

# Iteración 2 - FestivAndes

## Cambios del modelaje de la Iteración 1

Paula J. Alvarado Zabala, Ana C. Fandiño de la Hoz  
201313033, 201326407  
Sistemas Transaccionales  
Universidad de los Andes, Bogotá, Colombia  
{pj.alvarado10, ac.fandino10}@uniandes.edu.co

Fecha de presentación: Marzo 22 de 2017

### **Tabla de Contenido**

#### 0. Introducción

#### 1. Análisis

##### 1.a. Casos de Uso

##### 1.a.1. Casos de uso: Espectadores

##### 1.a.2. Casos de uso: Organizadores

##### 1.b. Modelo del Mundo

#### 2. Diseño de la Base de Datos

##### 2.a. Modelo Lógico y Modelo de Tablas Relacional

##### 2.a.1. Modelo Lógico

##### 2.a.2. Modelo de Tablas Relacional

##### 2.b. Documentación del Diseño y Definición de Tablas

##### 2.b.1. Documentación del Diseño

##### 2.b.2. Definición de tablas

#### 3. Conclusiones

#### 4. Bibliografía

### **0. Introducción**

El presente proyecto busca modificar el modelado de una aplicación transaccional llamada FestivAndes realizada en la Iteración 1 que servirá para la organización, ejecución y reserva de eventos de teatro, disponible para usuarios que sean organizadores y/o espectadores.

## 1. Análisis

### 1.a. Casos de Uso

#### 1.a.1. Casos de uso: Espectadores

Título: Reserva de Eventos

Descripción General del Caso de Uso		
La reserva es realizada por un usuario espectador para poder reservar un evento.		
Entidades Involucradas		
ID: id el usuario que pretende realizar la reserva Evento: Evento que quiere reservar, teniendo en cuenta que este cuenta con una fecha.		
Precondiciones		
El usuario no tiene que estar registrado por su id.		
Flujo normal de Eventos		
	Usuario	Sistema
1.	Manda la solicitud del evento por la aplicación	Encuentra disponibilidad del evento y retorna las sillas disponibles
2	Selecciona una silla y reserva	Guarda los cambios y compra la silla
Post-condiciones principales del caso de uso		
Retorna resultado de la reserva. Si fue exitosa o no. Booleano		
Caminos de Excepción		
Que el evento no exista, o no haya disponibilidad.		

Título: Eliminar Reserva

Descripción General del Caso de Uso		
El usuario espectador elimina la reserva que tiene en el momento.		
Entidades Involucradas		
ID: id del usuario que pretende eliminar la reserva Evento: Evento que tiene reservado		
Precondiciones		

El usuario tiene que estar registrado por su id y tiene que tener una reserva.		
Flujo normal de Eventos		
	<b>Usuario</b>	<b>Sistema</b>
1.	Manda la solicitud para eliminar la reserva	Elimina la reserva
Post-condiciones principales del caso de uso		
Retorna resultado de la eliminación de la reserva. Si fue exitoso o no. Booleano		
Caminos de Excepción		
Que el evento no exista o el id no se encuentre registrado o no tenga reserva ese usuario.		

Título: Búsqueda de una reserva

Descripción General del Caso de Uso		
El usuario espectador solicita la reserva que tiene en el momento		
Entidades Involucradas		
ID: id del usuario que tiene la reserva Reserva que está solicitando. Un usuario puede tener varias reservas.		
Precondiciones		
El usuario tiene que estar registrado por su id y tiene que tener una reserva.		
Flujo normal de Eventos		
	<b>Usuario</b>	<b>Sistema</b>
1.	Manda la solicitud de sus reservas	Devuelve las reservas que tiene el usuario por ese id
2.	Selecciona una reserva de la cual quiere obtener los detalles	Retorna la reserva con toda la información asociada a esta
Post-condiciones principales del caso de uso		
Retorna información de la reserva seleccionada		
Caminos de Excepción		
Que el evento no exista o el id del usuario no se encuentre registrado o el usuario no		

tenga reservas todavía.

#### Título: Búsqueda de eventos

##### Descripción General del Caso de Uso

El usuario espectador busca un evento

##### Entidades Involucradas

Nombre del evento que busca. Artista si busca el evento por artista. Fecha si busca el evento por fecha. Teatro si busca el evento por teatro.

##### Precondiciones

El usuario puede estar registrado por su id (aunque no es obligatorio).

##### Flujo normal de Eventos

	Usuario	Sistema
1.	Manda la solicitud de búsqueda	Devuelve los resultados de búsqueda de acuerdo a la selección del usuario
2.	Selecciona un evento	Retorna los detalles y la información del evento. Tiene la opción de reservar si el usuario está registrado.

##### Post-condiciones principales del caso de uso

Retorna información del evento seleccionado.

##### Caminos de Excepción

Que no hayan eventos con los datos especificados.

#### 1.a.2. Casos de uso: Representantes

##### Título: Creación de eventos

##### Descripción General del Caso de Uso

Un usuario organizador crea un evento.

##### Entidades Involucradas

ID: id el usuario que pretende crear el evento

Evento: Evento que quiere crear, teniendo en cuenta que este cuenta con una fecha.

Teatro: Teatro donde se realizará el evento.  
Sala: Sala del teatro donde se realizará el evento.  
Artista: Artistas que participarán en el evento.

#### Precondiciones

El usuario tiene que estar registrado por su id.

#### Flujo normal de Eventos

	Usuario	Sistema
1.	Crea la solicitud del evento por la aplicación	Devuelve los espacios para rellenar datos del evento.
2	Completa los datos de parámetros y también especifica los respectivos costos (de acuerdo a la silla) que tendrá el evento.	Guarda los cambios y crea el evento.

#### Post-condiciones principales del caso de uso

Retorna el evento que ahora debe estar público.

#### Caminos de Excepción

Que el evento no exista, el teatro, la sala y/o algún artista no esté disponible para esa fecha o el id del usuario no se encuentre registrado.

#### Título: Editar eventos

##### Descripción General del Caso de Uso

Un usuario organizador edita los datos de un evento.

##### Entidades Involucradas

ID: id el usuario que pretende crear el evento.  
Evento: Evento que quiere editar.  
Teatro: Teatro donde se realizará el evento.  
TeatroCambio: Teatro donde se hará el evento si es esto lo que quiere editar el usuario.  
Sala: Sala del teatro donde se realizará el evento.  
SalaCambio: Nueva sala donde se realizará el evento, si esto es lo que quiere ser editado.  
Artista: Artistas que participarán en el evento.  
ArtistaNuevo: Si hay nuevos artistas en el evento o si elimina y/o cambia un artista por otro.

Precondiciones		
El usuario tiene que estar registrado por su id.		
Flujo normal de Eventos		
	Usuario	Sistema
1.	Envía la solicitud de editar el evento por la aplicación	Devuelve los espacios para editar datos del evento.
2	Edita los datos que desee	Guarda los cambios.
Post-condiciones principales del caso de uso		
Retorna el evento con la información modificada.		
Caminos de Excepción		
Que el evento no exista, el teatro, la sala y/o algún artista no esté disponible para esa fecha o el id del usuario no se encuentre registrado.		

#### Título: Eliminar Evento

Descripción General del Caso de Uso		
El usuario organizador elimina un evento creado.		
Entidades Involucradas		
ID: id del usuario que pretende eliminar el evento Evento: Evento creado por el usuario		
Precondiciones		
El usuario tiene que estar registrado por su id y tiene que tener un evento.		
Flujo normal de Eventos		
	Usuario	Sistema
1.	Manda la solicitud para eliminar el evento.	Elimina el evento.
Post-condiciones principales del caso de uso		
Retorna resultado de la eliminación del evento. Si fue exitoso o no. Booleano		
Caminos de Excepción		

Que el evento no exista o el id del usuario no se encuentre registrado o el usuario no tenga eventos creados todavía.

Título: Búsqueda de un evento creado

#### Descripción General del Caso de Uso

El usuario espectador solicita un evento que tenga en el momento

#### Entidades Involucradas

ID: id del usuario que tiene el evento

Evento que está solicitando. Un usuario puede tener varios eventos.

#### Precondiciones

El usuario tiene que estar registrado por su id y tiene que tener un evento

#### Flujo normal de Eventos

	Usuario	Sistema
1.	Manda la solicitud de sus eventos.	Devuelve los eventos que tiene el usuario por ese id
2.	Selecciona un evento del cual quiere obtener los detalles.	Retorna el evento con toda la información asociada a este

#### Post-condiciones principales del caso de uso

Retorna información del evento seleccionado.

#### Caminos de Excepción

Que el evento no exista o el id del usuario no se encuentre registrado o el usuario no tenga eventos todavía.

## 1.b. Modelo del Mundo

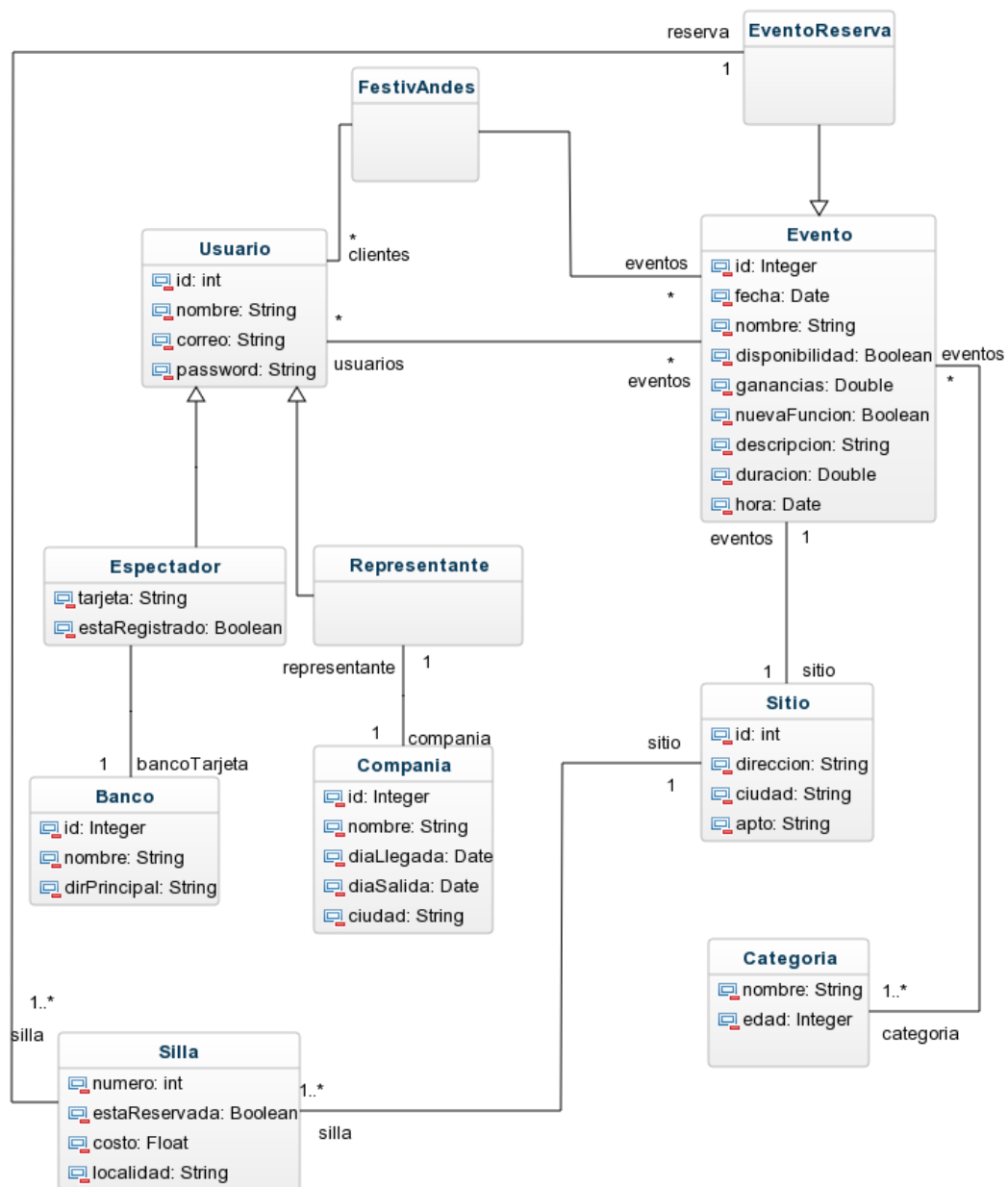


Imagen 1. Diagrama UML de FestivAndes

En este diagrama conceptual, excepto por la clase principal FestivAndes (que es primordialmente para realizar operaciones), todas las clases deben tener persistencia pues contienen información importante del festival (los eventos, las personas (usuarios y artistas), los teatros, las salas, las sillas y sus numeraciones). Adicionalmente, es importante mencionar que las relaciones tienen que persistir.



## 2. Diseño de la Base de Datos

### 2.a. Modelo Lógico y Modelo de Tablas Relacional

#### 2.a.1. Modelo Lógico

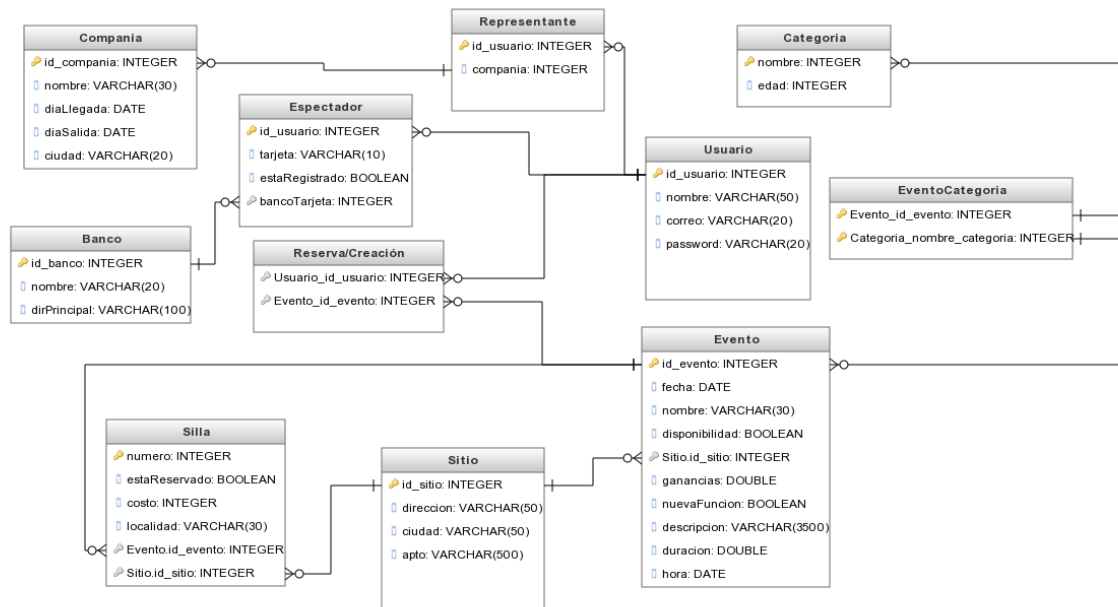


Imagen 2 - Modelo Lógico de FestivAndes

#### 2.a.2. Modelo de Tablas Relacional

Usuario			
id_usuario	nombre	correo	password
PK	NN	NN, ND	NN

Espectador			
id_usuario	tarjeta	estaRegistrado	bancoTarjeta
PK, FK(Usuario.id_usuario)	NN	NN	NN, FK(Banco.id_banco)

Representante	
id_usuario	compania
PK, FK(Usuario.id_usuario)	NN, FK(Compania.id_compania)

Compania				
id_compania	nombre	diaLlegada	diaSalida	ciudad
PK	NN	NN, CK(>hoy)	NN, CK(>diaLlegada)	NN

Banco		
id_banco	nombre	dirPrincipal
PK	NN	NN

reserva/creacion	
id_usuario	id_evento
PK1,FK(Usuarios.id_usuario)	PK2, FK(Eventos.id_evento)

Evento									
id_ev ento	fecha	nomb re	disp onibi lidad	gana ncias	nuevaF uncion	descrip cion	duracion	hora	sitio
PK	NN, CK(>Hoy)	NN	NN	NN	CK	NN	NN	NN	NN,FK(Sitio.i d_sitio)

Sitio			
id_sitio	direccion	ciudad	apto
PK	NN,ND	NN	NN

Silla					
numero	estaReservado	costo	localidad	sitio	eventoReserva
PK	NN	CK( $\geq 0$ )	NN	FK(Sitio.id_sitio), NN	FK(Evento.id_evento)

Categoria	
nombre	edad
PK	NN

EventoCategoria	
id_Evento	nombre_Categoria
PK1, FK(Evento.id_evento)	PK2, FK(Categoria.nombre)

## 2.b. Documentación del Diseño y Definición de Tablas

### 2.b.1. Documentación del Diseño

Nombre	Usuarios		
Descripción	Base de datos de los Usuarios		
Llave	id_usuario		
Atributo	Tipo	Descripción	Características/Restricciones
id_usuario	int	Identificador del usuario	Valor único. PK
nombre	String	Nombre del usuario	No puede estar vacío
correo	String	Correo electrónico del usuario	No puede estar vacío

password	String	Contraseña del usuario	No puede estar vacío
----------	--------	------------------------	----------------------

Nombre	Eventos		
Descripción	Base de datos de los eventos		
Llave	id_evento		
Atributo	Tipo	Descripción	Características/Restricciones
id_evento	int	Identificador del evento	Valor único. PK
fecha	Date	Fecha del evento	No puede estar vacío
nombre	String	Nombre del evento	No puede estar vacío
disponibilidad	Boolean	Indica si el evento está disponible	No puede estar vacío. True si está disponible, False de lo contrario.

Nombre	Sitio		
Descripción	Base de datos de los teatros		
Llave	id_sitio		
Atributo	Tipo	Descripción	Características/Restricciones
id_sitio	int	Identificador del teatro	Valor único. PK
direccion	String	Dirección donde sucede el evento	No puede estar vacío
ciudad	String	Ciudad donde sucede el evento	No puede estar vacío

Nombre	Sillas		
Descripción	Base de datos de las sillas		
Llave	numero_silla		
Atributo	Tipo	Descripción	Características/Restricciones
numero_silla	int	Número de la silla	Valor único. PK
estaReservada	Boolean	Indica si la silla se encuentra reservada	No puede estar vacío. True si esta reservada, false de lo contrario.
tipo	String	Tipo o ubicación de la silla	No puede estar vacío
costo	Float	Costo por seleccionar esa silla	No puede estar vacío.

Nombre	Categorías		
Descripción	Base de datos de las categorías		
Llave	nombre		
Atributo	Tipo	Descripción	Características/Restricciones
nombre	String	Nombre de la categoría	Valor único. PK
edad	int	Edad mínima permitida para esa categoría	No puede estar vacío

Nombre	Bancos		
Descripción	Base de datos de los bancos		
Llave	id_banco		
Atributo	Tipo	Descripción	Características/Restricciones
id_banco	int	Identificador del banco	Valor único. PK
nombre	String	Nombre del banco	No puede estar vacío
dirPrincipal	String	Dirección de la oficina principal del banco	No puede estar vacío

Nombre	Companias		
Descripción	Base de datos de las compañías		
Llave	id_compania		
Atributo	Tipo	Descripción	Características/Restricciones
id_compania	int	Identificador de la compañía	Valor único. PK
nombre	String	Nombre de la compañía	No puede estar vacío
diaLlegada	Date	Día en el que una compañía llega a una ciudad	No puede estar vacío
diaSalida	Date	Día en el que la compañía se va de una ciudad	No puede estar vacío.
ciudad	String	Ciudad a la que llega la compañía	No puede estar vacío.

### 2.b.2. Definición de tablas

La cardinalidad de todas las tablas en este momento es 1 pues no se han agregado tuplas de datos en este momento.

TABLA USUARIO	
<b>Tipo</b>	Tabla de una clase
<b>Anidad</b>	4
<b>Atributos</b>	id_usuario, nombre, correo, password.
<b>Descripción</b>	Su llave primaria es id_usuario, no acepta nulos y el correo no puede estar duplicado.

TABLA RESERVA/CREACION	
<b>Tipo</b>	Tabla de una relación
<b>Anidad</b>	2
<b>Atributos</b>	id_usuario, id_evento
<b>Descripción</b>	Relación entre un usuario y un evento, en el que un usuario puede reservar (en caso de los espectadores) o crear (en caso de los organizadores) un evento. Un usuario puede tener muchos eventos y un evento puede tener muchos usuarios. Tiene dos llaves foráneas (el id_usuario de la Tabla Usuario e id_evento de la Tabla Evento) que a su vez son sus dos llaves primarias.

TABLA EVENTO	
<b>Tipo</b>	Tabla de una clase
<b>Anidad</b>	10
<b>Atributos</b>	id_evento, fecha, nombre, disponibilidad, sitio, descripcion, ganancias, nuevaFuncion, duracion, hora.
<b>Descripción</b>	Su llave primaria es id_evento, la fecha tiene una restricción de que no puede ser una fecha menor a hoy, no acepta nulos y tiene una llaves foránea que la relaciona con un sitio (id_sitio). Evento es lo mismo que función.

TABLA SITIO	
<b>Tipo</b>	Tabla de una clase
<b>Anidad</b>	4
<b>Atributos</b>	id_sitio, direccion, ciudad, apto.
<b>Descripción</b>	Su llave primaria es id_sitio, y no acepta nulos. Apto significa si el lugar es apto para personas con condiciones especiales (niños, discapacitados, ancianos).

TABLA SILLA	
<b>Tipo</b>	Tabla de una clase
<b>Anidad</b>	5
<b>Atributos</b>	numero, estaReservado, costo, tipo, sitio.



<b>Descripción</b>	Su llave primaria es número, no acepta nulos, tiene una restricción en el costo y es que no puede ser menor a 0, y tiene una llave foránea que la relaciona con una sala (id_sitio).
--------------------	--

<b>TABLA CATEGORIA</b>	
<b>Tipo</b>	Tabla de una clase
<b>Anidad</b>	2
<b>Atributos</b>	nombre, edad.
<b>Descripción</b>	Su llave principal es el nombre, indica la edad de las personas que pueden verla.

<b>TABLA EVENTOCATEGORIA</b>	
<b>Tipo</b>	Tabla de una relación
<b>Anidad</b>	2
<b>Atributos</b>	id_evento, nombre_categoria.
<b>Descripción</b>	Sus atributos son llaves principales y relaciona varios eventos con varias categorías.

<b>TABLA BANCO</b>	
<b>Tipo</b>	Tabla de una clase
<b>Anidad</b>	3
<b>Atributos</b>	id_banco, nombre, dirPrincipal
<b>Descripción</b>	Su llave principal es id_banco, y tiene el nombre del banco y la

	dirección de la oficina principal.
--	------------------------------------

TABLA COMPANIA	
<b>Tipo</b>	Tabla de una clase
<b>Anidad</b>	5
<b>Atributos</b>	id_compania, nombre, diaLlegada, diaSalida, ciudad.
<b>Descripción</b>	Su llave principal es id_compania, y la tabla tiene el nombre de la compañía, el día que llega a la ciudad, el día que se va y la ciudad a la que llega.

### 3. Conclusiones

Las modificaciones que se han hecho es agregar una clase que se relacione con el banco de la tarjeta bancaria de la persona y que la persona ya no tenga que estar registrada si quiere comprar boletos de una función. Además la clase Reserva se refiere a una compra. Se cambia la clase Organizador por Representante y se agrega la clase Compañía que está relacionada con Representante. Se agrega la clase Categoría.

### 4. Bibliografía

Database Administrator's Guide. (2008, March 13). Retrieved February 18, 2017, from [https://docs.oracle.com/cd/B28359\\_01/server.111/b28310/create003.htm#ADMIN11073](https://docs.oracle.com/cd/B28359_01/server.111/b28310/create003.htm#ADMIN11073)

Universidad De Granada (n.d.). Diseño Lógico: Diseño de bases de datos relacionases. Retrieved February 17, 2017, from <http://elvex.ugr.es/idbis/db/docs/design/5-logical.pdf>

Universidad De Buenos Aires (n.d.). Ingeniería de Software: Casos de Uso. Retrieved February 17, 2017, from [http://www-2.dc.uba.ar/materias/isoft1/2001\\_2/apuntes/CasosDeUso.pdf](http://www-2.dc.uba.ar/materias/isoft1/2001_2/apuntes/CasosDeUso.pdf)