

Trabajo Practico N.º 4

Juan Manuel García 2º C

Atención al Cliente Clínica Odontológica

El proyecto genera un acceso público para los clientes y un acceso privado acciones administrativas.

Desde el acceso público los clientes pueden reservar un turno. El mismo les muestra los turnos disponibles para la fecha seleccionada y deben ingresar el tipo de turno que desean de una lista de tipos, su nombre, DNI y número de teléfono.

Desde el acceso privado, primero uno debe ingresar un usuario y contraseña. En el mismo uno puede:

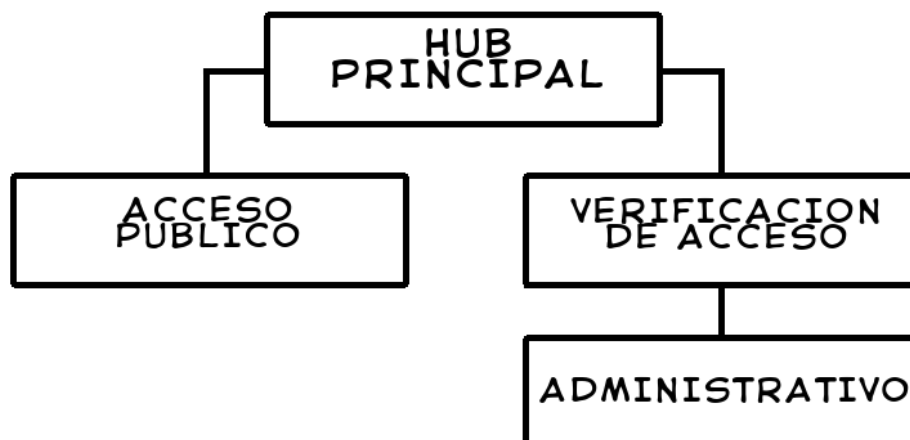
- Ver los turnos solicitados en las distintas fechas.
- Modificar un turno.
- Eliminarlo un turno.
- Indicar que fue atendido, el cual se guarda en su respectiva sección.
- Generar nuevos turnos con los datos correspondientes.
- Ver el historial de los turnos que ya fueron atendidos.
- Ver la recaudación según la fecha seleccionada, la última semana, mes y año.
- Modificar los precios de los distintos tratamientos
- Exportar los turnos a un archivo
- Obtener los turnos almacenados en un archivo seleccionado.

La base del proyecto es la clase Turno, la cual maneja la información respecto a los turnos de la clínica y la clase ManejadorDeTurnos que se encarga interactuar con la base de datos SQL que almacena los turnos pendiente y el historial ya atendido, de leer y guardar los turnos en archivos, poder obtener que horarios están disponibles, obtener turnos de un día específico, modificar, agregar y eliminar turnos.

Usuario: Odontologia

Contraseña: SatoshiNakamoto

Posee la siguiente estructura:



Utilizacion de los temas de las respectivas clases de la materia:

- Tema 10 – Excepciones:
La clase DatosInvalidosException genera una excepcion nueva.
La misma es lanzada por Turno cuando se intentan crear un Turno con datos invalidos
Tambien es lanzada por ManejadorDeTurnos al intentar asignar un precio negativo o invalido.
- Tema 11 – Pruebas Unitarias:
Un proyecto de pruebas unitarias llamdo TestUnitarios.
En el mismo se realizan pruebas a metodos de las clases Turno, ManejadorDeTurnos y Autenticador.
- Tema 12 – Tipos Genericos | Tema 13 – Interfaces:
La interfaz IArchivoDTO utiliza un tipo generico T que debe ser una clase.
Posee un metodo que toma un objeto de tipo T, con el fin de que deba ser convertido a un DataTransferObject para poder ser serializado sin perder el encapsulamiento en el objeto de tipo T
Posee un metodo para devolver una instancia del tipo T, cuando uno necesite leer de un DTO y volver al tipo original
- Tema 14 – Archivos | Tema 15 – Serialización:
El ManejadorDeTurnos se encargar poder de leer los turnos almacenados en un path especificado. Utiliza serializacion JSON.
El ManejadorDeTurnos se encargar de guardar la lista de turnos en el path a eleccion.
Se crea una lista de TurnoDTO, el cual tiene los requisitos para poder ser serializado como JSON.
El ManejadorDeTurnos se encargar de leer los precios de los tipos de tratamientos de un archivo serializado en XML, utiliza un path por defecto y si no existe el archivo lo crea con valores por defecto.
El ManejadotDeTurnos se encargar de guardar nuevamente los precios, ya que los mismos pueden ser modificados y se requiere que perduren al terminar el programa.
- Tema 16 - Introducción a bases de datos y SQL | Tema 17 - Conexión a bases de datos:
El ManejadorDeTurnos se encargar de administrar una base de datos con los turnos pendientes y el historial ya atendido:
 - Permitiendo agregar, eliminar y modificar elementos del mismo.
 - Actualizar el estado de un turno de noAtendido a atendido
 - Obtener información de la misma según sea necesario.

Para poder conectarse puede ser requerido modificar el stringConnection, que se encuentra en en la lista de constantes de la clase, justo despues de declararse la misma
- Tema 18 - Delegados y expresiones lambda:
El form Hub utiliza un delegado para poder modificar el estado actual de la musica de fondo desde otros form y a su vez cuando se vuelva a estar en el Hub el boton que posee el mismo para apagar la musica tenga el texto correcto.
Apagar Musica / Encender Musica. Segun corresponda.
Utilice expresiones lambda en programación multi-hilo

- Tema 19 - Programación multi-hilo y concurrencia:
El form de Administracion posee un recordatorio si hay un turno pendiente cada media hora, en caso de actualmente no estar en el form Administracion, si no en uno de los form secundarios asociados al mismo se espera a que la vista vuelva a estar centrada en el form Administración.
Para poder hacer esto utilice Task, y un bucle while que espera a que la vista vuelva al form principal, para posteriormente mostrar el recordatorio del turno pendiente.
Ya que de no utilizar multi hilo se congelaria la ejecución.
A su vez eventos tambien utiliza Task.
- Tema 20 – Eventos:
La clase RecordatorioEvent genera un evento el cual sucede cada hora según el minuto en el cual se le indica que suceda.
Ejemplo si indico minuto 30, sucedera a la 1:30, 2:30, 3:30, etc
Ejemplo si indico 15, sucedera a la 1:15, 2:15, 3:15, 4:15, etc.

Utilice dicho evento en conjunto con el form Administracion para que cada media hora se invoque al metodo del form que reviza si un un turno actualmente reservado en esa hora y si lo hay le muestra por pantalla la información del turno como un recordatorio.
- Tema 21 - Métodos de extensión:
Extendi las clases DateTime y Int64.
Esto es porque en la base de datos SQL para poder almacenar una fecha y hora considera que seria util tener dicho dato en una sola variable. Asi poder acceder a todos los turnos de un dia consultando unicamente una columna de la tabla en vez de una para el año, otra para el mes, otra para el dia, otra para la hora y otra para los minutos.
Con la extension de clase puedo convertir un dato de tipo DateTime en Int64 bajo el siguiente formato:
AAMMDDHHmm (año, mes, día, hora y minuto)
Ejemplo la fecha y hora 2022/06/18 15:30 pasa a ser 2206181530

A su vez la extension de Int64 permite desde un long en dicho formato volver a un DateTime.
Y la base de datos almacena un long, el cual es mas facil de comparar y obtener valores dentro de un rango.
Por ejemplo si quiero todos los turnos del dia de hoy 2022/18/06 solicito los turnos cuya fecha sea mayor a 2218060000 y menor a 2218070000

Correcciones indicades en el TP3:

“Un proyecto por formulario?

Al sacar turno, ser más específico con qué datos faltan y en qué formato se esperan
Refinarlo, pero viene bien”

-Ya no hay un proyecto por formulario, si no un unico proyecto con varios forms.

-Al sacar un turno si faltan datos el mensaje que se le muestra al usuario indica que datos faltan.

-Los TextBoxs no permiten ingresar datos invalidos, por ejemplo en el Telefono solo se pueden ingresar numeros o en el Nombre solo letras espacio y tildes. A su vez tienen un limite de la cantidad de caracteres que uno puede ingresar según lo esperado.