

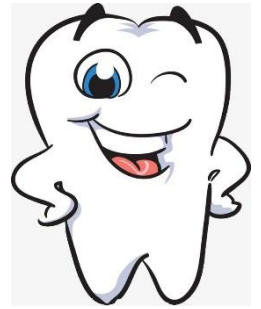
Apellido y Nombre: ..... Legajo: .....

## DentalSi

DentalSi es un centro de atención odontológico que posee varias sucursales a lo largo del país.

Hasta el momento un paciente estaba ligado a una única sucursal, debiendo atenderse y reservar turnos solamente en ella (de forma presencial únicamente).

DentalSi comenzó a notar que los pacientes preguntaban cada vez más por si podían atenderse en otra sucursal más cercana, o por lo menos realizar una reserva de turno en ella. Es por esto que dicha entidad nos contactó para realizar un sistema que permita que:



- Un **paciente** sea atendido en cualquier sucursal.
- Un paciente pueda sacar **turno** para un día, horario, especialidad y sucursal específico en cualquier sucursal o por la web.
- Un paciente pueda **cancelar** un turno cuando el horario del mismo esté fuera de las 24 horas anteriores a concretarse (presencialmente o por la web).

Además:

- Cada vez que se reserve un turno, se deberá enviar una *notificación adicional* al usuario por alguno de los siguientes medios:
  - Generar un evento nuevo en el calendario de Google del paciente (es condición necesaria que ahora todos los clientes registren al menos un mail) donde se especificará los detalles del turno. Para lograr esto se debe hacer uso de "Calendar API".
  - Enviar un SMS (será necesario tener un número de celular del paciente) detallando la información del turno.
  - Enviar un mail a una dirección establecida por el usuario (puede ser distinta a la obligatoria).
- Cuando un paciente llegue a la sucursal para ser atendido, algún recepcionista ingresará el código de turno en el sistema y el mismo pasará a estar "En espera".
- Cuando un doctor esté libre y pueda llamar a un paciente, deberá escoger al próximo turno que se encuentre en espera. Habrá una pantalla en la sala que indique el número de turno y el número de consultorio cada vez que esto suceda. Esta pantalla podría llegar a mostrar más información si así fuera requerido en un futuro.
- Cuando el paciente ya haya sido atendido, el doctor deberá marcar el turno como "cerrado", y también generar una **ficha de atención** donde especifique el resultado de la consulta.
- La ficha de atención puede contener **adjuntos**, como por ejemplo: radiografías, estudios clínicos necesarios, etc. Estos adjuntos pueden venir en *distintos formatos*.
- El paciente debe poder ver, únicamente, su propia *historia clínica*. Se entiende por historia clínica al conjunto de fichas de atención que posee el paciente. Esta historia puede ser *exportada* en distintos formatos.
- Los doctores deberán tener acceso a la historia clínica de cualquier paciente.
- El paciente debe poder *cambiar su turno* para otro día/horario/sucursal siempre y cuando el horario del mismo esté fuera de las 48 horas anteriores a concretarse. No podrá cambiarse la especialidad, ya que esto sería similar a solicitar un turno nuevo. NOTA: el horario límite para el cambio de turno podría variar.

### Punto 1 - Modelo de Dominio

A. Un equipo de desarrollo junior nos envió la siguiente implementación de la reserva de turno:

>>USUARIO

```
public void reservar(Turno unTurno){
    this.agregarTurno(unTurno);
    this.notificar(unTurno);
}
```

```
private void notificar(Turno unTurno){
    switch (this.medioDeNotificacion){
        case GOOGLE_CALENDAR:{
            ServiceCalendar unServicio = CalendarAPI::getInstance();
            Event event = new Event().setSummary("Turno con DentalSi").setLocation(unTurno.getSucursal())
                .setDescription(unTurno.getDescripcion());
            unServicio.events().insert(this.params[GOOGLE_CALENDAR]['id'], event).execute();
            }break;
        case SMS:{
            SMS unSMS = FactorySMS.crear(this.numeroDeCelular, unTurno.getDescripcion());
            HttpPost post = new HttpPost("http://www.altiria.net/smsSender");
            post.setEntity(new UrlEncodedFormEntity(unSMS, "UTF-8"));
            };break;
        case MAIL:{
            private String remitente = this.mail;
            Properties props = System.getProperties();
            props.put("mail.smtp.host", "smtp.gmail.com");
            ...
            String session = Session.getDefaultInstance(props);
            Mensaje unMensaje = FactoryMensaje.crear(unTurno.descripcion());
            This.enviar(unMensaje, session);
            };break;
        }
    }
```

¿Qué desventajas tiene el anterior código? Enumérelas.

- B. Modelar el dominio presentado utilizando el paradigma orientado a objetos, comunicando su solución mediante un diagrama de clases. Tenga en cuenta que su propuesta debe permitir agregar más medios para la notificación con facilidad, y que dicho parámetro pueda cambiar en cualquier momento. Si utiliza patrones de diseño, indíquelos y justifique su uso. Además, indique qué atributos de calidad cumplen su solución.  
NOTA: Puede ayudarse para comunicar, además, con código, pseudo-código, prosa u otros diagramas (diagrama de secuencia, de estados, entre otros).
- C. Nos han avisado (un poco tarde) que los especialistas de ortodoncia atienden, además, turnos por prioridades. ¿Su modelo soporta el nuevo requerimiento? ¿Qué cambiaría o agregaría? Explique su respuesta.  
NOTA: no es necesario mostrar esta respuesta con diagrama de clases.

## Punto 2 - Persistencia

Utilizando un DER, explique cómo persistirá el modelo del punto anterior indicando claramente:

- Qué elementos del modelo es necesario persistir.
- Las claves primarias, las foráneas y las restricciones según corresponda.
- Si fueran necesarias, estrategias de mapeo de herencia utilizadas. Explicar por qué fueron elegidas y compararlas con alguna otra.
- Justificaciones sobre las decisiones de diseño tomadas anteriormente.

## Punto 3 - Arquitectura

Proponer una arquitectura lógica del sistema a alto nivel, y comunicarla utilizando prosa y diagramas. Debe quedar en claro: componentes de alto nivel, sistemas externos (si los hubiera) y comunicación entre componentes. Si utiliza algún patrón o estilo arquitectónico, justifique su uso y explique en qué beneficia a toda su solución propuesta.

**Condiciones de aprobación:** Sumar como mínimo 60 puntos y no menos del 50 % en cada sección: (1) otorga 45 puntos, (2) 30 puntos y (3) 25 puntos. Explicar supuestos y justificar decisiones de diseño.