

# *Manual de tecnico – Formulario de Solicitud de Cita Odontológica*

---

en este proyecto implementamos un sistema web de agendas de citas para pacientes odontológicos esta página incluye:

1. solicitud de de citas en un formulario publico
2. panel administrativo de recepcionista
3. asignación a dentista registrado

## Arquitectura de archivos:

```
CLINICA/
  api/
    solicitar_cita.php      # Endpoint público para registrar citas desde el formulario

  config/
    db.php                  # Configuración de conexión PDO a la base de datos

  dentista/
    agregar_dentista.php   # Permite registrar nuevos dentistas en el sistema

  recepcionista/
    asignar_dentista.php   # Asigna un dentista a una cita pendiente
    auth_check.php          # Verifica sesión activa antes de mostrar el panel
    panel_citas.php         # Panel principal donde la recepcionista gestiona las citas

  form_citas.html          # Formulario público de solicitud de citas (clientes)
```

## Flujo General del Sistema

### 1. Cliente (Paciente)

Accede al formulario [form\\_citas.html](#) y completa sus datos personales junto con la fecha y hora deseadas.

### 2. API ([solicitar\\_cita.php](#))

Recibe los datos del formulario mediante `fetch()` en formato JSON.

Valida los campos y guarda la cita en la base de datos como *pendiente*.

### 3. Recepcionista

- Inicia sesión y accede a [recepcionista/panel\\_citas.php](#).
- Visualiza las citas pendientes.
- Puede **asignar un dentista** a cada cita desde [asignar\\_dentista.php](#).

### 4. Dentista

Puede registrarse mediante [dentista/agregar\\_dentista.php](#) (o ser registrado por el administrador).

Una vez asignado, queda vinculado a la cita del paciente.

## Archivos y Funciones Detalladas

### 1. form\_citas.html

Formulario de solicitud de cita accesible públicamente.

Contiene campos de datos personales, fecha, hora y motivo, enviados al backend mediante `fetch()` hacia [api/solicitar\\_cita.php](#).

#### ✓ Características:

- Validaciones nativas de HTML5.
- Envío asíncrono sin recargar la página.
- Mensajes dinámicos de éxito o error.
- Diseño adaptativo con Tailwind CSS.

### 2. api/solicitar\_cita.php

Archivo PHP que recibe los datos del formulario y los guarda en la base de datos.

#### Flujo interno:

1. Recibe JSON desde `php://input`.

2. Valida que los campos requeridos no estén vacíos.

3. Inserta un nuevo registro en la [tabla de citas](#).

4. Devuelve un objeto JSON indicando éxito o error.

#### código de confirmación

```
// 6. Respuesta de Éxito
echo json_encode(['success' => true, 'message' => 'Cita registrada con éxito.']);

} catch (PDOException $e) {
    $pdo->rollBack();
    error_log("Error de BD al solicitar cita: " . $e->getMessage());
    echo json_encode(['success' => false, 'message' => 'Error en el servidor al registrar la cita.']);
}
?>
```

#### 3. config/db.php

Contiene la configuración de conexión con la base de datos mediante **PDO**, con manejo de errores y codificación

```
try {
    $pdo = new PDO("mysql:host=$host;dbname=$dbname;charset=utf8", $user, $password);
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    // echo "Conexión exitosa";
} catch (PDOException $e) {
    die("Error de conexión a la base de datos: " . $e->getMessage());
}
?>
```

#### 4. dentista/agregar\_dentista.php

Permite registrar nuevos dentistas en el sistema.

##### Campos típicos:

- Nombre completo
- Especialidad

Los registros se guardan en la tabla [dentistas](#), que luego se utilizan en el panel para asignar citas.

 **Administración de Dentistas**

[Volver al Panel de Citas](#)

✓ ¡Dentista L.Joaquin agregado con éxito!

**Agregar Nuevo Dentista**

Nombre Completo	Especialidad
<input type="text" value=""/>	<input type="text" value="Ej: Ortodoncia, Cirugía Maxilofacial"/>
<b>Guardar Dentista</b>	

**Dentistas Registrados (3)**

ID	NOMBRE	ESPECIALIDAD
2	B.L.Ruiz Diaz	Odontologa
1	L.Joaquin	dentista
3	L.Joaquin	cirujano maxilofacial

## 5. recepcionista/panel\_citas.php

Panel de control para la **recepcionista**, donde puede:

- Ver todas las citas registradas.
- Asignar dentistas a las citas mediante un listado despegable.

Cada acción se comunica con el servidor mediante **Fetch API**, permitiendo actualizar la vista sin recargar la página.

**Gestión de Citas Pendientes**

CLIENTE	CONTACTO	FECHA/HORA SOL.	MOTIVO	ASIGNAR DENTISTA
joaquin leguizamon	Tel: 1231235456345 Email: joa@gmail.com	2025-10-28 @ 15:30:00	dolor de muela...	<input style="border: 1px solid #ccc; padding: 2px 10px; width: 150px; height: 20px; vertical-align: middle;" type="button" value="Selecciona Dentista"/> <span style="margin-left: 10px; border: 1px solid #ccc; padding: 2px 10px; width: 150px; height: 20px; vertical-align: middle; background-color: #c8f7e4;"></span>

## 6. recepcionista/auth\_check.php

Script de seguridad que se incluye al inicio de las páginas del panel.

### **Función principal:**

Verificar que el usuario tenga una sesión activa (recepcionista logueada).  
 En caso contrario, redirige automáticamente al login o muestra un error de acceso no autorizado.

```

$acceso_concedido = true;

if (!isset($acceso_concedido) || $acceso_concedido !== true) {
    // En producción, esto redirigiría a un login: header('Location: ../login.php');
    die("ACCESO DENEGADO. Este es un panel exclusivo. Implemente un sistema de autenticación real.");
}

require_once '../config/db.php';
?>

```

## 7. recepcionista/asignar\_dentista.php

Archivo encargado de actualizar la cita con el dentista seleccionado.

Flujo:

1. Recibe id\_cita y id\_dentista desde el panel.
2. Ejecuta un UPDATE en la tabla citas para asignar el dentista.
3. Devuelve un mensaje de confirmación o error.

```

<?php
// recepcionista/asignar_dentista.php

// Inclusión del chequeo de acceso y la conexión a la BD
include 'auth_check.php';

if ($_SERVER['REQUEST_METHOD'] !== 'POST') {
    header('Location: panel_citas.php');
    exit;
}

$cita_id = $_POST['cita_id'] ?? null;
$dentista_id = $_POST['dentista_id'] ?? null;

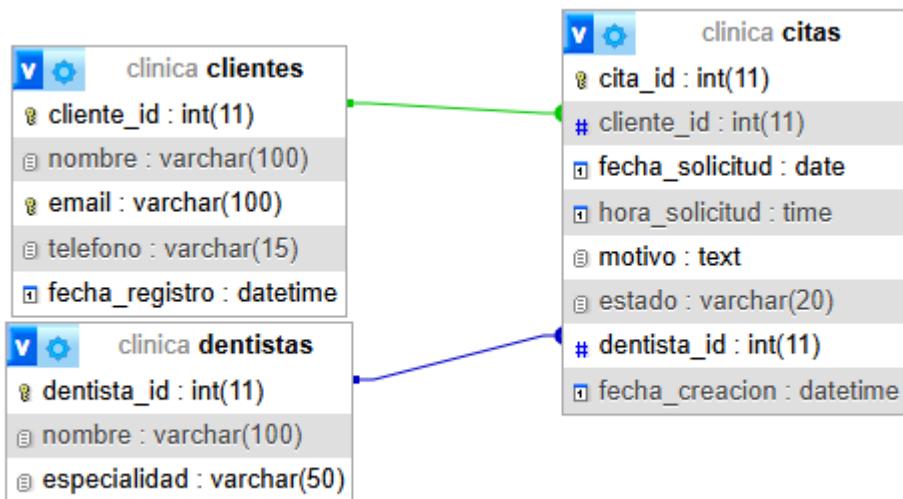
if (empty($cita_id) || empty($dentista_id)) {
    header('Location: panel_citas.php?error=' . urlencode('Error: Datos de cita o dentista incompletos.'));
    exit;
}

try {
    // 1. Actualizar la cita en la BD
    $sql = "UPDATE citas SET estado = 'Confirmada', dentista_id = :dentista_id WHERE cita_id = :cita_id AND estado = 'Pendiente'";
    $stmt = $pdo->prepare($sql);
    $stmt->execute([
        ':dentista_id' => $dentista_id,
        ':cita_id' => $cita_id
    ]);

    if ($stmt->rowCount() > 0) {
        // Éxito: Redirigir con un mensaje
        header('Location: panel_citas.php?mensaje=' . urlencode('Cita confirmada y asignada con éxito.'));
    } else {
        // Fallo: Cita no encontrada o ya procesada
        header('Location: panel_citas.php?error=' . urlencode('La cita no pudo ser confirmada (ya procesada o ID incorrecto).'));
    }
    exit;
} catch (PDOException $e) {
    // Error de BD
    error_log("Error al asignar dentista: " . $e->getMessage());
    header('Location: panel_citas.php?error=' . urlencode('Error de la base de datos: ' . $e->getMessage()));
}
?>

```

## Estructura de la Base de Datos (Resumen)



## Acceso y Roles

Rol	Descripción	Acceso
 Cliente	Solicita cita mediante <a href="#">form_citas.html</a> .	Público
 Recepcionista	Gestiona y asigna citas a dentistas.	Requiere login
 Dentista	Aparece en la lista de asignación y puede consultar sus citas.	Interno

## Ejecución Local

1. Coloca la carpeta **CLÍNICA** dentro de **htdocs** (XAMPP) o **www** (WAMP).
2. Crea la base de datos **clinica\_bd** y ejecuta el script SQL incluido.
3. Configura tus credenciales en **config/db.php**.
4. Accede desde tu navegador:

Sección	URL
 Solicitar Cita	<a href="http://localhost/CLINICA/form_citas.html">http://localhost/CLINICA/form_citas.html</a>
 Panel Recepcionista	<a href="http://localhost/CLINICA/recepcionista/panel_citas.php">http://localhost/CLINICA/recepcionista/panel_citas.php</a>

## **Mejoras Futuras**

- Sistema de autenticación completo con roles y contraseñas cifradas.
- Envío de correos automáticos de confirmación.
- Reporte de citas por día o por dentista.
- Integración con un calendario visual (FullCalendar.js).
- API REST extendida para aplicaciones móviles.