

Trabajo integrador

Sistema de reserva de turnos

Se desea implementar un sistema que permita administrar la reserva de turnos para una clínica odontológica. Este debe cumplir con los siguientes requerimientos:

- **Administración de datos de odontólogos:** listar, agregar, modificar y eliminar odontólogos. Registrar apellido, nombre y matrícula de los mismos.
- **Administración de datos de los pacientes:** listar, agregar, modificar y eliminar pacientes. De cada uno se almacenan: nombre, apellido, domicilio, DNI y fecha de alta.privat
- **Registrar turno:** se tiene que poder permitir asignar a un paciente un turno con un odontólogo a una determinada fecha y hora.
- **Login:** validar el ingreso al sistema mediante un login con usuario y password. Se debe permitir a cualquier usuario logueado (ROLE_USER) registrar un turno, pero solo a quienes tengan un rol de administración (ROLE_ADMIN) poder gestionar odontólogos y pacientes. Un usuario podrá tener un único rol y los mismos se ingresarán directamente en la base de datos.

Requerimientos técnicos

La aplicación debe ser desarrollada en capas:

- **Capa de entidades de negocio:** son las clases Java de nuestro negocio modelado a través del paradigma orientado a objetos.
- **Capa de acceso a datos (Repository):** son las clases que se encargarán de acceder a la base de datos.
- **Capa de datos (base de datos):** es la base de datos de nuestro sistema modelado a través de un modelo entidad-relación. Utilizaremos la base H2 por su practicidad.
- **Capa de negocio:** son las clases service que se encargan de desacoplar el acceso a datos de la vista.
- **Capa de presentación:** son las pantallas web que tendremos que desarrollar utilizando el framework de Spring Boot MVC con los controladores y alguna de estas dos opciones: HTML+JavaScript para la vista.

Es importante realizar el manejo de excepciones logueando cualquier excepción que se pueda generar y la realización de test unitarios para garantizar la calidad de los desarrollos.

Avances

El trabajo tendrá una única entrega final, pero para ayudarte a organizarte, te proponemos que con tu dupla vayas avanzando de la siguiente manera:

Sprint 0 (Comienzo)

Iniciada la materia con los conocimientos ya adquiridos en Programación Orientada a Objetos, Base de Datos I y Front End I, podrás comenzar a armar tu modelo UML de las

clases que necesitarás para el proyecto integrador como así también todo lo relacionado a las tablas de la base de datos relacional que necesitarás para persistir los datos y las pantallas HTML con sus estilos para ingresar los mismos. No te preocupes que durante toda la cursada aprenderás a integrar todas estas partes!!!

Sprint 1 (Inicio semana 1 hasta Final de la semana 2)

Con lo aprendido durante estas semanas podrás realizar los testeos unitarios de las clases Java que programaste. Para asegurarnos de acá en adelante que con cada cambio tu software sigue funcionando los testeos unitarios son muy importantes.

Sprint 2 (Inicio semana 3 al Final de la semana 4)

Durante este sprint con todo lo aprendido durante la cursada en la clase 18 podrás trabajar con Maven en tu proyecto para referenciar tus librerías y con lo que vimos en la clase 14 podrás armar tus clases DAO (capa de acceso a datos con JDBC) y las clases service (capa de negocio) para cada una de las entidades de tu proyecto, pudiendo siempre garantizar el funcionamiento de todo lo que vayas construyendo utilizando pruebas unitarias.

Sprint 3 (Inicio semana 5 al Final de la semana 6)

A lo largo de este sprint estarás refactorizando la capa de acceso a datos para poder acceder y recuperar los mismos a través de un ORM. Creando los mapeos y las clases Repository que serán reemplazadas por las DAO cumpliendo la misma función. Con todo lo aprendido en las clases 25, 27 y 28 podrás construir durante este sprint las APIs (a través del desarrollo de los controllers) y la integración con la capa de presentación, o sea las pantallas HTML, a través de javascript.

Sprint 4 (Inicio semana 7 al Final de la semana 8)

Queda lo más sencillo para el final. Con los conocimientos adquiridos en la clase 29 podrás agregarle muy fácilmente a tu proyecto un login con Spring Security. Entrega clase 34. Tendrás tiempo durante todo el día para entregar hasta las 23:59 horas.