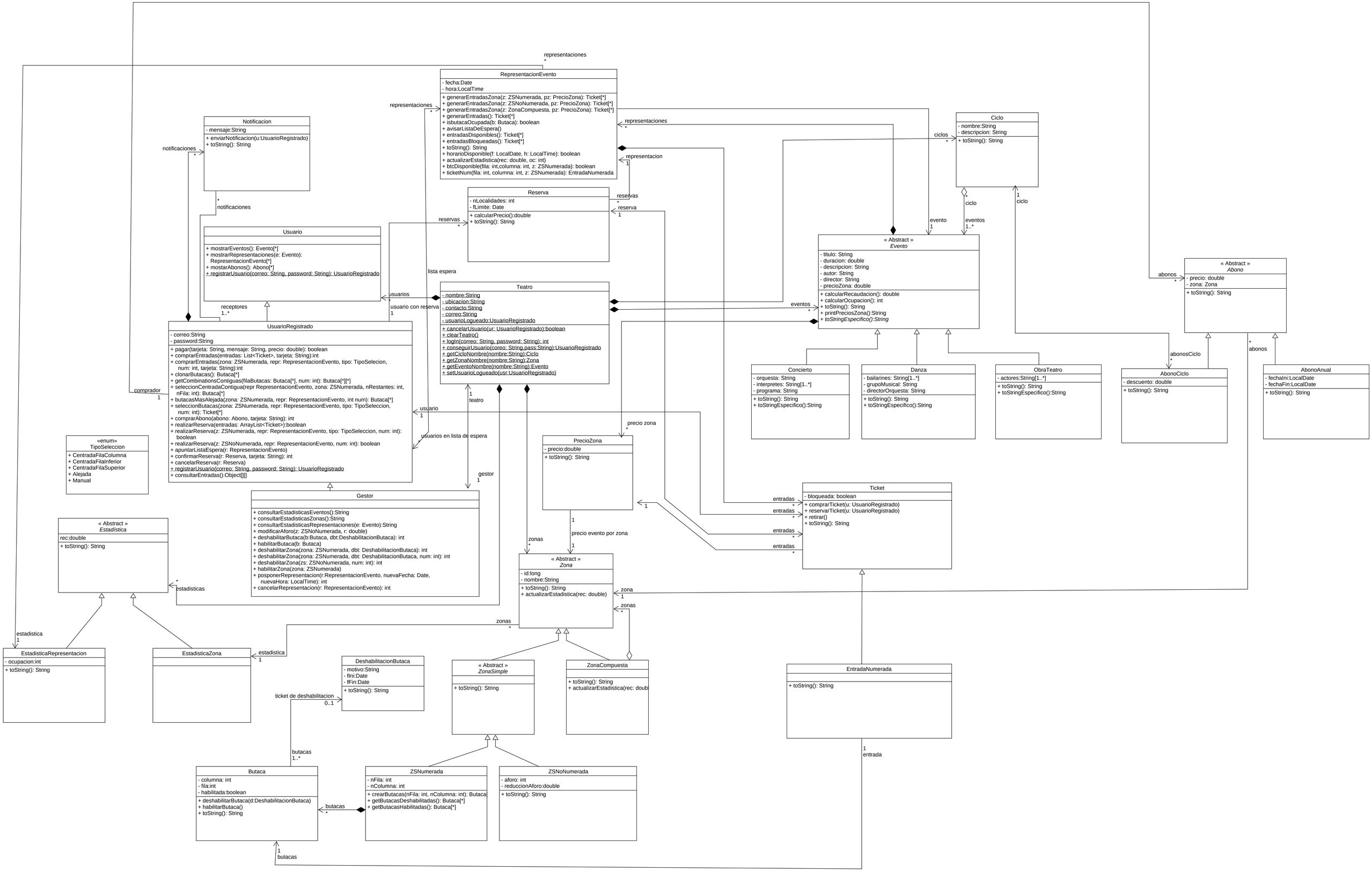
RepresentacionEvento
- fecha:Date
- hora:LocalTime
+ generarEntradasZona(z: ZSNumerada, pz: PrecioZona): Ticket[*]
+ generarEntradasZona(z: ZSNoNumerada, pz: PrecioZona): Ticket[*]
+ generarEntradasZona(z: ZonaCompuesta, pz: PrecioZona): Ticket[*]
+ generarEntradas(): Ticket[*]
+ isbutacaOcupada(b: Butaca): boolean
+ avisarListaDeEspera()
+ entradasDisponibles(): Ticket[*]
+ entradasBloqueadas(): Ticket[*]
+ toString(): String
+ horarioDisponible(f: LocalDate, h: LocalTime): boolean
+ actualizarEstadistica(rec: double, oc: int)
+ btcDisponible(fila: int,columna: int, z: ZSNumerada): boolean
+ ticketNum(fila: int, columna: int, z: ZSNumerada): EntradaNumerada
representaciones
representaciones
representaciones
representacion
Notificacion
- mensaje:String
+ enviarNotificacion(u:UsuarioRegistrado)
+ toString(): String
notificaciones
notificaciones
Ciclo
- nombre:String
- descripcion: String
+ toString(): String
ciclos
ciclo
ciclo
evento
eventos
Reserva
- nLocalidades: int
- fLimite: Date
+ calcularPrecio():double
+ toString(): String
reservas
reserva
reservas
Usuario
+ mostrarEventos(): Evento[*]
+ mostrarRepresentaciones(e: Evento): RepresentacionEvento[*]
+ mostarAbonos(): Abono[*]
+ registrarUsuario(correo: String, password: String): UsuarioRegistrado
« Abstract »
Evento
- titulo: String
- duracion: double
- descripcion: String
- autor: String
- director: String
- precioZona: double
+ calcularRecaudacion(): double
+ calcularOcupacion(): int
+ toString(): String
+ printPreciosZona():String
+ toStringEspecifico():String
« Abstract »
Abono
- precio: double
- zona: Zona
+ toString(): String
abonos
lista espera
Teatro
- nombre:String
- ubicacion:String
- contacto:String
- correo:String
- usuarioLogueado:UsuarioRegistrado
+ cancelarUsuario(ur: UsuarioRegistrado):boolean
+ clearTeatro()
+ login(correo: String, password: String): int
+ conseguirUsuario(correo: String,pass:String):UsuarioRegistrado
+ getCiclosNombre(nombre:String):Ciclo
+ getZonaNombre(nombre:String):Zona
+ getEventoNombre(nombre:String):Evento
+ setUsuarioLogueado(usr:UsuarioRegistrado)
usuarios
usuario con reserva
receptores
eventos
UsuarioRegistrado
- correo:String
- password:String
+ pagar(tarjeta: String, mensaje: String, precio: double): boolean
+ comprarEntradas(entradas: List<Ticket>, tarjeta: String):int
+ comprarEntradas(zona: ZSNumerada, repr: RepresentacionEvento, tipo: TipoSelecion, num: int, tarjeta: String):int
+ clonarButacas(): Butaca[*]
+ getCombinationsContiguas(filaButacas: Butaca[*], num: int): Butaca[*][*]
+ seleccionCentradaContigua(repr RepresentacionEvento, zona: ZSNumerada, nRestantes: int, nFila: int): Butaca[*]
+ butacasMasAlejada(zona: ZSNumerada, repr: RepresentacionEvento, int num): Butaca[*]
+ seleccionButacas(zona: ZSNumerada, repr: RepresentacionEvento, tipo: TipoSeleccion, num: int): Ticket[*]
+ comprarAbono(abono: Abono, tarjeta: String): int
+ realizarReserva(entradas: ArrayList<Ticket>):boolean
+ realizarReserva(z: ZSNumerada, repr: RepresentacionEvento, tipo: TipoSeleccion, num: int): boolean
+ realizarReserva(z: ZSNoNumerada, repr: RepresentacionEvento, num: int): boolean
+ apuntarListaEspera(r: RepresentacionEvento)
+ confirmarReserva(r: Reserva, tarjeta: String): int
+ cancelarReserva(r: Reserva)
+ registrarUsuario(correo: String, password: String): UsuarioRegistrado
+ consultarEntradas():Object[][]
comprador
usuario
usuarios en lista de espera
teatro
gestor
precio zona
«enum»
TipoSeleccion
+ CentradaFilaColumna
+ CentradaFilaInferior
+ CentradaFilaSuperior
+ Alejada
+ Manual
Gestor
+ consultarEstadisticasEventos():String
+ consultarEstadisticasZonas():String
+ consultarEstadisticasRepresentaciones(e: Evento):String
+ modificarAforo(z: ZSNoNumerada, r: double)
+ deshabilitarButaca(b:Butaca, dbt:DeshabilitacionButaca): int
+ habilitarButaca(b: Butaca)
+ deshabilitarZona(zona: ZSNumerada, dbt: DeshabilitacionButaca): int
+ deshabilitarZona(zona: ZSNumerada, dbt: DeshabilitacionButaca, num: int): int
+ deshabilitarZona(zs: ZSNoNumerada, num: int): int
+ habilitarZona(zona: ZSNumerada)
+ posponerRepresentacion(r:RepresentacionEvento, nuevaFecha: Date, nuevaHora: LocalTime): int
+ cancelarRepresentacion(r: RepresentacionEvento): int
PrecioZona
- precio:double
+ toString(): String
Concierto
- orquesta: String
- interpretes: String[1..*]
- programa: String
+ toString(): String
+ toStringEspecifico():String
Danza
- bailarines: String[1..*]
- grupoMusical: String
- directorOrquesta: String
+ toString(): String
+ toStringEspecifico():String
ObraTeatro
- actores:String[1..*]
+ toString(): String
+ toStringEspecifico():String
AbonoCiclo
- descuento: double
+ toString(): String
AbonoAnual
- fechaIni:LocalDate
- fechaFin:LocalDate
+ toString(): String
abonosCiclo
abonos
Ticket
- bloqueada: boolean
+ comprarTicket(u: UsuarioRegistrado)
+ reservarTicket(u: UsuarioRegistrado)
+ retirar()
+ toString(): String
entradas
entradas
entradas
entradas
« Abstract »
Estadística
rec:double
+ toString(): String
estadisticas
« Abstract »
Zona
- id:long
- nombre:String
+ toString(): String
+ actualizarEstadistica(rec: double)
zona
zonas
zonas
zonas
precio evento por zona
estadistica
EstadisticaRepresentacion
- ocupacion:int
+ toString(): String
EstadisticaZona
estadistica
DeshabilitacionButaca
- motivo:String
- fIni:Date
- fFin:Date
+ toString(): String
ticket de deshabilitacion
« Abstract »
ZonaSimple
+ toString(): String
ZonaCompuesta
+ toString(): String
+ actualizarEstadistica(rec: doub
EntradaNumerada
+ toString(): String
entrada
butacas
Butaca
- columna: int
- fila:int
- habilitada:boolean
+ deshabilitarButaca(d:DeshabilitacionButaca)
+ habilitarButaca()
+ toString(): String
butacas
butacas
ZSNumerada
- nFila: int
- nColumna: int
+ crearButacas(nFila: int, nColumna: int): Butaca
+ getButacasDeshabilitadas(): Butaca[*]
+ getButacasHabilitadas(): Butaca[*]
ZSNoNumerada
- aforo: int
- reduccionAforo:double
+ toString(): String