

SaludTinerfeñaDB



Grupo 6:

Concepción Peña, Evian
García Bautista, Alejandro
Plasencia Hernández, Edwin

Índice

Objetivos del proyecto.....	2
Contexto.....	3
Ruta /administra/.....	5
Ruta /trabajadores/.....	6
Ruta /consulta/.....	6
Ruta /paciente/.....	8

Objetivos del proyecto

El objetivo principal de este proyecto es simular la gestión de la sección de consultas de una cadena de hospitales privados que cuenta con sus centros situados a lo largo de la isla de Tenerife.

Además de la sección de consultas que será la más compleja de simular también se quiere simular la cantidad de material que hay en un hospital y su administración y uso a lo largo de las distintas plantas de cada hospital.

Para llevar a cabo este proyecto debemos realizar las siguientes tareas:

- Creación de un diagrama entidad relación que satisfaga los requisitos contemplados en el apartado de Contexto.
- Creación de un esquema relacional a partir de nuestro diagrama E-R.
- La creación de diferentes scripts que nos permitan crear la base de datos además de llenarla de información de una información básica.
- La creación de una API REST haciendo uso de Flask como se ha enseñado en las prácticas de la asignatura para poder hacer consultas a la base de datos. En esta API contaremos con distintos endpoints los cuales servirán para poder acceder a la información, se realizará el diseño de varios endpoints ya que la función de una API es hacer de intermediario entre la aplicación y la base de datos, es por eso que hemos decidido diseñar los distintos endpoints para simular varias sesiones es decir, tendremos las siguientes rutas:
 - Una para tener toda la información sobre los hospitales y su stock de material (/administra).
 - Una en donde accedemos a la información de los trabajadores (/trabajadores).
 - Una donde accedemos a todo lo relacionado con las consultas (/consultas).
 - Una donde se simule las acciones de para pacientes, pedir cita, anular, dar alta o baja por ejemplo (/paciente).

Contexto

Como se ha mencionado en el apartado anterior el objetivo de este proyecto es que con esta base de datos se espera poder gestionar los datos de los pacientes, trabajadores y los propios hospitales de una cadena de hospitales privados situados a lo largo de la isla de Tenerife.

Las bases de datos de los hospitales y centros médicos pueden ser muy complejas debido a la cantidad de información que se guarda y lo importante que es tenerlas bien organizadas para agilizar el trabajo de los empleados, es por eso que hemos decidido, para este proyecto, gestionar toda la parte de consultas, ya que creemos que es la más completa debido a que no solo cuenta con consultas si no con citas para futuras consultas, además contamos con que cada hospital tiene distintas plantas que hacen uso de distinto material médico el cual debe estar bien gestionado.

Para llevar a cabo el desarrollo correcto de este proyecto debemos conocer primero qué aspectos se deben tener en cuenta para nuestro proyecto:

- Como hemos indicado anteriormente vamos a modelar la gestión de una cadena de hospitales, es por eso que necesitamos tener un control de todos los hospitales que pertenecen a esta cadena, así como sus plantas, material, pacientes y empleados.
- Necesitamos tener una ficha clínica donde se guarden los datos de los pacientes.

Contamos con dos posibilidades en cuanto a los pacientes ya que pueden ser:

- Pacientes que cuentan con un seguro privado.
- Pacientes que no tienen seguro privado.
- Como solo estamos modelando el área de consultas debemos tener en cuenta que aquí principalmente trabajan los médicos y enfermeras.
- Los trabajadores podrán tener consultas en como máximo 2 centros a la vez, siempre y cuando la suma del número de horas en ambos centros sea como máximo 40 horas semanales y en cada contrato trabajar mínimo 5 horas semanales.

- Todos los centros de esta cadena hospitalaria cuentan con material para poder atender a los pacientes ya sean mascarillas, kits para poner puntos, medicación, etc. Además se deberá llevar a cabo el control de stock de los materiales.

Estas tablas que se muestran a continuación resumen los distintos endpoints que tratan de imitar una situación de la vida real de la cual venimos hablando a lo largo de este documento. El objetivo de esta tabla es mostrar las acciones que se pueden llevar a cabo en nuestra API a modo de una pequeña documentación. Es muy importante conocer que para algunas operaciones de POST, PATCH o DELETE hay que enviar un body en la petición en donde hay que especificar los campos de la consulta.

Ruta /administra/

Verbo	Ruta	Funcionalidad
GET	/administra/hospital	Vemos información sobre todos los hospitales
GET	/administra/hospital/<int:id_hospital>	Muestra información sobre un hospital
GET	/administra/stock_hospital/<int:id_hospital>	Muestra el stock de un único hospital
GET	/administra/material	Muestra todo el material que puede tener un hospital
POST	/administra/material/<int:id_material>/hospital/<int:id_hospital>/cantidad/<int:cantidad>	Añade un material a un hospital
POST	/administra/uso	Se hace uso de un material
PATCH	/administra/actualizar	Actualiza el stock

Ruta /trabajadores/

Verbo	Ruta	Funcionalidad
GET	/trabajadores	Devuelve todos los trabajadores activos
GET	/trabajadores/<int:colegiado>	Devuelve un trabajador específico
GET	/trabajadores/hospital/<int:id_hospital>	Devuelve todos los trabajadores de un hospital
GET	/trabajadores/<int:colegiado>/contratos	Devuelve todos los contratos de un solo trabajador
POST	/trabajadores/<int:colegiado>/contratos	Crea un contrato
PATCH	/trabajadores/<int:colegiado>/contratos	Modifica el contrato del trabajador especificado
DELETE	/trabajadores/<int:colegiado>	Elimina un trabajador
POST	/trabajadores/nuevo	Añade un nuevo trabajador

Ruta /consulta/

Verbo	Ruta	Funcionalidad
GET	/consulta/<int:colegiado>/paciente/<int:historia_clinica>	Devuelve todas las consultas realizadas por un trabajador a un paciente
GET	/consulta/paciente/<int:historia_clinica>	Devuelve todas las consultas realizadas a un paciente
POST	/consulta/nueva	Añade una nueva consulta
PATCH	/consulta/<int:colegiado>/paciente/<int:historia_clinica>/fecha/<date:fecha>	Se modifica una consulta de un paciente

Ruta /paciente/

Verbo	Ruta	Funcionalidad
GET	/paciente	Nos devuelve todas las citas futuras
GET	/paciente/todos/<string:tipo>	Nos devuelve a todos los pacientes de un tipo de seguro
GET	/paciente/<int:historia_clinica>	Nos devuelve las citas futuras de un paciente
GET	/paciente/pasadas/<int:historia_clinica>	Nos devuelve las citas pasadas de un paciente
POST	/paciente/nueva_cita	Añade una cita para un paciente en una fecha con un trabajador
POST	/paciente/alta	Añade un paciente a la base de datos
DELETE	/paciente/baja/<int:historia_clinica>/<string:tipo>	Elimina un paciente de la base de datos dada su historia clínica y el tipo de seguro
DELETE	/paciente/nueva_cita	Elimina la cita de un paciente para una fecha con un trabajador
PATCH	/paciente/nueva_cita	Nos permite modificar una cita de un paciente con un profesional