

Resumen del Sistema MiClinica

1. Estructura General del Proyecto

El sistema MiClinica está desarrollado siguiendo el patrón de arquitectura Modelo-Vista-Controlador (MVC), lo que permite una separación clara de responsabilidades y facilita el mantenimiento del código.

Organización de Carpetas

- **model/**: Contiene las clases que interactúan con la base de datos y definen la lógica de negocio.
- **view/**: Contiene las plantillas y archivos de presentación que se muestran al usuario.
- **controller/**: Contiene los controladores que procesan las solicitudes del usuario y coordinan el modelo y la vista.
- **ajax/**: Contiene archivos para manejar solicitudes asíncronas.
- **api/**: Implementación de API REST para integración con otros sistemas.
- **config/**: Archivos de configuración del sistema.
- **vendor/**: Dependencias de terceros gestionadas por Composer.

Flujo de Trabajo MVC

1. El usuario realiza una solicitud a través del navegador
2. El archivo `index.php` recibe la solicitud y la dirige al controlador apropiado
3. El controlador procesa la solicitud, interactúa con el modelo si es necesario
4. El modelo realiza operaciones en la base de datos y devuelve los resultados
5. El controlador pasa los datos a la vista
6. La vista genera la presentación HTML que se envía al navegador

2. Sistema de Autenticación y Gestión de Usuarios

El sistema implementa un mecanismo de autenticación basado en sesiones PHP para controlar el acceso a las diferentes áreas de la aplicación.

Características Principales

- Inicio de sesión seguro con validación de credenciales
- Registro de nuevos usuarios
- Control de acceso basado en roles
- Gestión de sesiones con PHP
- Protección contra ataques comunes (CSRF, XSS, inyección SQL)

Flujo de Autenticación

```
// Verificación de sesión en template.php
if (isset($_SESSION["iniciarSesion"]) && $_SESSION["iniciarSesion"]
== "ok") {
    // Usuario autenticado - mostrar interfaz de administración
} else {
    // Usuario no autenticado - mostrar interfaz pública o login
}
```

3. Módulo de Personas/Pacientes

Este módulo gestiona la información de pacientes y personas relacionadas con la clínica, permitiendo el registro, consulta, actualización y eliminación de datos.

Funcionalidades

- Registro de datos personales y médicos
- Historial médico de pacientes
- Gestión de consultas médicas
- Programación de citas
- Carga y gestión de archivos relacionados con pacientes

Estructura de Datos

Tabla	Descripción
personas	Almacena información básica de personas (pacientes, médicos, etc.)
consultas	Registra las consultas médicas realizadas
citas	Gestiona las citas programadas
archivos	Almacena referencias a documentos y archivos subidos al sistema

4. API REST y Controladores

El sistema incluye una API REST que permite la integración con otros sistemas y aplicaciones móviles, facilitando el acceso a los datos de forma estructurada.

Estructura de la API

- **api/controllers/**: Controladores específicos para la API
- **api/models/**: Modelos utilizados por la API
- **api/routes/**: Definición de rutas y endpoints
- **api/core/**: Componentes centrales como Router, Response, etc.

Endpoints Principales

```
// Ejemplos de endpoints disponibles
GET /api/patients - Obtener lista de pacientes
GET /api/patients/{id} - Obtener datos de un paciente específico
POST /api/patients - Crear nuevo paciente
PUT /api/patients/{id} - Actualizar datos de un paciente
DELETE /api/patients/{id} - Eliminar un paciente

GET /api/users - Obtener lista de usuarios
POST /api/register - Registrar nuevo usuario
```

5. Integración con AdminLTE 3.2.0

El sistema utiliza AdminLTE 3.2.0 como framework de interfaz de usuario, proporcionando una experiencia moderna y responsive para los usuarios.

Componentes Utilizados

- Plantillas de diseño responsive
- Componentes de UI como tablas, formularios, modales, etc.
- Plugins integrados (Select2, Summernote, Dropzone, etc.)
- Sistema de navegación con sidebar y navbar
- Temas y personalización visual

Estructura de Plantillas

```
// Estructura básica de plantillas
- template.php (Plantilla principal)
- nav/navbar.php (Barra de navegación superior)
- nav/sidebar.php (Menú lateral)
- nav/footer.php (Pie de página)
- modules/ (Módulos específicos de contenido)
```

6. Sistema de Enrutamiento

El sistema implementa un mecanismo de enrutamiento simple basado en parámetros GET para navegar entre diferentes módulos y funcionalidades.

Funcionamiento

```
// Ejemplo de enrutamiento en template.php
if(isset($_GET["ruta"])){
    if ($_GET["ruta"] == "home" || $_GET["ruta"] == "logout" ||
        $_GET["ruta"] == "consultas" || $_GET["ruta"] ==
"personas") {
        include "modules/".$_GET["ruta"].".php";
    } else {
        include "view/modules/404.php";
    }
} else {
    include "view/modules/home.php";
}
```

Rutas Principales

- **home:** Página principal del sistema
- **login:** Página de inicio de sesión
- **register:** Registro de nuevos usuarios
- **consultas:** Gestión de consultas médicas
- **personas:** Administración de personas/pacientes
- **start:** Página de inicio pública
- **web-servicios:** Página de servicios pública

