

8-11-2022

Proyecto

Intermodular bases de datos- Entornos de desarrollo

Eduardo Martín-Sonseca Alonso Mario Ortuñez Sanz

Noel Prieto Pardo

Contenido

[1. Introducción 2](#_Toc128231848)

[2. Diseño de la base de datos 2](#_Toc128231849)

[2.1. Diseño conceptual 2](#_Toc128231850)

[2.1.1. Especificación de requisitos 2](#_Toc128231851)

[2.1.2. Modelo Entidad-Relación 3](#_Toc128231852)

[2.2. Diseño lógico 3](#_Toc128231853)

[2.2.1. Modelo relacional 3](#_Toc128231854)

[2.2.2. Normalización 4](#_Toc128231855)

[2.3. Diseño físico 5](#_Toc128231856)

[2.3.1. Modelo físico 5](#_Toc128231857)

[2.3.2. Script de creación de la BD 5](#_Toc128231858)

[2.3.3. Diccionario de datos 5](#_Toc128231859)

[3. Consultas SQL 7](#_Toc128231860)

[4. Subprogramas PL/SQL 7](#_Toc128231861)

[5. Recursos empleados / necesarios 7](#_Toc128231862)

[6. Temporalización 9](#_Toc128231863)

# Introducción

Se desea realizar una base de datos para una empresa de parques infantiles (parque de bolas o similar). Normalmente estas empresas no tienen informatizadas las reservas, el acceso y las cuentas y funcionan con una simple hoja de datos. Nuestro objetivo es realizar la informatización de este sistema para una sencilla utilización. En este caso formalizamos una

base de datos basada en el modelo de negocio de “***DragonVall****”*.

# Diseño de la base de datos

## Diseño conceptual

### Especificación de requisitos

R1. Se desea almacenar a los niños/usuarios del parque con un nombre completo, alergia,

fecha de nacimiento, teléfono y email.

R2. Se desea almacenar de los eventos el nombre de la persona que lo organiza, el nombre del evento, su fecha de reserva, fecha en la que se va a organizar el evento, y las personas que acudirán al evento.

R3. Se desea almacenar un número de taquilla asociado a un niño para cada vez que un niño accede al recinto y este es obligatorio.

R4. Se desea guardar al personal como equipo de cocina o monitores y con cada ejemplar perteneciendo a solo uno de los grupos, almacenando, su nombre, teléfono, email, horario, DNI y dirección

R5. Se desea almacenar qué menú escoge cada niño cada vez que va al recinto, es decir cada vez que accede.

R6. Se puede ir a más de dos eventos en el mismo día, y solo se puede escoger un menú. R7. Un niño puede ir a la zona común, aunque no haya ningún evento.

R8. Se puede compartir un evento entre dos personas, es decir un evento puede estar organizado para 1 o más personas.

R9. Una persona organizadora del evento puede contratar por su cuenta otras atracciones (payasos, magos u otras atracciones extras), de estas se desea almacenar el nombre completo, el email, el teléfono y la dirección de la empresa contratada.

R10. Si es un cumpleaños, se añadirá al menú una tarta.

R11. Un usuario que sea menor de edad deberá tener a otro como responsable.

R12. Se necesitan al menos 10 niños para organizar un cumpleaños.

R13. Si asisten 20 o más niños se ofrece una tarta de chuches o bizcocho.

R14. El precio del evento depende del tipo de menú que el usuario escoge.

Relacionado con esto se desea almacenar el nombre del menú y el precio por usuario.

R15. Depende de la edad del usuario accede a zonas de juego distintas:

1. Si su edad se comprende entre 18 meses y 3 años accede al *baby-park* con la compañía de un usuario adulto (mayor de edad).
2. Si su edad se comprende entre 3 años y 12 años accede al parque grande.

R16.El equipo de cocina engloba a camareros y cocineros, pero no interesa conocer su función en la cocina.

### Modelo Entidad-Relación

Diagrama, Esquemático

Descripción generada automáticamente

Ilustración 1

## Diseño lógico

### Modelo relacional

### Diagrama, Esquemático Descripción generada automáticamente

Ilustración 2

### Normalización

**1FN:** Para que una tabla esté en primera forma normal tiene que cumplir 3 requisitos:

* + Cada columna es atómica.
  + Un solo valor en cada fila.
  + No hay grupos repetitivos.

**2FN:** Para que pase a estar en segunda forma normal tiene que estar en primera forma normal y que cada columna no clave dependa funcionalmente de forma completa de cualquiera de las claves.

**3FN:** Para que esté en tercera forma normal tiene que estar en segunda forma normal y que ningún atributo que no sea clave dependa transitivamente de las claves de la tabla.

**SOLUCIÓN**

**1FN:** La tabla **USUARIO** en la columna de **alergias\_usuario** no tiene un solo valor en cada fila, debido a que se puede descomponer en varias alergias.

La solución consiste en crear una nueva tabla **ALERGIA\_USUARIO** para separar los datos. Además, se crearía una nueva tabla de la relación llamada **ES\_ALERGICO** ya que un usuario puede tener varias alergias y las alergias pueden pertenecer a varios usuarios.

USUARIO(dni\_usuario(PK), nombre\_usuario, telefono\_usuario, correo\_usuario, direccion\_usuario, alergias\_usuario, fecha\_nacimiento\_usuario)

La tabla USUARIO pasa a ser:

USUARIO(dni\_usuario(PK), nombre\_usuario, telefono\_usuario, correo\_usuario, direccion\_usuario, fecha\_nacimiento\_usuario)

Se crea la nueva tabla ALERGIAS:

ALERGIAS(id\_alergia(PK), nombre\_alergia)

Y una tabla de la relación (varios a varios)

ES\_ALERGICO(dni\_usuario(FK)+id\_alergia(FK)(PK))

**1FN:** La tabla **USUARIO** en la columna de **nombre\_usuario** y **direccion\_usuario** no son atómicas ya que se pueden descomponer.

* + **nombre\_usuario:** nombre\_usuario y direccion\_usuario
  + **direccion\_usuario:** nombre\_calle, numero\_calle y detalles\_calle

La solución consiste en separar los datos en nuevos campos.

Esta solución es aplicable a todos los campos de la base de datos donde haya nombre de personas y direcciones, resolviéndolo de manera análoga.

USUARIO(dni\_usuario(PK), nombre\_usuario, telefono\_usuario, correo\_usuario, direccion\_usuario, fecha\_nacimiento\_usuario)

La tabla USUARIO pasa a ser:

USUARIO(dni\_usuario(PK), nombre\_usuario, apellidos\_usuario, telefono\_usuario, correo\_usuario, nombre\_calle, numero\_calle, detalles\_calle, fecha\_nacimiento\_usuario)

## Diseño físico

### Modelo físico

### 

Ilustración 3

### Script de creación de la BD

### Script SQL, realizado en PD mediante el modelo Físico, y almacenado en la carpeta **SQL**

### Diccionario de datos

ACCESO: es la tabla de los usuarios que acceden a una zona de juego sin asistir a ningún evento.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| numero\_acceso | Number | PK |
| numero\_taquilla | Number | FK -> taquilla (not null) |
| id\_zonajuego | Number | FK -> zona\_juego (not null) |
| dni\_usuario | varchar2 | FK -> dni\_usuario (not null) |
| id\_menu | Number | FK -> menu |
| precio\_acceso | Number | Al que se le tiene que añadir el precio del menú (si escoge uno) |
| fecha\_acceso | date |  |

ANIMACION: es la tabla de las personas que animan en los eventos.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| dni\_animador | varchar2 | PK |
| nombre\_animador | varchar2 |  |
| teléfono\_animador | Number |  |
| dirección\_animador | varchar2 |  |

ASISTE: es la tabla de la relación entre USUARIO y EVENTO (un usuario puede asistir a múltiples eventos y a un evento asisten varios usuarios).

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| dni\_usuario | varchar2 | PK, FK -> usuario |
| id\_evento | Number | PK, FK -> evento |

EVENTO: es la tabla que representa los eventos que se realizan en el centro.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| id\_evento | Number | PK |
| id\_menu | Number | FK -> menu (not null) |
| nombre\_evento | varchar2 |  |
| fecha\_evento | date |  |
| numero\_sala | Number | No es obligatorio que un evento esté asociado a una sala |
| es\_cumple | boolean | Si es un cumpleaños, se administrará un suplemento que está almacenado en la tabla menu |

INCLUYE: es la tabla de la relación entre EVENTO y ANIMADOR (un evento puede tener, o no, uno o varios animadores y los animadores van a distintos eventos).

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| id\_evento | Number | PK, FK -> evento |
| dni\_animador | varchar2 | PK, FK -> animador |

MENU: la tabla que almacena los distintos menús que se pueden pedir. Cabe destacar el menú tipo 0, que es el que se otorga a los eventos que no escogen un menú en específico.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| id\_menu | Number | PK |
| nombre\_menu | varchar2 |  |
| precio\_menu | Number | Por persona. |

PERSONAL: es la tabla que almacena el personal del centro. En el caso de los monitores (puesto: ‘MONITORES’) estos se asocian a una zona de juego.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| dni\_personal | varchar2 | PK |
| nombre\_personal | varchar2 |  |
| telefono\_personal | Number |  |
| direccion\_personal | varchar2 |  |
| correo\_personal | varchar2 |  |
| puesto\_personal | varchar2 |  |

RESERVA: es la tabla que almacena las reservas que se han hecho.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| id\_reserva | number | PK |
| dni\_usuario | varchar2 | FK -> usuario (not null) |
| id\_evento | varchar2 | FK -> evento (not null) |
| fecha\_reserva | date |  |

TAQUILLA: es la tabla que almacena las taquillas que hay en el centro.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| numero\_taquilla | number | PK |

TIENE: es la tabla que relaciona usuarios con sus tutores.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| dni\_usuario | varchar2 | PK, FK -> usuario (tutor) |
| usu\_dni\_usuario | varchar2 | PK, FK -> usuario (tutelado) |

TRABAJA: es la tabla de la relación entre personal y la zona de juego donde trabajan.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| id\_zonajuego | number | PK, FK -> zona\_juego |
| dni\_personal | varchar2 | PK, FK -> personal |

USUARIO: es la tabla que almacena los usuarios del centro.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| dni\_usuario | varchar2 | PK |
| nombre\_usuario | varchar2 |  |
| teléfono\_usuario | number |  |
| correo\_usuario | varchar2 |  |
| dirección\_usuario | varchar2 |  |
| alergias\_usuario | varchar2 |  |
| fecha\_nacimiento\_usuario | date |  |

ZONA\_JUEGOS: es la tabla donde se almacena la información de las zonas de juego del centro.

|  |  |  |
| --- | --- | --- |
| Nombre columna | Tipo de dato | Propiedades |
| id\_zonajuego | number | PK |
| nombre\_zonajuego | varchar2 |  |

# Consultas SQL

Las consultas estas almacenadas en una carpeta denominado **SQL.**

# Subprogramas PL/SQL

Los subprograms PL/SQL realizados están localizados en la capeta denominada **PL SQL.**

# Recursos empleados / necesarios

* **Sistemas Operativo:** Windows 10 Enterprise.
* **SGBD**: Oracle Database 19c.
* **Software**
  + **Oracle 19c:** Para la creación de las tablas y para la administración de la base de datos.
  + **Oracle Live SQL**: Para la realización y ejecución de las consultas.
* **Arquitectura cliente-servidor.**
* **Requerimientos técnicos:** Para instalar la BBDD tanto en Windows como en Linux, como mínimo deberemos disponer de:
  + 4GB de RAM.
  + 1,4 GHz de procesador.
  + 20GB de espacio en disco.
  + Si es Linux, se necesitará que tenga terminal o GUI (Graphics User Interfaces).

# Temporalización

|  |  |  |
| --- | --- | --- |
| **EDUARDO** | | |
| **Dia** | **Duración** | **Actividad realizada** |
| **08/10/2022** | 2 horas | Realización de los requisitos del proyecto y del modelo E-R |
| **15/10/2022** | 2 horas | Realización de los requisitos del proyecto y del modelo E-R |
| **24/10/2022** | 3 horas y 30 minutos | Reunión de mejora del modelo entidad relación y dudas |
| **05/12/2022** | 2 horas | Realización y ayuda del modelo Entidad Relación en Power Designer |
| **24/02/2023** | 1 hora | Realización de consultas SQL (tipo multitabla) |
| **15/03/2023** | 1 hora | Actualizar las imágenes de los modelos, realizar las consultas |
| **21/05/2023** | 2 horas | Realizar las consultas restantes, PowerPoint, y normalización |
|  |  |  |
| **MARIO** | | |
| **Dia** | **Duración** | **Actividad realizada** |
| **08/10/2022** | 2 horas | Realización de los requisitos del proyecto y del modelo E-R |
| **15/10/2022** | 2 horas | Realización de los requisitos del proyecto y del modelo E-R |
| **24/10/2022** | 3 horas y 30 minutos | Reunión de mejora del modelo entidad relación y dudas |
| **05/12/2022** | 2 horas | Realización y ayuda del modelo Entidad Relación en Power Designer |
| **24/02/2023** | 1 hora | Realización de consultas SQL (tipo agrupadas) |
| **15/03/2023** | 1 hora | Realizar las consultas |
| **21/05/2023** | 2 horas | Realizar las consultas restantes, PowerPoint, y normalización |
|  |  |  |
| **NOEL** | | |
| **Dia** | **Duración** | **Actividad realizada** |
| **08/10/2022** | 2 horas | Realización de los requisitos del proyecto y del modelo E-R |
| **15/10/2022** | 2 horas | Realización de los requisitos del proyecto y del modelo E-R |
| **24/10/2022** | 3 horas y 30 minutos | Reunión de mejora del modelo entidad relación y dudas |
| **05/12/2022** | 1 hora | Realización y ayuda del modelo Entidad Relación en Power Designer |
| **24/02/2023** | 1 hora | Realización de consultas SQL (tipo subconsultas) |
| **15/03/2023** | 2 horas | Realizar las consultas |
| **21/05/2023** | 2 horas | Realizar las consultas restantes (update, insert y PL SQL, PowerPoint) y comprobar el Script |