

Proyecto Final

Base de Datos:

“SAT GLOOBAL”



MARCOS PADILLA LLÀCER
1º ASIR

1. Índice

2. Introducción

3. Análisis de requerimientos

4. Análisis estático

- 4.1. Entidades, atributos asociados y dominios.
 - 4.1.1. Especializaciones
- 4.2. Relaciones, atributos propios y dominios.
 - 4.2.1. Agregaciones.
- 4.3. Diagrama EER con todos los atributos (sin los dominios).
- 4.4. RI's no representables.
- 4.5. Transacciones de usuario.

5. Diseño estático

- 5.1. Paso a tablas
- 5.2. Normalización
- 5.3. Transacciones de usuario sobre las relaciones normalizadas.

6. Implementación física que funcione con MySQL

- 6.1. Create database
- 6.2. Crear las tablas
- 6.3. Inserts de prueba
- 6.4. Creación de usuarios.
- 6.5. Crear todas las transacciones de usuarios correspondientes.

7. Diccionario de datos sobre las tablas normalizadas

8. Conclusiones finales del proyecto.

2. Introducción

Mi proyecto va a ser la base de datos de una “ *Red Social* ” enfocada a la promoción de música y eventos para lo que haré uso del motor ***MYSQL InnoDB***.

Este proyecto contiene:

- 7 entidades fuertes que de las cuales, 4 son de especialización.
- 1 especializaciones, que son **Totales** y **disjunta**.

Para el desarrollo del proyecto he utilizado los siguientes programas:

- **LibreOffice**, para documentación e información.
- **DIA**, para la creación del esquema Entidad-Relación.
- **XAMPP** (PhpMyAdmin), para crear la base de datos y la cargar información.
- **VISUAL Studio Code**, para introducir los comandos de MYSQL, de una forma más cómoda.

3. Análisis de Requerimientos.

Nuestro objetivo principal es acercar el Usuario al Artista haciendo que mediante filtros de géneros pueda conocer nueva música de una forma rápida y efectiva.

La especialización llamada **ARTISTAS** tiene dentro de ella, 4 tipos diferentes:

- **Cantantes:**

| artistas_ID | Nombre Usuario | email_privado | Contraseña |
|-------------|--------------------|--------------------------|------------------|
| 1001 | @badbunny | benito@universal.com | 1223456778 |
| 1002 | @ricosuaveyatusabe | yungbeef@lavendicion.com | lav3rdader4gr4sa |
| | | | |

- **Beatmakers:**

| artistas_ID | Nombre Usuario | email_privado | Contraseña |
|-------------|----------------|--------------------|------------------|
| 2001 | @bizarrap | bzrp@universal.com | elbisanoabisa23 |
| 2002 | @manubeats | elmanu@gmail.com | lav3rdader4gr4sa |
| | | | |

- **Diseñador:**

| artistas_ID | Nombre Usuario | email_privado | Contraseña |
|-------------|----------------|--------------------|------------------|
| 3001 | @adrian_alley | adri@alley.com | 1223456778 |
| 3002 | @wingwong_art | wingwong@gmail.com | lav3rdader4gr4sa |
| | | | |

• Eventos:

| artista_ID | Nombre Usuario | email_privado | Contraseña | Dirección |
|------------|----------------|----------------------------|---------------|---|
| 4001 | @maewest | Benito @universal.com | mayyerwestter | Centro Comercial Neptuno, Arabial, 18004 Granada |
| 4002 | @club.la3 | Contrataciones @la3.com | los333forever | Av. de Blasco Ibáñez, 111, 46022 València, Valencia |
| | | | | |

Para que los Artistas puedan compartir su arte hemos creado **PUBLICACIONES**:

| ID | artista_ID | Nombre Usuario | titulo | src_archivo |
|----|------------|----------------|------------------------|----------------|
| 1 | 1001 | @badbunny | Cancion nueva | Prerelease.wav |
| 2 | 3002 | @wingwong_art | Portada de badbunny | Port4da.png |
| | | | | |

| id_pais | genero | n_likes |
|---------|-----------|-----------|
| 100 | Reggaeton | 1.000.000 |
| 300 | Anime | 5.000 |
| | | |

La entidad **PERFIL_ARTISTA** será donde el **ARTISTA** podrá tener un espacio donde se visualicen sus publicaciones de una forma ordenada además de información extra y de interés.

| artista_ID | post_ID | Nombre Usuario | biografía | Email_ contrataciones | Links | profile_img |
|------------|---------|----------------|--------------------------------------|-------------------------|--|-------------|
| 1001 | 1, ... | BAD BUUUNNY! | Escucha mi nuevo álbum click abajo ▼ | Badbunny @universal.com | Link: spotify Link: youtube | Archivo.jpg |
| 3002 | 2, ... | W1NG W0NG | GFX ARTIST | inquires@wingwong.com | Link: OpenSea | Archive.jpg |
| | | | | | | |

Ahora vamos a ver la entidad **USUARIOS**, el cual es capaz de ver publicaciones, perfiles de artistas y seguirlos.

| user_ID | username | email | password | artista_ID |
|---------|----------------|-------------------|-------------|------------|
| 1 | K10LAE | k10lolo@gmail.com | kelokemanin | 3002 |
| 2 | mariacifuentes | Lamari@gmail.com | badbunitamo | 1001 |
| | | | | |

4. Análisis estático

4.1 Entidades, atributos asociados y dominios

• USUARIO (ID,username,email,password,artista_ID)

| | | |
|--------------------|-----------------|--------------------|
| ID | → INTEGER (255) | PRIMARY KEY |
| USERNAME | → VARCHAR (32) | |
| EMAIL | → VARCHAR (32) | |
| PASSWORD | → VARCHAR (32) | |
| ARTISTAS_ID | → INTEGER (255) | FOREIGN KEY |

• PUBLICACIONES

(ID,artista_ID,nombre_Usuario,titulo,id_pais,genero,n_likes,src_archivo)

| | | |
|--------------------|-----------------|--------------------|
| ID | → INTEGER (255) | PRIMARY KEY |
| ARTISTA_ID | → VARCHAR (32) | FOREIGN KEY |
| TITULO | → VARCHAR (32) | |
| ID_PAIS | → VARCHAR (32) | |
| GENERO | → VARCHAR (32) | |
| N_LIKES | → INTEGER (64) | |
| SRC_ARCHIVO | → VARCHAR (32) | |

• PERFIL_ARTISTA

(ID, artista_ID, username, biografia, id_pais, genero, n_likes, src_archivo)

| | | |
|-------------------------|-----------------|--------------------|
| ID | → INTEGER (255) | PRIMARY KEY |
| ARTISTA_ID | → VARCHAR (32) | FOREIGN KEY |
| PUBLICACIONES_ID | → VARCHAR (32) | FOREIGN KEY |
| USERNAME | → VARCHAR (32) | |
| BIOGRAFIA | → VARCHAR (128) | |
| EMAIL_PUBLICO | → VARCHAR (64) | |
| <u>LINKS</u> | → VARCHAR (32) | |
| N_FOLLOWERS | → INTEGER (64) | |
| PROFILE_IMG | → VARCHAR (32) | |

4.1.1 Especializaciones

• ARTISTAS (ID)

ID → INTEGER (255)

■ CANTANTES (username,email_privado,password)

USERNAME → VARCHAR (32)

EMAIL → VARCHAR (32)

PASSWORD → VARCHAR (32)

■ DISEÑADORES (username,email_privado,password)

USERNAME → VARCHAR (32)

EMAIL → VARCHAR (32)

PASSWORD → VARCHAR (32)

■ BEATMAKER (username,email_privado,password)

USERNAME → VARCHAR (32)

EMAIL → VARCHAR (32)

PASSWORD → VARCHAR (32)

■ EVENTOS (username,email_privado,password)

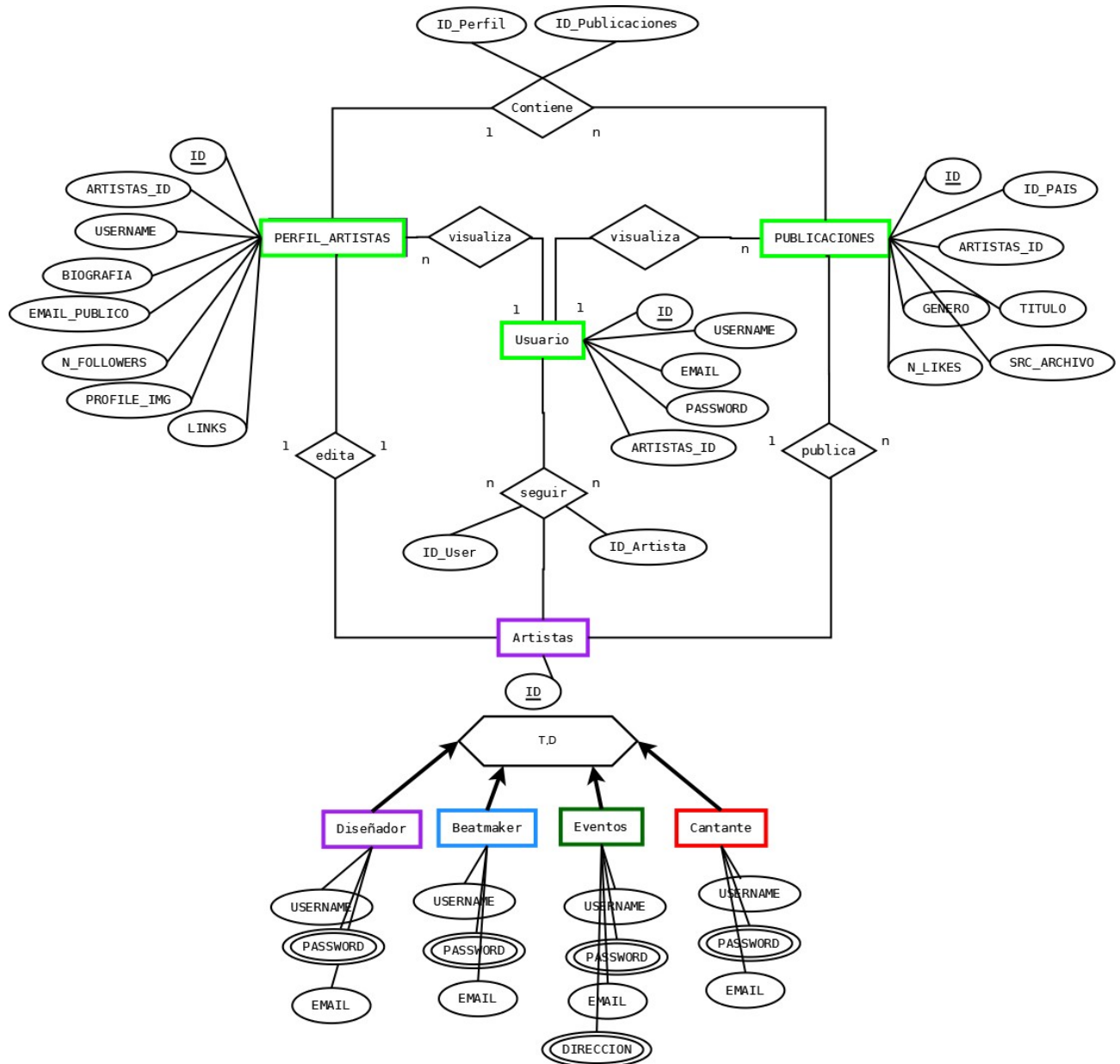
USERNAME → VARCHAR (32)

EMAIL → VARCHAR (32)

PASSWORD → VARCHAR (32)

DIRECCION → VARCHAR (64)

4.3 Diagrama EER



4.4 RI no representables

4.5 Transacciones de usuario

- El **usuario** puede **seguir** a los **artistas**.
- El **usuario** puede **visualizar** la entidad **publicaciones**.
- Los **artistas** publican en la entidad **publicaciones**.
- Las **publicaciones** son **guardadas** en la entidad **perfil_artistas**.
- El **usuario** puede **visualizar** los **perfiles de artistas**.
- Los **artistas editan** sus **perfil** para hacerlo atractivo al usuario.

5. Diseño Estático

5.1 Paso a tablas

Artista edita Perfil_artista – 1:1

Perfil_artista (ID, artistas_ID, username, biografia, email, followers)

Clave Ajena: artistas_ID → Artistas

Artista publica Publicaciones – 1:N

Publicaciones (ID, ID_pais, artistas_ID, genero, titulo, n_likes, src_archivo)

Clave Ajena: artista_ID → Artistas

Perfil_artista contiene Publicaciones – 1:N

Publicaciones (ID, ID_pais, artistas_ID, genero, titulo, n_likes, src_archivo)

Clave Ajena:

| | | |
|-------------------------|---|-----------------------|
| <i>artista_ID</i> | → | Artistas |
| <i>ID_perfil</i> | → | Perfil_artista |
| <i>ID_publicaciones</i> | → | Publicaciones |

Usuario sigue a Artistas – N:N

Artistas (ID)

Clave Ajena:

| | | |
|-------------------|---|-----------------|
| <i>ID_user</i> | → | Usuario |
| <i>ID_artista</i> | → | Artistas |

5.2 Normalización

Artista edita Perfil_artista – 1:1: *No necesita normalización.*

Artista publica Publicaciones – 1:N: *No necesita normalización.*

Perfil_artista contiene Publicaciones – 1:N: *No necesita normalización.*

Usuario sigue a Artistas – N:N: *No necesita normalización.*

5.3 Transacciones de usuarios sobre las relaciones normalizadas

- * Dar “LIKE” en las publicaciones de un artista.
- * Visualizar el perfil del artistas.
- * Seguir a los artistas.
- * Filtrar las publicaciones según el genero.
- * Añadir al perfil de usuario un artista favorito.
- * Filtrar las publicaciones según el ID_pais.

6. Implementación física

6.1 CREATE database

```
mysql -u root  
create database sat_gloobal
```

6.2 Crear las tablas

• Tabla ARTISTAS

```
CREATE TABLE ARTISTAS (  
    ID INT (255) NOT NULL  
    CONSTRAINT ARTIST_ID PRIMARY KEY (ID)  
);
```

• Tabla BEATMAKERS

```
CREATE TABLE BEATMAKERS (  
    ARTISTAS_ID INT (255) NOT NULL  
    USERNAME VARCHAR (32) NOT NULL  
    EMAIL VARCHAR (32) NOT NULL  
    PASSWORD VARCHAR (32) NOT NULL  
    CONSTRAINT PK_USERNAME PRIMARY KEY (USERNAME)  
);
```

• Tabla CANTANTES

```
CREATE TABLE CANTANTES (  
    ARTISTAS_ID INT (255) NOT NULL  
    USERNAME VARCHAR (32) NOT NULL  
    EMAIL VARCHAR (32) NOT NULL  
    PASSWORD VARCHAR (32) NOT NULL  
    CONSTRAINT PK_USERNAME PRIMARY KEY (USERNAME)  
);
```

• Tabla DISEÑADORES

```
CREATE TABLE DISEÑADORES (  
    ARTISTAS_ID INT (255) NOT NULL  
    USERNAME VARCHAR (32) NOT NULL  
    EMAIL VARCHAR (32) NOT NULL  
    PASSWORD VARCHAR (32) NOT NULL  
    CONSTRAINT PK_USERNAME PRIMARY KEY (USERNAME)  
);
```

• Tabla EVENTOS

```
CREATE TABLE EVENTOS (  
    ARTISTAS_ID INT (255) NOT NULL  
    USERNAME VARCHAR (32) NOT NULL  
    EMAIL VARCHAR (32) NOT NULL  
    PASSWORD VARCHAR (32) NOT NULL  
    DIRECCION VARCHAR (64) NOT NULL  
    CONSTRAINT PK_USERNAME PRIMARY KEY (USERNAME)  
);
```

• Tabla PERFIL_ARTISTA

```
CREATE TABLE PERFIL_ARTISTA (  
    ID INT(255) PRIMARY KEY,  
    ARTISTA_ID VARCHAR(32),  
    USERNAME VARCHAR(32),  
    BIOGRAFIA VARCHAR(128),  
    EMAIL_PUBLICO VARCHAR(64),  
    LINKS VARCHAR(32),  
    N_FOLLOWERS INT(64),  
    PROFILE_IMG VARCHAR(32),  
    FOREIGN KEY (ARTISTA_ID) REFERENCES ARTISTAS(ID)  
);
```

CREATE **PUBLICACIONES** (

| | | | |
|-------------|---------|------------------|----------|
| ID | INT | (255) | NOT NULL |
| ARTISTA_ID | INT | (255) | NOT NULL |
| TITULO | VARCHAR | (64) | NOT NULL |
| PAIS_ID | VARCHAR | (32) | NOT NULL |
| GENERO | VARCHAR | (64) | NOT NULL |
| EMAIL | VARCHAR | (64) | NOT NULL |
| LINKS | VARCHAR | (32) | NOT NULL |
| N_LIKES | VARCHAR | (64) | NOT NULL |
| SRC_ARCHIVO | VARCHAR | (64) | NOT NULL |
| CONSTRAINT | ID_POST | PRIMARY KEY (ID) | |

);

CREATE **USUARIOS** (

| | | | |
|---|---------|-------|----------|
| ID | INT | (255) | NOT NULL |
| ARTISTA_ID | INT | (255) | NOT NULL |
| USERNAME | VARCHAR | (32) | NOT NULL |
| EMAIL | VARCHAR | (64) | NOT NULL |
| PASSWORD | VARCHAR | (255) | NOT NULL |
| FOREIGN KEY (ARTISTA_ID) REFERENCES ARTISTAS(ID) | | | |

);


```
CREATE USUARIO_ARTISTA (  
  ARTISTA_ID          INT (255)  
  PUBLICACIONES_ID    INT (255)  
);
```

```
ALTER TABLE `usuario_artista` ADD INDEX(`artistas_ID`);
```

```
ALTER TABLE `usuario_artista` ADD INDEX(`user_ID`);
```

```
ALTER TABLE `usuario_artista` ADD CONSTRAINT `artista_ID` FOREIGN  
KEY (`artistas_ID`) REFERENCES `artistas`(`ID`) ON DELETE RESTRICT  
ON UPDATE RESTRICT;
```

```
ALTER TABLE `usuario_artista` ADD CONSTRAINT `usuario_ID`  
FOREIGN KEY (`user_ID`) REFERENCES `usuarios`(`ID`) ON DELETE  
RESTRICT ON UPDATE RESTRICT;
```

6.3 Inserts de prueba

```
INSERT INTO `artistas` (`ID`) VALUES ('0001'), ('0002'), ('0003'), ('0004'), ('0005'), ('0006'), ('0007'), ('0008'), ('0009'), ('0010');
```

```
INSERT INTO `beatmakers` (`artistas_ID`, `username`, `email`, `password`) VALUES ('3', 'zero', 'zero@gmail.com', '00passw0rd'), ('4', 'BMaster9000', 'beatmaster9000@gmail.com', 'pass1234'), ('5', 'DJMixer99', 'djmixer99@hotmail.com', 'securepass'), ('6', 'TheBeatWizard', 'thebeatwizard@yahoo.com', 'p@ssw0rd'), ('7', 'DJBeats', 'djbeats@gmail.com', '12345abc'), ('8', 'TheRealByte', 'therealbeatmaker@outlook.com', 'qwerty123');
```

```
INSERT INTO `cantantes` (`artistas_ID`, `username`, `email`, `password`) VALUES ('9', 'JuanCanta', 'juan.canta@gmail.com', 'mypass123'), ('10', 'MariaPastori', 'maria_cantautora@hotmail.com', 'singer456'), ('11', 'LuisPonzi', 'luissito@yahoo.com', 'rockstar777'), ('12', 'AnaCervantes', 'anabocervantes@gmail.com', 'mypass123'), ('13', 'CarlitosDobleA', 'carlos_aaaa@outlook.com', 'popstar12'), ('14', 'Laureto', 'laueraa@gmail.com', 'singer23'), ('15', 'ro.oberto', 'robertocantabien@yahoo.com', 'popstar12'), ('16', 'SofiSins', 'lasufiia@gmail.com', 'singer23');
```

```
INSERT INTO `diseñadores` (`artistas_ID`, `username`, `email`, `password`) VALUES ('17', 'chipkidd', 'ChipKidd@gmail.com', 'kidbarato234'), ('18', 'Janoff', 'robertojanoff@gmail.com', 'jajanoooff'), ('19', 'saville', 'elsavillo@gmail.com', 'savutosavut2'), ('20', 'bierutmaikel', 'elmaiky@gmail.com', 'kelokurabeirut'), ('21', 'massivignelli', 'vignelio@gmail.com', 'vigbignelly'), ('22', 'jbarnbrook', 'brook@gmail.com', 'passworld666'), ('23', 'airesmoro', 'moroaire@gmail.com', 'mononomono'), ('24', 'g.lois', 'georgelois@gmail.com', 'loelmejoris');
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (2, 'maria563ca', 'maria453ca@gmail.com', 'senyacoona', 15);
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (3, 'carlos33pintor', 'carlospintor@gmail.com', '33leloso', 10);
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (4, 'anaescala44', 'anaescala44@gmail.com', 'contrepessa4', 4);
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (5, 'pedro55bain', 'pedro55baila@gmail.com', 'ananas662', 5);
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (6, 'laura66es', 'laura6es@gmail.com', 'contraassa6', 16);
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (7, 'pablerascoci', 'pablococi@gmail.com', '777lotto777', 7);
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (8, 'sofia8tora', 'sofia8tora@gmail.com', 'toraxx2123', 18);
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (9, 'carolina9', 'carolina9@gmail.com', 'guiterrassa', 9);
```

```
INSERT INTO USUARIOS (ID, username, email, password, artista_ID)
VALUES (10, 'felipe10foto', 'felipe10foto@gmail.com', 'fotosso1001', 22);
```

6.4 Creación de usuarios. Por defecto tendremos el usuario ROOT. Crea un usuario con el nombre "TCI" con password "ToniCamarena" y asignarle permisos para todas las tablas.

```
GRANT ALL PRIVILEGES ON *.* TO 'tci'@'localhost' IDENTIFIED BY 'ToniCamarena';
```

```
GRANT ALL PRIVILEGES ON sat_global.* TO 'tci'@'localhost' IDENTIFIED BY 'ToniCamarena';
```

6.5 Crear todas las transacciones de usuarios correspondientes al apartado 5.3 y comprueba que "TCI" las pueda ejecutar.

7. DD (Diccionario de datos sobre las tablas normalizadas)

Artistas

| Columna | Tipo | Nulo | Enlaces a |
|---------------|----------|------|-----------|
| ID (Primaria) | int(255) | No | |

Beatmakers

| Columna | Tipo | Nulo | Enlaces a |
|---------------------|-------------|------|----------------|
| artistas_ID | int(255) | No | artistas -> ID |
| username (Primaria) | varchar(32) | No | |
| email | varchar(32) | No | |
| password | varchar(32) | No | |

Cantantes

| Columna | Tipo | Nulo | Enlaces a |
|---------------------|-------------|------|----------------|
| artistas_ID | int(255) | No | artistas -> ID |
| username (Primaria) | varchar(32) | No | |
| email | varchar(32) | No | |
| password | varchar(32) | No | |

Diseñadores

| Columna | Tipo | Nulo | Enlaces a |
|---------------------|-------------|------|----------------|
| artistas_ID | int(255) | No | artistas -> ID |
| username (Primaria) | varchar(32) | No | |
| email | varchar(32) | No | |
| password | varchar(32) | No | |

Eventos

| Columna | Tipo | Nulo | Predeterminado | Enlaces a |
|---------------------|-------------|------|----------------|----------------|
| artistas_ID | int(255) | No | | artistas -> ID |
| username (Primaria) | varchar(32) | No | | |
| email | varchar(32) | No | | |
| password | varchar(32) | No | | |
| direccion | varchar(64) | No | | |

Perfil_artista

| Columna | Tipo | Nulo | Predeterminado | Enlaces a |
|------------------|--------------|------|----------------|---------------------|
| ID (Primaria) | int(255) | No | | |
| artista_ID | int(255) | No | | artistas -> ID |
| publicaciones_ID | int(255) | No | | publicaciones -> ID |
| username | varchar(32) | No | | |
| biografia | varchar(128) | No | | |
| email | varchar(64) | No | | |
| links | varchar(32) | No | | |
| n_Followers | int(100) | No | | |
| profile_IMG | varchar(32) | No | | |

Publicaciones

| Columna | Tipo | Nulo | Enlaces a |
|---------------|-------------|------|----------------|
| ID (Primaria) | int(255) | No | |
| artista_ID | int(255) | No | artistas -> ID |
| titulo | varchar(64) | No | |
| pais_ID | varchar(32) | No | |
| genero | varchar(64) | No | |
| N_Likes | varchar(64) | No | |
| src_Archivo | varchar(64) | No | |

Usuarios

| Columna | Tipo | Nulo | Enlaces a |
|---------------|-------------|------|----------------|
| ID (Primaria) | int(100) | No | |
| username | varchar(32) | No | |
| email | varchar(64) | No | |
| password | varchar(32) | No | |
| artista_ID | int(255) | No | artistas -> ID |

8. Conclusiones finales de proyecto.

Decidí hacer el proyecto sobre esta “Red Social” ya que llevo tiempo pensando en una aplicación que acercará al artista al fan, además de una forma en la que dependiendo del país o género puedan filtrar entre muchos artistas y ver los que los fans quieran.

Además de facilitar al artista a tener su perfil donde pueda promocionar sus trabajos mediante las publicaciones, las cuales serán mostradas al otros usuarios que aun no sigan a ese artista pero si el género es del agrado del usuario se le mostrará esa publicación, de esta forma conseguiremos facilitar el acceso a los proyectos del artista.