

Universidad del Valle de Guatemala

Estudiante: Nathalie Valle

Ejercicio 2

Este ejercicio consiste en el desarrollo de un sistema orientado a objetos para gestionar la reserva de salones para eventos. El sistema permite registrar salones, agendar eventos, calcula costos y evitar conflictos de horario, cumpliendo con las buenas prácticas de encapsulamiento, modularidad y control de flujo.

El objetivo de este ejercicio es poder aplicar los conceptos de programación orientada a objetos en Java, diseñar clases que representen entidades reales como salones y eventos, la implementación de un sistema funcional que cumpla con restricciones específicas del enunciado y validar entradas y evitar errores comunes como conflictos de horarios o bucles infinitos.

Para ello se realizó el análisis y diseño del sistema en donde se incluyó el diagrama de clases, con su descripción de cada clase, los requisitos que opta el programa y así mismo las restricciones que tenía el sistema.

Análisis

1. Requisitos Funcionales

1. Registro de salones disponibles: Al iniciar el sistema, se deben ingresar al menos 4 salones con sus datos.
2. Solicitud de reserva: El sistema debe permitir ingresar los datos de un evento para solicitar un salón.
3. Asignación de salón: Se debe asignar un salón disponible que cumpla con las condiciones del evento.
4. Restricción por tipo de evento: Los salones grandes solo pueden ser asignados a eventos VIP.
5. Validación de reglas: Se deben cumplir reglas como el pago de depósito antes de confirmar la reserva.
6. Lista de espera: Si no hay salones disponibles, el evento debe agregarse a una lista de espera.
7. Estadísticas mensuales: Mostrar cantidad de eventos realizados e ingresados generados.
8. Interfaz gráfica: Mejorar la interacción con el usuario mediante una GUI.

2. Clases y propósito

1. Salon: Representa un salón del centro de eventos, incluyendo su tipo, capacidad y costo.
2. Evento: Contiene los datos de un evento que solicita una reserva, como fecha, tipo y encargado.

3. Reserva: Administra la asignación de salones a eventos y controla el estado de la reserva.
4. CentroEvento: Gestiona los salones, las reservas y la lista de espera. Es el núcleo del sistema.
5. Vista: Se encarga de mostrar información al usuario y recibir sus entradas.
6. Controlador: Coordina la interacción entre Vista y CentroEvento, siguiendo al patrón MVC.

3. Atributos y métodos

Clase:. Salon

Atributos:

1. numeroSalon: Identificador único del salón.
2. tipo: Define si el salón es pequeño, mediano o grande.
3. capacidad: Número máximo de personas que puede albergar.
4. costoPorHora: Precio de alquiler por hora.
5. disponible: Indica si el salón está libre para ser reservado.

Métodos:

1. esDisponible(fecha, horainicio, horafin): Verifica si el salón está libre en ese horario.
2. esApto(tipoEvento): Determina si el salón puede ser usado para el tipo de evento solicitado.

Clase: Evento

Atributos:

1. nombreEvento
2. encargado
3. tipoEvento
4. horaInicio
5. horaFin
6. depositoPagado

Métodos:

1. duración(): Calcula la duración del evento en horas.
2. requiereSalonGrande(): Retorna true si el evento es VIP y necesita salón grande.

Clase: Reserva

Atributos:

1. evento: Referencia al evento que se desea realizar.
2. salon: Salón asignado al evento.
3. estado: Estado actual de la reserva (Confirmada, Pendiente, En espera).

Métodos:

1. confirmar(): Cambia el estado de la reserva a Confirmada
2. rechazar(): Cambia el estado a En espera si no se puede asignar salón.

Clase: CentroEventos**Atributos:**

1. salones[]: Arreglo de salones disponibles.
2. reservas[]: Arreglo de reservas realizadas.
3. listaEspera[]: Eventos que no han podido ser asignados.

Métodos:

1. agregarSalon(Salones): Añade un nuevo salón al sistema.
2. recibirSolicitud(Evento e): Recibe una solicitud de reserva.
3. asignarSalon(Evento e): Busca y asigna un salón disponible que cumpla las reglas.
4. mostrarEstadisticasMensuales(): Muestra estadísticas de eventos e ingresos.

Clase: Vista**Atributos:**

1. vista: Instancia de la clase Vista.
2. modelo: Instancia de CentroEventos.

Métodos:

1. iniciarSistema(): Inicializa el sistema con salones disponibles.
2. procesarReserva(): Coordinar la solicitud y asignación de salones.

Diseño UML

