

**Instituto Tecnológico de Costa Rica (ITCR)**  
**Lenguajes de programación**  
**II Semestre 2023**  
**Tema: Programación orientada a objetos con Java**

**Tarea programada #2**

**El objetivo del presente trabajo** es poner en práctica el conocimiento adquirido en el curso sobre **programación orientada a objetos** y diseñar e implementar funcionalidad utilizando el lenguaje de programación **Java y el framework Spring Boot**.

Este objetivo está estrechamente relacionado con el objetivo específico del curso de utilizar un lenguaje ejemplo para cada uno de los cuatro paradigmas principales de programación, en este caso el paradigma de orientación a objetos (OO).

El proyecto permitirá a los estudiantes poner en práctica metodologías de análisis y diseño de aplicaciones orientadas a objetos y su implementación en Java. La aplicación deberá persistir los datos en una base de datos relacional de elección de las personas estudiantes utilizando **Hibernate** (una herramienta de mapeo objeto-relacional).

**Caso**

**1. Antecedentes:**

Se requiere desarrollar un sistema de información para el área de Medicina General del Hospital México que permita llevar el control de la citas de los pacientes de forma eficaz y eficiente, disminuyendo los tiempos de espera en la atención de pacientes.

El área de Medicina General del hospital brinda atención a pacientes ambulatorios de las especialidades medicas que ofrece el hospital. El servicio consiste en evaluar a los pacientes para generar un diagnóstico, si el diagnóstico es leve se atiende el caso en Medicina General, de lo contrario los pacientes son transferidos a especialistas.

Las personas estudiantes han estudiado metodologías de análisis y diseño de sistemas y técnicas de programación orientada a objetos para resolver este tipo de casos.

**2. Preguntas de reflexión:**

1. ¿Cómo se podría lograr mejorar el servicio en la consulta externa del hospital de forma que se disminuyan las filas que existen y los pacientes estén satisfechos con el servicio?

2. ¿Cómo se puede diseñar de forma modular el sistema que requiere el hospital teniendo en cuenta la herencia de clases?
3. ¿Cómo se puede lograr estandarizar el comportamiento de clases utilizando interfaces?
4. ¿Cómo se puede desarrollar el sistema en Java de forma eficiente y mantenible utilizando el patrón de diseño modelo-vista-controlador?

### **3. Delimitación del problema:**

El problema consiste en realizar todo el proceso de desarrollo de un sistema desde la etapa de análisis, diseño e implementación. El sistema debe solventar los requerimientos del área de Consulta Externa del Hospital México.

Algunos de los requerimientos se detallan a continuación:

- 1) Todas las personas en el sistema tienen asociada información como: nombre, apellido, cédula, correo electrónico. Las personas pueden ser:
  - Médicos con diversas especialidades: estos tienen asociado un número identificador además de todos los datos asociados a la persona. Los médicos pueden ser generales (médicos de cabeceras) o especialistas.
  - Pacientes: cada paciente tiene asignado un médico de cabecera (Medicina General).
- 2) Cada paciente tiene un expediente médico, en el que se ingresa información como fecha, padecimiento, procedimiento realizado y medicamentos.
- 3) Cada paciente tiene asociada una agenda. La agenda de un paciente muestra un listado de sus citas, con fecha, hora y nombre del médico.
- 4) Los pacientes pueden solicitar citas de Medicina General, pero para obtener una cita con un especialista, debe ser referido por su médico de cabecera.
- 5) Un paciente puede tener más de una referencia a especialistas y más de una cita a medicina general.
- 6) Cada cita tiene una duración de 30 minutos, un médico y un paciente asociados, así como fecha y hora.
- 7) Durante la cita el médico actualiza el expediente del paciente con datos como fecha, padecimiento, procedimiento realizado y medicamentos referidos.
- 8) Se requieren al menos los siguientes informes:

- Despliegue la agenda de un médico que muestra un listado de sus citas, con fecha, hora y paciente.
- Despliegue el expediente de un paciente.
- Despliegue la agenda de un paciente.
- Liste todos los pacientes y sus citas.
- Liste todos los médicos y su especialidad.

#### 4. Descripción de la aplicación:

Se debe implementar una aplicación para manejo de datos del hospital con los requerimientos mencionados en la delimitación del problema que cumpla con lo siguiente:

- Se **debe** utilizar la siguiente tecnología: **Java, Spring Boot, Hibernate** (una herramienta de mapeo objeto-relacional ) y una **base de datos relacional de su elección**. De no aplicar alguna de estas herramientas el proyecto tendrá un cero de nota final.
- El sistema a desarrollar debe tener funcionalidad para incluir, consultar, modificar y borrar todas las entidades del hospital (operaciones CRUD).
- Requerimientos para la programación en JAVA:
  - 1) Se debe usar modificadores de visibilidad de los atributos y métodos de las clases.
  - 2) Se debe definir al menos una variable y método de clase (static).
  - 3) Se debe implementar herencia.
  - 4) alguna de las clases debe ser abstracta.
  - 5) Se debe definir una interface que aplique a **todas** las clases que se implementen.
  - 6) Deben implementar polimorfismo por sobrecarga de métodos.
  - 7) Deben utilizar una clase genérica.
  - 8) Deben programar una aplicación web.
  - 9) Deben usar el patrón de diseño Modelo-Vista-Controlador.
  - 10) Se deben preparar los datos de prueba necesarios para ejecutar la aplicación.

#### 4. Documentación

El trabajo debe ir acompañado de documentación externa e interna.

**Documentación Interna:** Para la documentación interna, se deberán incluir comentarios descriptivos para cada clase, atributos y métodos implementados con sus respectivas entradas, salidas y restricciones.

#### **Documentación Externa:**

La documentación externa deberá incluir:

- Tabla de contenidos.
- Descripción del problema.
- Descripción de la solución
- Diagrama de casos de uso (utilizando UML)
- Descripción **completa** de cinco casos de uso.
- **Diagrama de clases (utilizando UML).**
- Diagrama Entidad-Relación (utilizando UML)
- Análisis de resultados: objetivos alcanzados, objetivos no alcanzados, y razones por las cuales no se alcanzaron los objetivos (en caso de haberlas).
- Conclusión personal.
- La documentación no debe tener errores de redacción u ortografía.
- Para la presentación deben incluir algunos registros ilustrativos para realizar las pruebas de la aplicación.

#### 5. Datos administrativos

- El trabajo se debe realizar en grupos de máximo dos personas.
- La aplicación y documentación se debe entregar en un archivo zip por medio del TecDigital y se van a programar citas para revisiones.

#### 5. Rúbrica para la evaluación

La rúbrica se adjunta en un archivo aparte.