



UNIVERSIDAD
NACIONAL DE
HURLINGHAM

PROGRAMACION DE OBJETOS II

TRABAJO PRÁCTICO FINAL

ALUMNOS

Bellone Gian Luca

Macagna Marcos

PROFESOR

Juarez, Andrés

Ciarallo, Cristian

GRUPO

CTRL + C

Comportamientos Asumidos

Para la elaboración de este tp tomamos algunas decisiones con respecto al comportamiento de la aplicación, para desarrollar primero las funcionalidades esenciales y luego mejorarlas.

Las decisiones que tomamos son:

1. Cada especialidad tiene su consultorio, y cada consultorio tiene dos médicos (uno en el horario de la mañana y otro en el horario de la tarde). Esto facilita la gestión de los consultorios ya que el administrador no tiene que elegirlo manualmente, sino que el sistema lo asigna automáticamente, junto con el médico correspondiente.
2. El horario de la mañana es de 7 a 15 hs. y el de la tarde de 16 a 20 hs.
3. Las especialidades tomadas en cuenta para tratamientos son odontología, cardiología, traumatología y clínica (como un general).
4. Las especialidades tomadas en cuenta para estudios son radiografía, resonancia magnética y extracción de sangre.
5. Los pacientes se encuentran en una base de datos, la cual es un archivo .txt. Los mismos se importan cuando se inicia la app y el administrador los deberá guardar manualmente antes de cerrar dicha app.
6. La base de pacientes fue implementada con el patron singleton para que solo se pueda tener una instancia de la misma y no tener problemas con pacientes duplicados
7. Un médico al generar un reporte, devolverá la cantidad de prestaciones brindadas, nombre de los pacientes y la fecha de realización de cada una. Si llegara a generar un nuevo reporte en el mismo día, actualizará el realizado anteriormente.
8. Los posibles estados de un turno son “disponible” indicando que el mismo puede tomarse en fecha y horario, “asignado” indicando que ya se ha tomado un turno en dicha fecha y horario, “ausente” indicando que el paciente faltó a dicho turno, “cumplido” indicando que el paciente asistió exitosamente y “cancelado” lo cual indica que el médico o paciente, canceló dicho turno.
9. La duración de los turnos es de media hora, por lo cual la hora de finalización automáticamente es la hora de inicio sumado treinta minutos.
- 10.El menú se divide en menú de médico y menú de administrador. En el caso administrativo puede ingresar cualquiera que tenga la contraseña, en cambio al menú de médico se ingresa identificándose con el id correspondiente
- 11.Si el turno fue creado exitosamente, el mismo devolverá un número de gestión.
- 12.Si se ingresa mal la clave de administrador 3 veces, se enviará al usuario al menú principal con un mensaje de error.

13. La superposición de turnos la manejamos con la clase CalendarioTurnos la cual posee cada médico, en la cual se guardan los turnos generados y se gestionan los mismos.

Cambios realizados con el tiempo

En un principio pensamos que a la hora de elegir el turno el administrativo pueda elegir el consultorio y el médico con el que el paciente desea atenderse. Además, tanto los consultorios como los médicos podrían brindar muchas prestaciones distintas. Esto incrementaba mucho el alcance y la dificultad del programa por lo que decidimos ir por el camino del ítem 1 mencionado en los comportamientos asumidos.

Para los turnos pensábamos crear los objetos turnos y listarlos para que el administrador pueda ver solo los que estaban disponibles y que sea imposible seleccionar un turno ya asignado. Esto nos trajo dificultades a la hora de generar cientos de turnos disponibles al iniciarse la aplicación, por lo que tomamos la decisión del ítem 12 mencionado anteriormente.

En el caso de las prestaciones, en un principio las creamos con mucha especificación, es decir, “Ergometría”, “Cirugía odontológica”, etc. Lo cual terminamos generalizando por especialidades para facilitar la gestión.

Planeábamos que todos los datos se guarden al cerrar la aplicación, pero llevaba mucho tiempo por lo que solo se implementó para los pacientes. Implementarlo para los Turnos y Médicos es una oportunidad de mejora.

Conclusión

Como conclusión de este trabajo práctico nos dimos cuenta que el tamaño de la aplicación puede cambiar dependiendo de todas las requerimientos que el cliente quiera, los cuales a veces no son todos necesarios para una versión inicial. Además las funcionalidades iniciales pueden ser más simples con el fin de que cumplan con su propósito, y luego mejorarlas con el tiempo para que faciliten la experiencia de usuario u otras necesidades.

Nos dimos cuenta que es importante preguntarle al cliente con detalle para saber que de todo lo que nos dice es lo indispensable para la versión inicial.