

A dark blue vertical bar is on the left. A blue arrow points right from it, containing the date.

2-11-2017

Grupo de ingeniería de software

EUGENIO HERNÁNDEZ
CARLOS QUINTERO
FERNANDO RHENALS
SANTIAGO VIDAL

Several thin, curved lines in shades of blue and grey sweep upwards from the bottom left corner.

[PROYECTO D&G - UNIVERSIDAD DE CORDOBA]

El proyecto se realizó y se llevó a cabo durante el segundo periodo semestral del año 2017 en la Facultad de Ciencias e Ingenierías, del programa de Ingeniería de Sistemas de la Universidad de Córdoba al Instructor, Señor Oswaldo Velez Langs en el curso de INGENIERIA DE SOFTWARE.

Tabla de contenido

1.	DESCRIPCION DEL PROBLEMA O REQUERIMIENTOS	2
2.	IDENTIFICANDO ENTIDADES PARTICIPANTES DEL SISTEMA	3
3.	TABLA DE ENTIDADES, ATRIBUTOS Y RELACIONES	4
4.	DIAGRAMA DE ENTIDAD/RELACION PARA LA BASE DE DATOS.	5
5.	DIAGRAMA DE ENTIDAD/RELACION EXTENDIDO PARA LA BASE DE DATOS.	6
6.	VIABILIDAD	6
7.	REPRESENTACION DE LA SITUACION EN EL ENTORNO REAL	7
8.	REQUISITOS FUNCIONALES DEL SISTEMA	7
9.	CASOS DE USOS	8
9.1.	Actor usuario	8
9.2.	Actor administrador de base de datos:.....	9
10.	DIAGRAMAS DE CLASE:	9

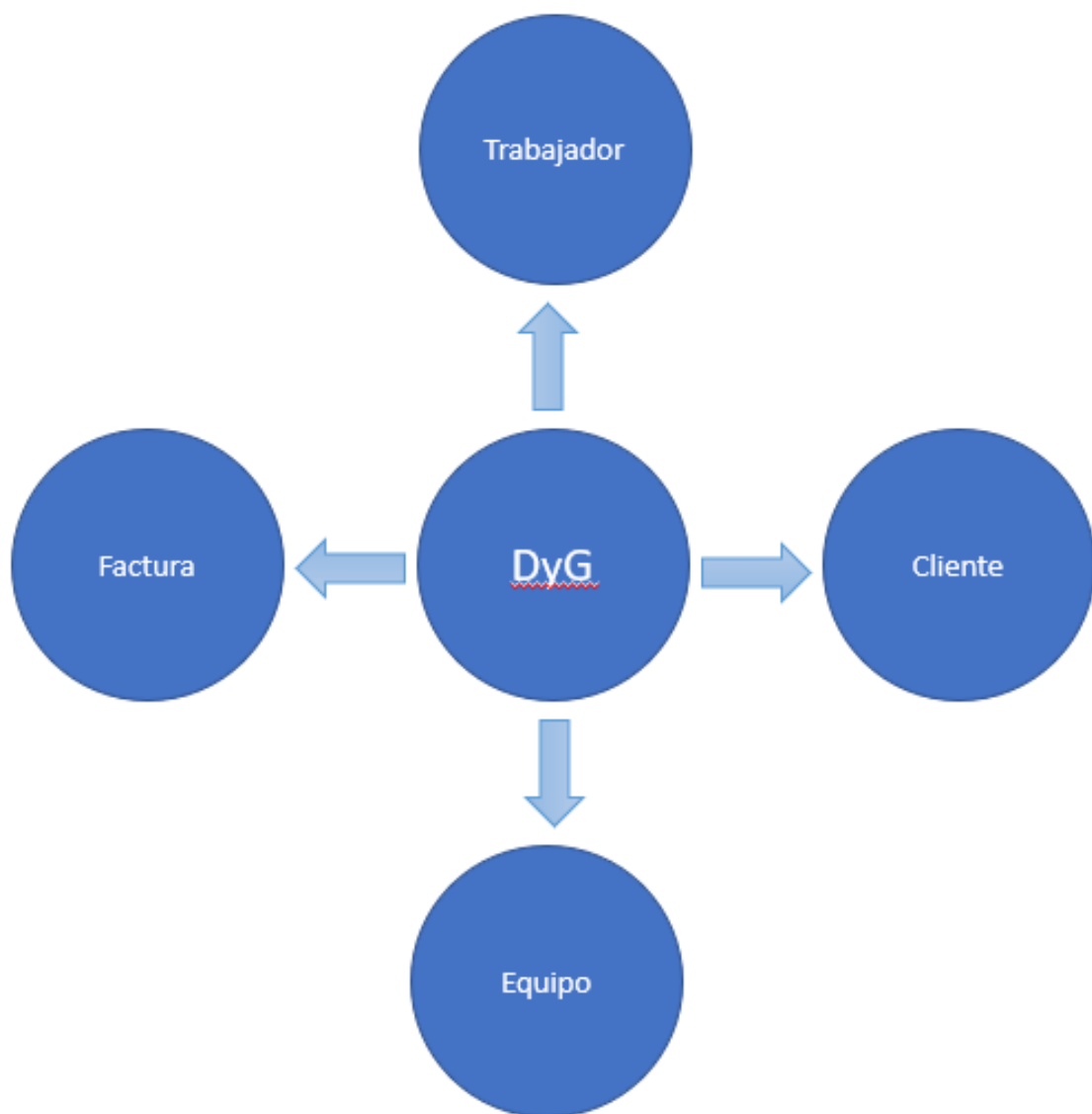
1. DESCRPCION DEL PROBLEMA O REQUERIMIENTOS

Una empresa llamada D&G se dedica a la **venta, alquiler, reparación** y mantenimiento de **equipos** de oficina tales como Impresoras, fotocopadoras, scanner, computadoras, tables y celulares. La empresa no cuenta con un sistema que almacene información de los equipos que se reciben para mantenimiento, diagnostico o reparación

Ellos manejan un recibo a mano que pide lo siguientes datos Pide fecha de ingreso y fecha en el que el equipo está listo para entregar, nombre de **cliente**, identificación, dirección, teléfono

Básicamente el sistema que ellos quieren debe permitir que agregue una nueva reparación, venta o alquiler y en ello debe exigir los datos del cliente, los datos del equipo, lo que tiene el equipo y la fecha de recibido y la fecha en el que se entregara el equipo Y también que tenga una casilla de observaciones en donde se coloca cosas a tener en cuenta del equipo como por ejemplo si el equipo que se recibió tiene algún defecto en las bisagras, o si le falta algunos tornillos etc. y también una casilla accesorias en donde se colocara si el equipo vino con cargador, forros o algo adicional.

2. IDENTIFICANDO ENTIDADES PARTICIPANTES DEL SISTEMA

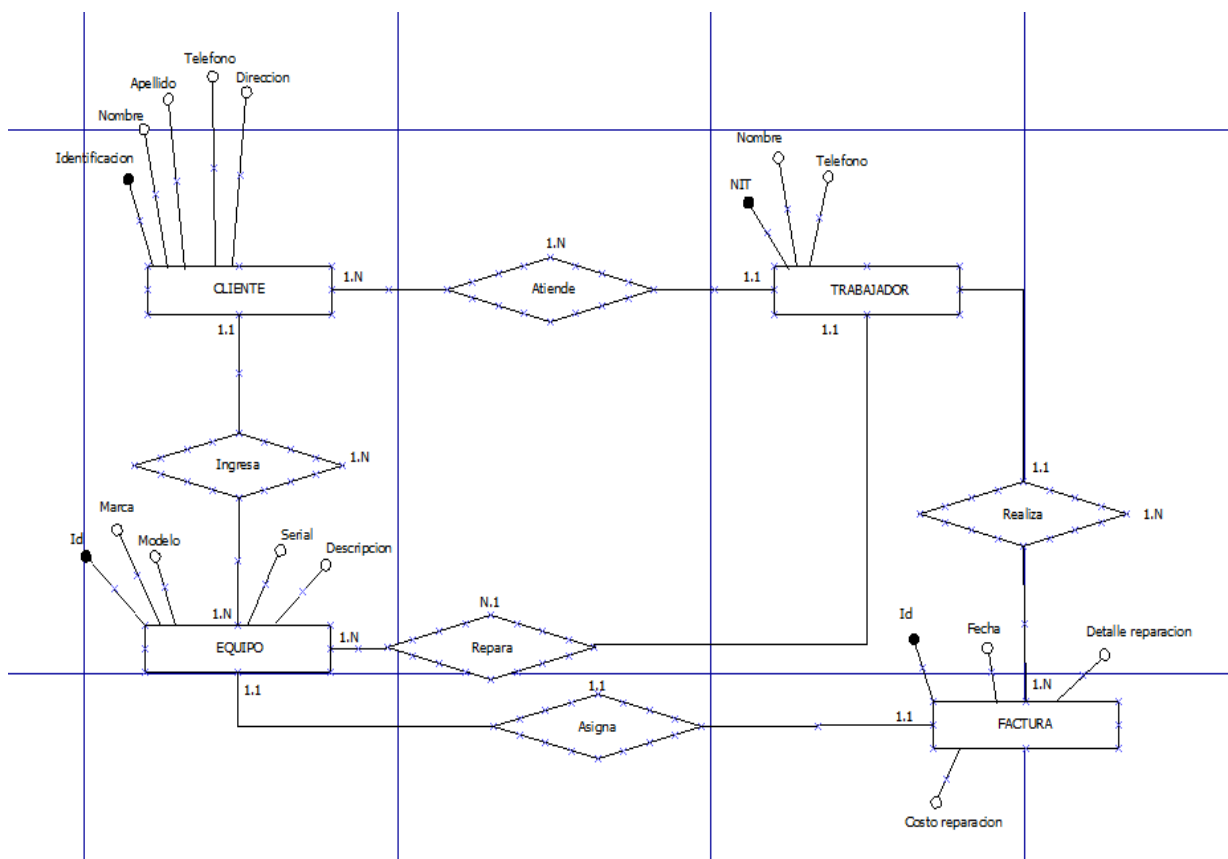


3. TABLA DE ENTIDADES, ATRIBUTOS Y RELACIONES

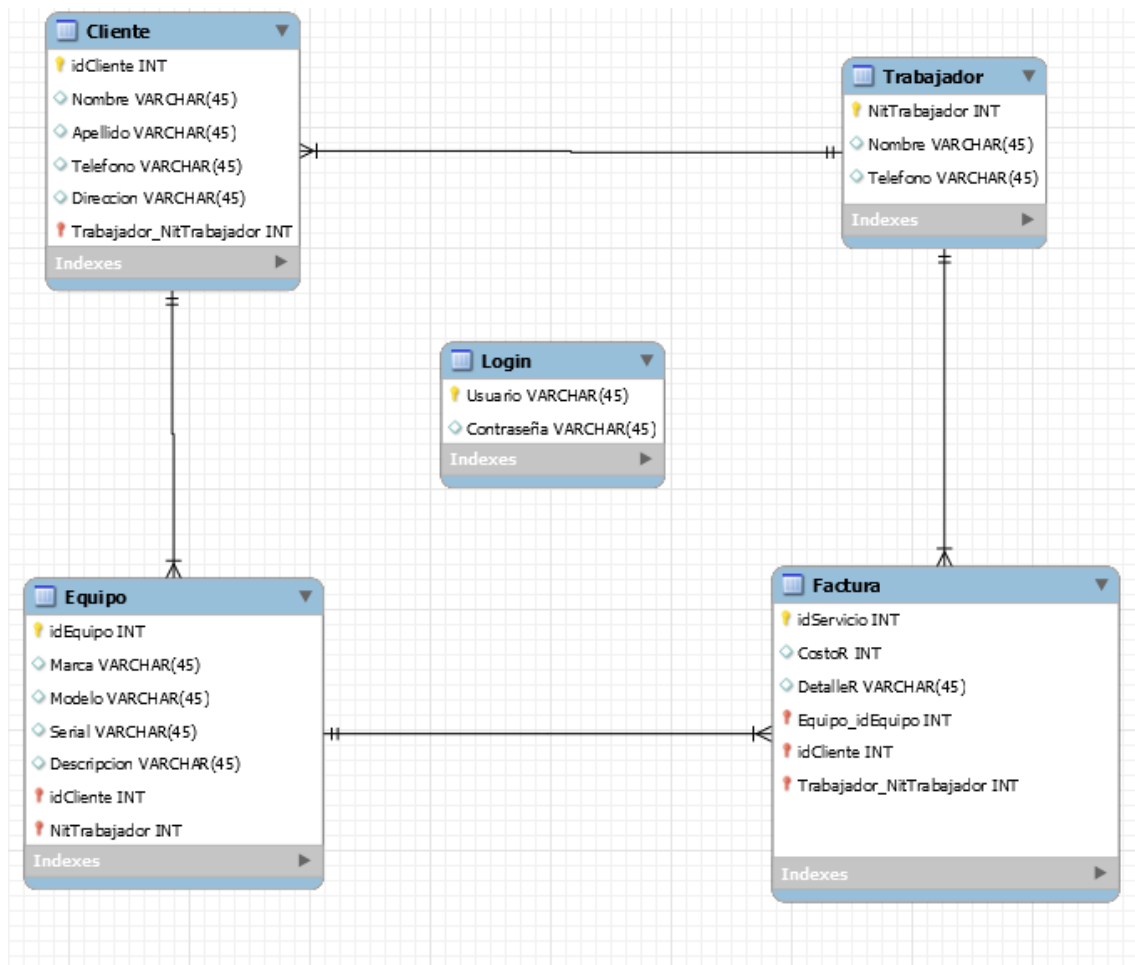
Entidad	Descripción	Atributos y relaciones
Almacén	Representa al almacén	<p>➤ Información del almacén</p> <ul style="list-style-type: none"> • Nombre: GyG • NIT: • Teléfono: • Dirección: <p>[?] Trabajador: Contiene los trabajadores del almacén</p> <p>[*] Cliente: Contiene los clientes del almacén</p>
Cliente	Representa a los clientes que compran en el almacén DyG	<p>➤ Información de cliente</p> <ul style="list-style-type: none"> • Nombre • Apellido • Identificación • Teléfono • Dirección • E-mail <p>[*]Reparación/mantenimiento: El cliente puede solicitar el servicio de reparación y mantenimiento en varias ocasiones.</p>
Trabajador	Representa a un trabajador en el almacén	<p>➤ Información personal</p> <ul style="list-style-type: none"> • Nombre • Apellido • Cargo • NIT • Teléfono <p>[*]Equipo: El técnico repara y hace mantenimiento a varios equipos, y el vendedor alquila y vende equipos.</p>
Factura	Representa a las facturas generadas en el almacén DyG	<p>➤ Información de servicio</p> <ul style="list-style-type: none"> • Id • Fecha • Detalle de reparación • Costo de reparación <p>[1] Equipo: cada factura es relacionada con un único equipo.</p> <p>[*]Trabajador: Uno o varios trabajadores pueden ir asociados a los servicios prestados.</p>

Equipo	Representa a los equipos ingresados por los clientes y los pertenecientes a DG	<p>➤ Informacion del equipo</p> <ul style="list-style-type: none"> • ID • Marca • Modelo • Estado • Descripcion <p>[1] Cliente: Se asocia un cliente al equipo, ya sea este el dueño o esté solicitando un servicio de alquiler.</p> <p>[1]Trabajador: En caso de reparacion o mantenimiento se le asigna un tecnico al equipo</p>

4. DIAGRAMA DE ENTIDAD/RELACION PARA LA BASE DE DATOS.



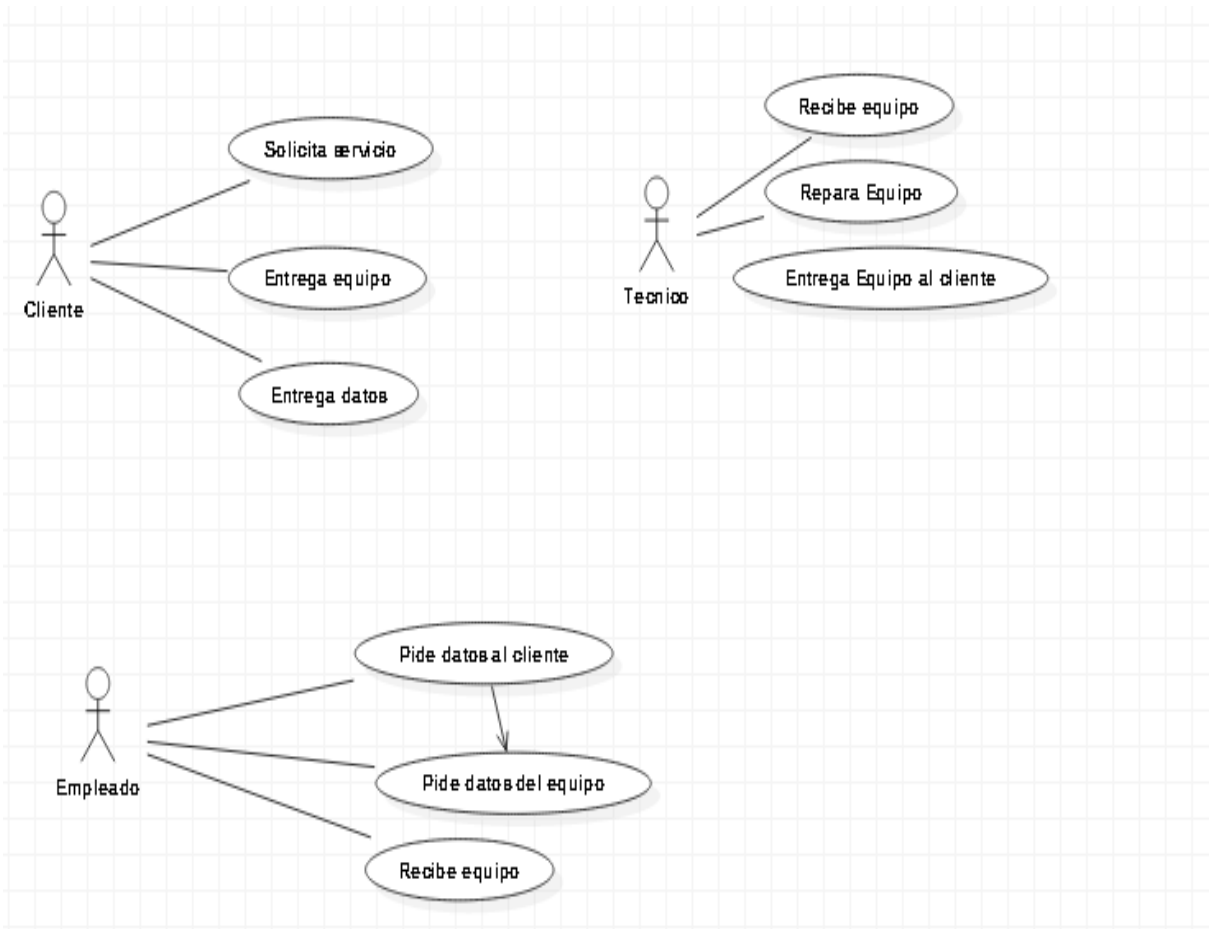
5. DIAGRAMA DE ENTIDAD/RELACION EXTENDIDO PARA LA BASE DE DATOS.



6. VIABILIDAD

Actualmente el almacén no cuenta con un software propio y el que están usando además de ser ilegal, es decir sin licencia y no cuenta con todos los requerimientos. Trabajar con un software sin licencia no expresa mucha confianza al momento del que el cliente haga la reclamación del equipo reparado o alquilado, ya que da la posibilidad de que los datos se pierdan o haya un mal manejo de la información.

7. REPRESENTACION DE LA SITUACION EN EL ENTORNO REAL



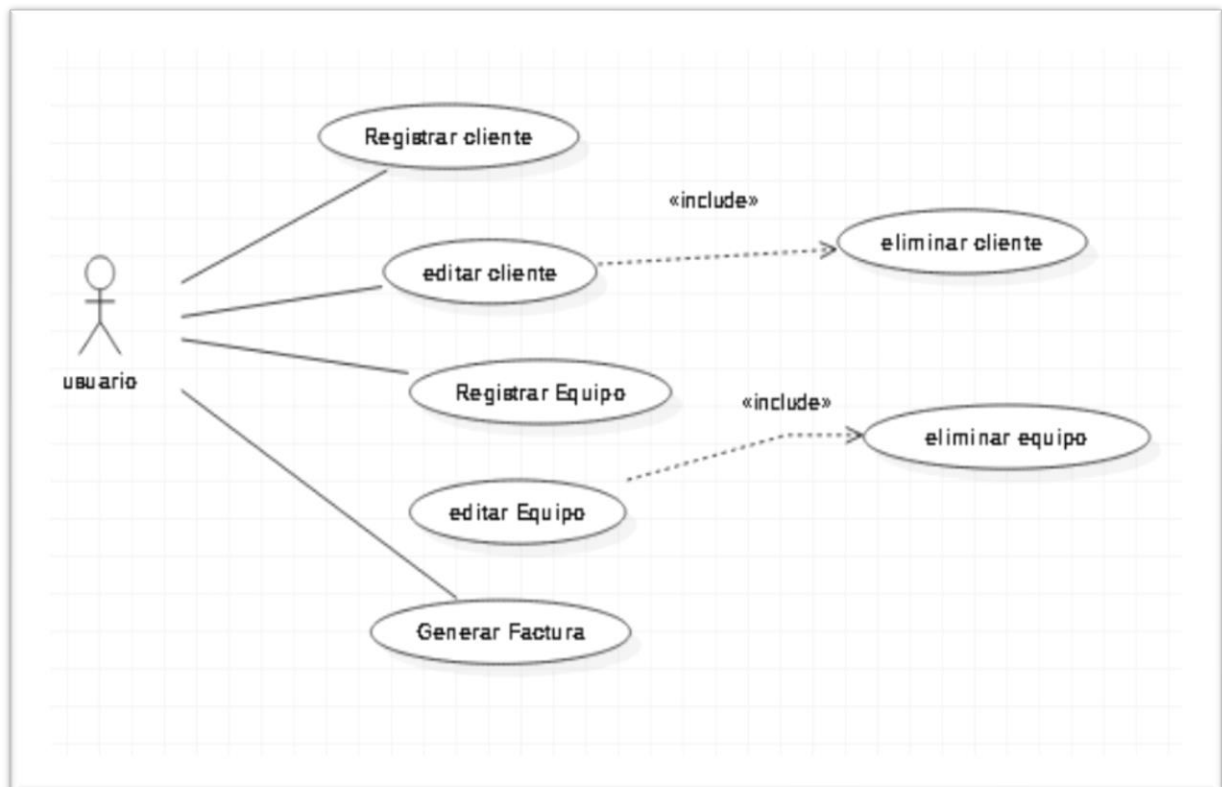
8. REQUISITOS FUNCIONALES DEL SISTEMA

Número	Requerimiento	Descripción	Prioridad
RF1	EL EMPLEADO TENDRÁ QUE INICIAR SESIÓN PARA REGISTRAR, VER, EDITAR O ELIMINAR INFORMACIÓN DE LOS EQUIPOS Y LOS CLIENTES,	Para el uso del sistema se pedirá que ingrese un usuario y una contraseña que solo el administrador del sistema podrá darle al empleado.	4
RF2	EL EMPLEADO TENDRÁ UN CONTROL DE REGISTRO DE EQUIPOS DE LOS CLIENTES	El sistema tendrá información de los datos de los equipos que traen los clientes a la empresa	5

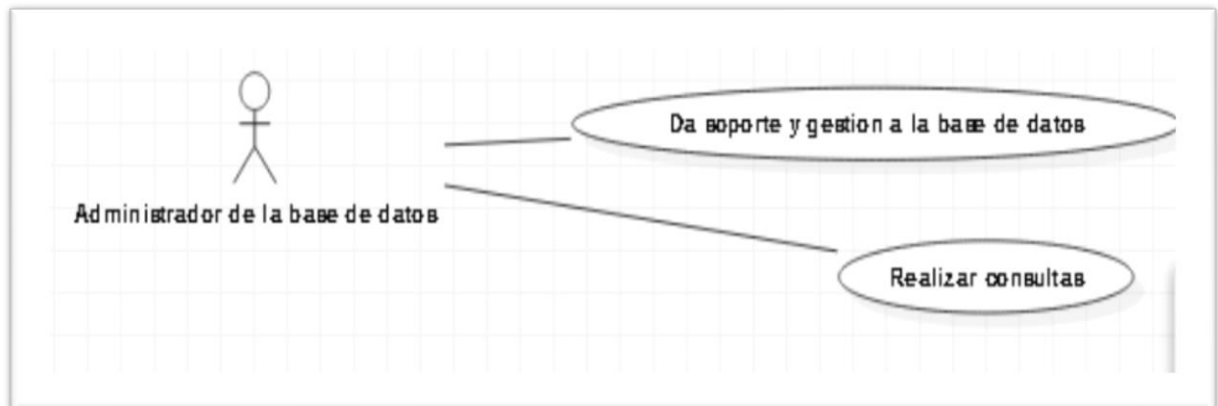
RF3	EL EMPLEADO TENDRÁ UN CONTROL DE REGISTRO DE LOS CLIENTES	El sistema tendrá información de los datos personales de los clientes.	5
RF4	EL EMPLEADO GENERARÁ UNA FACTURA DE RECIBIDO Y LO IMPRIMIRÁ	El sistema tendrá la opción de generar un reporte o un recibo que será impreso y entregado al cliente dando constancia de que el equipo fue recibo por el empleado o técnico.	5
RF5	EL ADMINISTRADOR PODRÁ GESTIONAR EL ACCESO AL SISTEMA.	El sistema tendrá la opción de agregar o borrar cuentas de usuario para poder dar uso al sistema.	3

9. CASOS DE USOS

9.1. Actor usuario

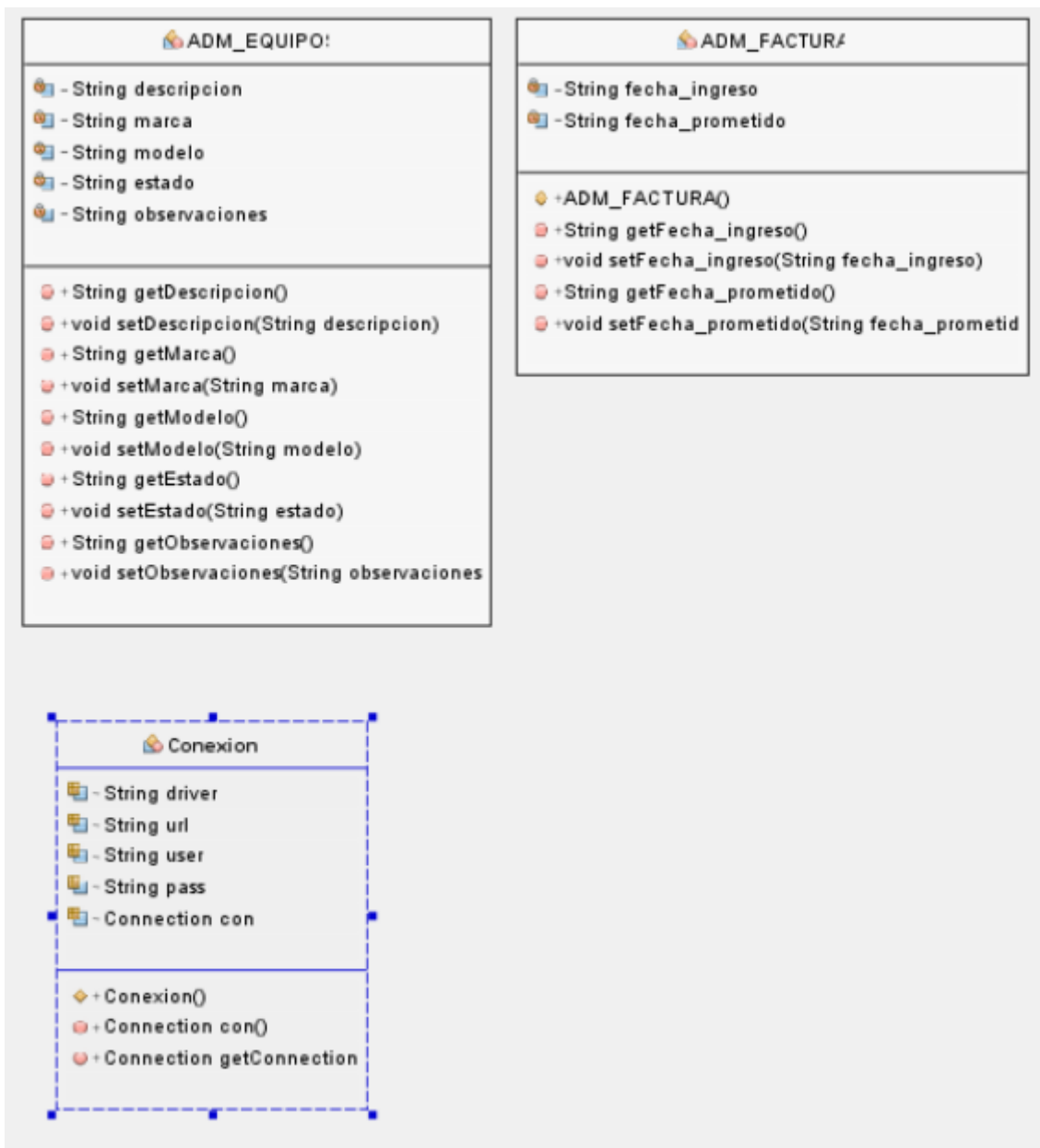


9.2. Actor administrador de base de datos:



10. DIAGRAMAS DE CLASE:





11. SCRIPT SQL:

```

CREATE DATABASE IF NOT EXISTS `bd` /*!40100 DEFAULT CHARACTER SET utf8
*/;
USE `bd`;

-- Volcando estructura para tabla bd.clientes
CREATE TABLE IF NOT EXISTS `clientes` (
  `RUT` varchar(50) NOT NULL,
  `Nombres` varchar(150) DEFAULT NULL,
  `Departamento` varchar(50) DEFAULT NULL,
  `Ciudad` varchar(50) DEFAULT NULL,
  `Direccion` varchar(50) DEFAULT NULL,
  `Telefono` varchar(50) DEFAULT NULL,
  PRIMARY KEY (`RUT`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
  
```

```

-- Volcando datos para la tabla bd.clientes: ~1 rows (aproximadamente)
/*!40000 ALTER TABLE `clientes` DISABLE KEYS */;
INSERT INTO `clientes` (`RUT`, `Nombres`, `Departamento`, `Ciudad`,
`Direccion`, `Telefono`) VALUES
('1067938445', 'FERNANDO RHENALS', 'CORDOBA', 'MONTERIA', 'LA GRANJA',
'3008474121');
/*!40000 ALTER TABLE `clientes` ENABLE KEYS */;

-- Volcando estructura para tabla bd.empleado
CREATE TABLE IF NOT EXISTS `empleado` (
`idEmpleado` int(11) NOT NULL AUTO_INCREMENT,
`em_Nombres` varchar(200) NOT NULL DEFAULT '0',
`em_Telefono` varchar(50) NOT NULL DEFAULT '0',
PRIMARY KEY (`idEmpleado`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8;

-- Volcando datos para la tabla bd.empleado: ~1 rows (aproximadamente)
/*!40000 ALTER TABLE `empleado` DISABLE KEYS */;
INSERT INTO `empleado` (`idEmpleado`, `em_Nombres`, `em_Telefono`)
VALUES
(1, 'kljhlk', '15645');
/*!40000 ALTER TABLE `empleado` ENABLE KEYS */;

-- Volcando estructura para tabla bd.equipo
CREATE TABLE IF NOT EXISTS `equipo` (
`tipo` varchar(50) DEFAULT NULL,
`marca` varchar(50) DEFAULT NULL,
`modelo` varchar(50) DEFAULT NULL,
`serial` varchar(50) NOT NULL,
`observaciones` varchar(1000) DEFAULT NULL,
`cli_rut` varchar(150) NOT NULL,
PRIMARY KEY (`serial`, `cli_rut`),
KEY `FK_cli_rut` (`cli_rut`),
CONSTRAINT `FK_cli_rut` FOREIGN KEY (`cli_rut`) REFERENCES `clientes`
(`RUT`) ON DELETE CASCADE ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- Volcando datos para la tabla bd.equipo: ~2 rows (aproximadamente)
/*!40000 ALTER TABLE `equipo` DISABLE KEYS */;
INSERT INTO `equipo` (`tipo`, `marca`, `modelo`, `serial`,
`observaciones`, `cli_rut`) VALUES
('COMPUTADOR PORTÁTIL', 'KLJ', 'JLKJKL', 'JKLJKL', 'JKLJLKJLK',
'1067938445'),
('COMPUTADOR PORTÁTIL', 'LLL', 'LL', 'LLL', ' LLLL', '1067938445');
/*!40000 ALTER TABLE `equipo` ENABLE KEYS */;

-- Volcando estructura para tabla bd.facturacion
CREATE TABLE IF NOT EXISTS `facturacion` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`cl_RUT` varchar(50) NOT NULL,
`eq_Serial` varchar(50) NOT NULL,
`em_Nombres` varchar(50) DEFAULT NULL,
`Fecha_Ingreso` varchar(50) DEFAULT NULL,
`txtFecha_Salida` varchar(50) DEFAULT NULL,
PRIMARY KEY (`id`, `cl_RUT`, `eq_Serial`),
KEY `FK_facturacion_clientes` (`cl_RUT`),
KEY `FK_facturacion_equipos` (`eq_Serial`),
CONSTRAINT `FK_facturacion_clientes` FOREIGN KEY (`cl_RUT`) REFERENCES
`clientes` (`RUT`) ON DELETE CASCADE ON UPDATE CASCADE,
CONSTRAINT `FK_facturacion_equipos` FOREIGN KEY (`eq_Serial`)
REFERENCES `equipo` (`serial`) ON DELETE CASCADE ON UPDATE CASCADE

```

```

) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8;

-- Volcando datos para la tabla bd.facturacion: ~2 rows
(aproximadamente)
/*!40000 ALTER TABLE `facturacion` DISABLE KEYS */;
INSERT INTO `facturacion` (`id`, `cl_RUT`, `eq_Serial`, `em_Nombres`,
`Fecha_Ingreso`, `txtFecha_Salida`) VALUES
(1, '1067938445', 'JKLJKL', 'kljhlk', '02/12/2017', '06/12/2017'),
(2, '1067938445', 'LLL', 'kljhlk', '02/12/2017', ' / / ');
/*!40000 ALTER TABLE `facturacion` ENABLE KEYS */;

-- Volcando estructura para tabla bd.login
CREATE TABLE IF NOT EXISTS `login` (
`id` int(100) NOT NULL AUTO_INCREMENT,
`usuario` varchar(50) NOT NULL DEFAULT '0',
`contraseña` varchar(50) NOT NULL DEFAULT '0',
PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=3 DEFAULT CHARSET=utf8;

-- Volcando datos para la tabla bd.login: ~2 rows (aproximadamente)
/*!40000 ALTER TABLE `login` DISABLE KEYS */;
INSERT INTO `login` (`id`, `usuario`, `contraseña`) VALUES
(1, 'admin', 'admin'),
(2, 'user', 'user');
/*!40000 ALTER TABLE `login` ENABLE KEYS */;

-- Volcando estructura para tabla bd.test
CREATE TABLE IF NOT EXISTS `test` (
`cm_codigo` int(11) NOT NULL AUTO_INCREMENT,
`cm_nombre` varchar(50) NOT NULL DEFAULT '0',
PRIMARY KEY (`cm_codigo`)
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8;

-- Volcando datos para la tabla bd.test: ~4 rows (aproximadamente)
/*!40000 ALTER TABLE `test` DISABLE KEYS */;
INSERT INTO `test` (`cm_codigo`, `cm_nombre`) VALUES
(1, 'Cargando BD'),
(2, 'Cargando Datos'),
(3, 'Iniciando interfaz de usuario'),
(4, 'Espere');
/*!40000 ALTER TABLE `test` ENABLE KEYS */;

-- Volcando estructura para tabla bd.tipo_equipo
CREATE TABLE IF NOT EXISTS `tipo_equipo` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`tipo` varchar(50) NOT NULL DEFAULT '0',
PRIMARY KEY (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=utf8;

-- Volcando datos para la tabla bd.tipo_equipo: ~11 rows
(aproximadamente)
/*!40000 ALTER TABLE `tipo_equipo` DISABLE KEYS */;
INSERT INTO `tipo_equipo` (`id`, `tipo`) VALUES
(1, 'COMPUTADOR PORTÁTIL'),
(2, 'COMPUTADOR ESCRITORIO'),
(3, 'TABLET'),
(4, 'CELULAR / SMARTPHONE'),
(5, 'CPU / TORRE'),
(6, 'PANTALLA / MONITOR'),
(7, 'UPS'),
(8, 'ESTABILIZADOR'),

```

```
(9, 'IMPRESORA'),  
(10, 'FOTOCOPIADORA'),  
(11, 'OTRO');  
/*!40000 ALTER TABLE `tipo_equipo` ENABLE KEYS */;  
  
/*!40101 SET SQL_MODE=IFNULL(@OLD_SQL_MODE, '') */;  
/*!40014 SET FOREIGN_KEY_CHECKS=IF(@OLD_FOREIGN_KEY_CHECKS IS NULL, 1,  
@OLD_FOREIGN_KEY_CHECKS) */;  
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
```