Makita

Jonathan Ramirez Sandoval

Juan Pablo Viscarra Acosta

Versión 1.0

Cochabamba, 18 de Septiembre de 2018

Registro Histórico de Cambios y Revisiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 03/septiembre/2018 | 0.1 | Presentación del modelo de datos | Juan Pablo Viscarra  Jonathan Ramirez | 03/septiembre/2018 |
| 04/septiembre/2018 | 0.1 | Presentación de los Objetivos | Juan Pablo Viscarra  Jonathan Ramirez | 04/septiembre/2018 |
| 05/septiembre/2018 | 0.1 | Presentación del Diagrama de Clases | Juan Pablo Viscarra  Jonathan Ramirez | 05/septiembre/2018 |
| 11/septiembre/2018 | 0.1 | Modificación del modelo de datos | Juan Pablo Viscarra  Jonathan Ramirez | 11/septiembre/2018 |
| 12/septiembre/2018 | 0.2 | Presentación del Primer Prototipo | Juan Pablo Viscarra  Jonathan Ramirez | 12/septiembre/2018 |
| 12/septiembre/2018 | 0.3 | Creación de la base de datos | Juan Pablo Viscarra  Jonathan Ramirez | 12/septiembre/2018 |
| 14/septiembre/2018 | 0.4 | Creación de la lógica para la funcionalidad de la aplicación, creación de la conexión con mysqul | Juan Pablo Viscarra  Jonathan Ramirez | 14/septiembre/2018 |
| 17/septiembre/2018 | 0.5 | Arreglo de error producido al desplegar los mensajes | Juan Pablo Viscarra  Jonathan Ramirez | 17/septiembre/2018 |
| 18/septiembre/2018 | 1.0 | Presentación del Proyecto Final | Juan Pablo Viscarra  Jonathan Ramirez | 18/septiembre/2018 |

Tabla de Contenidos

[ Introducción 4](#_Toc525005963)

[ Análisis de Requerimientos 4](#_Toc525005964)

[ Objetivo General 4](#_Toc525005965)

[ Objetivos Específicos 4](#_Toc525005966)

[ Límites y Alcances 5](#_Toc525005967)

[ Marco Teórico 6](#_Toc525005968)

[ Diagramas de Casos de Uso 7](#_Toc525005969)

[ Modelo de Datos 7](#_Toc525005970)

[ Diagrama de Clases 8](#_Toc525005971)

[ Pruebas y Validación 9](#_Toc525005972)

[ Conclusiones 9](#_Toc525005973)

[ Recomendaciones 10](#_Toc525005974)

[ Bibliografía 10](#_Toc525005975)

Informe Final del Proyecto

# Introducción

Gracias a los avances tecnológicos obtenidos actualmente, la optimización de procesos dentro de cualquier empresa adquiere un rol de gran relevancia, con la finalidad de tener los registros de cualquier actividad guardados en un sistema.

Razón por la cual la empresa *Makita*, desea desarrollar un sistema que le permita optimizar tanto la compra de muebles a sus proveedores, como la venta de los mismos a sus clientes. Con la finalidad de tener los registros almacenados en una base de datos.

# Análisis de Requerimientos

El usuario espera que el sistema realice las operaciones de registro de sus procesos centrales, que se describen a continuación:

* + **Requerimiento Registrar la compra de un mueble**
* Desea registrar todos los datos necesarios para la compra de un mueble, como ser: La fecha de la compra, el empleado que la registró, un detalle de los muebles comprados al proveedor que contenga la cantidad, los precios y el tipo de mueble.
  + **Requerimiento Registrar la venta de un mueble**
* Desea registrar todos los datos necesarios para la venta de un mueble, como ser: La fecha de la venta, el empleado que la registró, un detalle de los muebles vendidos al cliente, que contenga la cantidad, los precios y el tipo de mueble.

# Objetivo General

Desarrollar un sistema para la empresa *Makita*  que le permita registrar la compra y venta de muebles, con la finalidad de optimizar dichos procesos de registro.

# Objetivos Específicos

* + **Registrar a sus Empleados**
* Registrar los datos de un Empleado
* Restricciones o condiciones que se deban verificar
* Condiciones que pueden llevar a un error
* Almacenar los datos en la base de datos
* Mostrar a los empleados registrados en el sistema
  + **Registrar a sus Proveedores**
* Registrar los datos de un proveedor
* Restricciones o condiciones que se deban verificar
* Condiciones que pueden llevar a un error
* Almacenar los datos en la base de datos
* Mostrar a los proveedores registrados en el sistema
  + **Registrar a sus Clientes**
* Registrar los datos de un cliente
* Restricciones o condiciones que se deban verificar
* Condiciones que pueden llevar a un error
* Almacenar los datos en la base de datos
* Mostrar a los clientes registrados en el sistema
  + **Registrar la venta de uno o varios mueble(s) a un cliente**
* Registrar los datos de la venta
* Registrar los datos del detalle de la venta
* Restricciones o condiciones que se deban verificar
* Condiciones que pueden llevar a un error
* Almacenar los datos en la base de datos
* Mostrar las ventas registradas
  + **Registrar la compra de uno o varios mueble(s) a un proveedor**
* Registrar los datos de la compra
* Registrar los datos del detalle de la compra
* Restricciones o condiciones que se deban verificar
* Condiciones que pueden llevar a un error
* Almacenar los datos en la base de datos
* Mostrar las compras registradas

**4.6 Mostrar los tipos de muebles**

* La Empresa ya tiene definidos sus tipos de muebles y no pueden ser modificadoss, por lo cual sólo se visualizarán.

# Límites y Alcances

Para el desarrollo del sistema se establecen los siguientes límites y alcances:

* + **Límites**
* No se identificaron límites de gran magnitud de la aplicación al momento de realizarla, sin embargo, el factor tiempo sería uno de los factores principales para poder realizar este trabajo con todo lo que se esperaría.
* La aplicación solo está hecha para poder administrar una sola tienda (no incluye sucursales).
  + **Alcances**
* La aplicación tendrá el alcance de realizar el registro de datos de clientes, empresas proveedoras, muebles y sus características

# Marco Teórico

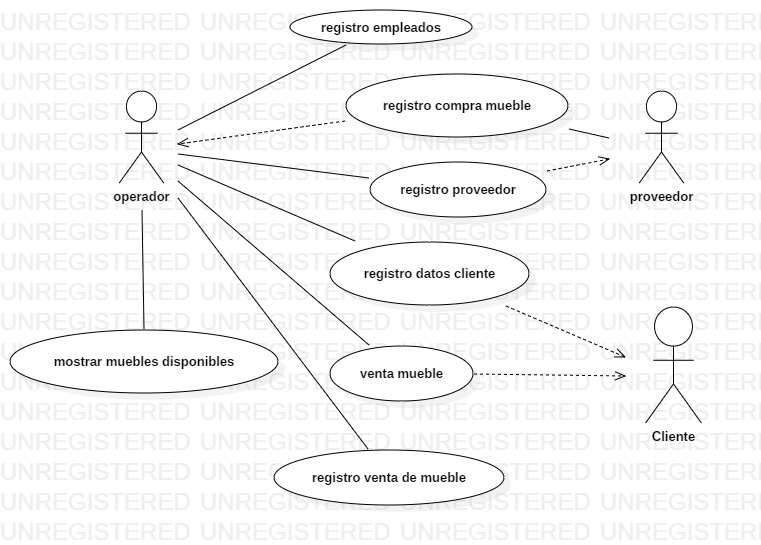
Los conocimientos para dar inicio al proyecto fueron principalmente el libro, los apuntes en clases, ejemplos que a medida del tiempo y curso fuimos realizando, las diapositivas del docente sirvieron de mucho.

Base de datos: Es un [sistema](http://www.monografias.com/trabajos11/teosis/teosis.shtml) que almacena [datos](http://www.monografias.com/trabajos11/basda/basda.shtml) que están relacionados. Es un repositorio en donde guardamos [información](http://www.monografias.com/trabajos7/sisinf/sisinf.shtml) integrada que podemos almacenar y recuperar. Un conjunto de información almacenada en [memoria](http://www.monografias.com/trabajos13/memor/memor.shtml) auxiliar que permite acceso directo y un conjunto de [programas](http://www.monografias.com/Computacion/Programacion/) que manipulan esos datos.

MARIADB: es un [sistema de gestión de bases de datos](https://es.wikipedia.org/wiki/Sistema_de_gesti%C3%B3n_de_bases_de_datos). Está desarrollado por [Michael (Monty) Widenius](https://es.wikipedia.org/wiki/Michael_Widenius) (fundador de [MySQL](https://es.wikipedia.org/wiki/MySQL" \o "MySQL)) y la comunidad de desarrolladores de [software libre](https://es.wikipedia.org/wiki/Software_libre). Tiene una alta compatibilidad con MySQL ya que posee las mismas órdenes, interfaces, APIs y bibliotecas, siendo su objetivo poder cambiar un servidor por otro directamente.

Modelo de Datos: Un modelo de base de datos es un tipo de modelo de datos que determina la estructura lógica de una base de datos y de manera fundamental determina el modo de almacenar, organizar y manipular los datos. Entre los modelos lógicos comunes para bases de datos se encuentran.

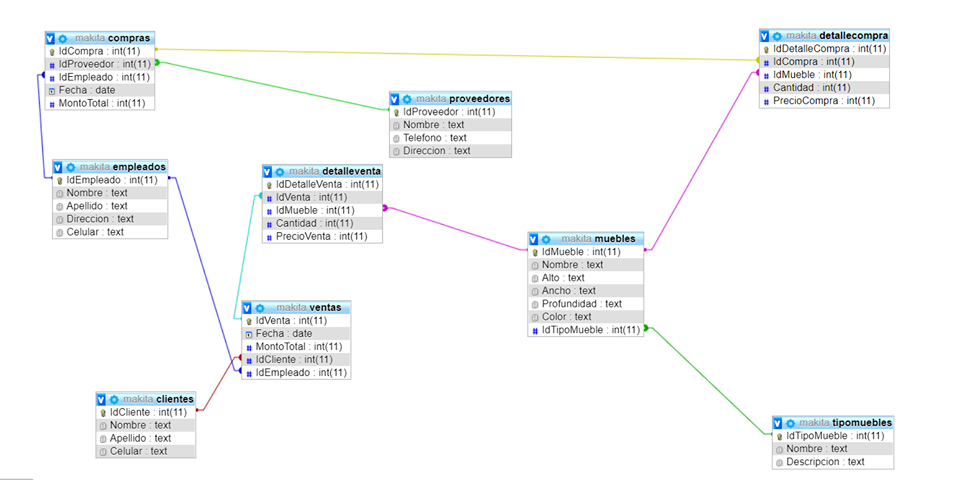
# Diagramas de Casos de Uso



El operador es capaz de manejar todo en cuanto a la aplicación se refiere, puede modificar eliminar, actualizar, listar y acceder a los muebles y sus descripciones.

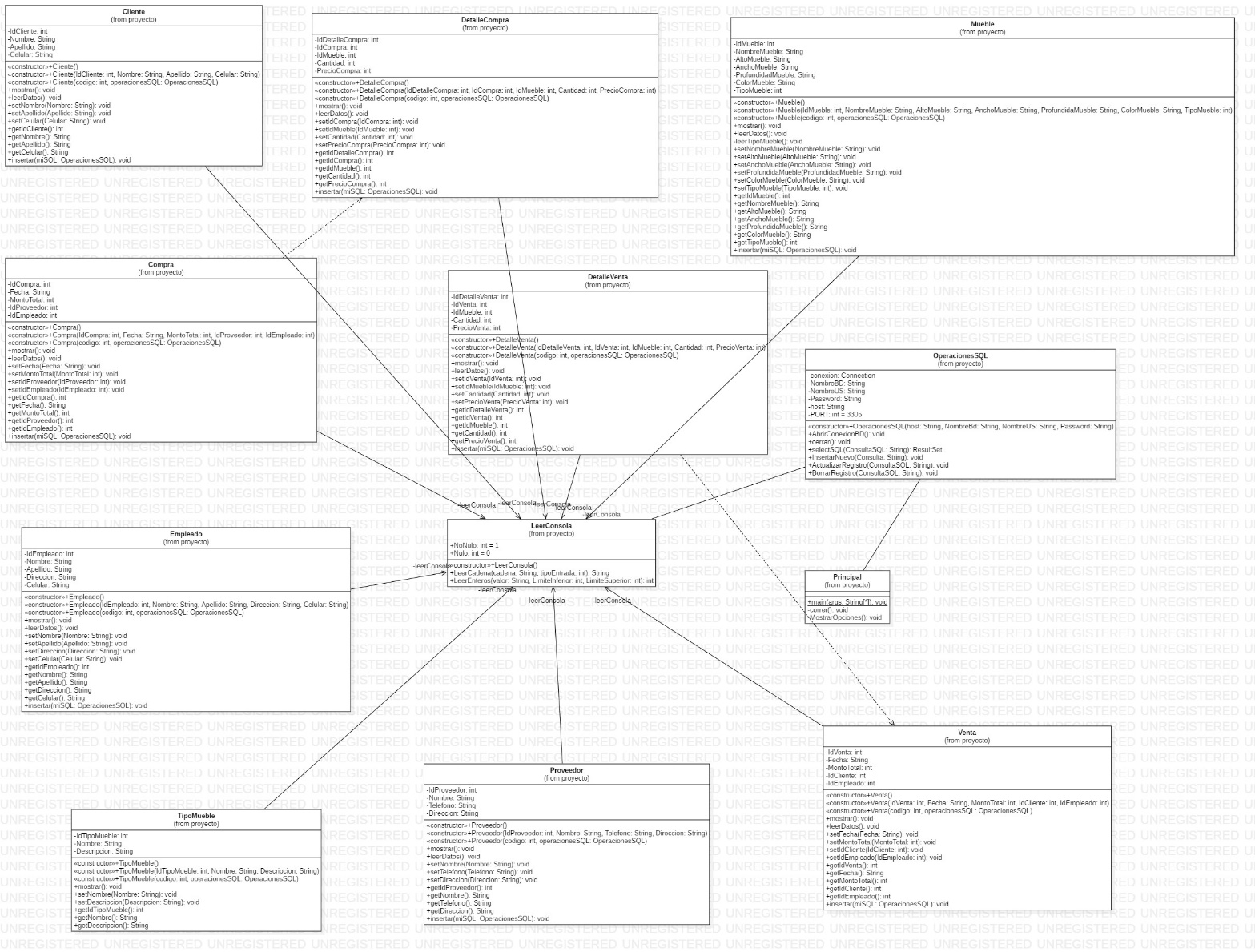
# Modelo de Datos

(Siguiente hoja)



# Diagrama de Clases

(Siguiente hoja)



# Pruebas y Validación

Se tuvo problemas al conectar la base de datos con el código de programación en JAVA, apareció un error al ingresar letras en vez de números en el carnet de identidad del cliente al momento de realizar la venta de los muebles, finalmente se pudo solucionarlo.

El principal cambio y problema que se tuvo fue el ordenamiento de las funciones de relación entre las clases del sistema de base de datos.

Se realizó la prueba de funcionalidad de la aplicación varias veces modificando el código y la base de datos hasta que no aparecieron más errores.

# Conclusiones

Logramos crear una base de datos fácilmente accesible e interactiva con el administrador.

Esta base tiene clases que permiten ser modificadas, listadas, eliminadas e insertadas, tal como se avanzó en la materia.

Hubo algunos percances con la implementación de algunos menús y submenús, debido al desorden pero luego se pudo controlar.

Se logró el objetivo de poder crear una base de datos aplicando los conocimientos obtenidos en el módulo y también revisando ayuda en el internet.

Se pudo solucionar todos los problemas que aparecieron durante la programación del sistema, como por ejemplo, la relación de base de datos con el código (MYSQL conector).

# Recomendaciones

Se podría mejorar la base de datos introduciendo una función que permita añadir muebles desde la interfaz de la aplicación, así como proveedores y empleados nuevos que ingresen a trabajar a la mueblería.

Para la mejora de esta base de datos podría pensarse en la implementación de sucursales y facturas.

# Bibliografía

-Texto “Programación para ingeniería II” – Marcel Barrero, Universidad Privada Boliviana

# -http://codejavu.blogspot.com/2013/06/ejemplo-conectando-java-con-mysql.html

# -http://solrive.com/blog/como-conectar-java-con-una-base-de-datos-mysql-conectar-java-y-mysql/

-<https://www.youtube.com/watch?v=Rn1v_SN3Wac>

-<https://www.youtube.com/watch?v=41bnHN-NnjI>