



APÉNDICE II

MODELO DE DATOS

1. Recogida de información.

2. Análisis de la información.

2.1. Diseño del Modelo Entidad - Relación.

2.2. Diseño del modelo relacional.

2.3. Tablas y campos de la BBDD.

3. Implementación de la Base de Datos.

3.1. Estructura.

3.2. Datos.

3.3. Sentencias SQL.

1. Recogida de información.

En la aplicación, se van a usar los siguientes datos:

POSICIÓN

#	NOMBRE
1	Líbero
2	Colocador
3	Punta
4	Central
5	Opuesto
6	Polivalente

CATEGORÍA

#	NOMBRE
1	Infantil
2	Cadete
3	Juvenil
4	Junior

COMPETICIÓN

#	NOMBRE
1	Liga regular
2	Copa de España
3	Campeonato de España de clubes
4	Campeonato de España de Selecciones
5	Campeonato de Mallorca
6	Campeonato de Baleares

7	Amistoso
---	----------

ENTRENAMIENTO

#	FECHA	HORA	DESCRIPCIÓN
1	2022-03-16	11:55:14	Entrenamiento1
2	2022-04-16	15:05:01	Defensa R2
3	2022-04-22	18:05:01	Nueva forma de ataque
4	2022-04-01	09:06:58	Ataque de zaguero
5	2022-04-12	06:06:58	Ataque central

ENTRENAMIENTO CON IMAGEN

#	ENTRENAMIENTO	IMAGEN
1	entrenamiento1	img1
2	Defensa R2	img2
3	Nueva forma de ataque	img3
4	Ataque de zaguero	img4
5	Ataque central	img5

JUGADOR

NÚMERO (id)	NOMBRE	DORSAL	EQUIPO	POSICIÓN_ID
1	Joan Feliu Feliu	1	1	1
2	Pere Joan Oliver	2	1	5
3	Santiago Rusiñol	3	1	3
4	Benito Pérez Galdós	4	1	4
5	Alberto Perez Juan	5	1	2

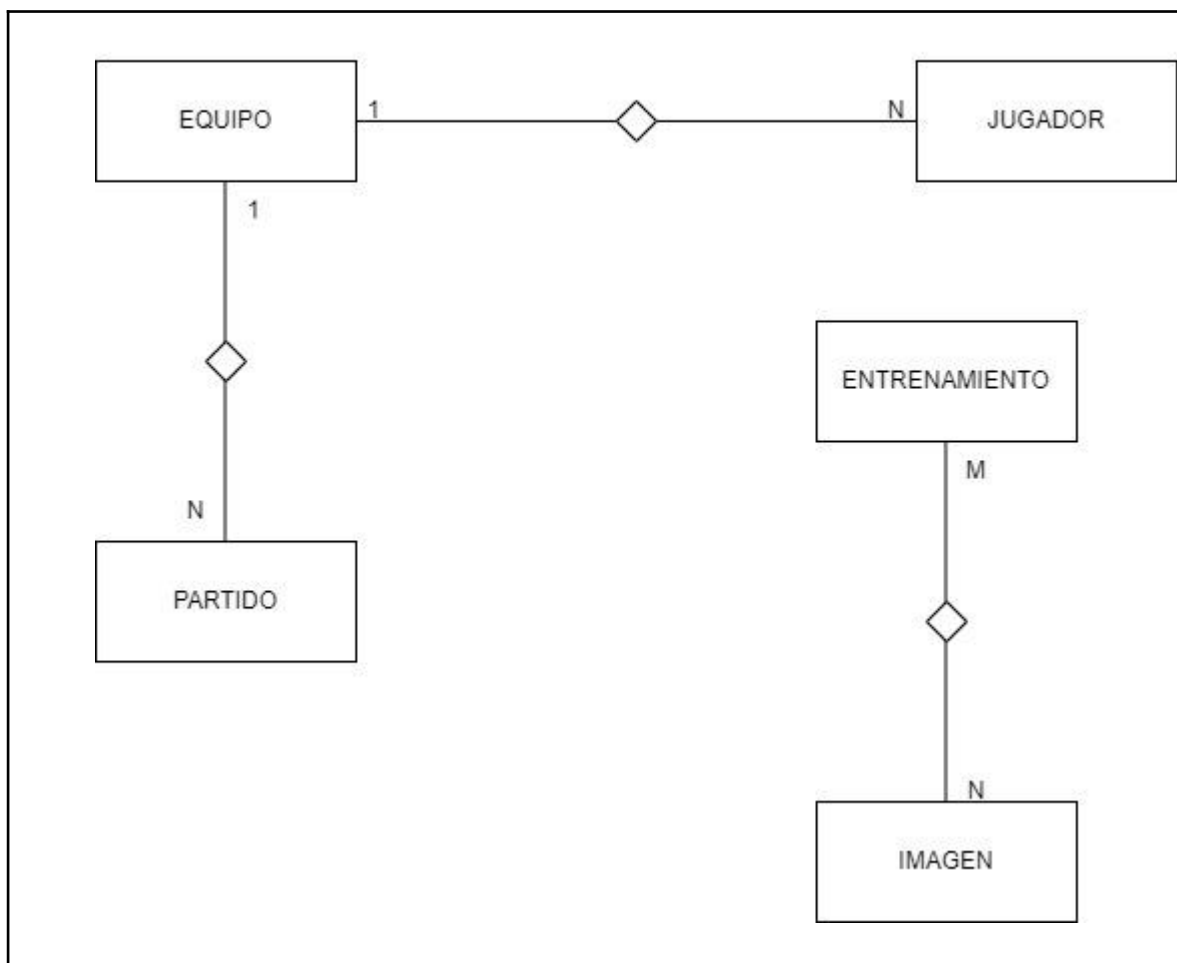
PARTIDO

NÚMERO (id)	FECHA	LOCAL	VISITANTE	SETS LOCAL	SETS VISITANTE	S1 LOCAL	S1 VISITANTE	S2 LOCAL	S2 VISITANTE	S3 LOCAL	S3 VISITANTE	S4 LOCAL	S4 VISITANTE	S5 LOCAL	S5 VISITANTE
1	2022-04-06	1	2	3	1	25	23	25	21	22	25	25	21	NULL	NULL
2	2022-04-06	1	2	3	2	25	20	25	23	25	27	22	25	15	12
3	2022-04-08	2	1	3	0	25	22	25	21	25	22	NULL	NULL	NULL	NULL
4	2022-04-04	3	1	1	3	25	20	20	25	19	25	19	25	NULL	NULL
5	2022-04-04	2	1	2	3	25	21	25	22	19	25	19	25	13	15

2. Análisis de la información.

2.1. Diseño del Modelo Entidad - Relación.

El **modelo entidad-relación** nos permite identificar las entidades de datos de las cuales deberemos guardar información y las interrelaciones entre ellas. El siguiente modelo se diseñó mediante la aplicación **diagrams.net**:



2.2. Diseño del modelo relacional.

El modelo relacional nos permite identificar los datos que se van a guardar en la aplicación. Éste se puede conseguir mediante la aplicación **MySQL Workbench** o trazar a mano mediante **diagrams.net**. Por ejemplo, el **modelo relacional de Keep The Ball Flying** diseñado mediante **Diagrams.net** es:

Lista paginada
+ r: Array de resultados + currentRes: int + numRes: int + numResVisibles: int + x: float + y: float + w: float + h: float
ListaPaginada(Resultado[] r, float x, float y, float w, float h): constructor display(): void next(): void prev(): void

Player
+ name: String + firstPos: String + secPos: String + age: int + numb: int + year: int + month: int + day: int
Player(String name, String firstPos, String secPos, int numb, int year, int month, int day): constructor display(): void getName(): String

Button
+ fillColor: color + strokeColor: color + fillColorOver: color + fillColorDisabled: color + enabled: boolean + x: float + y: float + w: float + h: float
Button(String text, float x, float y, float w, float h, boolean opaco): método constructor setEnabled(): void enable(): boolean display(): void mouseOverButton(): boolean

Select
+ x: float + w: float + y: float + selected: boolean + txt: String
Select(String txt, float x, float y, boolean selected, float w): método constructor display(): void setSelected(boolean select): void getSelected(): String mouseOverSelected(): boolean

Table
+ tableHeaders: String [][] + tableData: String [][] + columnWidths: float [][] + numCols: int + numRows: int + numPage: int + numTotalPages: int
Table(int nr, int nc): constructor display(float x, float y, float w, float h): void prevPage(): void nextPage(): void setHeaders(String[] h): void setData(String[][] d): void SetValueAt(String value, int nr, int nc): void setColumnWidths(float[] w): void

Guardado
+ img: imagen + titulo: String + day: int + month: int + year: int + numPage: int + numTotalPages: int + x: float + y: float + h: float + w: float
Guardado(PImage img, String titulo, int day, int month, int year, float x, float y): método constructor display(): void

Cone
+ x: float + y: float + w: float + h: float
Cone(float x, float y): método constructor display(): void mouseOverCone(): boolean

Select 2
+ x: float + w: float + w: float + h: float + texts: String[] + selectedValue: String + collapsed: boolean + enabled: boolean
Select2(String[] texts0, float x, float y, float h, float w): método constructor display(): void setCollapsed(boolean s): void toggle(): void mouseOverSelect2(): boolean clickedOption(): int

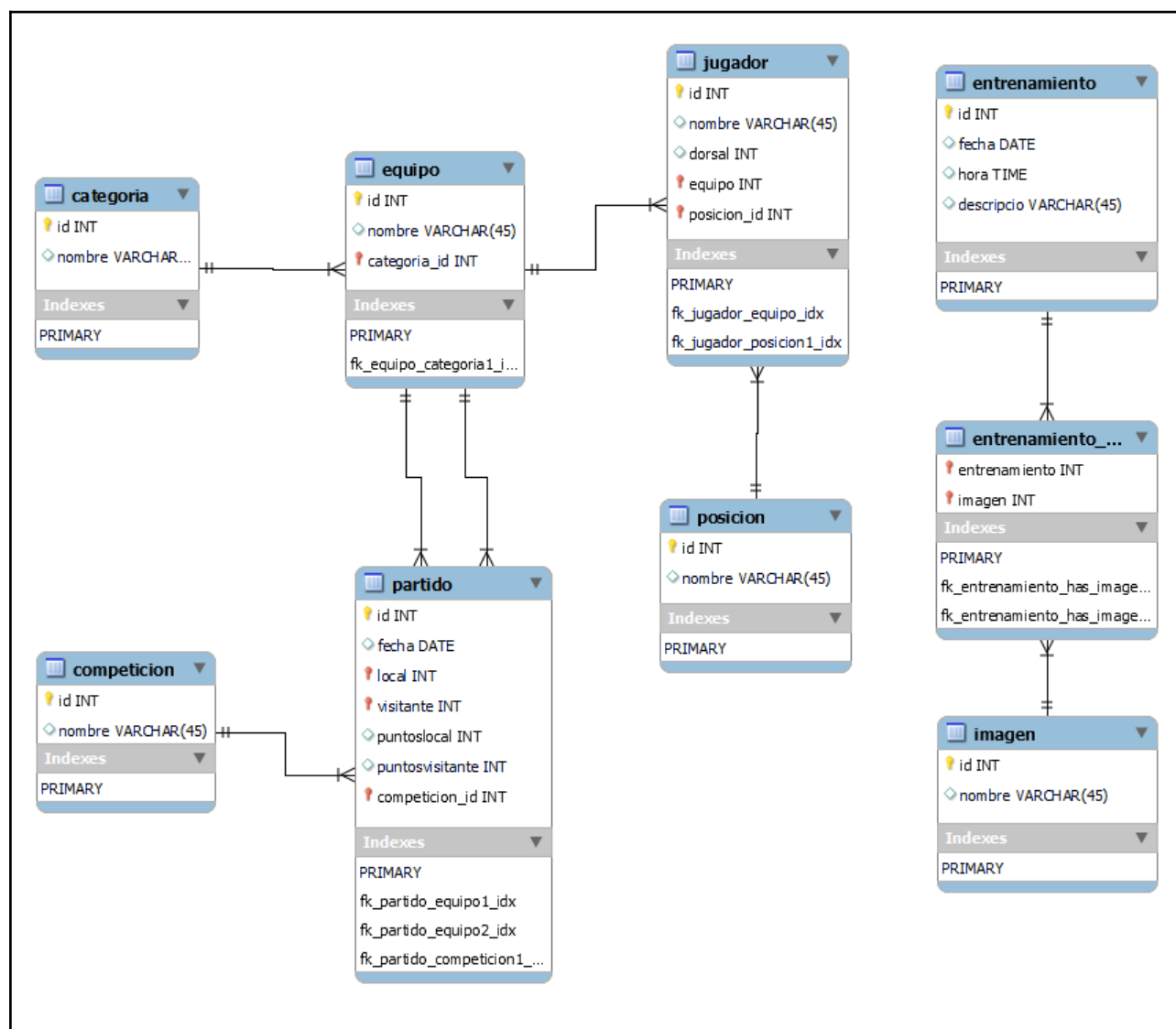
Pin
+ txt: String + r: float + color: c + enabled: boolean + x: float + y: float + w: float + h: float
Pin(float x, float y, float r, String t, color c, boolean enabled): método constructor display(): void mouseOver(): void void setEnable(boolean enable): void getEnabled(): boolean

Pines
+ pins1: Pin [] + pins2: Pin [] + x: float + y: float + w: float + h: float
Pines(float x, float y, float w, float h, float xPin1, float xPin2): método constructor displayPins1(): void displayPins2(): void mouseOver(): void checkPinsMotion(): void resetPinsPositions(): void

Resultado
+ nPartido: int + contrario: String + fecha: String + r1: string + s21_1: String + x: float + s21_1: String + s21_1: String + s21_1: String + s21_1: String + s21_1: String + s21_1: String
Resultado(float x, float y, int nPartido): método constructor display(): void checkContrario(): void setY(float n): void setX(float n): void

TextField
+ x: float + w: float + y: float + selected: boolean + txt: String
TextField(int x, int y, int w, int h): método constructor display(): void keyPressed(char key, int keyCode): void addText(char c): void mouseOverTextField(): boolean isPressed(): void

Por otro lado, el **modelo relacional de Keep The Ball Flying** diseñado con la herramienta **MySQL Workbench** es:



2.3. Tablas y campos de la BBDD.

Todos y cada uno de los campos y las tablas forman parte de la base de datos

a) Tabla partido

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
---	--------	------	------	----------------	-------

1	id	int(11)	No	None	AUTO_INCREMENT
2	fecha	fecha	Sí	NULL	
3	local	int(11)	No	None	
4	visitante	int(11)	No	None	
5	competicion_id	int(11)	No	None	
6	setslocal	int(11)	Sí	NULL	
7	setsvisitante	int(11)	Sí	NULL	
8	s1local	int(11)	Sí	NULL	
9	s1visitante	int(11)	Sí	NULL	
10	s2local	int(11)	No	None	
11	s2visitante	int(11)	No	None	
12	s3local	int(11)	No	None	
13	s3visitante	int(11)	No	None	
14	s4local	int(11)	Sí	NULL	
15	s4visitante	int(11)	Sí	NULL	
16	s5local	int(11)	Sí	NULL	
17	s5visitante	int(11)	Sí	NULL	

b) Tabla categoria

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
1	id	int(11)	No	Ninguna	AUTO_INCREMENT
2	nombre	varchar(45)	Sí	NULL	

c) Tabla competición

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
1	id	int(11)	No	Ninguna	AUTO_INCREMENT
2	nombre	varchar(45)	Sí	NULL	

d) Tabla entrenamiento

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
1	id	int(11)	No	Ninguna	AUTO_INCREMENT
2	fecha	date	Sí	NULL	
3	hora	time	Si	NULL	
4	descripción	varchar(45)	Si	NULL	

e) Tabla entrenamiento_has_imagen

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
1	entrenamiento	int(11)	No	Ninguna	AUTO_INCREMENT
2	fecha	date	No	Ninguna	

f) Tabla equipo

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
1	id	int(11)	No	Ninguna	AUTO_INCREMENT
2	nombre	varchar(45)	Sí	NULL	
3	categoria_id	int(11)	No	Ninguna	

g) Tabla guardado

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
1	id	int(20)	No	Ninguna	AUTO_INCREMENT
2	imagen	varchar(20)	No	Ninguna	
3	nombre	varchar(20)	No	Ninguna	
4	fecha	date	No	Ninguna	

h) Tabla jugador

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
1	id	int(20)	No	Ninguna	AUTO_INCREMENT
2	nombre	varchar(45)	Sí	NULL	
3	dorsal	int(11)	Sí	NULL	
4	equipo	int(11)	No	1	
5	posicion_id	int(11)	No	Ninguna	

i) Tabla posición

#	NOMBRE	TIPO	NULO	PREDETERMINADO	EXTRA
1	id	int(11)	No	Ninguna	AUTO_INCREMENT
2	nombre	varchar(45)	Sí	NULL	

3. Implementación de la Base de Datos.

El siguiente código SQL para implementar la base de datos incluye todas aquellas sentencias de creacion de tablas definicion de los campos y tipos. Se puede obtener a través de la aplicación **MySQL Workbench** o a través de la aplicación **PhpMyAdmin**.

3.1. Estructura.

Información de la estructura de las tablas de la Base de Datos:

```
-- phpMyAdmin SQL Dump
-- version 5.1.0
-- https://www.phpmyadmin.net/
--
-- Host: localhost:8889
-- Generation Time: Apr 08, 2022 at 05:48 PM
-- Server version: 5.7.34
-- PHP Version: 7.4.21

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `voley`
--

--
-- Table structure for table `categoria`
--

CREATE TABLE `categoria` (
  `id` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Table structure for table `competicion`
--

CREATE TABLE `competicion` (
  `id` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```

--
-- Table structure for table `entrenamiento`
--

CREATE TABLE `entrenamiento` (
  `id` int(11) NOT NULL,
  `fecha` date DEFAULT NULL,
  `hora` time DEFAULT NULL,
  `descripcion` varchar(45) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-----

--
-- Table structure for table `entrenamiento_has_imagen`
--

CREATE TABLE `entrenamiento_has_imagen` (
  `entrenamiento` int(11) NOT NULL,
  `imagen` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-----

--
-- Table structure for table `equipo`
--

CREATE TABLE `equipo` (
  `id` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL,
  `categoria_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-----

--
-- Table structure for table `guardado`
--

CREATE TABLE `guardado` (
  `id` int(20) NOT NULL,
  `imagen` varchar(20) NOT NULL,
  `nombre` varchar(20) NOT NULL,
  `fecha` date NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-----

--
-- Table structure for table `imagen`
--

CREATE TABLE `imagen` (
  `id` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-----

--
-- Table structure for table `jugador`
--

CREATE TABLE `jugador` (
  `id` int(11) NOT NULL,

```

```

`nombre` varchar(45) DEFAULT NULL,
`dorsal` int(11) DEFAULT NULL,
`equipo` int(11) NOT NULL DEFAULT '1',
`posicion_id` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- -----

--
-- Table structure for table `partido`
--

CREATE TABLE `partido` (
  `id` int(11) NOT NULL,
  `fecha` date DEFAULT NULL,
  `local` int(11) NOT NULL,
  `visitante` int(11) NOT NULL,
  `competicion_id` int(11) NOT NULL,
  `setslocal` int(11) DEFAULT NULL,
  `setsvisitante` int(11) DEFAULT NULL,
  `s1local` int(11) DEFAULT NULL,
  `s1visitante` int(11) DEFAULT NULL,
  `s2local` int(11) NOT NULL,
  `s2visitante` int(11) NOT NULL,
  `s3local` int(11) NOT NULL,
  `s3visitante` int(11) NOT NULL,
  `s4local` int(11) DEFAULT NULL,
  `s4visitante` int(11) DEFAULT NULL,
  `s5local` int(11) DEFAULT NULL,
  `s5visitante` int(11) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- -----

--
-- Table structure for table `posicion`
--

CREATE TABLE `posicion` (
  `id` int(11) NOT NULL,
  `nombre` varchar(45) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

--
-- Indexes for dumped tables
--

--
-- Indexes for table `categoria`
--
ALTER TABLE `categoria`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `competicion`
--
ALTER TABLE `competicion`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `entrenamiento`
--
ALTER TABLE `entrenamiento`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `entrenamiento_has_imagen`

```

```

--
ALTER TABLE `entrenamiento_has_imagen`
  ADD PRIMARY KEY (`entrenamiento`,`imagen`),
  ADD KEY `fk_entrenamiento_has_imagen_imagen1_idx` (`imagen`),
  ADD KEY `fk_entrenamiento_has_imagen_entrenamiento1_idx` (`entrenamiento`);

--
-- Indexes for table `equipo`
--
ALTER TABLE `equipo`
  ADD PRIMARY KEY (`id`,`categoria_id`),
  ADD KEY `fk_equipo_categorial_idx` (`categoria_id`);

--
-- Indexes for table `guardado`
--
ALTER TABLE `guardado`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `imagen`
--
ALTER TABLE `imagen`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `jugador`
--
ALTER TABLE `jugador`
  ADD PRIMARY KEY (`id`,`equipo`,`posicion_id`),
  ADD KEY `fk_jugador_equipo_idx` (`equipo`),
  ADD KEY `fk_jugador_posicion1_idx` (`posicion_id`);

--
-- Indexes for table `partido`
--
ALTER TABLE `partido`
  ADD PRIMARY KEY (`id`,`local`,`visitante`,`competicion_id`),
  ADD KEY `fk_partido_equipo1_idx` (`local`),
  ADD KEY `fk_partido_equipo2_idx` (`visitante`),
  ADD KEY `fk_partido_competicion1_idx` (`competicion_id`),
  ADD KEY `competicion_id` (`competicion_id`);

--
-- Indexes for table `posicion`
--
ALTER TABLE `posicion`
  ADD PRIMARY KEY (`id`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `categoria`
--
ALTER TABLE `categoria`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `competicion`
--
ALTER TABLE `competicion`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `entrenamiento`

```

```
--
ALTER TABLE `entrenamiento`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `equipo`
--
ALTER TABLE `equipo`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `guardado`
--
ALTER TABLE `guardado`
  MODIFY `id` int(20) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `imagen`
--
ALTER TABLE `imagen`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `jugador`
--
ALTER TABLE `jugador`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `partido`
--
ALTER TABLE `partido`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- AUTO_INCREMENT for table `posicion`
--
ALTER TABLE `posicion`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT;

--
-- Constraints for dumped tables
--

--
-- Constraints for table `entrenamiento_has_imagen`
--
ALTER TABLE `entrenamiento_has_imagen`
  ADD CONSTRAINT `fk_entrenamiento_has_imagen_entrenamiento1` FOREIGN KEY
  (`entrenamiento`) REFERENCES `entrenamiento` (`id`),
  ADD CONSTRAINT `fk_entrenamiento_has_imagen_imagen1` FOREIGN KEY (`imagen`)
  REFERENCES `imagen` (`id`);

--
-- Constraints for table `equipo`
--
ALTER TABLE `equipo`
  ADD CONSTRAINT `fk_equipo_categorial` FOREIGN KEY (`categoria_id`) REFERENCES
  `categoria` (`id`);

--
-- Constraints for table `jugador`
--
ALTER TABLE `jugador`
  ADD CONSTRAINT `fk_jugador_equipo` FOREIGN KEY (`equipo`) REFERENCES `equipo`
  (`id`),
  ADD CONSTRAINT `fk_jugador_posicion1` FOREIGN KEY (`posicion_id`) REFERENCES
```

```

`posicion` (`id`);

--
-- Constraints for table `partido`
--
ALTER TABLE `partido`
  ADD CONSTRAINT `fk_partido_competicion1` FOREIGN KEY (`competicion_id`)
REFERENCES `competicion` (`id`),
  ADD CONSTRAINT `fk_partido_equipo1` FOREIGN KEY (`local`) REFERENCES `equipo`
(`id`),
  ADD CONSTRAINT `fk_partido_equipo2` FOREIGN KEY (`visitante`) REFERENCES
`equipo` (`id`);
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

3.2. Datos.

Información de la base de datos:

```

-- phpMyAdmin SQL Dump
-- version 5.1.0
-- https://www.phpmyadmin.net/
--
-- Host: localhost:8889
-- Generation Time: Apr 08, 2022 at 05:50 PM
-- Server version: 5.7.34
-- PHP Version: 7.4.21

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `voley`
--

--
-- Dumping data for table `categoria`
--

INSERT INTO `categoria` (`id`, `nombre`) VALUES
(1, 'Infantil'),
(2, 'Cadete'),
(3, 'Juvenil'),
(4, 'Junior');

```



```
--
-- Dumping data for table `competicion`
--

INSERT INTO `competicion` (`id`, `nombre`) VALUES
(1, 'Liga regular'),
(2, 'Copa de España'),
(3, 'Campeonato de España de clubes'),
(4, 'Campeonato de España de selecciones'),
(5, 'Campeonato de Mallorca'),
(6, 'Campeonato de Baleares'),
(7, 'Amistoso');

--
-- Dumping data for table `entrenamiento`
--

INSERT INTO `entrenamiento` (`id`, `fecha`, `hora`, `descripcion`) VALUES
(1, '2022-03-16', '11:55:14', 'entrenamiento1'),
(2, '2022-04-16', '15:05:01', 'Defensa R2'),
(3, '2022-04-22', '18:05:01', 'Nueva forma de ataque'),
(4, '2022-04-01', '09:06:58', 'Ataque de zaguero'),
(5, '2022-04-12', '06:06:58', 'Ataque central');

--
-- Dumping data for table `equipo`
--

INSERT INTO `equipo` (`id`, `nombre`, `categoria_id`) VALUES
(1, 'Club Voley Palma', 1),
(2, 'Club Voley Palma', 2),
(3, 'Club Voley Palma', 3),
(4, 'Club Voley Palma', 4),
(5, 'Club Voley Manacor', 1),
(6, 'Club Voley Manacor', 2),
(7, 'Club Voley Manacor', 3),
(8, 'Club Voley Manacor', 4),
(9, 'Club Voley Sóller', 1),
(10, 'Club Voley Sóller', 2),
(11, 'Club Voley Sóller', 3),
(12, 'Club Voley Sóller', 4);

--
-- Dumping data for table `guardado`
--

INSERT INTO `guardado` (`id`, `imagen`, `nombre`, `fecha`) VALUES
(1, 'img0', 'k1', '2022-04-06'),
(2, 'img0', 'k2', '2022-04-07');

--
-- Dumping data for table `jugador`
--

INSERT INTO `jugador` (`id`, `nombre`, `dorsal`, `equipo`, `posicion_id`) VALUES
(1, 'Joan Feliu Feliu', 1, 1, 1),
(2, 'Pere Joan Oliver', 2, 1, 5),
(3, 'Santiago Rusiñol', 3, 1, 3),
```

```

(4, 'Benito Pérez Galdós', 4, 1, 4),
(5, 'Alberto Perez Juan', 5, 1, 2);

--
-- Dumping data for table `partido`
--

INSERT INTO `partido` (`id`, `fecha`, `local`, `visitante`,
`competicion_id`, `setslocal`, `setsvisitante`, `s1local`, `s1visitante`,
`s2local`, `s2visitante`, `s3local`, `s3visitante`, `s4local`,
`s4visitante`, `s5local`, `s5visitante`) VALUES
(1, '2022-04-06', 1, 5, 7, 3, 1, 25, 23, 25, 21, 22, 25, 25, 21, NULL,
NULL),
(2, '2022-04-06', 1, 9, 7, 3, 2, 25, 20, 25, 23, 25, 27, 22, 25, 15, 13),
(3, '2022-04-08', 2, 1, 5, 3, 0, 25, 22, 25, 21, 25, 22, NULL, NULL, NULL,
NULL),
(4, '2022-04-04', 3, 1, 6, 1, 3, 25, 20, 20, 25, 19, 25, 19, 25, NULL,
NULL),
(5, '2022-04-04', 2, 5, 3, 2, 3, 25, 21, 25, 22, 19, 25, 19, 25, 13, 15),
(6, '2022-04-15', 7, 4, 4, 0, 3, 19, 25, 18, 25, 22, 25, NULL, NULL, NULL,
NULL),
(7, '2022-04-18', 9, 5, 3, 1, 3, 17, 25, 20, 25, 25, 22, 22, 25, NULL,
NULL);

--
-- Dumping data for table `posicion`
--

INSERT INTO `posicion` (`id`, `nombre`) VALUES
(1, 'Libero'),
(2, 'Colocador'),
(3, 'Punta'),
(4, 'Central'),
(5, 'Opuesto'),
(6, 'Polivalente');
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

3.3. Sentencias SQL.

Las sentencias SQL más utilizadas para la consulta e inserción de datos a la base de datos de la aplicación són:

SELECTS

```
"SELECT p.id AS id, p.fecha AS fecha, c.nombre AS competicion,
e1.nombre AS local, e2.nombre AS visitante, p.setslocal AS setslocal,
p.setsvisitante AS setsvisitante, p.s1local AS s1local, p.s1visitante AS
s1visitante, p.s2local AS s2local, p.s2visitante AS s2visitante, p.s3local
AS s3local, p.s3visitante AS s3visitante, p.s4local AS s4local,
p.s4visitante AS s4visitante, p.s5local AS s5local, p.s5visitante AS
s5visitante FROM partido p , equipo e1, equipo e2, competicion c
WHERE p.local=e1.id AND p.visitante=e2.id AND p.competicion_id=c.id
ORDER BY `fecha` DESC"
```

```
SELECT j.id AS id, j.nombre AS nombre, p.nombre AS categoria FROM
equipo j, categoria p WHERE j.categoria_id=p.id ORDER BY `id` ASC
```

```
"SELECT j.nombre AS nombre, j.dorsal AS dorsal, p.nombre AS posicion
FROM jugador j, posicion p WHERE j.posicion_id=p.id"
```

INSERTS

```
INSERT INTO posicion (id,nombre) VALUES (NULL, '"'+n+'"')
```