Solicitante:
   I.T.S. – Instituto Tecnológico Superior Arias - Balparda

Nombre de Fantasía del Proyecto: BINDEV
Grupo de Clase: 3° IF
Turno: Nocturno
Materia: Base de Datos 2
Nombre de los Integrantes del Grupo: Alvarez Nicolas,
Armand-ugon Ignacio, Estefan German, Rivera Fabricio.

   Fecha de entrega: 12/ 09 /2020

Instituto Tecnológico Superior Arias Balparda
Gral. Flores 3591 esq.   Bvar. José Batlle y Ordoñez - Montevideo

Objetivo

El objetivo del siguiente documento es dejar en clara evidencia lo trabajado en la materia Base de datos 2 por parte del equipo con el foin de cumplir con los puntos estipulados para esta segunda entrega

Alcance

Este documento tiene como objetivo llegar a personas vinculadas de alguna manera con el sistema de base de datos, en el mismo se detallan conceptos de DER - Normalizacion - Modelos físicos - Sentencias DDl y DML - Estudio de permisos entre otros.

Índice

1. INTRODUCCIÓN

El siguiente documento de análisis fue creado por la empresa Binary Development (desde ahora BinDev) con el fin de aportar valor al área de DATOS sobre el proyecto a desarrollar.

En el mismo se detallan: Entidades - Atributos - Claves - Tipos de datos y Aspectos a considerar fundamentales para el control y acceso a datos.
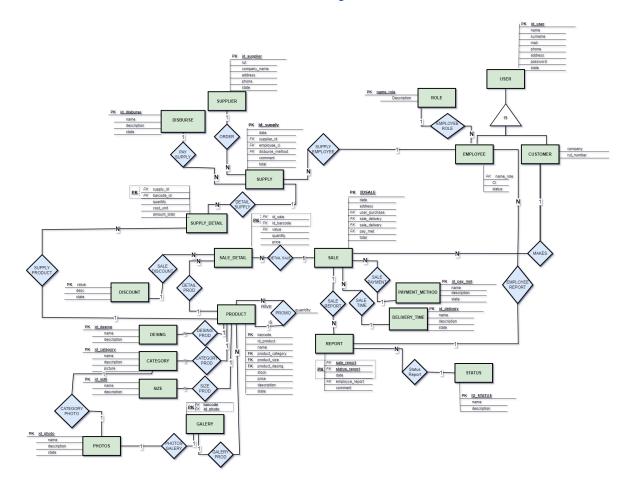
2. DER SITUACIÓN PLANTEADA

Se corrigieron algunos aspectos mal ingresados en el primer documento. Pensamos en cambiar el idioma a Inglés para darle mayor alcance y poder ayudar al equipo de desarrollo frente a las terminologías que se manejan en dicho sector

Enlace: DER FINAL- situación planteada

3. NORMALIZACION

Estamos en condiciones de afirmar y garantizar que nuestro diseño de solución para el desarrollo de la base de datos está normalizado hasta la 3FN cumpliendo este las siguientes condiciones:

1ra Forma Normal:
➢ No existen dos filas idénticas en los registros
➢ Todos los Atributos son Atómicos

2da Forma Normal:
➢ Cumplir la 1ra Forma Normal
➢ Todos los Atributos (no primos) dependen únicamente de la clave primaria PK

3ra Forma Normal:
➢ Cumplir la 2da Forma Normal
➢ No existen dependencias funcionales transitivas entre atributos no primos

4. MODELO FÍSICO

4.1 DDL y DML para creacion de BD

DDL y DML: [ArchivoAdjunto](#)

```sql
    -- MySQL Workbench Forward Engineering


    SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
    SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS,
FOREIGN_KEY_CHECKS=0;
    SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_Z
ERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
    -- -----------------------------------------------------------
    -- Schema bindev
    -- -----------------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `bindev` DEFAULT CHARACTER SET utf8 ;
    USE `bindev` ;


    -- -----------------------------------------------------------
    -- Table `bindev`.`category`
    -- -----------------------------------------------------------
    CREATE TABLE IF NOT EXISTS `bindev`.`category` (
      `id_category` INT NOT NULL,
      `name` VARCHAR(100) NOT NULL,
      `description` VARCHAR(500) NOT NULL,
      `picture` int NOT NULL,
      PRIMARY KEY (`id_category`),
      constraint `category_UNIQUE` UNIQUE  (`name`),
       CONSTRAINT `FK_picture_category`
        FOREIGN KEY (`picture`)
        REFERENCES `bindev`.`photos` (`id_photo`)
        ON DELETE cascade
        ON UPDATE cascade)
    ENGINE = InnoDB;


    -- -----------------------------------------------------------
    -- Table `bindev`.`design`
    -- -----------------------------------------------------------
```

```sql
CREATE TABLE IF NOT EXISTS `bindev`.`design` (
  `id_desing` INT NOT NULL,
  `name` VARCHAR(150) NOT NULL,
  `description` VARCHAR(500) NOT NULL,
  PRIMARY KEY (`id_desing`),
  constraint `desing_UNIQUE` UNIQUE (`name`))
ENGINE = InnoDB;


-- -----------------------------------------------------------
-- Table `bindev`.`size`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`size` (
  `id_size` INT NOT NULL,
  `name` VARCHAR(150) NOT NULL,
  `description` VARCHAR(500) NOT NULL,
  PRIMARY KEY (`id_size`),
  constraint `size_UNIQUE` UNIQUE (`name` ))
ENGINE = InnoDB;


-- -----------------------------------------------------------
-- Table `bindev`.`disburse`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`disburse` (
  `id_disburse` INT NOT NULL,
  `name` VARCHAR(150) NOT NULL,
  `description` VARCHAR(500) NOT NULL,
  `state` TINYINT default 1,
  PRIMARY KEY (`id_disburse`),
    constraint `CH_state_disburse` CHECK (`state`<=1 and
`state`>=0),
    constraint `disburse_UNIQUE` UNIQUE(`name`))
ENGINE = InnoDB;


-- -----------------------------------------------------------
-- Table `bindev`.`supplier`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`supplier` (
```

```sql
    `id_supplier` INT AUTO_INCREMENT NOT NULL,
    `rut` varchar(12) NOT NULL,
    `company_name` VARCHAR(150) NOT NULL,
    `address` VARCHAR(500) NOT NULL,
    `phone` VARCHAR(200) NOT NULL,
    `state` TINYINT default 1,
    PRIMARY KEY (`id_supplier`),
        constraint `CH_state_supplier` CHECK (`state`<=1 and
`state`>=0),
    constraint `UN_rut` UNIQUE  (`rut`))
ENGINE = InnoDB;




-- -----------------------------------------------------------
-- Table `bindev`.`role`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`role` (
    `name_role` VARCHAR(100) NOT NULL,
    `description` VARCHAR(500) NOT NULL,
    PRIMARY KEY (`name_role`))
ENGINE = InnoDB;




-- -----------------------------------------------------------
-- Table `bindev`.`user`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`user` (
    `id_user` INT NOT NULL AUTO_INCREMENT,
    `email` VARCHAR(200) NOT NULL,
    `name` VARCHAR(150) NOT NULL,
    `surname` VARCHAR(150) NOT NULL,
    `address` VARCHAR(500) NULL,
    `phone` VARCHAR(50) NULL,
    `password` VARCHAR(200) NOT NULL,
    `state` TINYINT default 1,
    PRIMARY KEY (`id_user`),
    constraint `CH_state_user` CHECK (`state`<=1 and `state`>=0),
    constraint `email_UNIQUE` UNIQUE (`email`))
ENGINE = InnoDB;
```

```sql
ALTER Table user
AUTO_INCREMENT=5000;
-- -----------------------------------------------------------
-- Table `bindev`.`status`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`status` (
  `id_status` INT NOT NULL,
  `name` VARCHAR(150) NOT NULL,
  `description` VARCHAR(500) NOT NULL,
  PRIMARY KEY (`id_status`),
  constraint `UN_status` UNIQUE (`name`))
ENGINE = InnoDB;


-- -----------------------------------------------------------
-- Table `bindev`.`delivery_time`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`delivery_time` (
  `id_delivery` INT NOT NULL,
  `name` VARCHAR(150) NOT NULL,
  `description` VARCHAR(500) NOT NULL,
  `state` TINYINT default 1,
  PRIMARY KEY (`id_delivery`),
    constraint `CH_state_delivery` CHECK (`state`<=1 and
`state`>=0))
ENGINE = InnoDB;


-- -----------------------------------------------------------
-- Table `bindev`.`payment_method`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`payment_method` (
  `id_pay_meth` INT NOT NULL,
  `name` VARCHAR(150) NOT NULL,
  `description` VARCHAR(500) NOT NULL,
  `state` TINYINT default 1,
  PRIMARY KEY (`id_pay_meth`),
    constraint `CH_state_PayMet` CHECK (`state`<=1 and
`state`>=0),
```

```sql
    constraint `UN_pay_meth` UNIQUE (`name`))
  ENGINE = InnoDB;



  -- -----------------------------------------------------------
  -- Table `bindev`.`photos`
  -- -----------------------------------------------------------
  CREATE TABLE IF NOT EXISTS `bindev`.`photos` (
    `id_photo` INT NOT NULL,
    `name` VARCHAR(150) NOT NULL,
    `description` VARCHAR(500) NOT NULL,
    `state` TINYINT default 1,
    PRIMARY KEY (`id_photo`),
  constraint `CH_state` CHECK (`state`<=1 and `state`>=0))
  ENGINE = InnoDB;



  -- -----------------------------------------------------------
  -- Table `bindev`.`discount`
  -- -----------------------------------------------------------
  CREATE TABLE IF NOT EXISTS `bindev`.`discount` (
    `value` DECIMAL(4, 2) NOT NULL,
    `description` VARCHAR(500) NOT NULL,
    `state` TINYINT default 1,
    PRIMARY KEY (`value`),
      constraint `CH_state_Discount` CHECK (`state`<=1 and
`state`>=0))
  ENGINE = InnoDB;



  -- -----------------------------------------------------------
  -- Table `bindev`.`employee`
  -- -----------------------------------------------------------
  CREATE TABLE IF NOT EXISTS `bindev`.`employee` (
    `ci` INT NOT NULL,
    `employee_user` INT NOT NULL,
    `employee_role` VARCHAR(100) NOT NULL,
    `state` TINYINT default 1,
    PRIMARY KEY (`employee_user`),
```

```sql
    constraint `UN_ci` unique (`ci`),
      constraint `CH_state_employee` CHECK (`state`<=1 and
`state`>=0),
    CONSTRAINT `FK_employee_role`
      FOREIGN KEY (`employee_role`)
      REFERENCES `bindev`.`role` (`name_role`)
      ON DELETE RESTRICT
      ON UPDATE RESTRICT,
    CONSTRAINT `FK_employee_user`
      FOREIGN KEY (`employee_user`)
      REFERENCES `bindev`.`user` (`id_user`)
      ON DELETE RESTRICT
      ON UPDATE RESTRICT)
  ENGINE = InnoDB;



  -- -----------------------------------------------------------
  -- Table `bindev`.`customer`
  -- -----------------------------------------------------------
  CREATE TABLE IF NOT EXISTS `bindev`.`customer` (
    `customer_user` INT NOT NULL,
    `company_name` VARCHAR(300) NULL,
    `rut_nr` varchar(12) NULL,
    PRIMARY KEY (`customer_user`),
    constraint `UN_rut` UNIQUE (`rut_nr`),
    CONSTRAINT `FK_user_id`
      FOREIGN KEY (`customer_user`)
      REFERENCES `bindev`.`user` (`id_user`)
      ON DELETE RESTRICT
      ON UPDATE RESTRICT)
  ENGINE = InnoDB;



  -- -----------------------------------------------------------
  -- Table `bindev`.`supply`
  -- -----------------------------------------------------------
  CREATE TABLE IF NOT EXISTS `bindev`.`supply` (
    `id_supply` INT NOT NULL AUTO_INCREMENT,
    `date` DATETIME NOT NULL,
```

```sql
    `supplier_id` INT NOT NULL,
    `employee_ci` INT NOT NULL,
    `disburse_method` INT NOT NULL,
    `comment` VARCHAR(500) NULL,
    `total` DECIMAL(10,2) NOT NULL default 0,
  PRIMARY KEY (`id_supply`),
  CONSTRAINT `FK_supply_supplier`
    FOREIGN KEY (`supplier_id`)
    REFERENCES `bindev`.`supplier` (`id_supplier`)
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
  CONSTRAINT `FK_supply_employee`
    FOREIGN KEY (`employee_ci`)
    REFERENCES `bindev`.`employee` (`ci`)
    ON DELETE RESTRICT
    ON UPDATE cascade,
  CONSTRAINT `FK_supply_disburse`
    FOREIGN KEY (`disburse_method`)
    REFERENCES `bindev`.`disburse` (`id_disburse`)
    ON DELETE RESTRICT
    ON UPDATE cascade)
ENGINE = InnoDB;

ALTER TABLE supply
AUTO_INCREMENT = 300000;
-- -----------------------------------------------------------
-- Table `bindev`.`product`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`product` (
  `barcode` INT NOT NULL AUTO_INCREMENT,
  `id_product` INT NOT NULL,
  `name` VARCHAR(200) NOT NULL,
  `product_category` INT NOT NULL,
  `product_desing` INT NOT NULL,
  `product_size` INT NOT NULL,
  `stock` INT NOT NULL,
  `price` DECIMAL(10,2) NOT NULL,
  `description` VARCHAR(500) NOT NULL,
  `state` TINYINT default 1,
```

```sql
      PRIMARY KEY (`barcode`),
        constraint `CH_state_product` CHECK (`state`<=1 and
`state`>=0),
          constraint `UN_product` UNIQUE (`id_product`,
`product_category`, `product_desing`, `product_size`),
      CONSTRAINT `FK_category_product`
        FOREIGN KEY (`product_category`)
        REFERENCES `bindev`.`category` (`id_category`)
        ON DELETE RESTRICT
        ON UPDATE CASCADE,
      CONSTRAINT `FK_desing_product`
        FOREIGN KEY (`product_desing`)
        REFERENCES `bindev`.`design` (`id_desing`)
        ON DELETE RESTRICT
        ON UPDATE CASCADE,
      CONSTRAINT `FK_size_product`
        FOREIGN KEY (`product_size`)
        REFERENCES `bindev`.`size` (`id_size`)
        ON DELETE RESTRICT
        ON UPDATE CASCADE)
    ENGINE = InnoDB;
    ALTER TABLE product
    AUTO_INCREMENT = 12312300;


    -- -------------------------------------------------------
    -- Table `bindev`.`galery`
    -- -------------------------------------------------------
    CREATE TABLE IF NOT EXISTS `bindev`.`galery` (
      `product_galery` INT NOT NULL,
      `photo_galery` INT NOT NULL,
      PRIMARY KEY (`product_galery`, `photo_galery`),
      CONSTRAINT `FK_product_galery`
        FOREIGN KEY (`product_galery`)
        REFERENCES `bindev`.`product` (`barcode`)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
      CONSTRAINT `FK_photo_product`
        FOREIGN KEY (`photo_galery`)
        REFERENCES `bindev`.`photos` (`id_photo`)
```

```sql
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB;



-- -----------------------------------------------------------
-- Table `bindev`.`supply_detail`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`supply_detail` (
  `supply_id` INT NOT NULL,
  `barcode_id` INT NOT NULL,
  `quantity` INT NOT NULL,
  `cost_unit` DECIMAL(10,2) NOT NULL,
  `amount_total` DECIMAL(10,2) NOT NULL,
  PRIMARY KEY (`supply_id`, `barcode_id`),
CONSTRAINT `FK_product_reference`
    FOREIGN KEY (`barcode_id`)
    REFERENCES `bindev`.`product` (`barcode`)
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
  CONSTRAINT `FK_supply_reference`
    FOREIGN KEY (`supply_id`)
    REFERENCES `bindev`.`supply` (`id_supply`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT)
ENGINE = InnoDB;



-- -----------------------------------------------------------
-- Table `bindev`.`sale`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`sale` (
  `id_sale` INT NOT NULL AUTO_INCREMENT,
  `date` DATETIME NOT NULL ,
  `address` VARCHAR(500) NOT NULL,
  `user_purchase` INT NOT NULL,
  `sale_delivery` INT NOT NULL,
  `pay_met` INT NOT NULL,
  `total` DECIMAL(10,2) NOT NULL default 0,
```

```sql
    PRIMARY KEY (`id_sale`),
    CONSTRAINT `FK_customer_user`
      FOREIGN KEY (`user_purchase`)
      REFERENCES `bindev`.`customer` (`customer_user`)
      ON DELETE RESTRICT
      ON UPDATE RESTRICT,
    CONSTRAINT `FK_delivery_sale`
      FOREIGN KEY (`sale_delivery`)
      REFERENCES `bindev`.`delivery_time` (`id_delivery`)
      ON DELETE RESTRICT
      ON UPDATE CASCADE,
    CONSTRAINT `FK_pay_sale`
      FOREIGN KEY (`pay_met`)
      REFERENCES `bindev`.`payment_method` (`id_pay_meth`)
      ON DELETE RESTRICT
      ON UPDATE RESTRICT)
  ENGINE = InnoDB;
  ALTER TABLE sale
  AUTO_INCREMENT=700000;


  -- -----------------------------------------------------------
  -- Table `bindev`.`report`
  -- -----------------------------------------------------------
  CREATE TABLE IF NOT EXISTS `bindev`.`report` (
    `sale_report` INT NOT NULL,
    `status_report` INT NOT NULL,
    `employee_report` INT NOT NULL,
    `date` DATETIME NOT NULL ,
    `comment` VARCHAR(500) NULL,
    PRIMARY KEY (`sale_report`, `status_report`),
        CONSTRAINT    `UN_report_sale`    UNIQUE    (`sale_report`,
`status_report`, `date`),
    CONSTRAINT `FK_sale_report`
      FOREIGN KEY (`sale_report`)
      REFERENCES `bindev`.`sale` (`id_sale`)
      ON DELETE RESTRICT
      ON UPDATE RESTRICT,
    CONSTRAINT `FK_status_report`
      FOREIGN KEY (`status_report`)
```

```sql
    REFERENCES `bindev`.`status` (`id_status`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT,
  CONSTRAINT `FK_employye_report`
    FOREIGN KEY (`employee_report`)
    REFERENCES `bindev`.`employee` (`ci`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT)
ENGINE = InnoDB;


-- -----------------------------------------------------------
-- Table `bindev`.`sale_detail`
-- -----------------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`sale_detail` (
  `sale_id` INT NOT NULL,
  `product_sale` INT NOT NULL,
  `quantity` INT NOT NULL,
  `sale_discount` DECIMAL(4,2) NULL,
  `total` DECIMAL(10,2) NOT NULL,
  PRIMARY KEY (`sale_id`, `product_sale`),
  CONSTRAINT `FK_id_sale`
    FOREIGN KEY (`sale_id`)
    REFERENCES `bindev`.`sale` (`id_sale`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT,
  CONSTRAINT `FK_prod_sale`
    FOREIGN KEY (`product_sale`)
    REFERENCES `bindev`.`product` (`barcode`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT,
  CONSTRAINT `FK_value_discount`
    FOREIGN KEY (`sale_discount`)
    REFERENCES `bindev`.`discount` (`value`)
    ON DELETE RESTRICT
    ON UPDATE RESTRICT)
ENGINE = InnoDB;
```

```sql
-- -----------------------------------------------------
-- Table `bindev`.`promo`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `bindev`.`promo` (
  `is_product` INT NOT NULL,
  `have_product` INT NOT NULL,
  `quantity` INT NOT NULL,
  PRIMARY KEY (`is_product`, `have_product`),
  CONSTRAINT `FK_is_product`
    FOREIGN KEY (`is_product`)
    REFERENCES `bindev`.`product` (`barcode`)
    ON DELETE RESTRICT
    ON UPDATE CASCADE,
  CONSTRAINT `FK_have_product`
    FOREIGN KEY (`have_product`)
    REFERENCES `bindev`.`product` (`barcode`)
    ON DELETE RESTRICT
    ON UPDATE CASCADE)
ENGINE = InnoDB;


DROP TRIGGER IF EXISTS `bindev`.`sale_detail_VALIDATION`;
DELIMITER $$
USE `bindev`$$
CREATE DEFINER = CURRENT_USER TRIGGER `bindev`.`sale_detail_VALIDATION` BEFORE INSERT ON `sale_detail` FOR EACH ROW
BEGIN
declare stockTemp int;
declare totalParc decimal(10,2);
declare totalDesc decimal(10,2);
set stockTemp = (select stock from product where new.product_sale = barcode)-new.quantity;
if (stockTemp >=0) then
update product set stock = stockTemp where barcode = new.product_sale;
set totalParc = new.quantity * (select price from product where new.product_sale = barcode);
set totalDesc = totalParc * new.sale_discount;
set new.total = totalParc - totalDesc;
```

```sql
    update sale set total = ((total + new.total) * 1.22) where
id_sale = new.sale_id;
    else
    SIGNAL SQLSTATE '45000' SET message_text = 'NO HAY STOCK
SUFICIENTE PARA REALIZAR LA OPERACIÓN';
    END if;
    END$$
    DELIMITER ;

    DROP                TRIGGER              IF              EXISTS
`bindev`.`supply_detail_AMOUNT_TOTAL_AUTO`;
    DELIMITER $$
    USE `bindev`$$
    CREATE       DEFINER       =       CURRENT_USER       TRIGGER
`bindev`.`supply_detail_AMOUNT_TOTAL_AUTO`    BEFORE    INSERT    ON
`supply_detail` FOR EACH ROW
    BEGIN
    set new.amount_total = new.cost_unit * new.quantity;
    update supply set total = ((total + new.amount_total) * 1.22)
where id_supply = new.supply_id;
    END$$
    DELIMITER ;

    DROP TRIGGER IF EXISTS `bindev`.`supply_detail_Actualiza_stock`;
    DELIMITER $$
    USE `bindev`$$
    CREATE       DEFINER       =       CURRENT_USER       TRIGGER
`bindev`.`supply_detail_Actualiza_stock`    AFTER    INSERT    ON
`supply_detail` FOR EACH ROW
    BEGIN
    update product set stock = stock + new.quantity where barcode =
new.barcode_id;
    END$$
    DELIMITER ;

    DROP TRIGGER IF EXISTS `bindev`.`Auto_insertado`;
    DELIMITER $$
    USE `bindev`$$
```

```
      CREATE DEFINER = CURRENT_USER TRIGGER `bindev`.`Auto_insertado`
AFTER INSERT ON `user` FOR EACH ROW
      BEGIN
      insert customer set customer_user = new.id_user;
      END$$
      DELIMITER ;


      SET SQL_MODE=@OLD_SQL_MODE;
      SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
      SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

INSERT Básicos para probar la base

```
INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`,
`password`) VALUES ('correo1@gmail.com', 'Nicolas', 'Alvarez', 'direccion 1',
'098776554', 'password1');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`,
`password`) VALUES ('correo2@gmail.com', 'Rodrigo', 'Gimenez', 'direccion2',
'267482', 'password2');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`,
`password`) VALUES ('correo3@gmail.com', 'Pablo ', 'Gimenez', 'direccion 2',
'098098098', 'password3');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`,
`password`) VALUES ('correo4@gmail.com', 'ana maria', 'libertad', 'direcc genérica',
'457', 'password4');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`,
`password`) VALUES ('correo5@gmail.com', 'emilia ', 'de guzman', 'direccion salto al
fondo', '25443', 'password5');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`,
`password`) VALUES ('correo6@gmail.com', 'clara', 'hiliarse', 'Al fondo que hay
lugar', '45988864', 'password6');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`,
`password`) VALUES ('correo7@seguridadcorporal.com', 'patricia', 'gutiérrez',
'dirección otra', '2334513', 'password7');
```

```sql
INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('correo8@seguridadcorporal.com', 'lucas', 'rodriguez', '18 de julio 2377', '22334455', 'password8');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('correo9@seguridadcorporal.com', 'esteban ', 'quito', 'no me acuerdo', '09828848', 'password9');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('correo10@seguridadcorporal.com', 'maximilian', 'fija', 'rio negro entre j carlos rodriguez y marcelino gutierrez', '56778883', 'password10');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('Lcarambula@seguridadcorporal.com', 'Leonardo', 'Carámbula', 'Programación Web', '22114433', 'lcarambula');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('Lacuna@seguridadcorporal.com', 'Laura', 'Acuna', 'Diseño Web', '098465773 098887766', 'lacuna');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('Cbarrios@seguridadcorporal.com', 'Christian', 'Barrios', 'Sist Ope y Gest de Proy', '098098098', 'cbarrios');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('Agonzalez@seguridadcorporal.com', 'Ana', 'Gonzalez', 'ADA', '123123321', 'agonzalez');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('Nhernandez@seguridadcorporal.com', 'Nora', 'Hernandez', 'UTU', '123123123', 'nhernandez');

INSERT INTO `bindev`.`user` (`email`, `name`, `surname`, `address`, `phone`, `password`) VALUES ('Bgates@seguridadcorporal.com', 'Bill', 'Gates', 'Estados Unidos', '1234567890000', 'bgates');

INSERT INTO `bindev`.`role` (`name_role`, `description`) VALUES ('VENDEDOR', 'EL VENDEDOR');

INSERT INTO `bindev`.`role` (`name_role`, `description`) VALUES ('COMPRADOR', 'EL COMPRADOR');

INSERT INTO `bindev`.`role` (`name_role`, `description`) VALUES ('JEFE', 'EL JEFE');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
```

```sql
('911', 'Remera roja', 'Foto de una remera');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('912', 'Remera azul', 'Foto de una remera');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('913', 'Remera verde', 'Foto de una remera');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('914', 'Remera amarilla', 'Foto de una remera');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('915', 'Remera bordo', 'Foto de una remera');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('916', 'Remera rosa', 'Foto de una remera');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('917', 'Remera blanca', 'Foto de una remera');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('918', 'Remera negra', 'Foto de una remera');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('919', 'Categoría REMERAS', 'Foto de categoria');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('920', 'Categoría BUZOS', 'Foto de categoria');

INSERT INTO `bindev`.`photos` (`id_photo`, `name`, `description`) VALUES
('921', 'Categoría PANTALÓN', 'Foto de categoria');

INSERT INTO `bindev`.`category` (`id_category`, `name`, `description`, `picture`)
VALUES ('1', 'REMERAS', 'LAS MEJORES REMERAS','919');

INSERT INTO `bindev`.`category` (`id_category`, `name`, `description`, `picture`)
VALUES ('2', 'BUZOS', 'LOS MEJORES BUZOS','919');

INSERT INTO `bindev`.`category` (`id_category`, `name`, `description`, `picture`)
VALUES ('3', 'CHALECOS', 'LOS MEJORES CHALECOS','919');

INSERT INTO `bindev`.`category` (`id_category`, `name`, `description`, `picture`)
VALUES ('4', 'PANTALONES', 'LOS MEJORES PANTALONES','919');

INSERT INTO `bindev`.`category` (`id_category`, `name`, `description`, `picture`)
VALUES ('5', 'CASCOS ', 'LOS MEJORES CASCOS','920');

INSERT INTO `bindev`.`category` (`id_category`, `name`, `description`, `picture`)
VALUES ('6', 'GUANTES', 'LOS MEJORES GUANTES','919');
```

```sql
INSERT INTO `BINDEV`.`CATEGORY` (`ID_CATEGORY`, `NAME`, `DESCRIPTION`, `PICTURE`)
VALUES ('7', 'ZAPATOS', 'LOS MEJORES ZAPATOS','920');

INSERT INTO `BINDEV`.`CATEGORY` (`ID_CATEGORY`, `NAME`, `DESCRIPTION`, `PICTURE`)
VALUES ('8', 'LENTES', 'LOS MEJORES LENTES DE PROTECCIÓN','919');

INSERT INTO `BINDEV`.`CATEGORY` (`ID_CATEGORY`, `NAME`, `DESCRIPTION`, `PICTURE`)
VALUES ('9', 'GENERICA', 'CATEGORIA GENERICA','921');

INSERT INTO `BINDEV`.`CATEGORY` (`ID_CATEGORY`, `NAME`, `DESCRIPTION`, `PICTURE`)
VALUES ('1000', 'PROMO BASICA', 'PROMO GENERICA','921');

INSERT INTO `BINDEV`.`CATEGORY` (`ID_CATEGORY`, `NAME`, `DESCRIPTION`, `PICTURE`)
VALUES ('1001', 'PROMO PYME', 'PROMO PARA PEQUEÑAS EMPRESAS','920');

INSERT INTO `BINDEV`.`CATEGORY` (`ID_CATEGORY`, `NAME`, `DESCRIPTION`, `PICTURE`)
VALUES ('1002', 'PROMO MEDIUM', 'PROMO PARA EMPRESAS
RECONOCIDAS','921');

INSERT INTO `BINDEV`.`CATEGORY` (`ID_CATEGORY`, `NAME`, `DESCRIPTION`, `PICTURE`)
VALUES ('1003', 'BIG PROMO', 'PROMO PARA EMPRESAS LÍDERES','920');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('1',
'XS', 'EXTRA SMALL');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('2',
'S', 'SMALL');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('3',
'M', 'MEDIUM');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('4',
'L', 'LARGE');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('5',
'XL', 'EXTRA LARGE');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('6',
'XXL', 'EXXTRA LARGE');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('7',
'CHICO', 'ALGO CHICO');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('8',
'MEDIO', 'ALGO MEDIO');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('9',
'GRANDE', 'ALGO GRANDE');
```

```sql
INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('38', 'TALLE 38', 'CALZADO 38');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('39', 'TALLE 39', 'CALZADO 39');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('40', 'TALLE 40', 'CALZADO40');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('41', 'TALLA 41', 'CALZADO TALLE 41');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('42', 'TALLA 42', 'CALZADO TALLE 42');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('43', 'TALLA 43', 'CALZADO TALLE 43');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('44', 'TALLA 44', 'CALZADO TALLE 44');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('45', 'TALLE 45', 'CALZADO 45');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('1000', 'PROMO BASIC ', 'PROMO BASICAS');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('1001', 'PROMO PYMES', 'PROMO PEQUEÑAS EMPRESAS');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('1002', 'PROMO MEDIUM', 'PROMO PARA EMPRESAS EN GRAL');

INSERT INTO `BINDEV`.`SIZE` (`ID_SIZE`, `NAME`, `DESCRIPTION`) VALUES ('1003', 'PROMO BIG ', 'PROMO PARA GRANDES EMPRESAS');

INSERT INTO `BINDEV`.`DESIGN` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES ('0', 'BLANCO', 'COLOR BLANCO');

INSERT INTO `BINDEV`.`DESING` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES ('1', 'NEGRO', 'COLOR NEGRO');

INSERT INTO `BINDEV`.`DESIGN` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES ('2', 'GRIS', 'COLOR GRIS');

INSERT INTO `BINDEV`.`DESIGN` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES ('3', 'VERDE', 'COLOR VERDE');

INSERT INTO `BINDEV`.`DESING` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES ('4',
```

```sql
'AMARILLO', 'COLOR AMARILLO');

INSERT INTO `BINDEV`.`DESIGN` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES ('5',
'ANARANJADO', 'COLOR ANARANJADO');

INSERT INTO `BINDEV`.`DESIGN` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES ('6',
'ROJO', 'COLOR ROJO');

INSERT INTO `BINDEV`.`DESIGN` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES ('7',
'GENERICO', 'CATEGORÍA COLOR GENÉRICA');

INSERT INTO `BINDEV`.`DESIGN` (`ID_DESING`, `NAME`, `DESCRIPTION`) VALUES
('1000', 'PROMO', 'PROMO');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('1000',
'PROMO CUIDADOS', '1000', '1000', '1000', '2', '700', 'PROMO DE
CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('1001',
'PROMO MANOS A LA OBRA', '1000', '1000', '1000', '10', '750', 'PROMO
DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('1002',
'PROMO PYMES', '1001', '1000', '1001', '10', '1000', 'PROMO DE
CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('1003',
'PROMOS PARA TODOS', '1002', '1000', '1002', '0', '1500', 'PROMO DE
CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('1004',
'BIG PROMO BIG', '1003', '1000', '1003', '1', '3000', 'BIG PROMO DE
CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('2000',
'REMERA CLASICA TALLE S COLOR ROJO', '1', '6', '2', '200', '380',
'REMERA DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
```

```sql
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('2000',
'REMERA CLASICA TALLE M COLOR ROJO', '1', '6', '3', '99', '380',
'REMERA DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('2000',
'REMERA CLASICA TALLE L COLOR ROJO', '1', '6', '4', '100', '380',
'REMERA DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('2000',
'REMERA CLASICA TALLE S COLOR NEGRA', '1', '1', '2', '101', '380',
'REMERA DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('2000',
'REMERA CLASICA TALLE M COLOR NEGRA', '1', '1', '3', '0', '380',
'REMERA DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('2000',
'REMERA CLASICA TALLE XL COLOR BLANCA', '1', '0', '5', '5', '380',
'REMERA DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('2001',
'REMERA PREMIUM TALLE S COLOR ROJO', '1', '6', '2', '300', '450', 'LA
MEJOR REMERA DEL MERCADO');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('3000',
'PANTALON CLASICO TALLE S COLOR GRIS', '4', '2', '2', '20', '1200',
'PANTALONES DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('3000',
'PANTALON CLASICO TALLE M COLOR GRIS', '4', '2', '3', '70', '1200',
'PANTALONES DE CALIDAD');

INSERT INTO `BINDEV`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('3000',
'PANTALON CLASICO TALLE L COLOR GRIS', '4', '2', '4', '71', '1200',
'PANTALONES DE CALIDAD');
```

```sql
INSERT INTO `bindev`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('4000',
'ZAPATOS TALLE 39', '7', '7', '39', '21', '3000', 'ZAPATOS DE
CONSTRUCCIÓN DE CALIDAD');

INSERT INTO `bindev`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('4000',
'ZAPATOS TALLE 40', '7', '7', '40', '0', '3000', 'ZAPATOS DE
CONSTRUCCIÓN DE CALIDAD');

INSERT INTO `bindev`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('4000',
'ZAPATOS TALLE 41', '7', '7', '41', '23', '3000', 'ZAPATOS DE
CONSTRUCCIÓN DE CALIDAD');

INSERT INTO `bindev`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('4001',
'ZAPATOS TALLE 44', '7', '7', '44', '46', '3000', 'ZAPATOS DE
CONSTRUCCIÓN DE CALIDAD');

INSERT INTO `bindev`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('5000',
'CASCO GRANDE', '5', '4', '9', '120', '1000', 'LA MEJORES CASCOS DEL
MERCADO');

INSERT INTO `bindev`.`PRODUCT` (`ID_PRODUCT`, `NAME`, `PRODUCT_CATEGORY`,
`PRODUCT_DESING`, `PRODUCT_SIZE`, `STOCK`, `PRICE`, `DESCRIPTION`) VALUES ('5000',
'CASCO CHICO', '5', '4', '7', '150', '1000', 'LA MEJORES CASCOS DEL
MERCADO');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `NAME`, `DESCRIPTION`)
VALUES ('1', 'Mercado Libre', 'Pago por mercado libre');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `NAME`, `DESCRIPTION`)
VALUES ('2', 'PayPal', 'Pago por PayPal');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `NAME`, `DESCRIPTION`)
VALUES ('3', 'Transferencia', 'Pago por Transferencia');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `NAME`, `DESCRIPTION`)
VALUES ('4', 'Efectivo', 'Pago en efectivo');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `NAME`, `DESCRIPTION`)
VALUES ('5', 'Débito', 'Pago por débito en local');
```

```sql
INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `name`, `description`)
VALUES ('6', 'Crédito VISA', 'Crédito por plataforma visa');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `name`, `description`)
VALUES ('7', 'Crédito Master', 'Crédito por plataforma Master');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `name`, `description`)
VALUES ('200', 'Nota de Crédito ', 'Nota de Crédito emitida por la empresa');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `name`, `description`)
VALUES ('300', 'Nota de Débito', 'Nota de Débito emitida por la empresa');

INSERT INTO `bindev`.`PAYMENT_METHOD` (`id_pay_meth`, `name`, `description`)
VALUES ('400', 'Ajuste', 'Ajuste contable');

INSERT INTO `bindev`.`delivery_time` (`id_delivery`, `name`, `description`) VALUES
('1', 'Matutino', 'L a V De 8 a 11 AM');

INSERT INTO `bindev`.`delivery_time` (`id_delivery`, `name`, `description`) VALUES
('2', 'Medio Dia', 'L a V De 13 a 16 HS');

INSERT INTO `bindev`.`delivery_time` (`id_delivery`, `name`, `description`) VALUES
('3', 'Tarde', 'L a V De 17 a 20 HS');

INSERT INTO `bindev`.`delivery_time` (`id_delivery`, `name`, `description`) VALUES
('4', 'Finde', 'Sabados de 8 a 13 hs');

INSERT INTO `bindev`.`delivery_time` (`id_delivery`, `name`, `description`) VALUES
('5', 'Pick-UP', 'Levanta en local');

INSERT INTO `bindev`.`discount` (`value`, `description`) VALUES ('0.20',
'20%OFF');

INSERT INTO `bindev`.`discount` (`value`, `description`) VALUES ('0.30',
'30%OFF');

INSERT INTO `bindev`.`discount` (`value`, `description`) VALUES ('0.50',
'50%OFF');

INSERT INTO `bindev`.`discount` (`value`, `description`) VALUES ('0', 'Sin
Descuento');

INSERT INTO `bindev`.`disburse` (`id_disburse`, `name`, `description`, `state`)
VALUES ('1', 'cheque','pago mediante cheque','1');

INSERT INTO `bindev`.`disburse` (`id_disburse`, `name`, `description`, `state`)
VALUES ('2', 'A 30 dias','Pago a 30 días','1');

INSERT INTO `bindev`.`disburse` (`id_disburse`, `name`, `description`, `state`)
```

```sql
VALUES ('3', 'Por Mercadería Mal Estado','Por devolucion','1');

INSERT INTO `BINDEV`.`SUPPLIER` (`RUT`, `COMPANY_NAME`, `ADDRESS`, `PHONE` )
VALUES ('211223224223','MONTEVIDEO UNIFORMES','18 DE JULIO
123','098098098');

INSERT INTO `BINDEV`.`SUPPLIER` (`RUT`, `COMPANY_NAME`, `ADDRESS`, `PHONE` )
VALUES ('212234522467','FUPI','WWW.FUPI.COM','240588667');

INSERT INTO `BINDEV`.`SUPPLIER` (`RUT`, `COMPANY_NAME`, `ADDRESS`, `PHONE` )
VALUES ('211222344412','VICAS','WWW.VICAS.COM.UY','EL NUM DE CARLOS
22334455');

INSERT INTO `BINDEV`.`SUPPLIER` (`RUT`, `COMPANY_NAME`, `ADDRESS`, `PHONE` )
VALUES ('222573323342','QARIMPORT','MALDONADO 123','22334455');

INSERT INTO `BINDEV`.`STATUS` (`ID_STATUS`,`NAME`, `DESCRIPTION`) VALUES ('1',
'PENDIENTE', 'COMPRA EN ESTADO PENDIENTE DE PAGO');

INSERT INTO `BINDEV`.`STATUS` (`ID_STATUS`,`NAME`, `DESCRIPTION`) VALUES ('2',
'CONFIRMADO', 'COMPRA EN ESTADO CONFIRMADO DE PAGO');

INSERT INTO `BINDEV`.`STATUS` (`ID_STATUS`,`NAME`, `DESCRIPTION`) VALUES ('3',
'EN VIAJE', 'COMPRA EN ESTADO EN VIAJE A DESTINO');

INSERT INTO `BINDEV`.`STATUS` (`ID_STATUS`,`NAME`, `DESCRIPTION`) VALUES ('4',
'CONFIRMADA', 'COMPRA CONFIRMADA Y ENTREGADA AL CLIENTE');

INSERT INTO `BINDEV`.`STATUS` (`ID_STATUS`,`NAME`, `DESCRIPTION`) VALUES ('5',
'PICK-UP', 'COMPRA PARA LEVANTAR EN LOCAL');

INSERT INTO `BINDEV`.`STATUS` (`ID_STATUS`,`NAME`, `DESCRIPTION`) VALUES ('6',
'CANCELADA', 'COMPRA EN ESTADO CANCELADA');

INSERT INTO `BINDEV`.`EMPLOYEE` (`CI`, `EMPLOYEE_USER`, `EMPLOYEE_ROLE`, `STATE`)
VALUES ('12345010', '5010', 'JEFE', '1');

INSERT INTO `BINDEV`.`EMPLOYEE` (`CI`, `EMPLOYEE_USER`, `EMPLOYEE_ROLE`, `STATE`)
VALUES ('12345011', '5011', 'COMPRADOR', '1');

INSERT INTO `BINDEV`.`EMPLOYEE` (`CI`, `EMPLOYEE_USER`, `EMPLOYEE_ROLE`, `STATE`)
VALUES ('12345012', '5012', 'COMPRADOR', '0');

INSERT INTO `BINDEV`.`EMPLOYEE` (`CI`, `EMPLOYEE_USER`, `EMPLOYEE_ROLE`, `STATE`)
VALUES ('12345013', '5013', 'VENDEDOR', '1');
```

4.2 Estudio de Roles y Permisos

En el siguiente análisis se detallan los roles y las posibles acciones que podrán efectuar en la base de datos sobre determinadas tablas los perfiles previamente identificados

Referencia CRUD (Create - Read - Update - Delete)

Tabla	Rol Master - Jefe	Rol Vendedor	Rol Comprador
User	CRU	CRU	R
Employee	CRUD	R	R
Role	CRUD	R	R
Customer	CRU	CRU	R
Sale	CRU	CRU	R
Pay Meth	CRUD	CRU	R
Delivery Time	CRUD	CRU	R
Report	CRU	CRU	RU
Status	CRUD	CRU	R
Sale Detail	CRU	CRU	R
Discount	CRUD	CRU	R
Product	CRUD	CRU	RU
Promo	CRUD	CRU	RU
Design	CRUD	CRU	R
Category	CRUD	CRU	R
Size	CRUD	CRU	R
Gallery	CRUD	CRU	RU
Photos	CRUD	CRUD	RU
Supply Detail	CRU	R	CRU
Supply	CRU	R	CRU
Supplier	CRUD	R	CRU
Disburse	CRUD	R	CRU

HOJA TESTIGO

MATERIA: (Base de Datos 2)

Nombre del Profesor: (Marcos Rodriguez)

_____

_____

_____

_____

_____

_____

_____

_____

_____

Nota Final