

Fastqueue

Propuesta de sistema

E09

V1.6

09/11/2019

Santo Domingo

Republica Dominicana

Historial del documento

Versión No.	Fecha	Autor(es)	Rol
1.0	05/11/2019	Ever Cuevas	CEO
1.2	06/11/2019	Ever Cuevas	CEO
1.3	08/11/2019	Ever Cuevas	CEO
1.4	10/11/2019	Ever Cuevas	CEO
1.5	11/11/2019	Ever Cuevas	CEO
1.6	13/11/2019	Ever Cuevas	CEO

Responsables

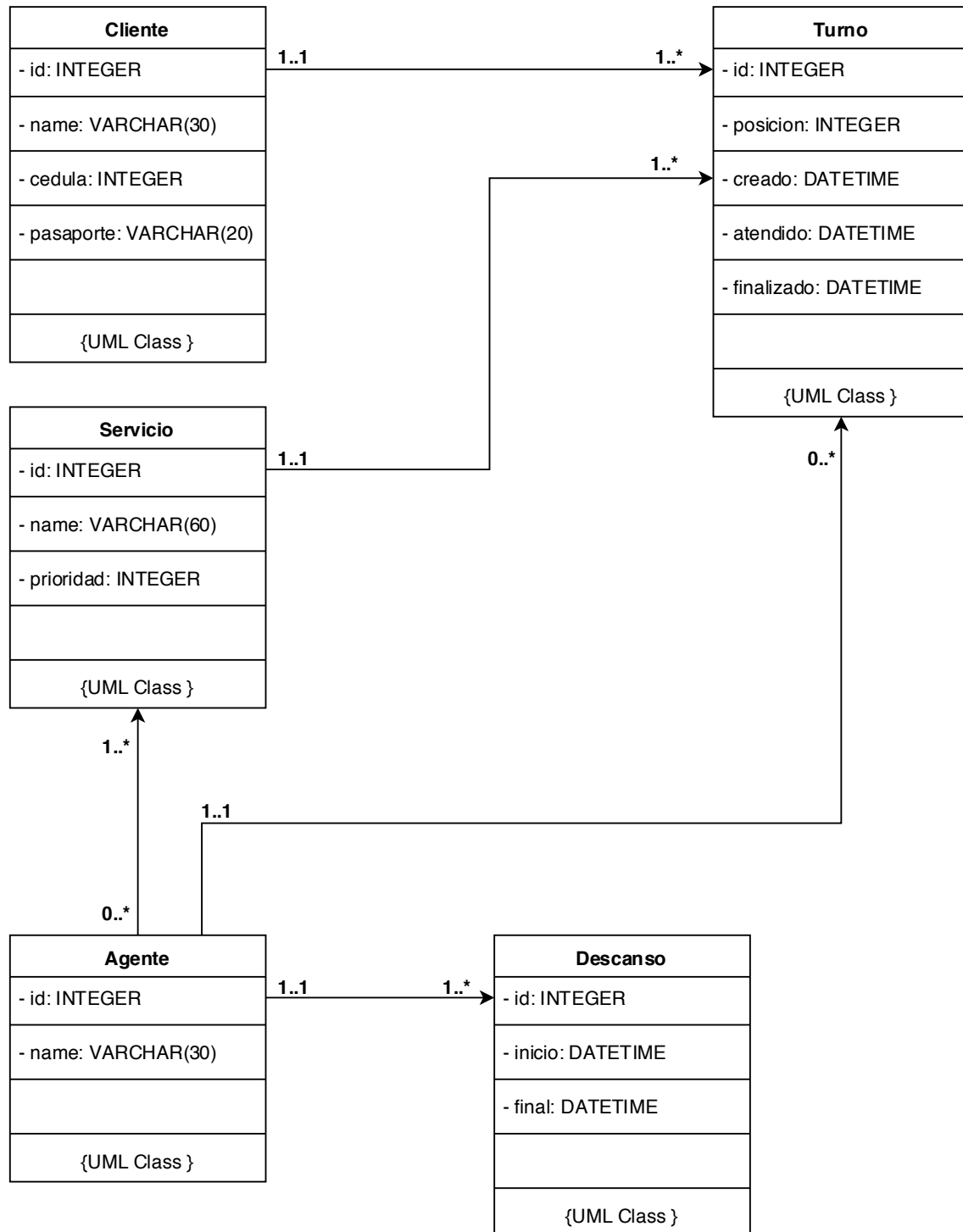
Nombre	Rol	Medio para contacto
Ever Cuevas	CEO	Ever_c@fastqueue.com
María Robles	Dir. Desarrollo	Maria_r@fastqueue.com
Pedro Sánchez	Dir. Ventas	Pedro_s@fastqueue.com
Jennifer Domínguez	Accionista	Jennifer_d@fastqueue.com

Contenido

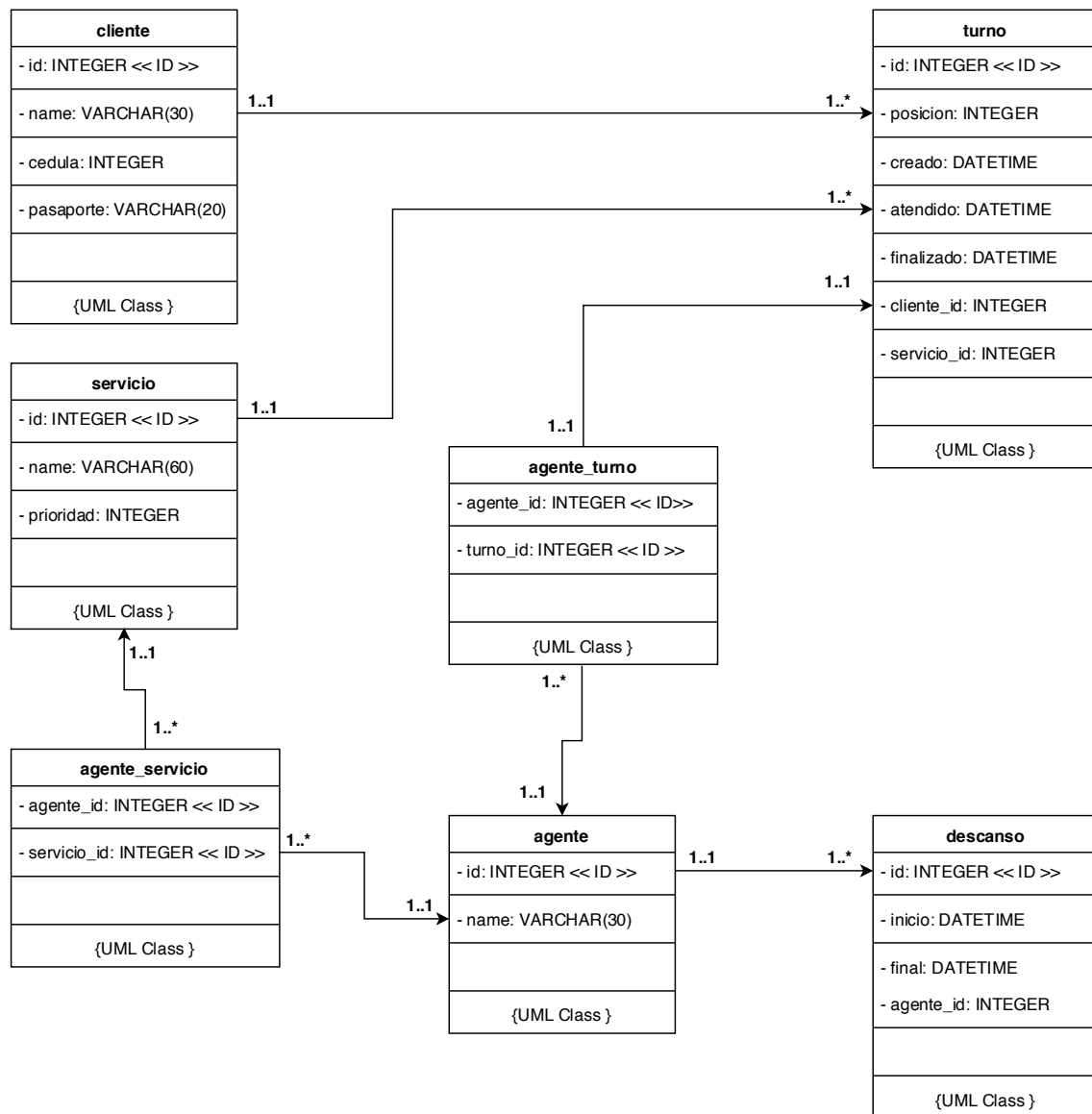
Historial del documento	ii
Responsables.....	iii
1.0 – Diseño de la base de datos	5
1.1 – Diagrama UML de Clases del modelo conceptual del dominio	5
1.2 – Diagrama UML de Clases del Modelo Lógico Puente.....	6
1.3 – Modelo Relacional de cada una de las clases en el Modelo Lógico Puente	7
1.4 – Normalizar (si usted lo considera necesario) por lo menos hasta la 3NF,	7
como le enseñaron en la asignatura Base de Datos, o usted aprendió de.....	7
alguna otra manera;	7
2.0 – Top-Level UML Use Case Diagram.....	8
3.0 – Lista de actores	9
4.0 – Seleccionar un RDBMS.....	10
4.1 – Autenticación, Creación de usuarios, Encriptación	10
4.2 – Escribir Scripts para	11
5.0 – Corridas de los scripts.....	12
6.0 – Security Trigger.....	13

1.0 – Diseño de la base de datos

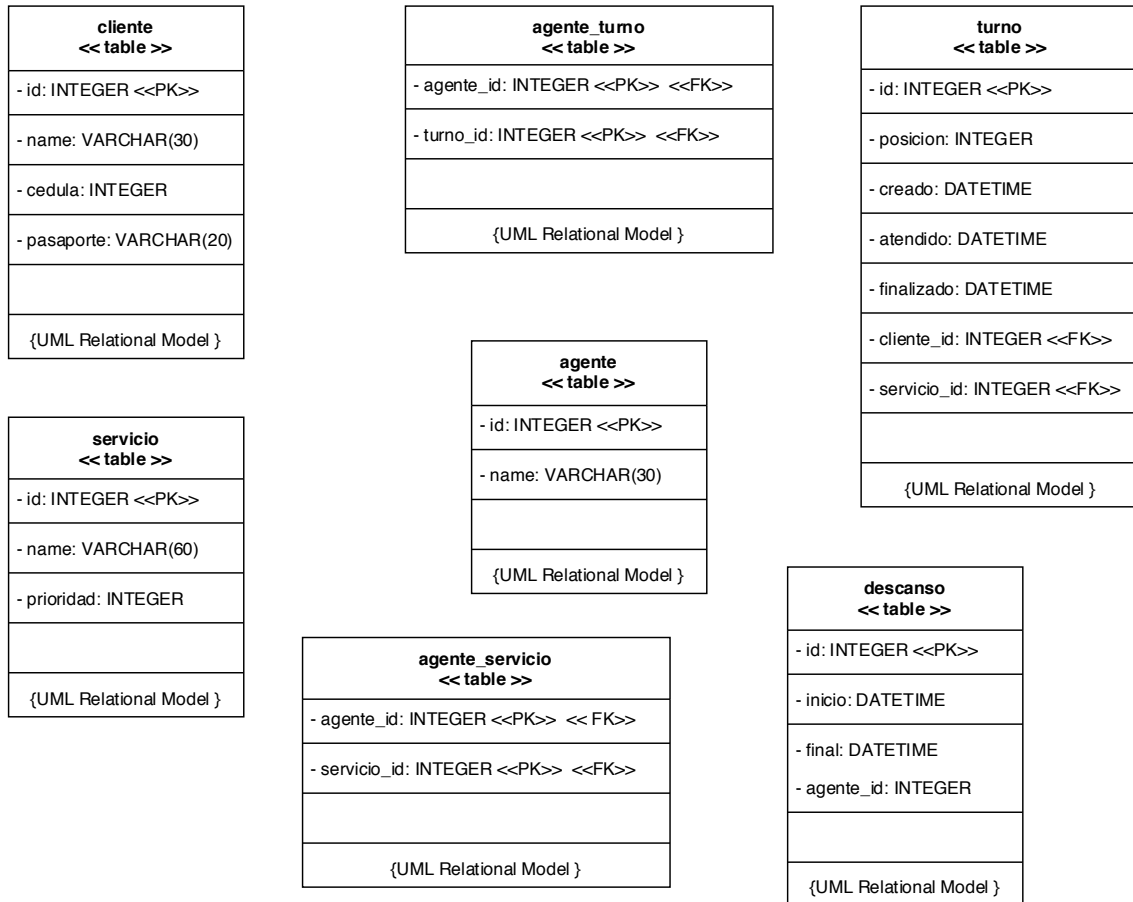
1.1 – Diagrama UML de Clases del modelo conceptual del dominio



1.2 – Diagrama UML de Clases del Modelo Lógico Puente



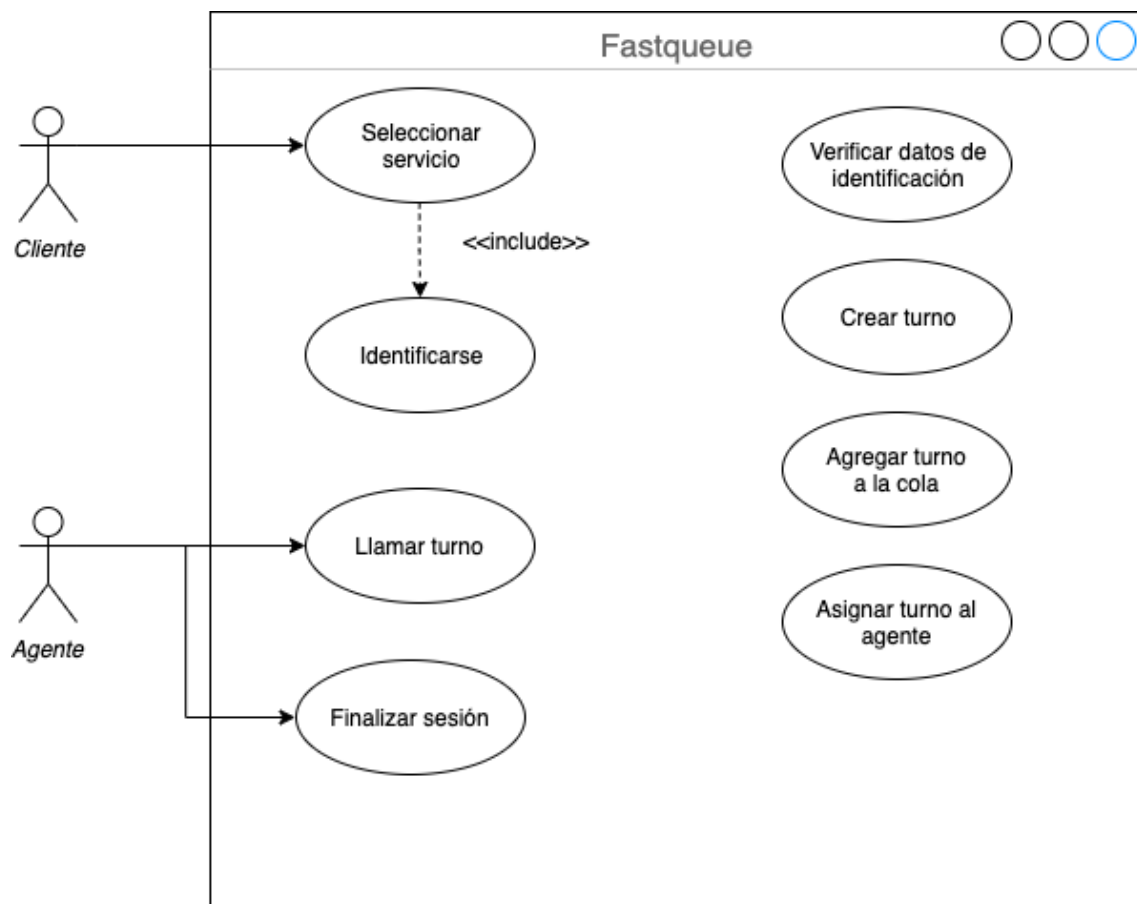
1.3 – Modelo Relacional de cada una de las clases en el Modelo Lógico Puente



1. 4 – Normalizar (si usted lo considera necesario) por lo menos hasta la 3NF, como le enseñaron en la asignatura Base de Datos, o usted aprendió de alguna otra manera;

No es necesario, ya que el diagrama 1.3 esta normalizado.

2.0 – Top-Level UML Use Case Diagram



3.0 – Lista de actores

Actor		Use Case
Name/Role	Type	
Cliente	Primario	- Selecciona servicio - Se identifica en el sistema
Agente	Primario	- Ver lista de turnos - Llamar turno - Solicitar descanso - Finalizar sesión
Sistema	Primario	- Verificar datos de cliente - Crear turno - Agregar turno a la cola - Asignar turno a Agente

4.0 - Seleccionar un RDBMS



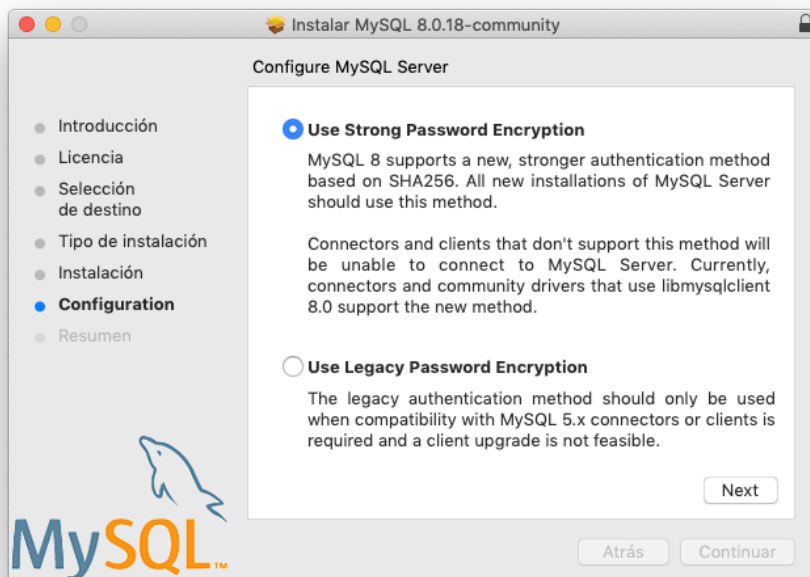
4.1 – Autenticación, Creación de usuarios, Encriptación

Autenticación

```
./mysql -u root -p
```

Iniciar sesión con root para crear el DBA de Fastqueue

Encriptación



Creación de usuarios

```
CREATE USER 'fastqueueDBA'@'localhost' IDENTIFIED BY '123456';
CREATE USER 'fastqueueCliente'@'localhost' IDENTIFIED BY '123456';
CREATE USER 'fastqueueAgente'@'localhost' IDENTIFIED BY '123456';
```

```
mysql> SELECT User FROM mysql.user;
+-----+
| User          |
+-----+
| root          |
| root          |
| root          |
| fastqueueAgente |
| fastqueueCliente |
| fastqueueDBA   |
| root          |
+-----+
7 rows in set (0.00 sec)

mysql>
```

4.2 – Escribir Scripts para

4.2.1 – Crear la base de datos con la seguridad implementada como indicado en 4.1 (user, pass, encrypting); El nombre debe terminar en DB, p. e., ferromDB;

Encriptación

```
SET GLOBAL default_table_encryption=ON;
```

```
CREATE DATABASE fastqueueDB;
```

```
use fastqueueDB;
```

Crear tablas

```
CREATE TABLE cliente(
  id BIGINT NOT NULL AUTO_INCREMENT,
  name VARCHAR(30) NOT NULL DEFAULT '',
  cedula VARCHAR(15) NOT NULL DEFAULT '',
  pasaporte VARCHAR(20) NOT NULL DEFAULT '',
  created_at TIMESTAMP,
  updated_at TIMESTAMP,
  CONSTRAINT PK_CLIENTE PRIMARY KEY(id)
);
```

```

CREATE TABLE agente(
  id BIGINT NOT NULL AUTO_INCREMENT,
  name VARCHAR(30) NOT NULL DEFAULT '',
  email VARCHAR(60) NOT NULL UNIQUE,
  password TEXT NOT NULL,
  created_at TIMESTAMP,
  updated_at TIMESTAMP,
  CONSTRAINT PK_AGENTE PRIMARY KEY(id)
);

CREATE TABLE servicio(
  id BIGINT NOT NULL AUTO_INCREMENT,
  name VARCHAR(60) NOT NULL DEFAULT '',
  prioridad INTEGER UNIQUE NOT NULL,
  created_at TIMESTAMP,
  updated_at TIMESTAMP,
  CONSTRAINT PK_SERVICIO PRIMARY KEY(id)
);

CREATE TABLE agente_servicio(
  agente_id BIGINT NOT NULL,
  servicio_id BIGINT NOT NULL,
  created_at TIMESTAMP,
  updated_at TIMESTAMP,
  CONSTRAINT PK_AGE_SERV PRIMARY KEY(agente_id,servicio_id),
  CONSTRAINT FK_AS_AGENTE FOREIGN KEY (agente_id) REFERENCES
agente(id),
  CONSTRAINT FK_AS_SERVICIO FOREIGN KEY (servicio_id) REFERENCES
servicio(id)
);

CREATE TABLE descanso(
  id BIGINT NOT NULL AUTO_INCREMENT,
  inicio TIMESTAMP,
  final TIMESTAMP,
  agente_id BIGINT NOT NULL,
  created_at TIMESTAMP,
  updated_at TIMESTAMP,
  CONSTRAINT PK_DESCANSO PRIMARY KEY(id),

```

```

    CONSTRAINT FK_DESCANSO_AGENTE FOREIGN KEY (agente_id) REFERENCES
agente(id)
);

CREATE TABLE turno(
    id BIGINT NOT NULL AUTO_INCREMENT,
    posicion BIGINT NOT NULL DEFAULT 0,
    creado TIMESTAMP,
    atendido TIMESTAMP,
    finalizado TIMESTAMP,
    cliente_id BIGINT NOT NULL,
    servicio_id BIGINT NOT NULL,
    created_at TIMESTAMP,
    updated_at TIMESTAMP,
    CONSTRAINT PK_TURN0 PRIMARY KEY (id),
    CONSTRAINT FK_TUR_CLIE FOREIGN KEY (cliente_id) REFERENCES
cliente(id),
    CONSTRAINT FK_TUR_SER FOREIGN KEY (servicio_id) REFERENCES
servicio(id)
);

CREATE TABLE agente_turno(
    agente_id BIGINT NOT NULL,
    turno_id BIGINT NOT NULL,
    created_at TIMESTAMP,
    updated_at TIMESTAMP,
    CONSTRAINT PK_AGE_TURN PRIMARY KEY(agente_id,turno_id),
    CONSTRAINT FK_AT_AGENTE FOREIGN KEY (agente_id) REFERENCES
agente(id),
    CONSTRAINT FK_AT_TURN0 FOREIGN KEY (turno_id) REFERENCES turno(id)
);

```

4.2.3 – Poblar las tablas con 10 tuplas

```
INSERT INTO cliente (name,cedula,created_at) VALUES
("Pedro",'00101458712',NOW()),
('Maria','00901458712',NOW()),
('Rocio','75515275315',NOW()),
('Jennifer','74267034436',NOW()),
('Kike','76464970863',NOW()),
('Pichilino','28801494040',NOW()),
('Petra','82983512192',NOW()),
('China','33563902063',NOW()),
('Zara','16117615730',NOW()),
('Katuska','82846348773',NOW());

INSERT INTO agente (name,email,password,created_at) VALUES
('Fior','f.martinez@grupopopular.com',SHA1(123456),NOW()),
('Tito','t.fernandez@grupopopular.com',SHA1(123456),NOW()),
('Ever','e.cuevas@grupopopular.com',SHA1(123456),NOW()),
('Rosa','r.rodriquez@grupopopular.com',SHA1(123456),NOW()),
('Magdalena','m.ozuna@grupopopular.com',SHA1(123456),NOW()),
('Yiyo','y.sarante@grupopopular.com',SHA1(123456),NOW()),
('Yandel','y.febrero@grupopopular.com',SHA1(123456),NOW()),
('Sofia','s.marquez@grupopopular.com',SHA1(123456),NOW()),
('Pintor','p.tropical@grupopopular.com',SHA1(123456),NOW()),
('Abram','a.lincIn@grupopopular.com',SHA1(123456),NOW());

INSERT INTO servicio (name,prioridad,created_at) VALUES
("Información",1,NOW()),
("Deposito",2,NOW()),
("Cobros",3,NOW()),
("Solicitud de tarjetas",4,NOW()),
("Prestamos",5,NOW()),
("Seguros",6,NOW()),
("Reclamaciones",7,NOW()),
("Servicios generales",8,NOW());

INSERT INTO agente_servicio (agente_id,servicio_id,created_at) VALUES
(1,9,NOW()),
(1,10,NOW()),
(2,9,NOW()),
```

```

(2,11,NOW()),
(3,11,NOW()),
(4,12,NOW()),
(4,13,NOW()),
(5,9,NOW()),
(5,12,NOW()),
(6,10,NOW()),
(6,14,NOW()),
(7,15,NOW()),
(7,16,NOW()),
(8,13,NOW()),
(9,16,NOW()),
(10,10,NOW());

INSERT INTO descanso (inicio,final,agente_id,created_at) VALUES
(NOW(),NOW(),2,NOW()),
(NOW(),NOW(),3,NOW()),
(NOW(),NOW(),1,NOW()),
(NOW(),NOW(),5,NOW()),
(NOW(),NOW(),6,NOW()),
(NOW(),NOW(),6,NOW()),
(NOW(),NOW(),8,NOW()),
(NOW(),NOW(),7,NOW()),
(NOW(),NOW(),9,NOW()),
(NOW(),NOW(),4,NOW()),
(NOW(),NOW(),10,NOW());

INSERT INTO turno
(posicion,creado,atendido,finalizado,cliente_id,servicio_id,created_at
) VALUES
(1,'2019-09-21 14:32:30','2019-09-21 14:32:30','2019-09-21
14:32:30',11,9,NOW()),
(2,'2019-09-21 14:45:30','2019-09-21 14:32:30','2019-09-21
14:32:30',12,16,NOW()),
(3,'2019-09-21 14:32:30','2019-09-21 14:32:30','2019-09-21
14:32:30',13,10,NOW()),
(4,'2019-09-21 14:32:30','2019-09-21 14:32:30','2019-09-21
14:32:30',14,11,NOW()),
(5,'2019-09-21 14:32:30','2019-09-21 14:32:30','2019-09-21
14:32:30',15,15,NOW()),

```

```

(6, '2019-09-21 14:32:30', '2019-09-21 14:32:30', '2019-09-21
14:32:30', 16, 13, NOW()),
(7, '2019-09-21 14:32:30', '2019-09-21 14:32:30', '2019-09-21
14:32:30', 17, 15, NOW()),
(8, '2019-09-21 14:32:30', '2019-09-21 14:32:30', '2019-09-21
14:32:30', 18, 12, NOW()),
(9, '2019-09-21 14:32:30', '2019-09-21 14:32:30', '2019-09-21
14:32:30', 19, 11, NOW()),
(10, '2019-09-21 14:32:30', '2019-09-21 14:32:30', '2019-09-21
14:32:30', 20, 9, NOW());

INSERT INTO agente_turno (agente_id, turno_id, created_at) VALUES
(1, 1, NOW()),
(9, 2, NOW()),
(6, 3, NOW()),
(2, 4, NOW()),
(7, 5, NOW()),
(8, 6, NOW()),
(7, 7, NOW()),
(4, 8, NOW()),
(2, 9, NOW()),
(5, 10, NOW());

```

4.2.4 – Hacer algunas operaciones en la base de datos, usando las sentencias del Lenguaje de manipulación de datos (DML), INNER JOIN, OUTER JOIN, LEFT OUTER JOIN, FULL OUTER JOIN, SELF JOIN, UNION, UNION ALL, del DBMS seleccionado.

Ver los clientes que han sido atendidos, que agente lo atendió, servicio solicitado, y ordenados según la posición de su turno.

```

SELECT c.name AS cliente, c.cedula, s.name AS servicio
, t.posicion, a.name AS agente, t.finalizado FROM cliente AS c
INNER JOIN turno AS t ON c.id = t.cliente_id
INNER JOIN agente_turno AS at ON t.id = at.turno_id
INNER JOIN agente AS a ON at.agente_id = a.id
INNER JOIN servicio AS s ON t.servicio_id = s.id
WHERE finalizado IS NOT NULL ORDER BY t.posicion;

```


5.0 – Corridas de los scripts

```
mysql> SELECT c.name AS cliente,c.cedula,s.name AS servicio ,t.posicion,a.name AS agente, t.finalizado
-> INNER JOIN turno AS t ON c.id = t.cliente_id
-> INNER JOIN agente_turno AS at ON t.id = at.turno_id
-> INNER JOIN agente AS a ON at.agente_id = a.id
-> INNER JOIN servicio AS s ON t.servicio_id = s.id
-> WHERE finalizado IS NOT NULL ORDER BY t.posicion;
```

cliente	cedula	servicio	posicion	agente	finalizado
Pedro	101458712	Información	1	Fior	2019-09-21 14:32:30
Maria	00901458712	Servicios generales	2	Pintor	2019-09-21 14:32:30
Rocio	75515275315	Deposito	3	Yiyo	2019-09-21 14:32:30
Jennifer	74267034436	Cobros	4	Tito	2019-09-21 14:32:30
Kike	76464970863	Reclamaciones	5	Yandel	2019-09-21 14:32:30
Pichilino	28801494040	Prestamos	6	Sofia	2019-09-21 14:32:30
Petra	82983512192	Reclamaciones	7	Yandel	2019-09-21 14:32:30
China	33563902063	Solicitud de tarjetas	8	Rosa	2019-09-21 14:32:30
Zara	16117615730	Cobros	9	Tito	2019-09-21 14:32:30
Katiuska	82846348773	Información	10	Magdalena	2019-09-21 14:32:30

```
10 rows in set (0.00 sec)

mysql>
```

Mostrar tablas (show tables)

```
mysql> show tables;
```

Tables_in_fastqueuedb
agente
agente_servicio
agente_turno
cliente
descanso
servicio
turno

```
7 rows in set (0.00 sec)

mysql>
```

6.0 – Security Trigger

No aplica.