

Soy Richard Rodrigo y les voy a presentar el caso de estudio de un sistema de gestión odontológica para implementarse en un consultorio odontológico privado cuyo nombre es M&M.

Este consultorio está situado en calle Miguel David, está a cargo de 2 profesionales odontológicos, cuenta con un solo consultorio compartido por ambos profesionales.

Los pacientes pueden solicitar turnos telefónicamente o presencialmente y estos son anotados en una agenda.

Si el paciente asiste por primera vez a una consulta, se registran sus datos personales en una ficha personal.

Luego se hace una revisión de toda la dentadura, si no fue por una urgencia.

Esta información se anota en el odontograma del paciente.

Cada procedimiento llevado a cabo es registrado en una tabla

La columna Código se utiliza para las Obras Sociales

En ocasiones se realizan radiografías odontológicas o fotografías dentales.

La ficha personal, el odontograma y las imágenes se guardan en un folio del paciente. Conforman la historia clínica del paciente.

El paciente puede pedir una copia de esa historia clínica.

El organismo no lleva ninguna estadística de las prestaciones realizadas o de los ingresos.

La solución propuesta es un sistema para la gestión de las consultas odontológicas.

El lenguaje de programación: Python.

Instalar las librerías: matplotlib, reportlab, Pillow (PIL)

Librerías: tkinter, re, sqlite3, os,

- **shutil**: el módulo de utilidades de shell, manipulación de archivos y directorios.
- **platform.system()**. Detectar el sistema operativo
- **subprocess**: Ejecutar programas y comandos externos desde Python. Abrir carpeta
- **ReportLab**: genera documentos PDF de manera programática.
- **datetime**: para manejar fechas, horas e intervalos de tiempo.
-

LOGIN

Comprueba BD. Crea BD si no existe. Comprueba el usuario y contraseña. También el tipo de usuario.

No acepta caracteres que no sean alfanuméricos

```
pattern = r'^[A-Za-z_]+$'
```

```
pattern = r'^[A-Za-z0-9_]+$'
```

UTIL - CONFIG

define las fuentes y los colores de fondo

tiene las funciones de validar contraseña y email

```
def leer_imagen(nombre_archivo, tamaño, mantener_proporciones=True)
```

Esta función permite dimensionar la imagen proporcionalmente y confirmar que se pudo cargar la imagen

```
def centrar_ventana(ventana, aplicacion_ancho, aplicacion_largo):  
esta función permite centrar cualquier ventana de tkinter
```

CONEXION

comprueba la BD, conecta la BD, obtiene el cursor, cierra la BD

Tiene 2 clases decorador para cuando cambiamos de base de datos:

- `cerrar_todas_conexiones` es un patrón de gestión de recursos que implementa un cierre seguro y controlado de múltiples conexiones a base de datos.
- `reconectar_todas`: Reconectar todas las instancias de la clase a la base de datos después de un cambio en la misma, cerrando y reabriendo las conexiones existentes.

Tiene la función de crear la base de datos cuando no existe, agregando el usuario admin.

BACKUP

- Crea la tabla
- Busca la carpeta del script para buscar las BD
- `eliminar_backup`: no deja eliminar la BD principal, elimina la BD seleccionada de la lista
- Carga el backup reemplazando la BD principal por la BD seleccionada, luego cierra la app.
- Tiene una función para comprobar la integridad de la BD.

- Crea el backup con el nombre de la BD principal y le agrega al fecha y hora (debe pasar 1 minuto para crear una nueva)
- Hace la lista de BD, con el nombre y la fecha
- Tiene una función para determinar la BD seleccionada

Tiene la función `verificar_y_actualizar_conexiones()` que no se llama en el mismo script

PRINCIPAL

Crea la interfaz gráfica

Contiene el menú principal que es desplegable

Carga las diferentes pestañas del menú

Llama a los distintos módulos: usuarios, pacientes, etc.

Hay módulos que se cargan en este script porque fue lo primero que se hizo, otros se carga la interfaz desde su script.

- Contiene la función para abrir la AYUDA. Es un archivo chm (formato de archivo de Ayuda HTML Compilada). Creado con HelpNDoc

USUARIO

- Crea la ventana de usuario
- Guarda o modifica usuarios con controles de nombre y contraseña válidos. Cuando se actualiza, carga los datos del usuario pero no deja modificar el tipo de usuario.
- Elimina el usuario desde el menú principal

ODONTÓLOGO

- Crea la ventana de odontólogo
- Guarda o modifica odontólogos con controles de nombre, apellido y matrícula válidos. Cuando se actualiza, carga los datos del odontólogo.
- Tiene controles para evitar que se repita la matrícula.
- Elimina el odontólogo desde el menú principal

PACIENTES

- Crea la ventana para agregar o actualizar el paciente
- Tiene campos obligatorios
- Tiene controles de formato de cada campo
- Comprueba que no se repita el DNI que es la PK

CALENDARIO

- Crea la interfaz del calendario
- Se marca con colores el día actual y los días con turnos asignados

DayTopLevel

Ventana con la lista de turnos para el día elegido en el calendario

Se puede cambiar de día

- Crea la ventana del turno
- Carga los odontólogos guardados
- Edita y elimina el turno

ODONTOGRAMA

- Crea la interfaz del odontograma
- Carga los datos del paciente elegido y la lista de odontólogos
- Crea el PDF
- Con Canvas se crea el odontograma completo y cada diente

GALERÍA

- Crea la interfaz gráfica de la galería
- Crea la carpeta del paciente para guardar las imágenes
- Carga las imágenes de la carpeta
- Carga los datos del paciente
- Muestra la imagen con tamaño proporcional
- Puede mover y agrandar las imágenes con el mouse
- Muestra datos de la imagen

INFORMES

- Crea la lista de informes disponibles
- Crea la ventana. Habilitando los meses y años
- Crea el gráfico de barras dependiendo del tipo de informe
- Obtiene los datos necesarios de la BD
- Crea el PDF del informe elegido
-