

# Manual técnico del proyecto: Proyecto-clip

GA8-220501096-AA1-EV02 módulos integrados

Harol Daniel Maldonado

Ficha: 2977342

ADSO

12 / 06 /2025

## 1. Introducción al Sistema

### Descripción General

Clip Techs Sistem es una plataforma web diseñada específicamente para la gestión de peluquerías y salones de belleza. El sistema permite automatizar procesos críticos como la reserva de citas, gestión de clientes, administración de servicios, productos y generación de reportes.

### Misión del Sistema

Brindar a los salones de belleza y peluquerías una herramienta web intuitiva, eficiente y segura que permita gestionar citas, clientes y servicios de forma automatizada, mejorando la experiencia tanto para administradores como para clientes.

### Visión del Sistema

Convertirnos en una plataforma líder en gestión de peluquerías a nivel local y regional, innovando constantemente en tecnología accesible y moderna que impulse la transformación digital del sector de belleza.

### Objetivos principales

- Automatizar la gestión de usuarios, citas y servicios
- Mejorar la organización interna del negocio
- Ofrecer una interfaz sencilla, clara y adaptable a cualquier dispositivo
- Permitir la administración segura y eficiente del sistema
- Facilitar el acceso de los clientes a servicios y reservas en línea

## Arquitectura del Sistema

```
Proyecto-clip/
├── app/                                # Núcleo de la aplicación (MVC)
│   ├── Controllers/                   # Controladores de la aplicación
│   │   ├── AdminController.php        # Control de funciones administrativas
│   │   ├── AuthController.php         # Gestión de autenticación
│   │   ├── CitaController.php         # Gestión de citas
│   │   ├── EmpleadoController.php     # Funciones específicas de empleados
│   │   ├── HomeController.php        # Controlador de página principal
│   │   ├── ServicioController.php     # Gestión de servicios
│   │   └── UserController.php        # Gestión de usuarios
│   ├── Core/                         # Componentes fundamentales del sistema
│   │   ├── Database.php              # Conexión y operaciones de BD
│   │   ├── Router.php                # Sistema de enrutamiento
│   │   ├── Session.php               # Gestión de sesiones
│   │   └── Validator.php             # Validación de formularios
│   ├── Models/                      # Modelos de datos
│   │   ├── CitaModel.php             # Modelo para gestión de citas
│   │   ├── EmpleadoServicioModel.php # Relación empleados-servicios
│   │   ├── ServicioModel.php         # Gestión de servicios
│   │   └── UserModel.php             # Gestión de usuarios
│   ├── Views/                       # Plantillas de presentación
│   │   ├── admin/                   # Vistas del panel administrativo
│   │   ├── auth/                   # Vistas de autenticación
│   │   ├── citas/                   # Vistas relacionadas con citas
│   │   ├── empleado/               # Vistas para el área de empleados
│   │   ├── servicios/              # Vistas de catálogo de servicios
│   │   ├── templates/              # Plantillas reutilizables
│   │   └── usuarios/               # Vistas de perfiles de usuario
│   ├── config.php                   # Configuración principal
│   └── config.example.php           # Configuración de ejemplo
├── documentacion/                   # Documentación del proyecto
│   ├── acta_aprobacion_requerimientos.pdf
│   ├── documentacion_modular.pdf
│   ├── manual_tecnico.pdf
│   ├── registro_pruebas.pdf
│   └── urls_sistema.txt             # Mapa de rutas del sistema
├── public/                          # Archivos públicamente accesibles
│   ├── css/                        # Hojas de estilo
│   │   ├── admin.css               # Estilos del panel administrativo
│   │   ├── main.css                # Estilos principales
│   │   └── responsive.css           # Ajustes para dispositivos
│   ├── img/                       # Imágenes estáticas
│   │   ├── logo.png                # Logo del sistema
│   │   └── backgrounds/            # Fondos e imágenes decorativas
│   ├── js/                        # Scripts de JavaScript
│   │   └── calendar.js              # Gestión de calendario de citas
│   ├── uploads/                   # Archivos subidos por usuarios
│   │   ├── profiles/               # Fotos de perfil
│   │   └── services/               # Imágenes de servicios
│   ├── .htaccess                   # Configuración de Apache
│   └── index.php                   # Punto de entrada principal
├── sql/                            # Scripts de base de datos
│   ├── schema.sql                  # Estructura de tablas
│   ├── data.sql                    # Datos iniciales
│   └── updates/                     # Scripts de actualización
└── leeme.txt                       # Instrucciones de instalación
```

Descripción detallada:

**app/** - Núcleo de la aplicación

Contiene todos los componentes principales siguiendo el patrón MVC:

- **Controllers/**: Gestionan las peticiones del usuario y coordinan el flujo
- **Core/**: Clases fundamentales para el funcionamiento del sistema
- **Models/**: Manejan el acceso a datos y la lógica de negocio
- **Views/**: Plantillas para la presentación visual
- **config.php**: Configuración del sistema (conexión a BD, rutas, etc.)
- **config.example.php**: Archivo de configuración de ejemplo para implementación

**public/** - Archivos accesibles

Único directorio expuesto públicamente en el servidor web:

- **css/**: Estilos de la aplicación
- **js/**: Funcionalidades del lado del cliente (solo calendar.js)
- **img/**: Recursos visuales estáticos
- **uploads/**: Archivos subidos por usuarios
- **index.php**: Punto de entrada que carga el sistema
- **documentacion/** - Documentación

Documentos técnicos y de usuario:

- **manual\_tecnico.pdf**: Documentación técnica completa
- **urls\_sistema.txt**: Listado de rutas disponibles
- Otros documentos del proyecto

**sql/** - Base de datos

Scripts necesarios para crear y mantener la base de datos:

- **schema.sql**: Estructura completa de tablas
- **data.sql**: Datos iniciales para el funcionamiento básico

### 3. Requerimientos Técnicos

#### Servidor

- **XAMPP:** Versión 7.4 o superior
- **Apache:** Versión 2.4 o superior
- **PHP:** Versión 7.4 o superior
- **MySQL/MariaDB:** Versión 5.7/10.4 o superior
  - Puerto configurado: 3307
- **Memoria recomendada:** 4GB mínimo
- **Espacio en disco:** 500MB mínimo (sin contar archivos de usuario)

#### Cliente

- **Navegadores compatibles:** Chrome, Firefox, Edge, Safari en sus versiones actualizadas
- **Resolución mínima:** 1280x720px (responsive para dispositivos móviles)
- **JavaScript:** Habilitado

### 4. Base de datos

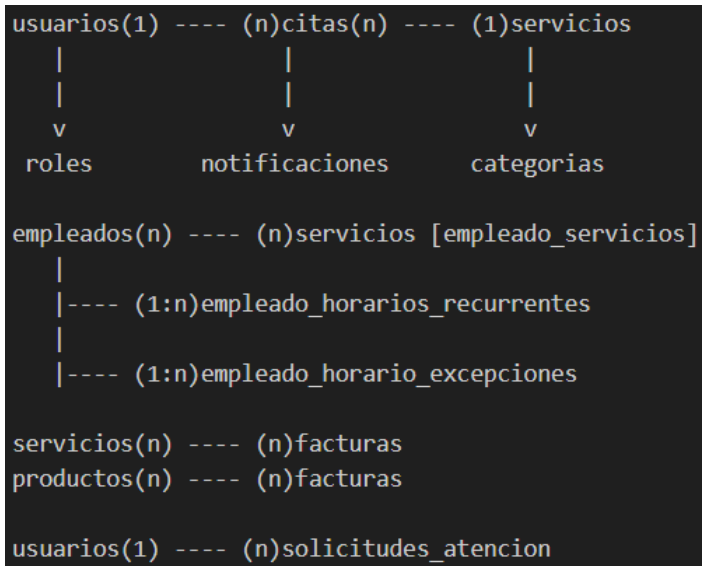
#### 4.1. Estructura General

La base de datos del sistema está compuesta por 12 tablas principales que almacenan toda la información necesaria para el funcionamiento de la aplicación:

Tabla	Registros	Tamaño	Codificación	Descripción
categorias	7	32.0 KB	utf8mb4_general_ci	Clasificación de servicios
citas	2	80.0 KB	utf8mb4_unicode_ci	Reservas de servicios
empleado_horarios_recurrentes	1	32.0 KB	utf8mb4_general_ci	Horarios regulares de trabajo
empleado_horario_excepciones	0	48.0 KB	utf8mb4_general_ci	Excepciones a horarios normales
empleado_servicios	8	48.0 KB	utf8mb4_general_ci	Relación empleados-servicios
facturas	1	80.0 KB	utf8mb4_general_ci	Transacciones económicas
notificaciones	5	32.0 KB	utf8mb4_unicode_ci	Sistema de alertas
productos	1	48.0 KB	utf8mb4_general_ci	Inventario de productos
roles	3	32.0 KB	utf8mb4_general_ci	Niveles de acceso al sistema
servicios	9	32.0 KB	utf8mb4_unicode_ci	Servicios ofrecidos
solicitudes_atencion	1	64.0 KB	utf8mb4_unicode_ci	Sistema de PQR
usuarios	3	64.0 KB	utf8mb4_general_ci	Usuarios del sistema

**Total:** 41 registros en 12 tablas (592.0 KB)

## 4.2. Diagrama Entidad-Relación



## 4.3. Descripción de Tablas Principales

### usuarios

Almacena todos los usuarios del sistema (clientes, empleados, administradores).

- **Campo clave:** id (PK)
- **Relaciones:** rol\_id (FK a roles), puede tener múltiples citas y solicitudes
- **Campos sensitivos:** password (almacenado con hash seguro)

### citas

Registra todas las citas programadas en el sistema.

- **Campo clave:** id (PK)
- **Relaciones:** cliente\_id, empleado\_id (FK a usuarios), servicio\_id (FK a servicios)
- **Estados posibles:** pendiente, completada, cancelada, reprogramada

### servicios

Catálogo de servicios ofrecidos por el salón de belleza.

- **Campo clave:** id (PK)
- **Relaciones:** categoria\_id (FK a categorías)
- **Indicadores:** precio, duración\_minutos, activo

## **empleado\_servicios**

Tabla de relación que asocia cada empleado con los servicios que puede ofrecer.

- **Campos clave:** empleado\_id, servicio\_id (PK compuesta)
- **Performance:** Indexada para búsquedas rápidas de empleados por servicio

## **5. Módulos Principales**

### **Sistema de Autenticación**

- Registro de nuevos usuarios
- Inicio de sesión con validación de roles
- Recuperación de contraseñas
- Gestión de perfiles

### **Gestión de Citas**

- Reserva de nuevas citas
- Visualización de calendario
- Gestión de cancelaciones y reprogramaciones
- Confirmación y seguimiento de estado

### **Gestión de Servicios**

- Catálogo de servicios
- Asignación de servicios a empleados
- Configuración de precios y duración

### **Gestión de Horarios**

- Configuración de horarios recurrentes
- Gestión de excepciones y disponibilidad
- Validación de conflictos

### **Atención al Cliente**

- Sistema de PQR (Peticiónes, Quejas, Reclamos)
- Seguimiento de solicitudes
- Notificaciones

## **6. Seguridad del Sistema**

### **6.1. Autenticación y Control de Acceso**

**El sistema implementa un riguroso control de acceso basado en roles:**

- Validación de sesiones activas en cada solicitud a rutas protegidas
- Verificación de permisos según el rol del usuario autenticado
- Expiración automática de sesiones por inactividad

### **6.2. Protección de Datos**

- Validación de entradas mediante la clase Validator
- Prevención de inyección SQL mediante consultas preparadas (PDO)
- Sanitización de salidas para prevenir XSS
- Hasheo de contraseñas con algoritmos seguros

### **6.3. Protección contra Ataques Comunes**

- **CSRF:** Tokens de protección en formularios
- **XSS:** Escape de datos de salida en plantillas
- **Fuerza bruta:** Limitación de intentos de inicio de sesión
- **Validación de rutas:** Protección contra acceso no autorizado

## **7. Procedimientos de Instalación**

### **7.1. Requisitos Previos**

- XAMPP 7.4 o superior
- Navegador web moderno
- Mínimo 4GB RAM y 500MB espacio en disco

### **7.2. Pasos de Instalación**

1. **Clonar o descomprimir** el código fuente en la carpeta htdocs de XAMPP

2. **Crear la base de datos** mediante importación del archivo SQL
3. **Configurar** el archivo config.php con las credenciales apropiadas

Variables de Configuración (config.php)

```
<?php

// Configuración de la base de datos

define('DB_HOST', 'localhost');

define('DB_USER', 'root');

define('DB_PASS', '');

define('DB_NAME', 'mi_base');

define('DB_PORT', 3307);


// URL base de la aplicación

define('BASE_URL', 'http://localhost/Proyecto-clip/public/');


// Directorio para subidas de archivos

define('APP_UPLOAD_DIR', dirname(__DIR__) . '/public/uploads/');


// Ruta pública para acceder a los archivos subidos

define('APP_UPLOAD_DIR_PUBLIC_PATH', 'uploads/');


// Configuración de IDs de Roles

define('ID_ROL_ADMIN', 1);

define('ID_ROL_CLIENTE', 2);

define('ID_ROL_EMPLEADO', 3);


// Configuración del modo de depuración
```



```

define('DEBUG_MODE', true);

if (DEBUG_MODE) {
    ini_set('display_errors', 1);
    ini_set('display_startup_errors', 1);
    error_reporting(E_ALL);
} else {
    ini_set('display_errors', 0);
    ini_set('display_startup_errors', 0);
    error_reporting(0);
}

// Crear el directorio de subidas si no existe
if (!is_dir(APP_UPLOAD_DIR)) {
    if (!@mkdir(APP_UPLOAD_DIR, 0775, true)) {
        $error = error_get_last();

        $errorMessage = "Error crítico: No se pudo crear el directorio de subidas: " .
APP_UPLOAD_DIR;

        if ($error !== null) {
            $errorMessage .= " - Detalles del sistema: " . $error['message'];
        }

        error_log($errorMessage);

        if (DEBUG_MODE) {
            die($errorMessage . ". Por favor, verifica los permisos de la carpeta 'public'");
        }
    }
}
}

```

?>

4. Verificar permisos de escritura en la carpeta uploads
5. Acceder a la aplicación a través de <http://localhost/Proyecto-clip/public/>

### 7.3 Credenciales Iniciales

Tipo de usuario	Email	Contraseña	Rol
Administrador	<a href="mailto:admin@cliptechs.com">admin@cliptechs.com</a>	admin123	admin
Empleado	<a href="mailto:empleado@cliptechs.com">empleado@cliptechs.com</a>	empleado123	empleado
Cliente	<a href="mailto:cliente@ejemplo.com">cliente@ejemplo.com</a>	cliente123	cliente

## 8. Mantenimiento y Desarrollo Futuro

### 8.1. Backups y Recuperación

- **Respaldo diario** de base de datos mediante script automatizado
- **Backup mensual** completo (código fuente + datos + uploads)
- **Plan de recuperación** ante desastres documentado

### 8.2. Monitoreo y Logs

- Log de errores PHP en xampp/apache/logs
- Registro de operaciones críticas en base de datos
- Monitoreo de espacio en disco de uploads

### 8.3. Ampliaciones Futuras

- Integración de pasarela de pagos
- Sistema de fidelización de clientes
- Aplicación móvil complementaria
- Módulo de inventario avanzado
- Integración con marketing por email

## 9. Soporte Técnico

Para asistencia técnica contactar al equipo de desarrollo:

- **Email:** harolmaldonado14@gmail.com
- **Teléfono:** +57 301 767 8950
- **Horario de atención:** Lunes a Viernes 8am - 6pm