



Final Project

Professor: Daniel Perez
Advanced Database Systems
Instituto Tecnológico y de Estudios Superiores de Monterrey Campus
Puebla

February - June 2021
21/05/2021

Agustín García Rodríguez	A01378823
José Eduardo Hernández Morales	A01732165
Dylan Ordaz Zamora	A01327199

Introduction

In the following project a distributed database system will be developed with a graphical interface shown in the form of a web application running in a linux environment, for this it will make use of two DBMS, mysql and postgresql, however the user should not be able to distinguish from which place the information is obtained

Problem definition

Our problem is obtained by thinking about the needs that a dental office may have as it grows, managing the information of a large number of patients, their types of treatment, the daily dates, the information of each of the dentists who work there, as well as payment information and various dental offices distributed in different locations. It's believed that a database system can facilitate each of these processes, at the same time that we apply in a practical way the concepts seen throughout the course

Concepts used in the Project

- **DBMS (Database Management System):**

It is a software used to optimize and manage the storage, definition and retrieval of the data of a database, as well as provides a centralized view of data that can be accessed by multiple users, from multiple locations, in a controlled manner

- **Bootstrap:**

It is considered the most popular framework for developing responsive websites, which allows you to build websites more easily and simply, since it has a large amount of reusable HTML, CSS and JavaScript code.

- **Framework:**

It is a platform for the development of applications, this can include classes or predefined and reusable functions that can help when developing to not have to start a project from scratch

- **JavaScript:**

It is an object-oriented programming language, usually used to give some functionality to a website such as video players, animations, maps. It is complemented in the frontend part along with CSS and HTML

- **HTML (Hypertext Markup Language):**

It is a markup language used to structure and define the content of a web page, for example, defining paragraphs, headings and data tables, or inserting images and videos on the page.

- **CSS (Cascading Style Sheets)**

CSS is a style language used to describe the presentation of HTML or XML documents. Through this language it is possible to give appearance to a website, with options such as adding shapes, colors, as well as the way in which the elements are displayed

- **PostgreSQL:**

The most advanced open-source DBMS. Part of our database is stored in this manager. Postgres uses relational-object models.

- **MySQL:**

It is an open-source relational database management system (RDBMS). This is a relational model in which we are going to store a part of the project database.

- **PHP:**

It is a multi purpose scripting language especially suited to web development. We are going to use this language to connect the database with the frontend.

- **GitHub:**

It is a provider of Internet hosting for software development and version control using git.

Implementation

Create

```
34 // insert data
35 if ($valid) {
36     var_dump($_POST);
37     $pdo = Database::connect();
38     $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
39     $sql = "INSERT INTO pacientes(paciente_id, paciente_nombre, paciente_apellido, paciente_telefono, paciente_fecha_nacimiento) values(
40     $q = $pdo->prepare($sql);
41     $q->execute(array($paciente_nombre, $paciente_apellido, $paciente_telefono, $paciente_fecha_nacimiento));
42     Database::disconnect();
43     header("Location: tablePacientes.php");
44 }
45 }
```

In this part we insert a customer in the database.

Delete

```

1  <?php
2      require 'database.php';
3      $id = 0;
4      if ( !empty($_GET['id'])) {
5          $id = $_REQUEST['id'];
6      }
7
8      if ( !empty($_POST)) {
9          // keep track post values
10         $id = $_POST['id'];
11         // delete data
12         $pdo = Database::connect();
13         $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
14         $sql = "DELETE FROM pacientes WHERE paciente_id = ?";
15         $q = $pdo->prepare($sql);
16         $q->execute(array($id));
17         Database::disconnect();
18         header("Location: tablePacientes.php");
19     }
20 ?>

```

In this part we delete a customer from the database.

Read

```

33  include 'database.php';
34  include 'funcs.php';
35  $pdo = Database::connect();
36  $sql = 'SELECT paciente_id, paciente_nombre, paciente_apellido as apellido, paciente_telefono, paciente_fecha_nacimiento from pacientes';
37  foreach ($pdo->query($sql) as $row) {
38      echo '<tr>';
39      echo '<td>'. $row['paciente_nombre'] . '</td>';
40      echo '<td>'. $row['apellido'] . '</td>';
41      echo '<td>'. $row['paciente_telefono'] . '</td>';
42      echo '<td>'. calculaedad($row['paciente_fecha_nacimiento']) . '</td>';
43      echo '<td width=390>';
44      //echo '<div class="btn-toolbar" role="group">';
45      echo '<a class="btn btn-outline-info btn-sm" onClick="loadDynamicContentModal('. $row['paciente_id']. ')">Cuadro Clinico</a>';
46      echo '&nbsp;';
47      echo '<a class="btn btn-outline-light-green btn-sm" href="actPaciente.php?id=' . $row['paciente_id']. '">Actualizar</a>';
48      echo '&nbsp;';
49      echo '<a class="btn btn-outline-danger btn-sm" href="deletePaciente.php?id=' . $row['paciente_id']. '">Eliminar</a>';
50      //echo '</div>';
51      echo '</td>';
52  echo '</tr>';
53 }
Database::disconnect();
54 ?>

```

Here we retrieve the information from the database so the user can read it on the website.

Update

```

34 // insert data
35 if ($valid) {
36     var_dump($_POST);
37     $pdo = Database::connect();
38     $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
39     $sql = "INSERT INTO pacientes(paciente_id, paciente_nombre, paciente_apellido, paciente_telefono, paciente_fecha_nacimiento) values(40
40     $q = $pdo->prepare($sql);
41     $q->execute(array($paciente_nombre, $paciente_apellido, $paciente_telefono, $paciente_fecha_nacimiento));
42     Database::disconnect();
43     header("Location: tablePacientes.php");
44 }
45 }

```

In this part we update the information of the customer.

Trigger

```

11 CREATE OR REPLACE FUNCTION addAccount()
12 RETURNS TRIGGER
13 LANGUAGE PLPGSQL
14 AS
15 $$
16 BEGIN
17     INSERT INTO cuentaPaciente VALUES(DEFAULT, NEW.paciente_id, 0);
18
19 RETURN NEW;
20 END;
21 $$;;
22
23 CREATE TRIGGER after_pacientes_insert AFTER INSERT ON pacientes
24 FOR EACH ROW
25 EXECUTE PROCEDURE addAccount();

```

In this Trigger what it is supposed to do is to create a new cuentaPaciente every time a cuenta is inserted

Trigger

```

12 DELIMITER //
13 CREATE TRIGGER before_citas_insert
14 BEFORE INSERT ON citas FOR EACH ROW
15 BEGIN
16     set @temp = (SELECT consultorio_id from dentistas where dentista_id = NEW.dentista_id);
17     IF NEW.consultorio_id != @temp THEN
18         SET NEW.consultorio_id = @temp;
19     END IF;
20 END//;
21
22 DELIMITER ;

```

In this trigger we are going to insert information on the table citas, using a temporary variable to save the select query. With an IF statement we are going to check if the consultorio is the same as the variable temp, in case that it is different it is going to be set.

View

```

CREATE VIEW info_cita AS
SELECT CONCAT(pacientes.paciente_nombre, ' ', pacientes.paciente_apellido) AS Paciente, citas.fecha_cita,
       consultorios.consultorio_nombre, consultorios.consultorio_telefono, consultorios.consultorio_calle,
       CONCAT(dentistas.dentista_nombre, ' ', dentistas.dentista_apellido_paterno) AS Dentista,
       dentistas.dentista_telefono, citas.comentarios FROM citas
  LEFT JOIN pacientes ON citas.paciente_id = pacientes.paciente_id
  LEFT JOIN dentistas ON citas.dentista_id = dentistas.dentista_id
  LEFT JOIN consultorios ON citas.consultorio_id = consultorios.consultorio_id;

```

In this view we are going to display the information of the cita, the information that we are going to retrieve from the database is general information about the tables pacientes, dentista, and consultorio. Joining the information of these 3 tables using a view.

Function

```

41  DELIMITER $$ 
42  CREATE FUNCTION getTimeLeft(idCita int)
43  RETURNS datetime
44  DETERMINISTIC
45  BEGIN
46  DECLARE
47    dt TIMESTAMP;
48    SELECT fecha_cita INTO dt FROM citas WHERE cita_id = idCita;
49    return timediff(dt, NOW());
50 END
51 $$ 
52 DELIMITER ;

```

In this function it is to get the time left before the date of the cita arrives

Stored procedure

```

56  DELIMITER $$ 
57  CREATE PROCEDURE registraPago(IN idPaciente integer, IN metodoPagoId integer, IN amnt decimal(8,2))
58  BEGIN
59  START TRANSACTION;
60  IF amnt < 1 THEN
61  ROLLBACK;
62  END IF;
63  INSERT INTO pagos(paciente_id, metodo_pago_id, monto, fecha_pago, comentarios)
64  VALUES (idPaciente,
65    metodoPagoId,
66    amnt,
67    (SELECT CURRENT_DATE),
68    'Procedure'
69  );
70
71  UPDATE cuentaPacientes
72  SET saldo = [saldo - amnt]
73  WHERE paciente_id = idPaciente;
74  COMMIT;
75 END $$ 
76 DELIMITER ;

```

In this stored procedure it is used to register the payment from the customer.

Explanation of implementation of the distributed queries described in [3] with screenshots
Include technical details (technologies, frameworks, tools, used)

```
1  *****INSTALAR FDW*****
2
3  sudo apt-get install libmysqlclient-dev
4  sudo apt install postgresql-server-dev-12
5
6  locate pg_config
7  locate mysql_config
8
9  git clone https://github.com/EnterpriseDB/mysql_fdw.git
10 cd mysql_fdw
11
12 export PATH=/usr/lib/postgresql/12/bin:/usr/bin:$PATH make USE_PGXS=1
13 sudo PATH=/usr/lib/postgresql/12/bin:/usr/bin:$PATH make USE_PGXS=1 install
14
```

For the distributed queries it was used the tool foreign data wrappers(fdw)

localhost/consultorio/app/addPacientes.php

Registra un nuevo paciente

Nombre del paciente
Dan

Apellido
Perez

Numero de telefono (...
268976567821

Fecha de Nacimiento
1992-03-10

Elige un consulto ▾
Elige un consultorio
Cero Caries
Dientes Sanos

REGISTRAR

```
eduardo@eduardo-HP-14-Notebook-PC: ~
+-----+
|      9 |          2 | Pedro          | Garcia          | 2465324551      | 1986-07-12      |
+-----+
2 rows in set (0.00 sec)

mysql> select * from pacientes;
+-----+-----+-----+-----+-----+-----+
| paciente_id | consultorio_id | paciente_nombre | paciente_apellido | paciente_telefono | paciente_fecha_nacimiento |
+-----+-----+-----+-----+-----+-----+
|      6 |          2 | Juan          | Salmon          | 2219124367      | 1991-11-14      |
|      9 |          2 | Pedro          | Garcia          | 2465324551      | 1986-07-12      |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from pacientes;
+-----+-----+-----+-----+-----+-----+
| paciente_id | consultorio_id | paciente_nombre | paciente_apellido | paciente_telefono | paciente_fecha_nacimiento |
+-----+-----+-----+-----+-----+-----+
|      6 |          2 | Juan          | Salmon          | 2219124367      | 1991-11-14      |
|      9 |          2 | Pedro          | Garcia          | 2465324551      | 1986-07-12      |
|     15 |          2 | Juan          | Salmon          | 2219124367      | 1991-11-14      |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> 
```

Here we are introducing a new user "Dan Perez"

```

eduardo@eduardo-HP-14-Notebook-PC: ~
consultoriode> select * from pacientes;
paciente_id | consultorio_id | paciente_nombre | paciente_apellido | paciente_telefono | paciente_fecha_nacimiento
-----+-----+-----+-----+-----+-----+
 10 |           1 | Jualiana       | Diz             | 2546542214 | 1990-05-12
  7 |           1 | Juan            | Banana          | 2432443234 | 1983-06-01
  6 |           2 | Juan            | Salmon          | 2219124367 | 1991-11-14
  9 |           2 | Pedro           | Garcia          | 2465324551 | 1986-07-12
 15 |           2 | Juan            | Salmon          | 2219124367 | 1991-11-14
 17 |           2 | Dan              | Perez           | 2689765672 | 1992-03-10
(6 rows)

consultoriode> select * from consultorios;
consultoriode> select * from "pacientes-PSQL"
;
paciente_id | consultorio_id | paciente_nombre | paciente_apellido | paciente_telefono | paciente_fecha_nacimiento
-----+-----+-----+-----+-----+-----+
 10 |           1 | Jualiana       | Diz             | 2546542214 | 1990-05-12
  7 |           1 | Juan            | Banana          | 2432443234 | 1983-06-01
(2 rows)

consultoriode> select * from "pacientes-PSQL"
;
paciente_id | consultorio_id | paciente_nombre | paciente_apellido | paciente_telefono | paciente_fecha_nacimiento
-----+-----+-----+-----+-----+-----+
 10 |           1 | Jualiana       | Diz             | 2546542214 | 1990-05-12
  7 |           1 | Juan            | Banana          | 2432443234 | 1983-06-01
 18 |           1 | Marco           | Solis           | 1323382912 | 1991-11-11
(3 rows)

consultoriode> █

```

18	Cero Caries	Marco Solis	1323382912	29	EDIT	DELETE
----	-------------	-------------	------------	----	----------------------	------------------------

And in here we can see that the user is "Dan Perez" is in the database

```

consultoriode# BEGIN;
C BEGIN
consultoriode# INSERT INTO dentistasA values(DEFAULT,1,'Poncho','Jarin','2361109221','poncho@gmail.com');
B INSERT 0 1
I consultoriode# INSERT INTO dentistasB values(DEFAULT,'2131485217529931','PEGM9007151HO');
I INSERT 0 1
C consultoriode# COMMIT;
  COMMIT
B consultoriode# █

consultoriode# BEGIN;
BEGIN
consultoriode# INSERT INTO dentistasA values(DEFAULT,1,'Poncho','Jarin','2361109221','poncho@gmail.com');
INSERT 0 1
consultoriode# INSERT INTO dentistasB values(DEFAULT,'2131485217529931','PEGM9007151HO');
INSERT 0 1
consultoriode# COMMIT;
  COMMIT
consultoriode# select * from dentistasA;
dentista_id | consultorio_id | dentista_nombre | dentista_apellido_paterno | dentista_telefono | dentista_correo
-----+-----+-----+-----+-----+-----+
  1 |           1 | Poncho         | Jarin           | 2361109221 | poncho@gmail.com
  2 |           2 | Jaen           | Jori            | 6327812665 | jaen@gmail.com
  3 |           1 | Poncho         | Jarin           | 2361109221 | poncho@gmail.com
(3 rows)

consultoriode# █

```

```

eduardo@eduardo-HP-14-Notebook-PC: ~
3 rows in set (0.00 sec)

mysql> select * from pacientes;
+-----+-----+-----+-----+-----+-----+
| paciente_id | consultorio_id | paciente_nombre | paciente_apellido | paciente_telefono | paciente_fecha_nacimiento |
+-----+-----+-----+-----+-----+-----+
| 6 | 2 | Juan | Salmon | 2219124367 | 1991-11-14 |
| 9 | 2 | Pedro | Garcia | 2465324551 | 1986-07-12 |
| 15 | 2 | Juan | Salmon | 2219124367 | 1991-11-14 |
| 17 | 2 | Dan | Perez | 2689765672 | 1992-03-10 |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select * from dentistas;
+-----+-----+
| dentista_id | no_cuenta_bancaria | dentista_rfc |
+-----+-----+
| 1 | 2131485217529931 | PEGM9007151HO |
| 2 | 0087571597776431 | GAG07671211AE |
| 3 | 2131485217529931 | PEGM9007151HO |
+-----+-----+
3 rows in set (0.00 sec)

mysql>

```

```

eduardo@eduardo-HP-14-Notebook-PC: ~
INSERT 0 1
consultorioid=# INSERT INTO dentistasB values(DEFAULT,'2131485217529931','PEGM9007151HO');
INSERT 0 1
consultorioid=# COMMIT;
COMMIT
consultorioid=# select * from dentistasA;
+-----+-----+-----+-----+-----+
| dentista_id | consultorio_id | dentista_nombre | dentista_apellido_paterno | dentista_telefono | dentista_correo |
+-----+-----+-----+-----+-----+
| 1 | 1 | Poncho | Jarin | 2361109221 | ponchog@gmail.com |
| 2 | 2 | Jaen | Jori | 6327812665 | jaen@gmail.com |
| 3 | 1 | Poncho | Jarin | 2361109221 | ponchog@gmail.com |
+-----+-----+-----+-----+-----+
(3 rows)

consultorioid=# SELECT a.dentista_id, consultorio_nombre, CONCAT(dentista_nombre, ' ', dentista_apellido_paterno) as Nombre,dentista_telefono,dentista_correo, no_cuenta_bancaria, dentista_rfc from dentistasA a INNER JOIN dentistasB b ON a.dentista_id = b.dentista_id INNER JOIN consultorios c ON a.consultorio_id = c.consultorio_id;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| dentista_id | consultorio_nombre | Nombre | dentista_telefono | dentista_correo | no_cuenta_bancaria | dentista_rfc |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1 | Cero Caries | Poncho Jarin | 2361109221 | ponchog@gmail.com | 2131485217529931 | PEGM9007151HO |
| 2 | Dientes Sanos | Jaen Jori | 6327812665 | jaen@gmail.com | 0087571597776431 | GAG07671211AE |
| 3 | Cero Caries | Poncho Jarin | 2361109221 | ponchog@gmail.com | 2131485217529931 | PEGM9007151HO |
+-----+-----+-----+-----+-----+-----+-----+-----+
(3 rows)

consultorioid=#

```

Include public repository link

<https://github.com/DylanOrdaz/Proyecto-Bases-Datos.git>

Team conclusions

It was a really hard project in which we had to apply knowledge from the semester and from the internet, because we had to do a lot of research.