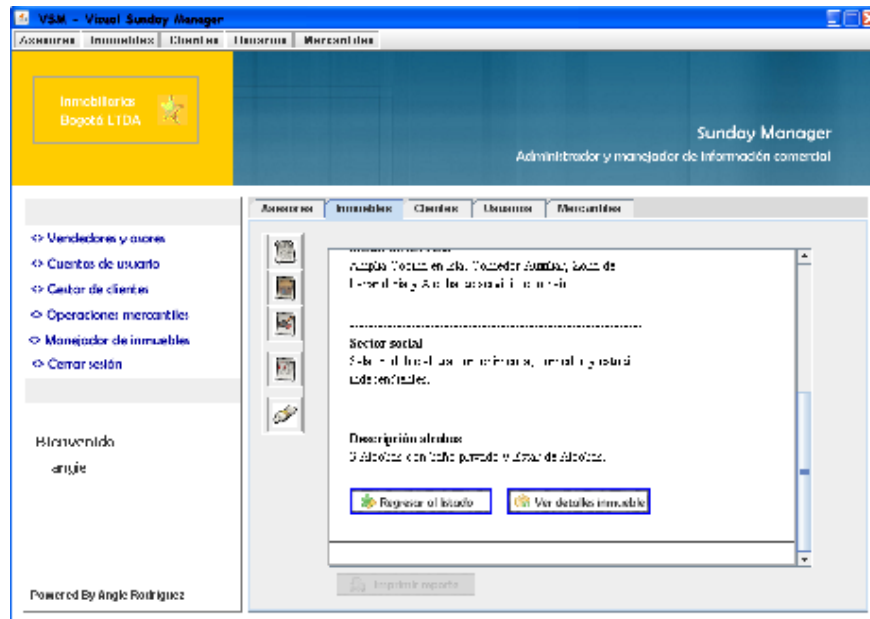
Figura 55 Otras Características del Inmueble



La siguiente opción que se encontrará en el módulo Inmuebles es: Buscar inmueble, allí se puede realizar una búsqueda del inmueble deseado, una vez se tenga identificado, se procede a ingresar en el link consultar.

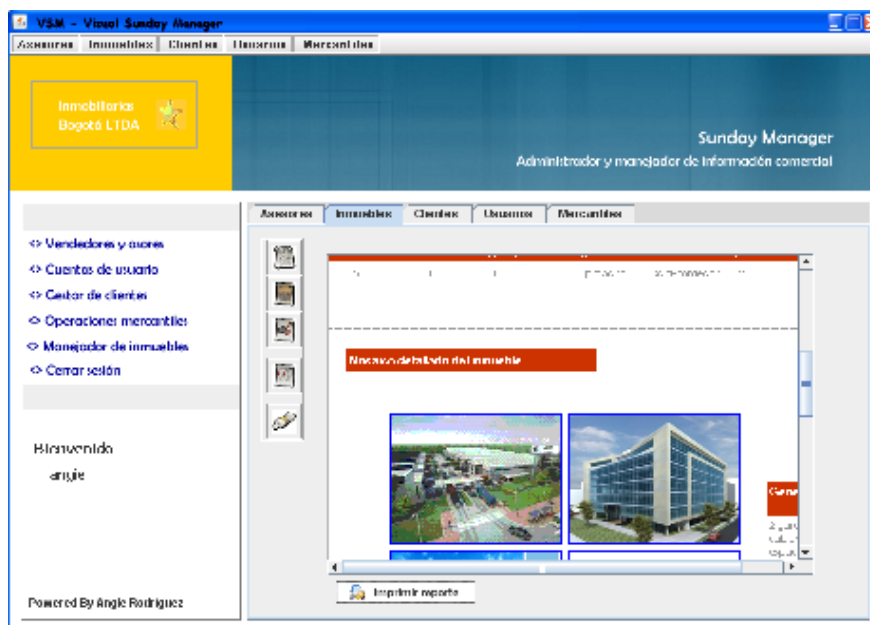


Figura 58 detalles del inmueble



Aquí se puede ver la primera opción de reporte que ofrece la aplicación.

Figura 59 Opción de Reporte



Se pulsa el botón “Impresión de reporte” y automáticamente se abre el explorador y muestra el informe para proceder con la impresión.



Figura 60 Reporte para Imprimir

<b>Inmobiliaria</b> Bogotá LTDA		<b>Inmueble Registrado Código :</b> #0003 Casa en el sector de Las aguas			
------------------------------------	--	---	--	--	--

---

Caracterización del inmueble (esquema externo)					
Dirección	Barrio	Tipo Inmueble	Valor (\$)	Estrato	Area del inmueble
diagonal 3ra #12-96a	Las aguas	Casa	850000	4	4 M <sup>2</sup>

Caracterización del inmueble (esquema interno)					
Total cuartos	Total Baños	Número de garajes	Tipo de gas	Tipo comedor	Nm de pisos
5	2	0	propano	sala-comedor	1

---

Mosaico detallado del inmueble	
	
<p align="center"><b>NO FOTO</b> recurso no disponible</p>	<p align="center"><b>NO FOTO</b> recurso no disponible</p>

Generales	
Magnífica vista al valle de Sopó y jardín privado. 4 Garajes y depósito. Completo clubhouse con zonas para niños, piscina, gimnasio y pesesbreras	

---

Descripción del medio	
<b>Zona de servicios</b> Cocina a Gas Natural con barra hacia el comedor y puerta al jardín, Zona de Lavandería y Alcobas de servicio con baño.	<b>Zona social</b> Sala con chimenea y comedor independiente con salida al jardín.
<b>Zona de alcobas</b> 4 Alcobas con baño privado o 3 alcobas y Estudio, Alcoba principal con terraza, Estar de Alcobas.	<b>Tiempo de construido</b> 6 años

---

Datos del actual propietario	
<b>Nombres propietario</b>	Miguel Andres Arias Romero
<b>Número Teléfono</b>	31245852
<b>Número Fax</b>	96584223
<b>Email</b>	viejomigue@hotmail.com
<b>Tipo de negocio</b>	Amiende

Reporte generado en la fecha	
2009-5-14	

El siguiente icono muestra la opción de eliminar los registros de los inmuebles creados, pero antes pregunta al usuario si está seguro de eliminar, ya que, no sólo se estaría eliminando el inmueble sino que también los datos del cliente publicador.

Figura 61 Selección del Inmueble – Para Eliminar

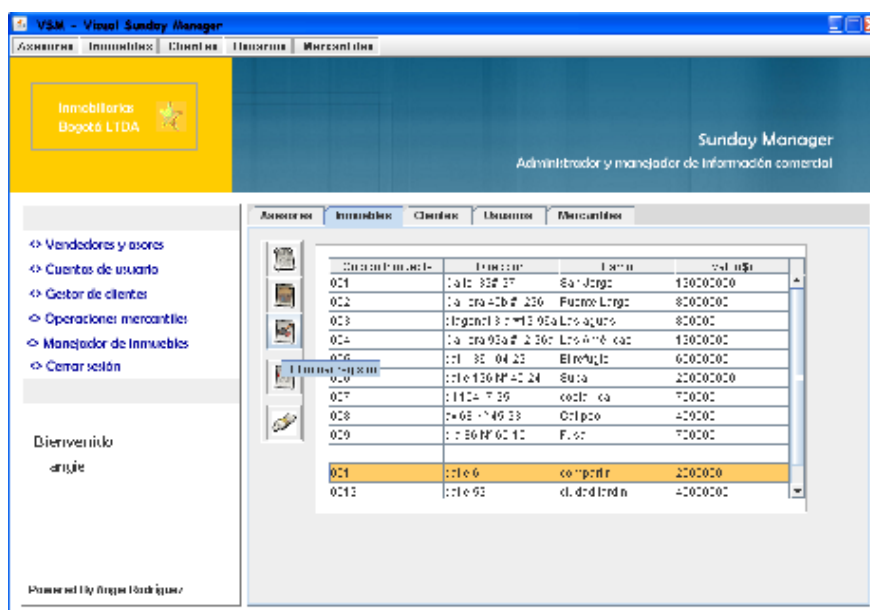


Figura 62 Selección del Inmueble – Para Eliminar

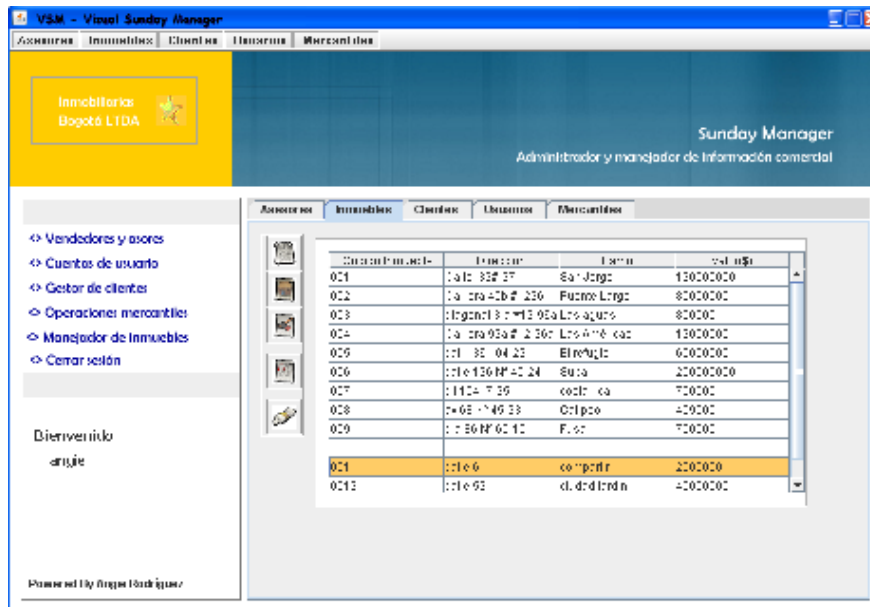
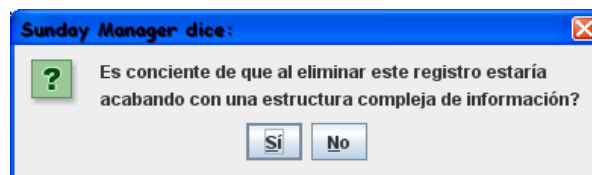
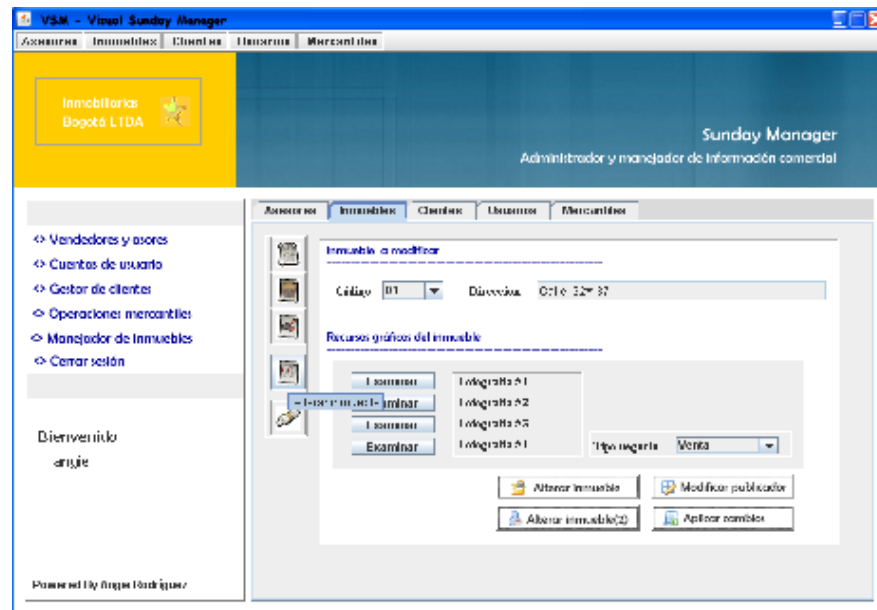


Figura 63 Verificar Borrado de Información



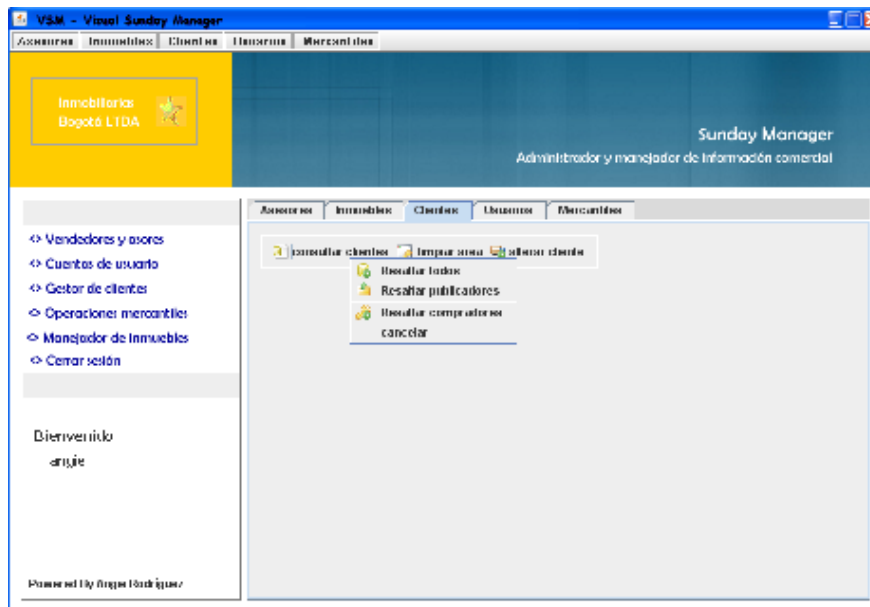
Quando se quiere modificar un inmueble, se hace únicamente desde la cuenta del administrador, se pulsa la opción “Alterar inmueble” e inmediatamente la aplicación vuelve a traer todos los datos que componen el inmueble a modificar, allí se pueden hacer los cambios necesarios.

Figura 64 Botón Modificar Inmuebles



### 3. MÓDULO CLIENTES

Figura 65 Módulo Clientes

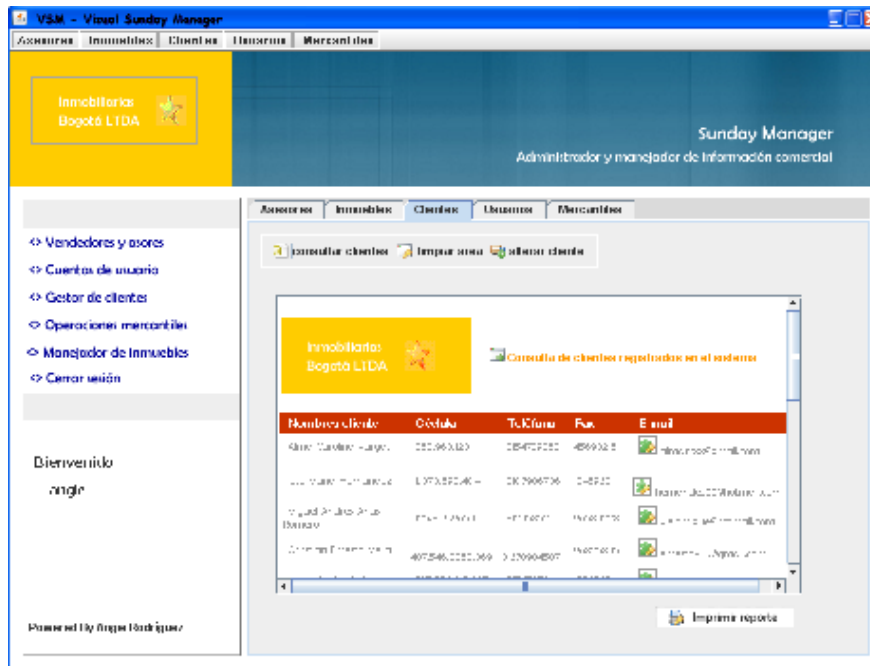


La primera opción que se puede ver en esta pantalla, es “consultar clientes”, se da un clic en esa opción y la aplicación mostrará cuatro opciones:

- Resaltar todos: Esta opción lista todos los clientes vinculados con la compañía, tanto compradores como publicadores.
- Resaltar publicadores: Aquí lista únicamente los clientes que han publicado algún bien.
- Resaltar compradores: Lista de los clientes que han comprado o tomado en arriendo algún inmueble.
- Cancelar: Deja de mostrar las opciones anteriores.


Todas estas listas se podrán imprimir como reporte.

Figura 66 Selección de lista de Clientes



El reporte de los clientes vinculados con la compañía, será visto de la siguiente forma:

Figura 67 Reporte de Clientes

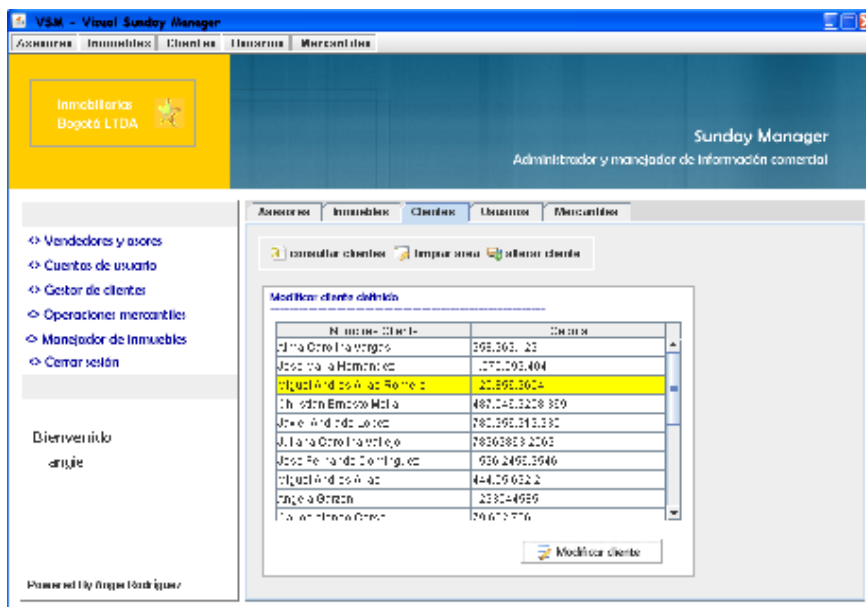
Inmobiliaria Bogotá LTDA  **Consulta de clientes registrados en el sistema**

Nombres cliente	Cédula	Teléfono	Fax	E-mail
Alma Carolina Vargas	256.963.123	3154789852	45693215	alma123@gmail.com
Jose Maria Hernandez	1.070.593.404	3107986706	1245963	hernandez88@hotmail.com
Miguel Andres Arias Romero	125.856.9654	31245852	96584223	viejomigue@hotmail.com
Christian Ernesto Mejia	487.546.3258.369	31678984587	96325845	ernesto859@gmail.com
Javier Andrade Lopez	785.256.312.235	31545652	1236968	andrade456@yahoo.com
Juliana Carolina Vallejo	78969863.2563	3124585236	31245963	julianita@gmail.com
Jose Fernando Dominguez	1596.2456.2546	3001245632	1236522	pinto@gmail.com
Miguel Andres Arias	444.55.632.2	3124512123	123456	arias115@yahoo.com
Angela Garzón	1225544585	6858574	7784596	agarzo@dominio.com
Carlos Alonso Corso	79.652.756	2365896	6352536	Carlitos69@dominio.com
Angela María Duarte	1956789566	6877455	6879878	angel6@gmail.com
Ana María Contreras		2455689	6589966	anma@yahoo.com
Sebastián Martínez		2565687	1546877	Sebas@hotmail.com
Andrés Hernandez	52858696	6844575	2365989	Andrus@yahoo.com
Javier Comedor	1234568	6589965	236689	jcomedor@gmail.com
Soley Casas	25663566	6565589	6556896	soleyhm@gmail.com
Sara Margarita Cubillos adfafafds	41596325	6585775	6585777	saritita@yahoo.com
carlos aquilar	4646546	6546546	654654	cuenta@yahoo.com
carlos aquilar	44444444444444	11111111111	00000000	carlos@dominio.com
tdtasa	445653	452455434	434354	cuenta@yahoo.com
Juan Manuel	Cubillos	6598775	6879985	jmaqu@hotmail.com

Reporte generado en la fecha 2009-5-14

La siguiente opción es, “alterar cliente”, en esta opción se podrán hacer los cambios que se requieran para el cliente.

Figura 68 Modificar Cliente



#### 4. MÓDULO USUARIOS

Esta opción le permite al administrador, crear cuentas de usuario para los empleados, se inicia llenando todos los campos y confirmando al terminar la operación.

Figura 69 Crear Nueva Cuenta de Usuario

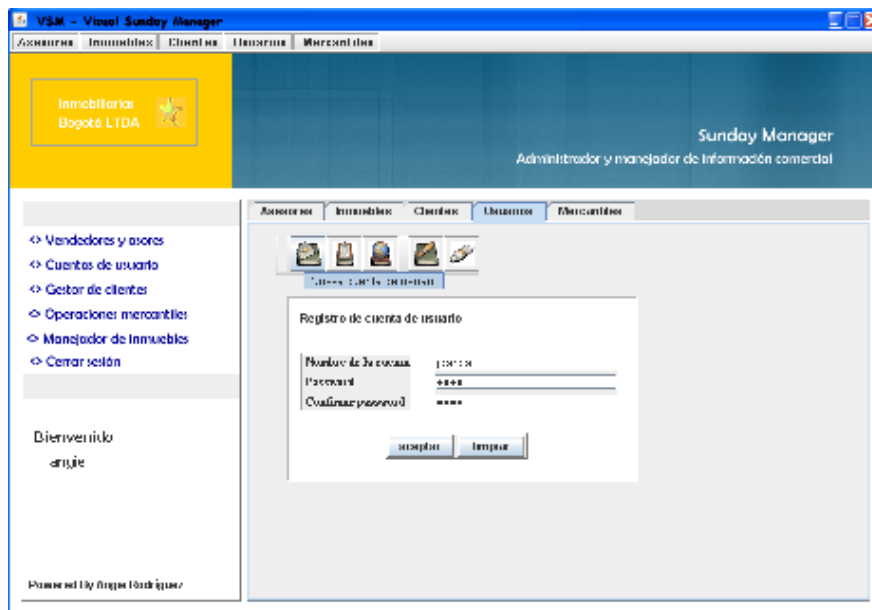
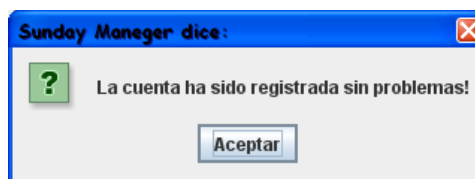


Figura 70 Confirmación Nueva Cuenta de Usuario



El segundo icono lista las cuentas de usuario creadas en la base de datos



Figura 71 Listado Cuentas de Usuario

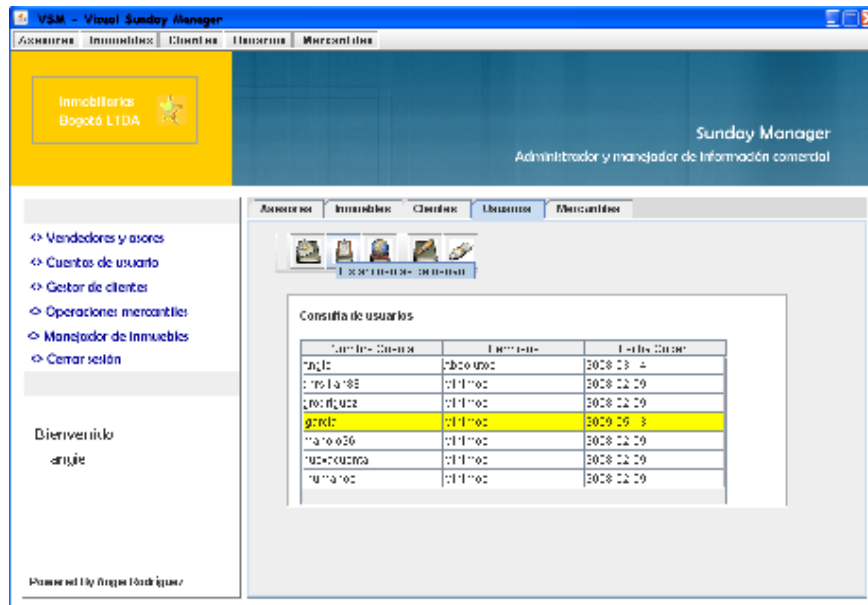
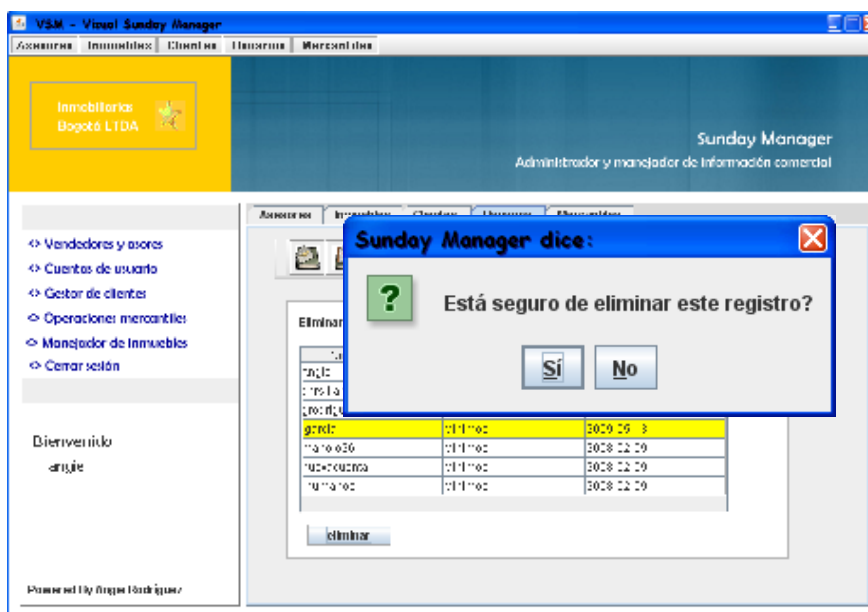
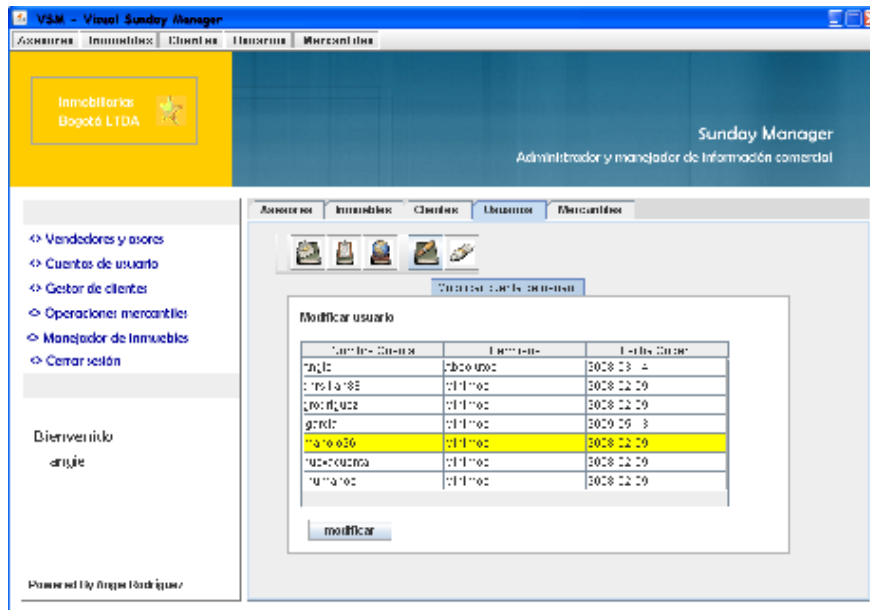


Figura 72 Eliminar Cuentas de Usuario



Eliminar registro.

Figura 73 Seleccionar Usuario - Modificación Usuario

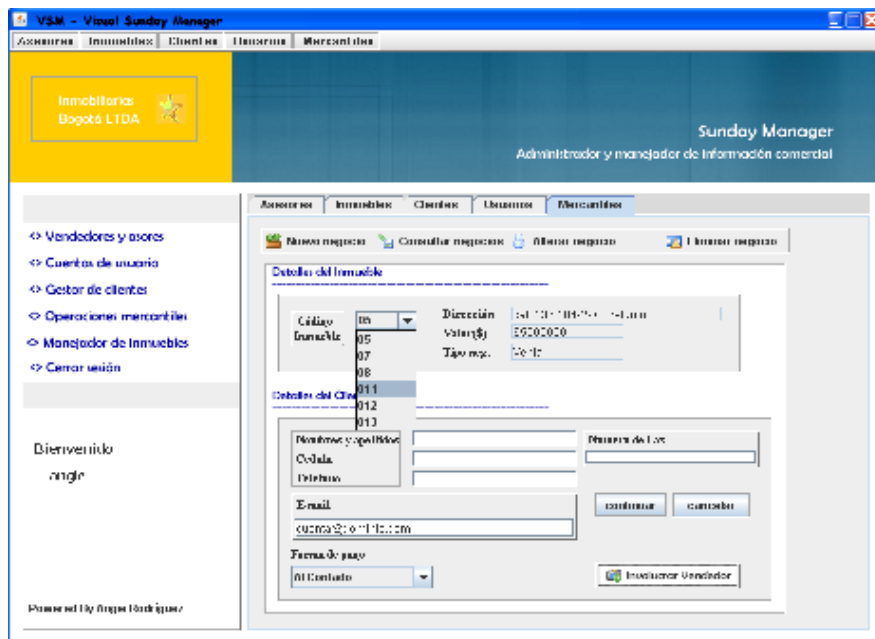


Modificar cuenta

## 5. MÓDULO MERCANTILES

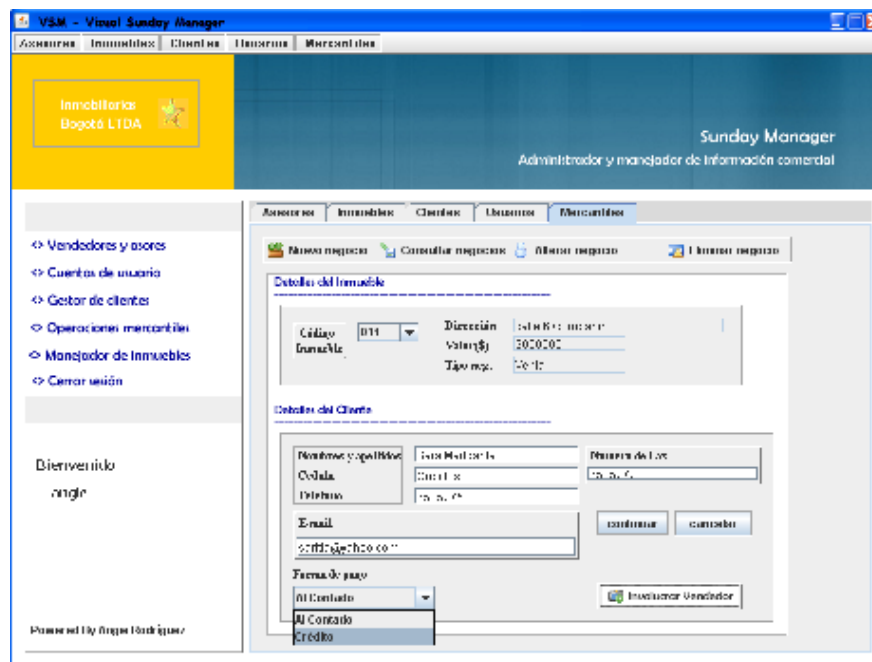
Prácticamente en este punto se centra toda la atención del aplicativo, ya que aquí se registrarán todos los negocios que se logren concretar. La primera opción que se muestra en esta pantalla es “Nuevo negocio”, se debe escoger el inmueble por código y automáticamente va a traer toda la información de ese registro.

Figura 74 Módulo Mercantiles



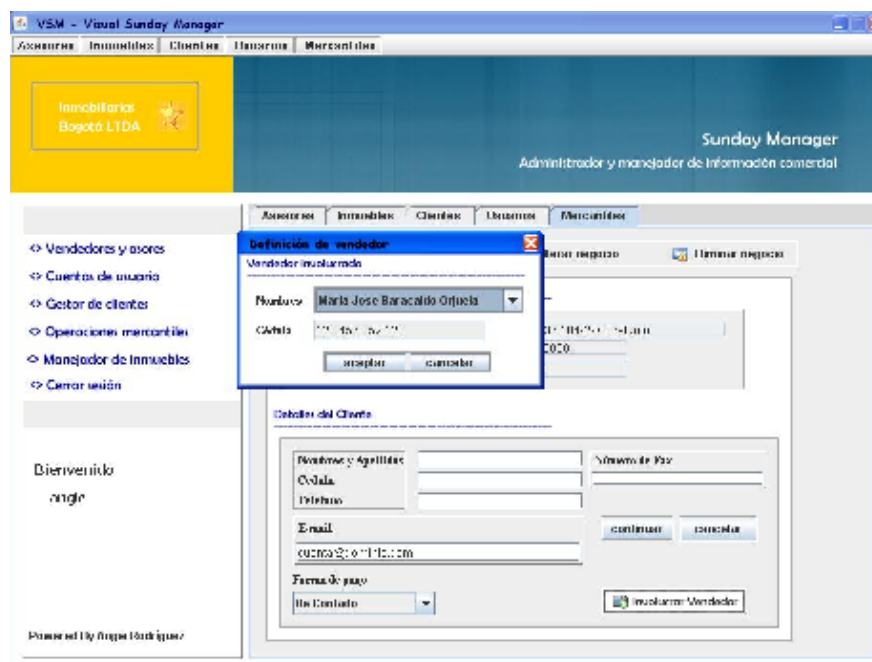
Posteriormente se deben llenar los datos del nuevo cliente, eligiendo la forma de pago, Al contado o a Crédito.

Figura 75 Nuevo Cliente



Luego hay una opción denominada “Involucrar Vendedor”, es necesario elegir el asesor que hizo posible cerrar el negocio, este se selecciona por el nombre y automáticamente trae la cédula.

Figura 76 Adicionar Asesor



La siguiente opción es para consultar los negocios que ya fueron cerrados.

Figura 77 Consulta de Negocios Cerrados



Aquí se puede generar otro reporte, en donde se muestra el inmueble, el cliente publicador, el cliente comprador y el asesor.

Figura 78 Reporte Clientes - Comprador - Asesor

		Negocio Registrado Código : M0006 Apartamento en el sector de Las Américas			
<b>Caracterización del inmueble (esquema externo)</b>					
Dirección	Barrio	Tipo Inmueble	Valor (\$)	Estrato	Area del inmueble
Camera 92a #12-96a	Las Américas	Apartamento	12000000	5	6 M <sup>2</sup>
<b>Caracterización del inmueble (esquema interno)</b>					
Total cuartos	Total Baños	Número de garajes	Tipo de gas	Tipo comedor	Nm de pisos
4	2	2	propano	sala-comedor	5
<b>Recurso gráfico del inmueble</b>					
		<b>Generales</b> 2 Garajes y Depósito, 9 parqueos de visitantes. Pisos en madera maciza comedor y hall y alfombra acústica en sala, estudio y alcobas. Mesones en granito en baños y cocina.			
<b>Datos del cliente involucrado</b>					
<b>Nombre Cliente</b>	Andrés Hernandez				
<b>Número Teléfono</b>	6844575				
<b>Número Fax</b>	2365989				
<b>Email</b>	Andrus@yahoo.com				
<b>Asesor(a) involucrado</b>					
<b>Nombre Vendedor</b>	Javier Alexander Arias Romero				
<b>Celular</b>	3168596001				
<b>Forma de pago</b>	Crédito				
<b>Detalles del publicador</b>					
<b>Nombre propietario</b>	Christian Ernesto Mejia				
<b>Número Teléfono</b>	31678984587				
<b>Número Fax</b>	96325845				
<b>Email</b>	emesto859@gmail.com				
<b>Tipo de negocio</b>	Venta				
					Reporte generado en la fecha 2009-5-14

Para modificar un negocio, se pulsa el botón alterar negocio y se generará un listado con la información existente en la base de datos. Allí se selecciona el negocio a modificar y se realizan los cambios necesarios.

Figura 79 Selección Modificar Negocio

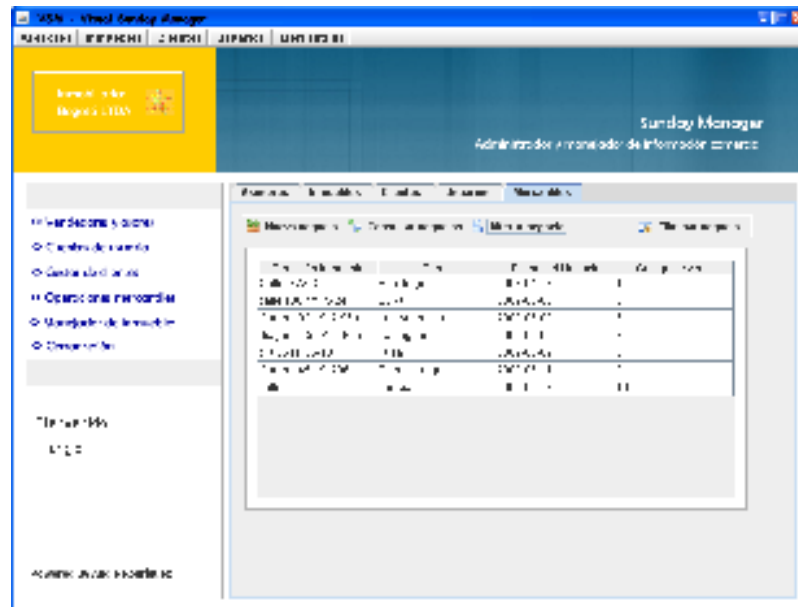
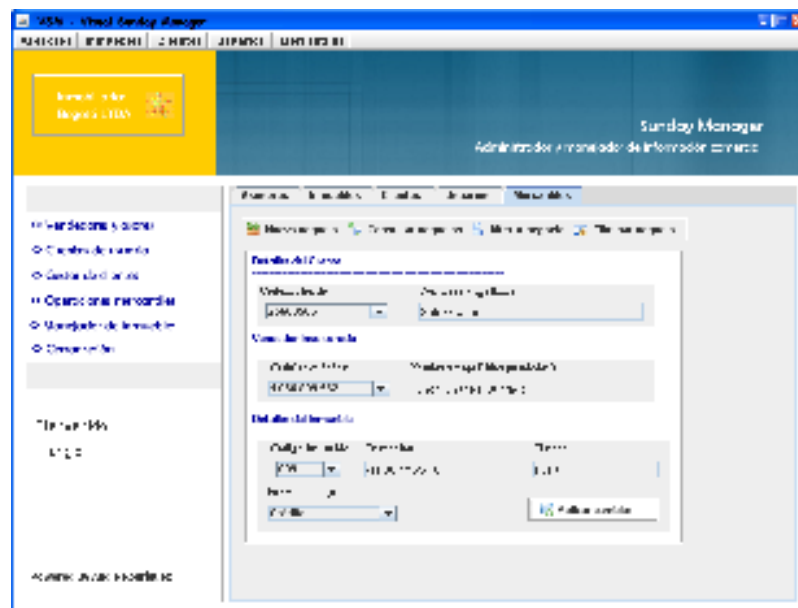


Figura 80 Modificar Negocio



La última opción que se visualiza en la pantalla, es Eliminar negocio.

Figura 81 Eliminar Negocio

