



Parcial – Evaluación de UML + Codificación Java

Sistema de Gestión de Parque de Diversiones

Duración total: 2 horas



Contexto

Se propone el desarrollo parcial de un sistema orientado a objetos que modela el funcionamiento de un parque de diversiones.

Este sistema registra atracciones, visitantes, entradas, tickets, empleados de seguridad, incidentes reportados, mantenimientos técnicos y más.

El modelo UML correspondiente se encuentra adjunto y debe ser utilizado como guía principal para implementar la solución en Java.



Consigna

◆ Punto 1 – Codificación e instanciación del modelo (40%)

A partir del diagrama UML entregado:

1. Codificá las clases que aparecen en el modelo, respetando atributos, relaciones y jerarquías.
 2. Instanciá los siguientes objetos como parte del método `main`:
 - Un Visitante con dos Entradas, una para cada tipo de ticket (General y VIP).
 - Al menos tres Atracciones, de distintos tipos (MECANICA, INFANTIL, ACUATICA), cada una con su lista de Horarios disponibles.
 - Un Empleado de Seguridad con dos especialidades.
 - Al menos dos Incidentes: cada uno asociado a una Atracción, un Visitante, y atendido por el Empleado de Seguridad.
 - Un Reporte que contenga esos dos Incidentes.
 - Un Mantenimiento programado para una atracción y un Técnico asignado.
-

◆ Punto 2 – Método `calcularTiempoDeEspera()` (10%)

Implementá el siguiente método en la clase `Atraccion`:

```
public int calcularTiempoDeEspera()
```

Este método debe devolver el tiempo estimado de espera en minutos, calculado en base a la cantidad de visitantes en cola y la duración de la atracción.

◆ Punto 3 – Método `agregarEspecialidad()` (10%)

Implementá el método en la clase `EmpleadoSeguridad`:

```
public void agregarEspecialidad(String especialidad)
```

Este método debe permitir agregar una nueva especialidad a la lista de especialidades del empleado (evitando duplicados).

◆ Punto 4 – Método `mostrarIncidentesReportados()` (20%)

1. Implementá en la clase `EmpleadoSeguridad` el método:

```
public void mostrarIncidentesReportados()
```

Este método debe imprimir por consola los incidentes en los que participó el empleado, mostrando: atracción, fecha, descripción y estado.

◆ Punto 5 – Método `obtenerAtraccionMasVisitada()` (20%)

2. Implementá en la clase `Parque`:

```
public Atraccion obtenerAtraccionsMasVisitada()
```

Este método debe devolver la atracción que más visitantes recibió (simulado por el campo `visitantesEnCola`).



Criterios de evaluación

- Correcta interpretación del UML (herencia, composición, agregación, asociaciones).
- Codificación limpia y coherente con Java (encapsulamiento, uso de listas, tipos, enums).
- Lógica funcional en los métodos requeridos.
- Capacidad de recorrer y procesar estructuras (listas de atracciones, incidentes, tickets, etc).