Sistema de Gestión de Eventos y Ventas de Tiquetes

1. INTRODUCCIÓN

El siguiente documento describe un modelo de dominio para un sistema de gestión de eventos y venta de tiquetes. El cual el sistema permite a los clientes comprar tiquetes para diferentes tipos de eventos (individuales, múltiples o temporadas completas) ,manejando transacciones, ofertas y configuraciones de cobros.

2. ENTIDADES PRINCIPALES

2.1 Usuario (Clase Base Abstracta)

Descripción: Clase abstracta que representa cualquier usuario del sistema. Contiene los atributos comunes a todos los tipos de usuarios.

Atributos:

- login: String - Identificador único para acceso al sistema
- password: String - Contraseña cifrada del usuario
- estado: double - Estado actual del usuario (activo/inactivo)

Subclases:

- Cliente
- Administrador

Justificación: Se usa herencia porque tanto Cliente como Administrador comparten atributos básicos de autenticación, pero tienen comportamientos y responsabilidades diferentes.

2.2 Cliente

Descripción: Representa a un usuario que puede comprar tiquetes y realizar transacciones en el sistema.

Atributos heredados de Usuario:

- login, password, estado

Métodos:

- realizarTransaccion(idTransaccion: int): void - Ejecuta una transacción de compra
- comprarTaquilla(idEvento: int, idLocalidad: int, idTiquete: int): void - Compra tiquetes para un evento específico
- tranferirTiquete(idTiquete: int, destino: Cliente): void - Transfiere un tiquete a otro cliente

- recibirTiquete(idTiquete: int): void - Recibe un tiquete transferido
- consultarSaldo(): double - Consulta el saldo disponible

Relaciones:

- Realiza múltiples Transacciones
- Posee múltiples Tiquetes

2.3 Administrador

Descripción: Usuario con privilegios especiales para gestionar el sistema, configurar eventos, venues y organizadores.

Atributos heredados de Usuario:

- login, password, estado

Métodos:

- aprobarVenue(idVenue: int): void - Aprueba un nuevo venue en el sistema
- cancelarEvento(idEvento: int): void - Cancela un evento programado
- configurarCobro(configuracionCobros: ConfiguracionCobros): void - Establece las tarifas de cobro del sistema
- configurarColaboradorVenue(idVenue: int, idOrganizador: int): void - Asigna organizadores a venues

Relaciones:

- Cancela Eventos
- Configura ConfiguracionCobros

3. GESTIÓN DE EVENTOS

3.1 Evento

Descripción: Representa un evento que se realiza en un venue específico. Puede ser de diferentes tipos según su naturaleza.

Atributos:

- idEvento: String - Identificador único del evento
- nombre: String - Nombre del evento
- descripcion: String - Descripción detallada
- hora: String - Hora de inicio
- tipo: String - Tipo de evento
- estado: String - Estado actual (activo, cancelado, finalizado)

Tipo de Eventos (Especialización):

3.1.1 TiqueteIndividual

Descripción: Evento que vende tiquetes individuales para una sola presentación.

- numeroAsiento: String - Identificador del asiento específico

3.1.2 TiqueteMultipleEvento

Descripción: Paquete que incluye acceso a múltiples eventos.

- cantidadEntradas: int - Número de entradas incluidas en el paquete

3.1.3 TiqueteTemporada

Descripción: Tiquete que da acceso a todos los eventos de una temporada completa.

- eventosIncluidos: List<Evento> - Lista de eventos incluidos en la temporada

3.2 Localidad

Descripción: Representa una sección o zona específica dentro de un evento, con características y precios particulares.

Atributos:

- idLocalidad: String - Identificador único de la localidad
- nombre: String - Nombre de la localidad (ej: "Platea", "Palco VIP")
- precio: double - Precio base de la localidad
- numerada: boolean - Indica si los asientos están numerados
- capacidad: int - Capacidad total de la localidad
- tiquetesVendidos: int - Cantidad de tiquetes ya vendidos

Relaciones:

- Pertenece a un Evento (1 a 1)
- Genera múltiples Tiquetes (* a 1)

3.3 Tiquete

Descripción: Representa un boleto de entrada a un evento en una localidad específica.

Atributos:

- idTiquete: String - Identificador único del tiquete
- precioFinal: double - Precio final después de aplicar descuentos/recargos
- fechaCompra: String - Fecha en que se realizó la compra

- estado: String - Estado del tiquete (válido, usado, transferido, cancelado)
- transferible: boolean - Indica si puede ser transferido

Relaciones:

- Pertenece a una Localidad
- Es poseído por un Cliente
- Está incluido en una Transacción

4. GESTIÓN DE TRANSACCIONES

4.1 Transaccion

Descripción: Representa una operación de compra realizada por un cliente.

Atributos:

- idTransaccion: String - Identificador único de la transacción
- fecha: String - Fecha y hora de la transacción
- estado: String - Estado de la transacción (completada, pendiente, cancelada)
- total: double - Monto total de la transacción

Relaciones:

- Generada por un Cliente
- Contiene múltiples Tiquetes

4.2 Transferencia

Descripción: Operación especial donde un cliente transfiere un tiquete a otro cliente.

Atributos:

- idTransferencia: String - Identificador único de la transferencia
- fecha: String - Fecha de la transferencia
- estado: String - Estado de la transferencia
- fromLogin: String - Login del cliente que transfiere
- toLogin: String - Login del cliente que recibe

Relaciones:

- Está asociada a Tiquete

5. GESTIÓN DE VENUES Y ORGANIZADORES

5.1 Venue

Descripción: Lugar físico donde se realizan los eventos.

Atributos:

- idVenue: String - Identificador único del venue
- nombre: String - Nombre del venue
- ubicacion: String - Dirección física
- capacidad: int - Capacidad total del venue
- tipo: String - Tipo de venue (teatro, estadio, auditorio, etc.)
- estado: String - Estado del venue (activo, en mantenimiento, cerrado)

Relaciones:

- Alberga múltiples Eventos (1 a *)
- Es sugerido por un Organizador (* a 1)
- Tiene múltiples ReportesFinancieros (1 a *)

5.2 Organizador

Descripción: Entidad o persona responsable de crear y gestionar eventos.

Atributos:

- No se especifican atributos adicionales en el diagrama

Métodos:

- crearEvento(idEvento: int): void - Crea un nuevo evento
- sugerirVenue(idVenue: int): void - Sugiere un venue para un evento
- configurarLocalidad(idLocalidad: int): void - Configura las localidades de un evento
- crearObjeto(idObjeto: int): void - Crea objetos relacionados con eventos
- crearReporteFinanciero(idReporte: int, fechaInicio: Date, fechaFin: Date): void - Genera reportes financieros

Relaciones:

- Crea múltiples Eventos
- Genera múltiples ReportesFinancieros
- Es consultado por ConfiguracionCobrs

6. SISTEMA FINANCIERO Y COBROS

6.1 ConfiguracionCobros

Descripción: Define los porcentajes y tarifas que se cobran por los diferentes servicios del sistema.

Atributos:

- porcentajeServicioMusical: double - Porcentaje cobrado en eventos musicales
- porcentajeServicioDeportivo: double - Porcentaje para eventos deportivos
- porcentajeServicioCultural: double - Porcentaje para eventos culturales
- porcentajeServicioReligioso: double - Porcentaje para eventos religiosos
- cargosPorEmision: double - Cargo fijo por emisión de tiquetes

Relaciones:

- Configurada por Administrador
- Consulta información del Organizador

6.2 ReporteFinanciero

Descripción: Informe que detalla los ingresos, gastos y comisiones de eventos en un periodo específico.

Atributos:

- idReporte: String - Identificador único del reporte
- tipo: String - Tipo de reporte (diario, mensual, anual)
- fechaGeneracion: String - Fecha en que se generó el reporte
- datos: String - Datos del reporte en formato estructurado

Relaciones:

- Generado por Organizador

7. OFERTAS Y PROMOCIONES

7.1 Oferta

Descripción: Promoción especial que aplica descuentos o beneficios a ciertos tiquetes.

Atributos:

- idOferta: String - Identificador único de la oferta
- porcentajeDescuento: double - Porcentaje de descuento aplicado
- fechaInicio: String - Fecha de inicio de la oferta
- fechaFin: String - Fecha de finalización de la oferta

- estado: boolean - Estado activo/inactivo de la oferta

Relaciones:

- Genera descuentos en Transacciones

8. OTROS COMPONENTES

8.1 PaqueteDeluxe

Descripción: Paquete premium que incluye beneficios adicionales y múltiples tiquetes.

Atributos:

- beneficios: String - Descripción de beneficios incluidos
- incluyeTiquetesAdicionales: boolean - Indica si incluye tiquetes extra

Relaciones:

- Se vincula con un Evento específico

9. REGLAS DE NEGOCIO

9.1 Reglas de Tiquetes

1. Un tiquete solo puede pertenecer a una localidad específica
2. Los tiquetes marcados como no transferibles no pueden cambiar de propietario
3. Un tiquete usado no puede ser reutilizado
4. La cantidad de tiquetes vendidos en una localidad no puede exceder su capacidad

9.2 Reglas de Eventos

1. Un evento cancelado no puede vender más tiquetes
2. Un evento debe tener al menos una localidad configurada
3. Solo el organizador que creó el evento puede modificarlo
4. Un evento debe estar asociado a un venue aprobado

9.3 Reglas de Transacciones

1. Una transacción debe incluir al menos un tiquete
2. El total de la transacción debe considerar los cargos de configuración de cobros
3. Las ofertas activas se aplican automáticamente si el periodo es válido
4. Una transacción completada no puede ser modificada

9.4 Reglas de Usuarios

1. Un cliente solo puede transferir tiquetes que le pertenezcan

2. Un administrador puede cancelar cualquier evento
3. Los organizadores solo pueden crear reportes de sus propios eventos
4. Un venue debe ser aprobado por un administrador antes de usarse

9.5 Reglas de Localidades

1. Las localidades numeradas deben tener asientos específicos asignados
2. El precio final de un tiquete incluye el precio base de la localidad más cargos
3. Una localidad sin capacidad disponible no puede vender más tiquetes

10. PATRONES DE DISEÑO IDENTIFICADOS

10.1 Herencia

- **Usuario** es la clase base para Cliente y Administrador
- **Tiquete** tiene especializaciones según el tipo de evento

10.2 Composición

- Un **Evento** está compuesto por múltiples **Localidades**
- Una **Localidad** contiene múltiples **Tiquetes**

10.3 Asociación

- **Cliente** se asocia con **Transaccion** y **Tiquete**
- **Organizador** se asocia con **Evento** y **Venue**

11. FLUJOS PRINCIPALES DEL SISTEMA

11.1 Flujo de Compra de Tiquetes

1. Cliente explora eventos disponibles
2. Selecciona un evento y una localidad
3. Verifica disponibilidad de tiquetes
4. Sistema aplica ofertas vigentes
5. Sistema calcula precio final con configuración de cobros
6. Cliente realiza la transacción
7. Sistema genera el tiquete
8. Actualiza tiquetes vendidos en la localidad

11.2 Flujo de Creación de Evento

1. Organizador sugiere un venue
2. Administrador aprueba el venue
3. Organizador crea el evento
4. Configurar localidades con capacidades y precios
5. Evento queda disponible para venta
6. Sistema registra el evento en el venue

11.3 Flujo de Transferencia de Tiquete

1. Cliente verifica que el tiquete sea transferible
2. Cliente inicia transferencia especificando destinatario
3. Sistema valida que el tiquete pertenezca al cliente
4. Sistema crea registro de transferencia
5. Destinatario recibe el tiquete
6. Sistema actualiza propietario del tiquete

12. CONSIDERACIONES TÉCNICAS

12.1 Persistencia

- Todas las entidades principales requieren almacenamiento persistente
- Las transacciones deben ser atómicas para garantizar integridad
- Los reportes financieros deben mantener histórico completo

12.2 Seguridad

- Las contraseñas deben almacenarse cifradas
- Los administradores tienen acceso privilegiado
- Las transacciones deben validar autenticación del cliente

12.3 Escalabilidad

- El sistema debe soportar múltiples eventos concurrentes
- La venta de tiquetes debe manejar acceso simultáneo
- Los venues pueden tener grandes capacidades
-

13. CARDINALIDADES IMPORTANTES

14. CONCLUSIONES

Este modelo de dominio representa un sistema completo de gestión de eventos y venta de tiquetes que:

- Permite múltiples tipos de usuarios con diferentes responsabilidades
- Soporta diversos tipos de eventos y modalidades de tiquetes
- Incluye un sistema flexible de precios y comisiones
- Facilita la gestión financiera mediante reportes
- Permite transferencia de tiquetes entre usuarios
- Integra un sistema de ofertas y promociones