

## ITERACION 1 SISTEMAS TRANSACCIONALES

Integrantes:

- Carlos Mendoza  
Ca.mendoza  
201621144
- Federico Marroquin  
f.marroquin10  
201615499

### 3.A casos de uso

Descripción General del Caso de Uso		
REGISTRAR LOS OPERADORES DE ALOJAMIENTO PARA ALOHANDES		
Entidades Involucradas		
Operadores, Sistema		
Precondiciones		
Los datos o atributos del operador deben seguir las reglas de negocio		
Flujo normal de Eventos		
	Usuario	Sistema
1	El usuario ingresa a la aplicación de Alohandes	
2	El usuario se registra proporcionando los datos necesarios.	El sistema valida los datos recibidos por el usuario
3		Si están correctos notifica al operador que su usuario ha sido creado
Post-condiciones principales del caso de uso		
Fue creado un nuevo usuario operador.		
Caminos de Excepción		
Si los datos registrados por el usuario no son correctos su cuenta no es creada. O si el Operador ya se a registrado con anterioridad		

Descripción General del Caso de Uso		
REGISTRAR PROPUESTAS DE ALOJAMIENTOS PARA ALOHANDES.		
Entidades Involucradas		
Operador, Oferta, Inmueble, servicio		
Precondiciones		
Debe existir un operador para crear dicha oferta. Los datos deben ser correctos		
Flujo normal de Eventos		
	<b>Usuario</b>	<b>Sistema</b>
<b>1</b>	El usuario operador ingresa los datos de la oferta, los inmueble y los servicios de dicha oferta	El sistema valida la información otorgada por el usuario
<b>2</b>		Crea la oferta con toda la información suministrada si esta es correcta
Post-condiciones principales del caso de uso		
Se creo una propuesta de alojamiento( Oferta)		
Caminos de Excepción		
Si los datos suministrados violan las reglas de negocio establecidas por la Aplicación la propuesta no es creada		

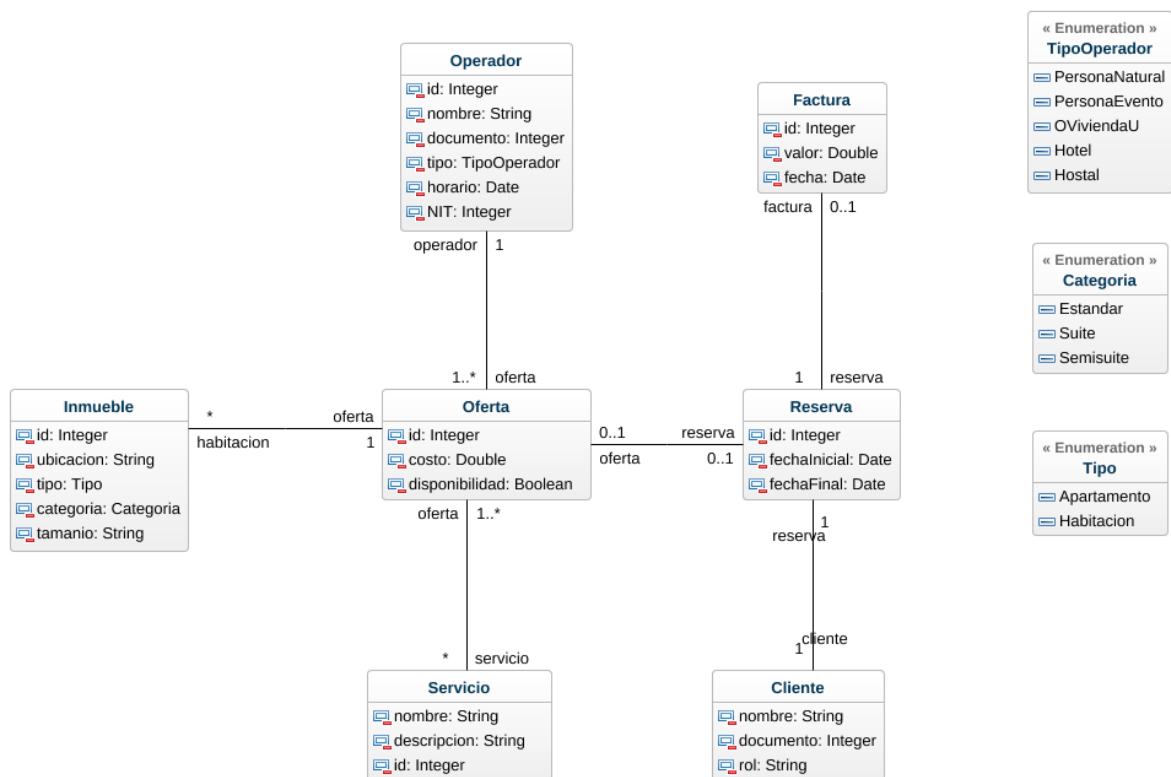
Descripción General del Caso de Uso		
REGISTRAR LAS PERSONAS HABILITADAS PARA UTILIZAR LOS SERVICIOS		
Entidades Involucradas		
Cliente		
Precondiciones		
Los datos y atributos del usuario cliente deben ser correctos		
Flujo normal de Eventos		
	Usuario	Sistema
1	El usuario ingresa a la aplicación de Alohandes	
2	El usuario se registra proporcionando los datos necesarios.	El sistema valida los datos recibidos por el usuario
3		Si están correctos notifica al operador que su usuario ha sido creado
Post-condiciones principales del caso de uso		
Fue creado un nuevo usuario Cliente		
Caminos de Excepción		
No es creado el usuario si los datos proporcionados están errados o si el usuario ya a sido registrado		

Descripción General del Caso de Uso		
REGISTRAR UNA RESERVA		
Entidades Involucradas		
Operadores, Oferta, Reserva, Cliente		
Precondiciones		
Flujo normal de Eventos		
	Usuario	Sistema
1	El usuario cliente busca hacer una reserva en el sistema. Abre sesión en el sistema	El sistema valida al usuario
2	El usuario identifica la oferta para la cual desea hacer la reserva	
3	El usuario registra los datos para realizar la reserva	El sistema valida los datos. Si la oferta está disponible, si las fechas son correctas y en general si los datos son correctos
4		El sistema crea la reserva relacionado al usuario Cliente
5		El sistema notifica al operador de la reserva
6		Cambia la disponibilidad de la oferta
Post-condiciones principales del caso de uso		
Se crea una reserva por un usuario cliente		
Caminos de Excepción		
Si no existe la oferta para la dicha reserva esta no se crea. Si los datos proporcionados son incorrectos la reserva no es creada		

Descripción General del Caso de Uso		
CANCELAR UNA RESERVA		
Entidades Involucradas		
Oferta, Reserva, Factura, Cliente		
Precondiciones		
La reserva a cancelar existe y es del usuario que desea cancelarla. La fecha de cancelación debe ser antes de la fecha final de la reserva		
Flujo normal de Eventos		
	Usuario	Sistema
1	El usuario cliente busca cancelar una reserva en el sistema. Abre sesión en el sistema	El sistema valida al usuario
2	El usuario busca la reserva	Proporcionar reserva
3	El usuario intenta eliminar la reserva	El sistema elimina la reserva
4		Borra la reserva en la base de datos
5		Actualiza la disponibilidad de la oferta asociada a dicha reserva
6		Genera la factura asociada a dicha reserva
7		Notifica al cliente que la reserva ha sido cancelada
Post-condiciones principales del caso de uso		
La reserva fue cancelada, se actualizo la base de datos.		
Caminos de Excepción		
Si el usuario intenta cancelar la reserva después de la fecha final de esta entonces no es cancelada y debe pagar por ella		

Descripción General del Caso de Uso		
RETIRAR UNA OFERTA DE ALOJAMIENTO		
Entidades Involucradas		
Operadores, Oferta, Inmueble, Servicio, reserva		
Precondiciones		
El operador debe existir, la oferta, sus servicios e inmuebles deben estar creados		
Flujo normal de Eventos		
	Usuario	Sistema
1	El usuario operador ingresa en la plataforma	El sistema valida el ingreso
	El operador decide si retirar la oferta de alojamiento inmediatamente o programada	Si el usuario intenta retirarla de forma inmediata se verifica que está disponible.
		Su el usuario intenta retirarla de forma programada la oferta es eliminada un día después de la fecha final de la reservación asociada
		Se elimina, se actualiza la información en la base de datos, en las tablas PRESTAN, OFERTA, INMUEBLE, OPERADORES
Post-condiciones principales del caso de uso		
Se retiró una oferta de alojamiento		
Caminos de Excepción		
No se retira una si esta tiene una reservación y dicha reservación aún está vigente		

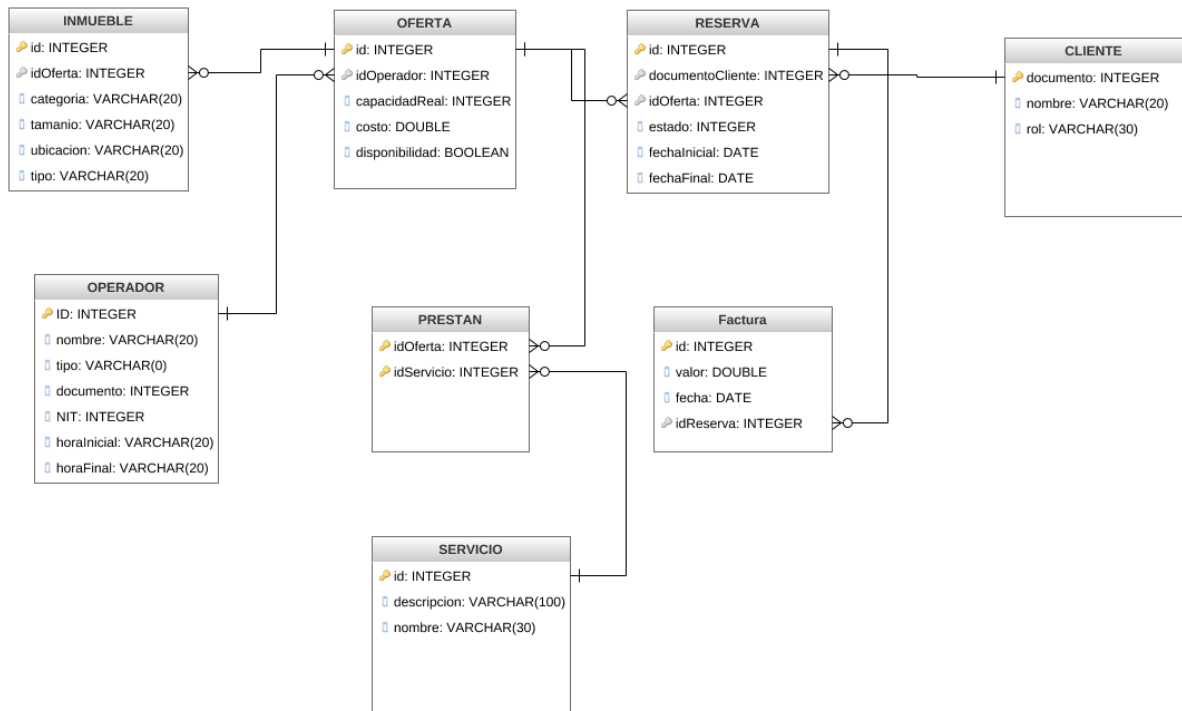
### 3.B Modelo conceptual



EL modelo conceptual conciste de 7 clases. La clase operador representa a los operadores de Alohandes, entre ellos estan los hoteles, los hostales y los particulares. La clase oferta representa la oferta de un operador, esta puede estar o no disponible para reserva, así mismo tiene asociados los inmuebles del operador y los servicios que presta. La clase inmueble modela un inmueble, ya sea un apartamento o una habitación. La clase reserva tiene la información de la fecha en la cual un cliente hace una reservación. El cliente es el que hace uso de los servicios de alojamiento así como quien hace las reservaciones. La clase factura representa la factura propia de la transacción que permite hacer una reserva exitosa de una oferta dada. En nuestro modelo todas las clases deben ser persistidas debido a que es necesario la información de cada componenete para mantener la calidad de los datos.



## 4.A Modelo de datos relacional



### Oferta:

Nombre	OFERTA		
Descripción	Modela una oferta hecha por un operador		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	INTEGER	identificación única de la oferta	PK
idOperador	INTEGER	identificador del operador que hace la oferta	FK(OPERADOR.id), NN
idReserva	INTEGER	identificador de la reserva sobre la oferta hecha por un usuario	FK(RESERVA.id)
capacidadReal	INTEGER	la capacidad o numero de personas que puede acoger un inmueble	NN, CK(>0)
costo	DOUBLE	precio que tiene la oferta	NN CK(>=0)
disponibilidad	BOOLEAN	indica si la oferta puede ser reservada o no por un cliente	NN

### Operador:

Nombre	OPERADOR		
Descripción	Representa un ente generador de servicios, puede ser un hotel, hostel, persona natural o empresas de viviendas universitarias		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	INTEGER	identificador de un operador	PK
horaInicial	VARCHAR	indica la hora inicial en el horario de recepción de clientes	NN
nombre	VARCHAR	nombre del operador	NN
tipo	tipoOperador	ptipo: persona natural, hotel, hostel, persona evento o empresa viviendaU	NN
documento	INTEGER	documento de identificación de una persona natural	NN
NIT	INTEGER	número de identificación tributaria	NN
horaFinal	VARCHAR	indica la hora final en el horario de recepción de clientes	NN

### Cliente:

Nombre	CLIENTE		
Descripción	define la relación de una oferta que		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
documento	INTEGER	identificación única de un cliente	PK
nombre	VARCHAR	nombre del cliente	NN
rol	VARCHAR	rol que tiene un cliente en alohandes	FK(RESERVA.id)

### Reserva:

Nombre	RESERVA		
Descripción	modela una reserva hecha por un cliente sobre la oferta de un operador		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	INTEGER	identificador de la reserva	PK
documentoCliente	INTEGER	identificación única de un cliente	FK(CLIENTE.DOCUMENTO), NN
estado	INTEGER	estado de la reserva	NN
fechaInicial	DATE	fecha inicial de la reserva	NN ck(fechaInicial>new date())
fechaFinal	DATE	fecha final de la reserva	NN, CK( fechaFinal>fechaInicial)

### Factura:

Nombre	FACTURA		
Descripción	modela una factura propia de una reservación		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	INTEGER	identificador de la factura	PK
valor	DOUBLE	valor de la factura	NN, CK(valor>0 )
idReserva	INTEGER	identificador de una reserva	FK(RESERVA.id), nn
fecha	DATE	fecha de la factura	NN

Presta:

Nombre	PRESTAN		
Descripción	define la relación de una oferta que tiene servicios		
Llave	idOferta, idServicio		
Atributo	Tipo	Descripción	Características/Restricciones
idOferta	INTEGER	identificación única de la oferta	PK
idServicio	INTEGER	identificador del servicio	FK(SERVICIO.id), NN

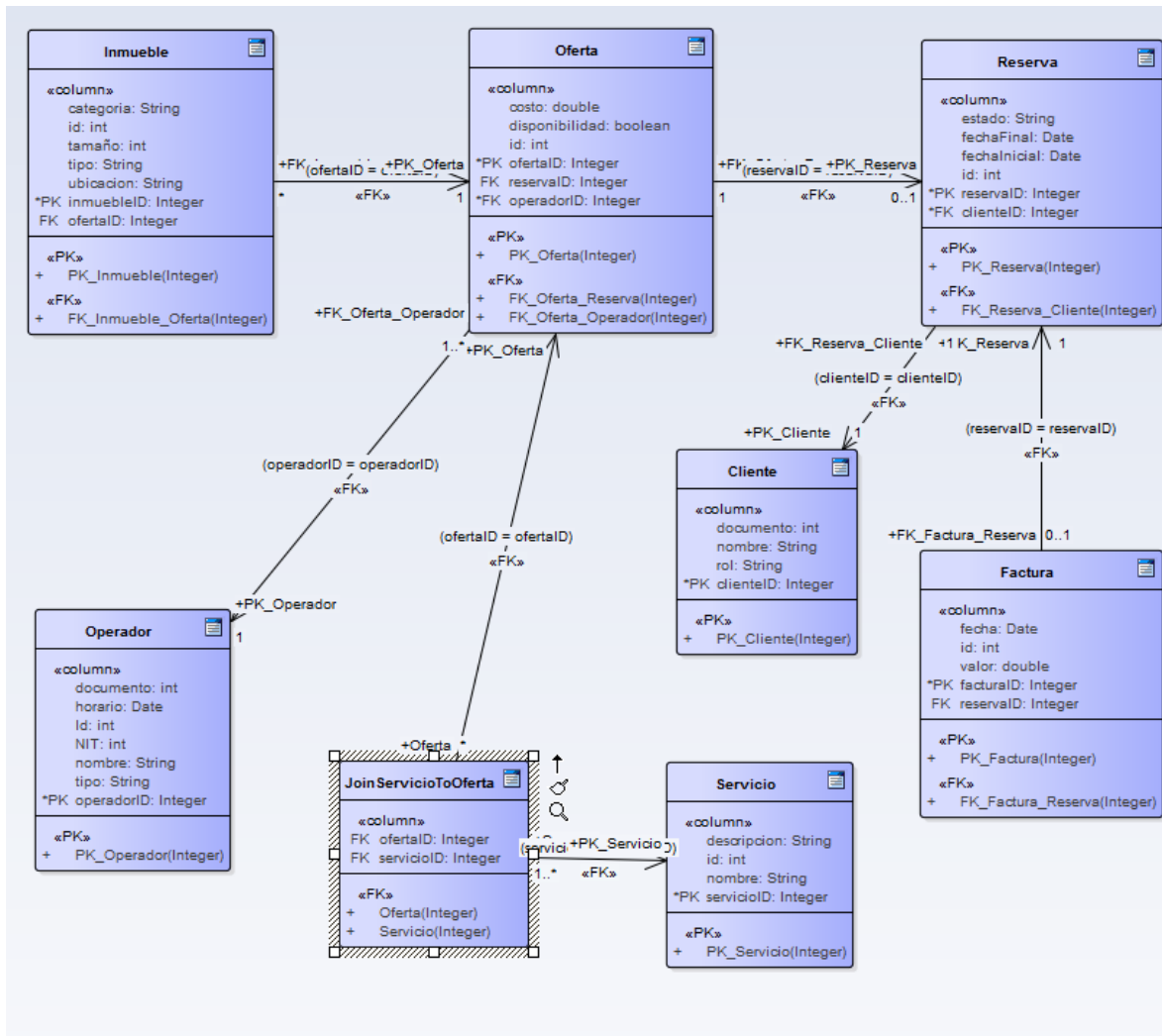
Inmueble:

Nombre	INMUEBLE		
Descripción	representa un apartamento o habitación el cual es puesto en oferta por un operador		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	INTEGER	identificador del inmueble	PK
idOferta	INTEGER	identificador de la oferta, la cual ofrece este inmueble	FK(OFERTA.id), NN
tipo	tipo	tipo de inmueble: Apartamento o habitación	NN
categoria	Categoria	categoria del inmueble: Estandar, suite o semisuite	NN
tamano	VARCHAR	tamaño del inmueble en metros cuadrados	NN
ubicacion	VARCHAR	dirección del inmueble	NN

Servicio:

Nombre	SERVICIO		
Descripción	representa un servicio que puede tener incluido un inmueble en su oferta		
Llave	id		
Atributo	Tipo	Descripción	Características/Restricciones
id	INTEGER	identificador de un servicio	PK
nombre	VARCHAR	nombre que se le da a un servicio, debe estar relacionado con la naturaleza de este	NN
descripcion	VARCHAR	descripción detallada del servicio en cuestión	NN

### 3.B Modelo de datos relacional generado en Enterprise Architect.



Al generar el modelo de datos relacional a partir del modelo conceptual notamos que tanto las tablas como las relaciones son las mismas a las de del modelo de datos relacional que implementamos nosotros. Se creó una nueva tabla llamada JoinServicioToOferta la cual es equivalente a nuestra tabla prestan donde se relacionan las ofertas con los servicios. Las llaves foráneas generadas coinciden con las nuestras.

#### Resultados logrados y no logrados:

La implementación de todos los requerimientos funcionales son logrados satisfactoriamente exceptuando, el requerimiento de crear una oferta, puesto que en el momento de agregar una oferta de alojamiento se logra crear la información de la oferta, la información del inmueble (habitación o apartamento) pero no se logra generar la información de la tabla prestan para relacionar los servicios con la oferta. Desconocemos la naturaleza del bug presentado, ya que este puede ser inherente a la implementación del código y asociado a un error de SQL developer.

Se crearon los modelos conceptual y relacional con sus debidos comentarios según el formato dado, así mismo se desarrollaron los casos de uso.