**Sistemas Transaccionales, Iteración 2**

Iván Felipe García Laverde, Juan Sebastián Espitia Acero

Universidad de los Andes, Bogotá, Colombia

Fecha de presentación: marzo 21 de 2018

Contenido

[1 Análisis de reestructuración 1](#_Toc512333427)

[2 Diseño de la Aplicación 2](#_Toc512333428)

[4 Generación de Tablas 8](#_Toc512333429)

[5 Población 8](#_Toc512333430)

# Análisis de reestructuración

**Cambios del modelo conceptual**

Debido a la adaptabilidad y complexión del modelo de AlohAndes ya planteado para la iteración anterior, no existió la necesidad de hacer cambios en las clases o las relaciones entre ellas. Los nuevos requerimientos funcionan de manera adecuada con la información que ya se tuvo en cuenta para realizar el programa y la base de datos, pues no exigen un manejo diferente de la información como tal, sino de la forma en que es presentada ante las consultas de un usuario.

**Funcionalidades principales:**

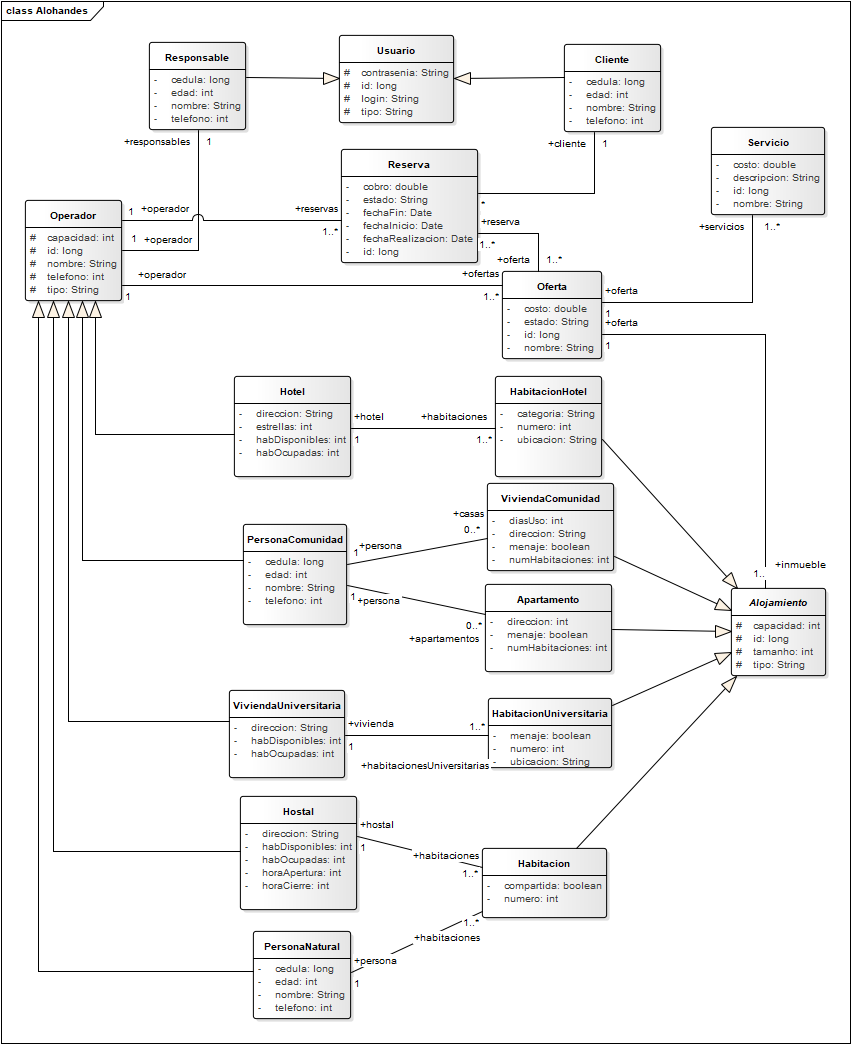
Las funcionalidades principales agregadas para la segunda versión de la aplicación incluyen requerimientos de transaccionalidad, lo cual implica la habilidad de los usuarios de manejar reservas de un mismo operador en un mismo o varios alojamientos a nombre de uno o más usuarios, también implica que los proveedores deben poder consultar el estado de dichas reservas y ofertas para garantizar el bien funcionamiento financiero de etas.

**Reglas de negocio adicionales**

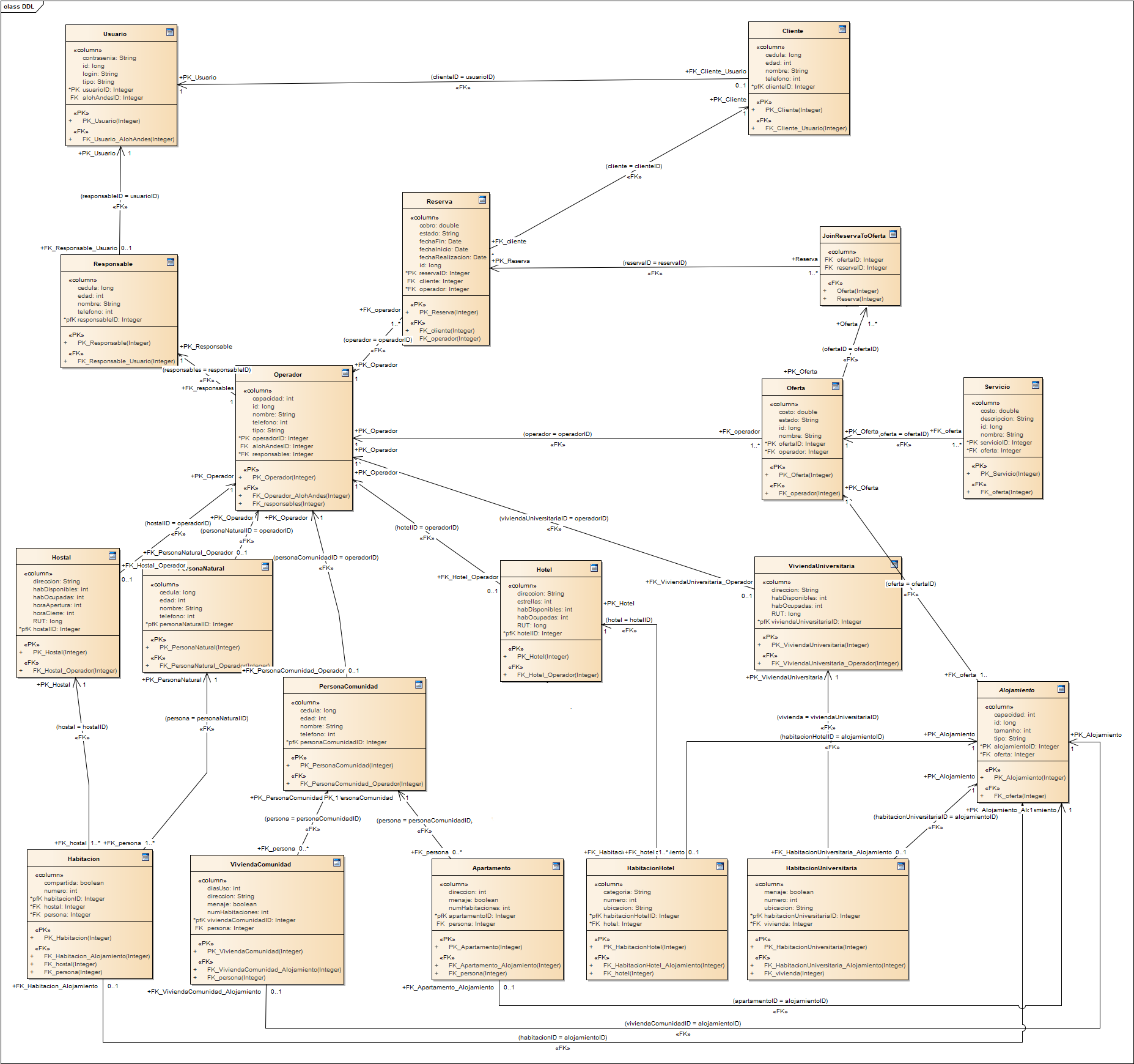
* Las ofertas colectivas que se deben poder deshabilitar y habilitar.
* Las reservas colectivas correspondientes a una oferta colectiva deshabilitada deben reasignarse en su mayor proporción posible, e informarse completamente cuales reservas fueron reasignadas y cuáles no.

# Diseño de la Aplicación

**2.1 Diagrama de clases UML:**

****

**2.2 Modelo relacional:**

****

**2.3 Impacto de los nuevos requerimientos:**

colectivas correspondientes a una oferta colectiva deshabilitada deben reasignarse en su mayor proporción

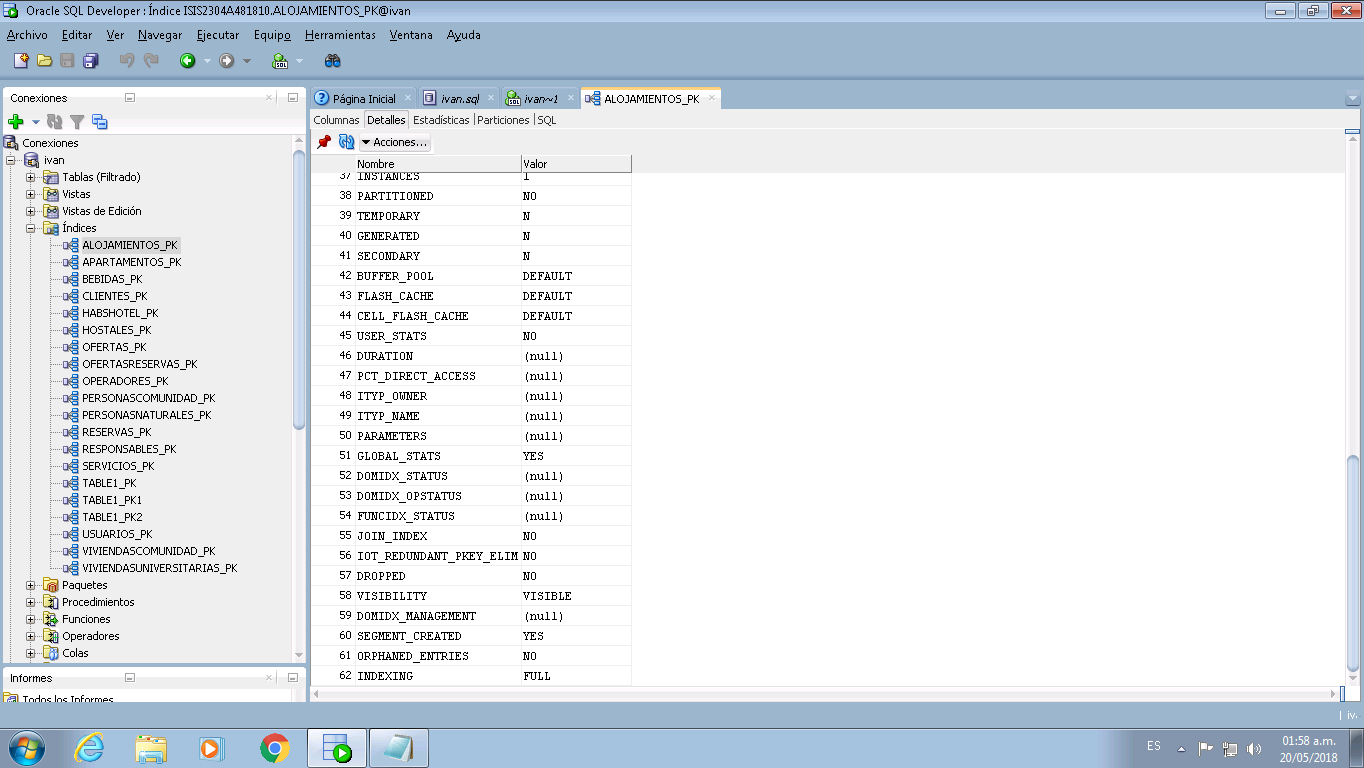
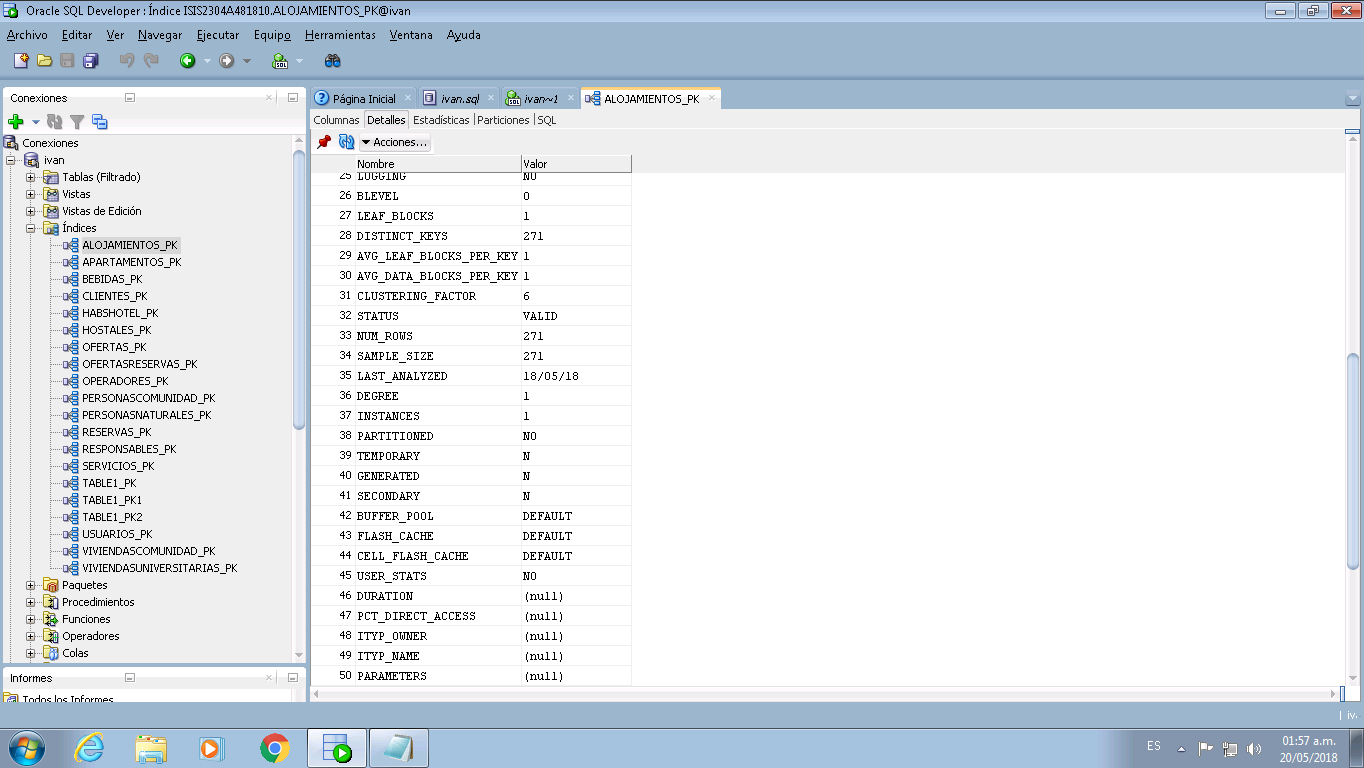
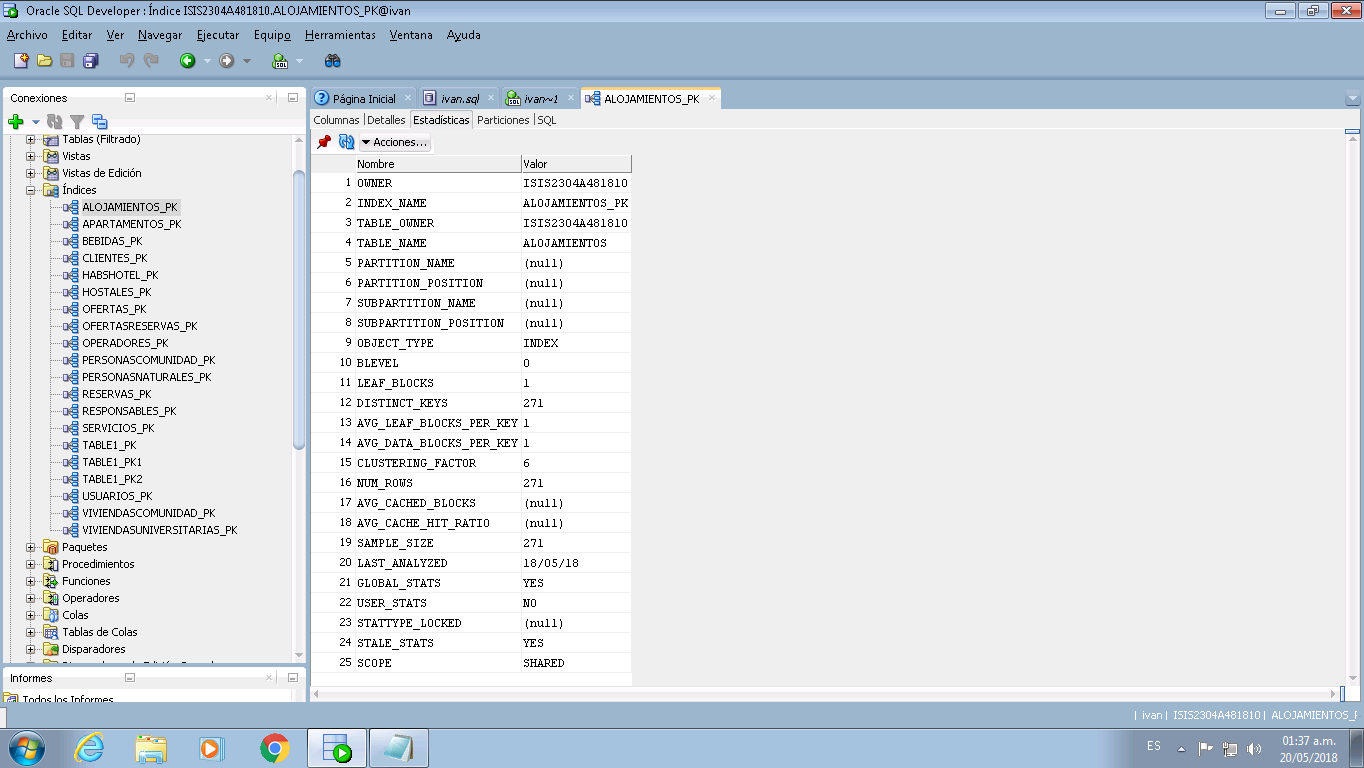
**2.4 Detalle de índices:**

**2.4.1 Justificación de los índices:**

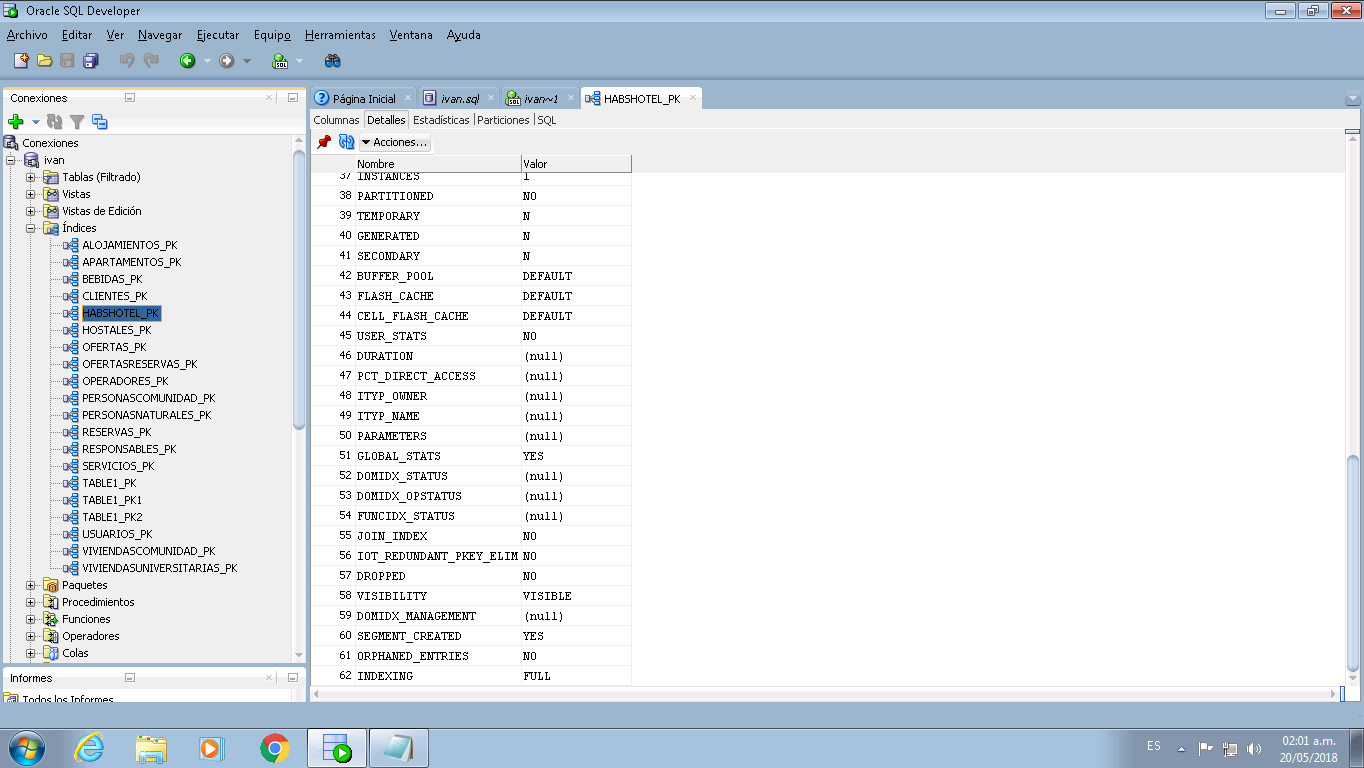
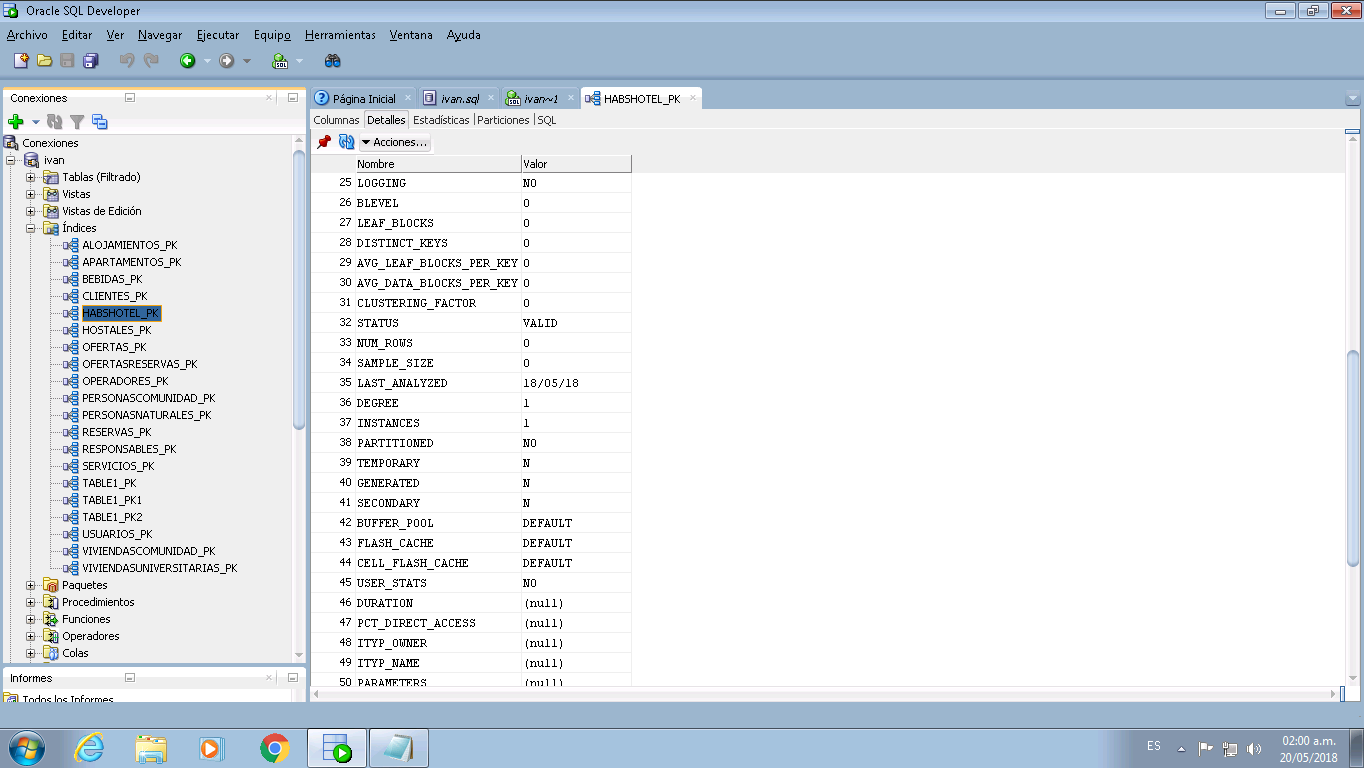
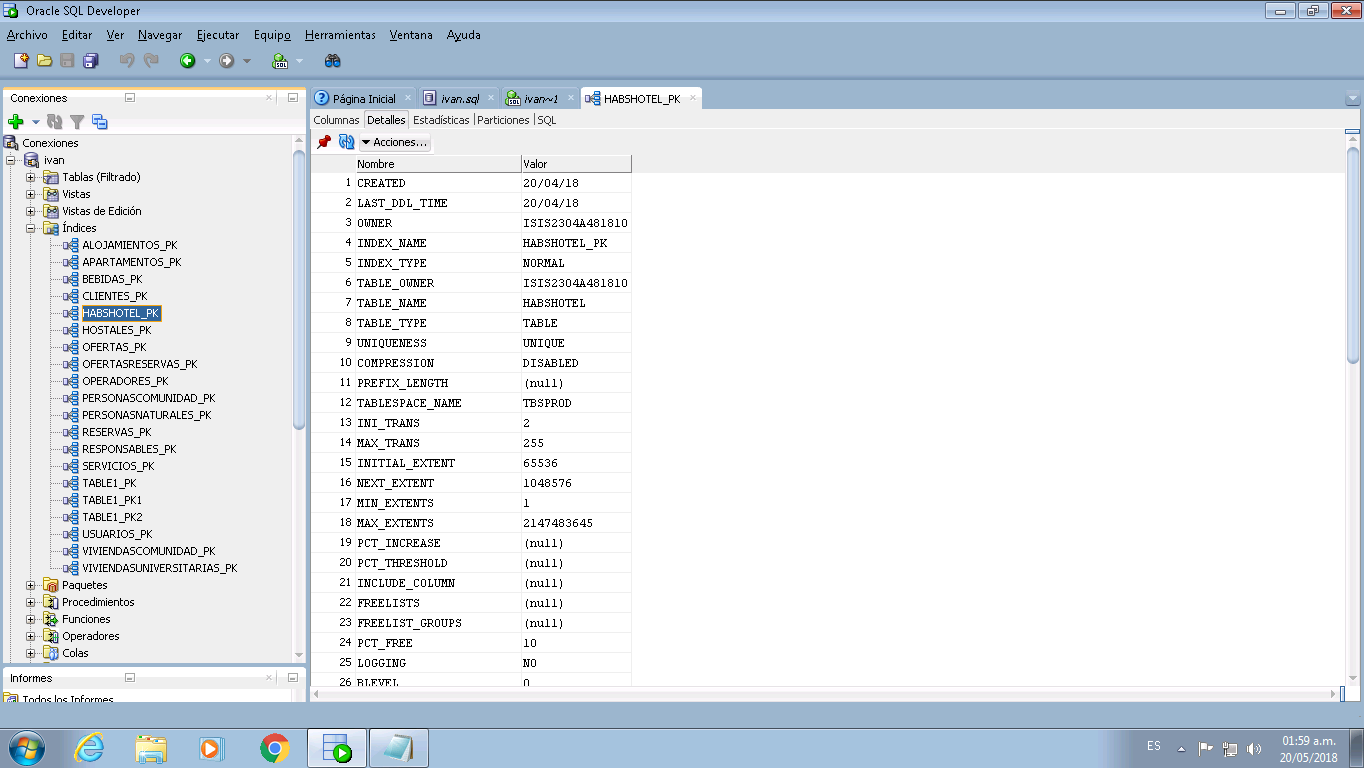
**2.4.2 Índices autogenerados:**

Los índices que genera Oracle de forma automática son los que se encargan de manejar las llaves primarias de todas las tablas, estos se deben a que las columnas de identidad deben ser valores únicos, no repetibles, y es más fácil crearlos de manera secuencial con un índice que, a su vez, permite que haya una selectividad alta y que al hacer una consulta de una tupla en particular, se obtenga una respuesta muy rápida.

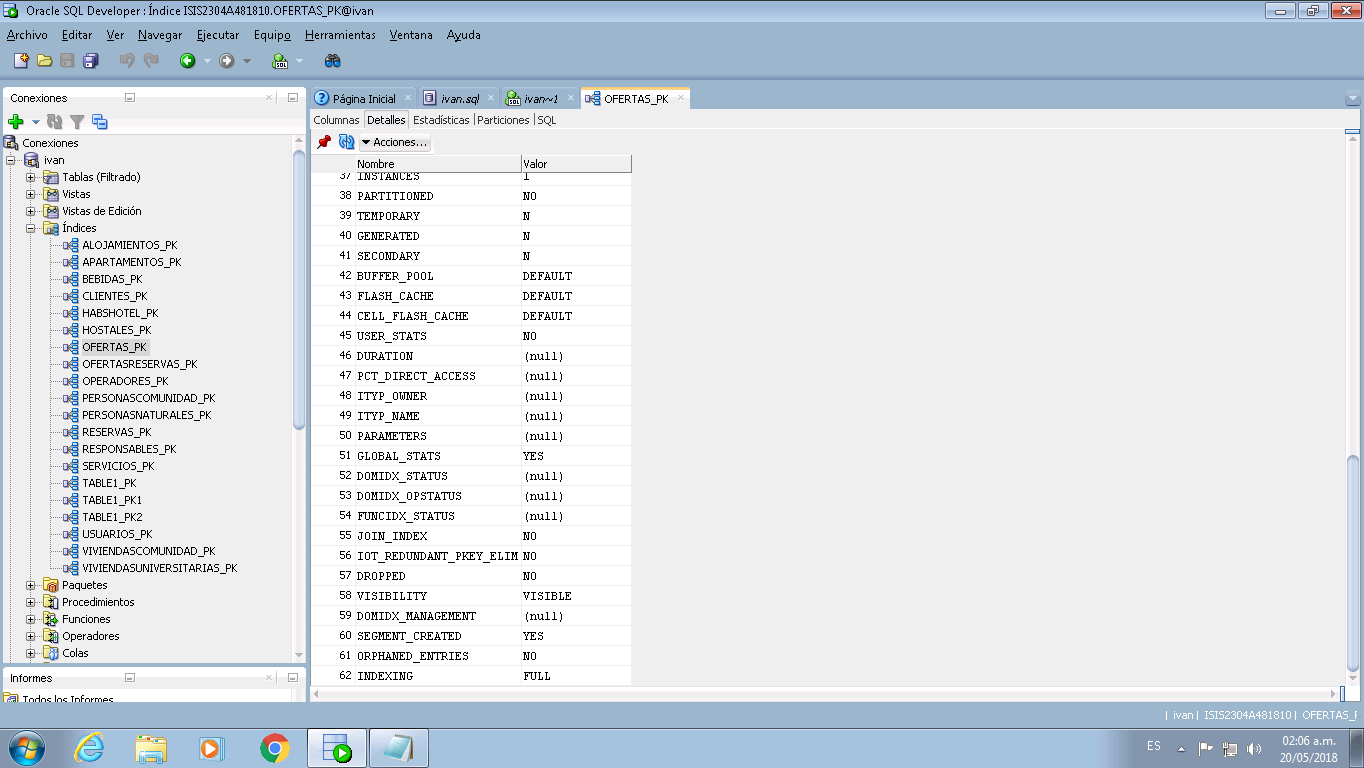
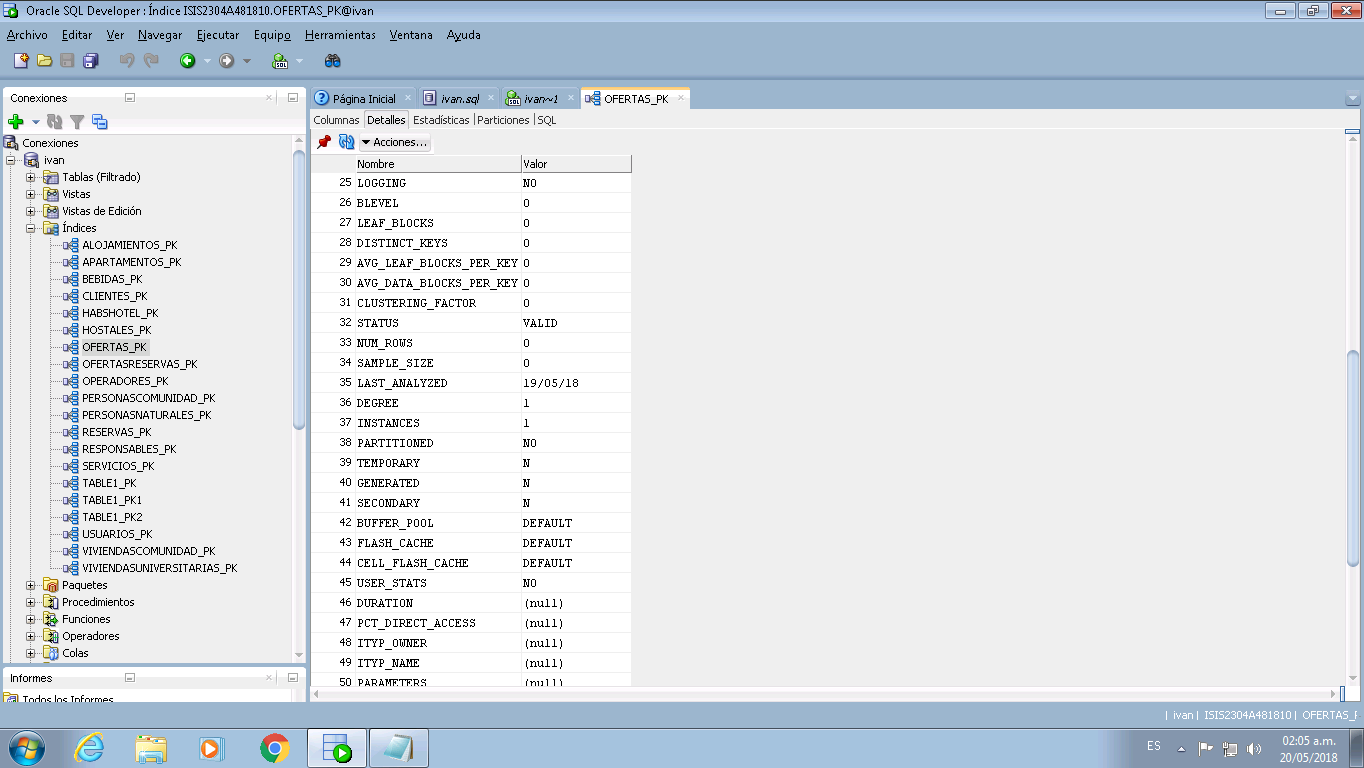
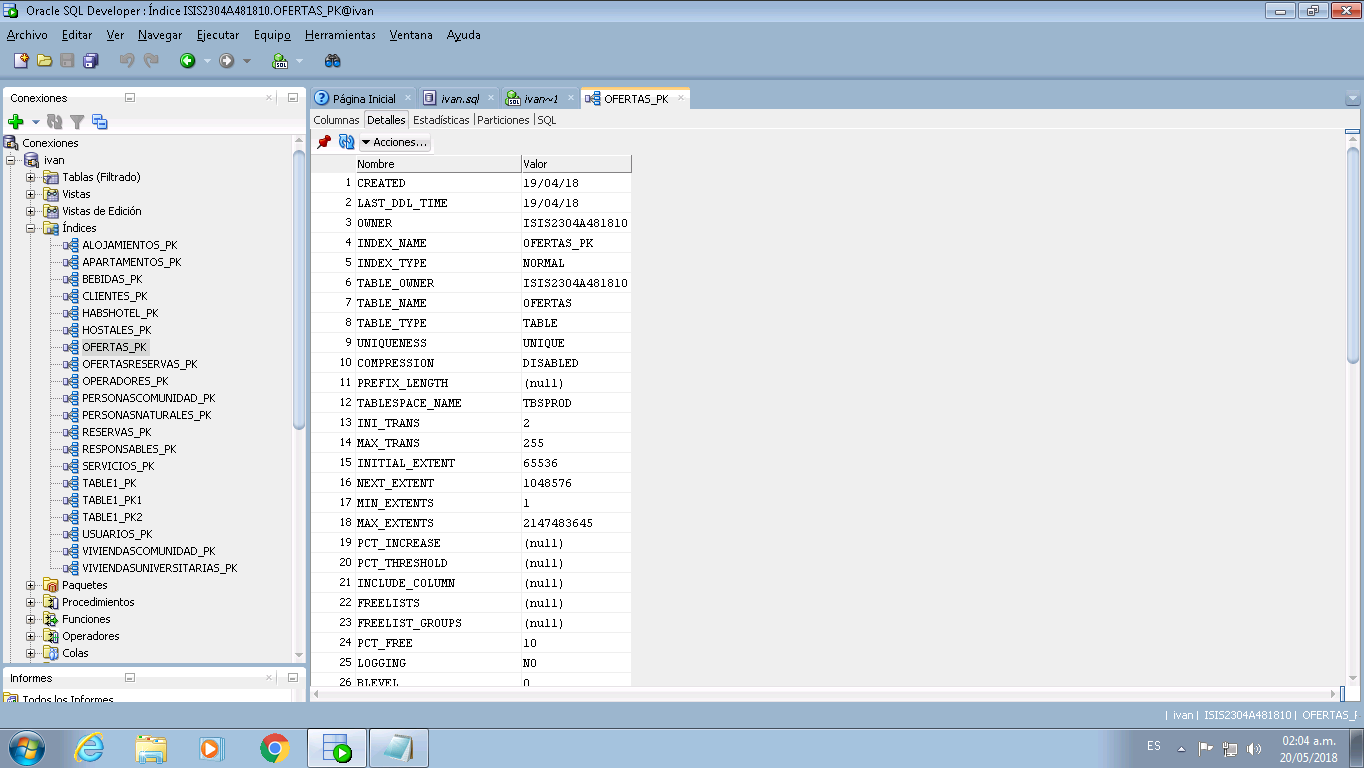
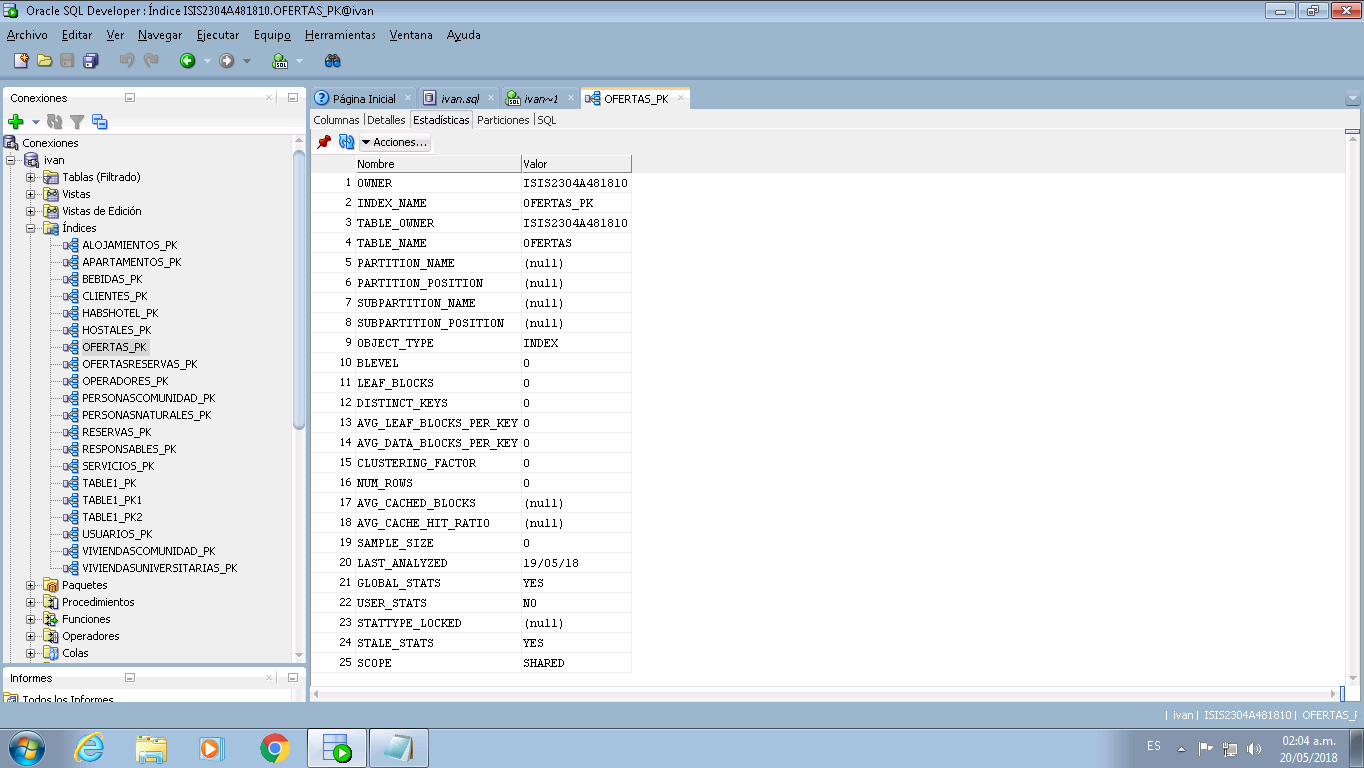
**alojamientos**



**habshotel**



**Ofertas**



Las ofertas colectivas que se deben poder deshabilitar y habilitar.

**3 Casos de uso adicionales:**

**Registrar una reserva de alojamiento colectiva**

|  |  |  |
| --- | --- | --- |
| Descripción General del Caso de Uso | | |
| Se debe poder registrar una reserva de varios usuarios para un operador | | |
| Entidades Involucradas | | |
| Usuario, Reserva | | |
| Precondiciones | | |
| Existen las reservas necesarias para satisfacer la solicitud del usuario | | |
| Flujo normal de Eventos | | |
|  | **Usuario** | **Sistema** |
|  | Genera una solicitud para algún operador | El Sistema verifica la existencia de alojamientos disponibles para dicho operador |
|  |  | En caso de que existan empieza a recompilar la información de los alojamientos respecto a las solicitudes del usuario cliente |
| Post-condiciones principales del caso de uso | | |
| Se ha registrado en la tabla de reservas la solicitud del/los usuarios . | | |
| Caminos de Excepción | | |
| Si no existen alojamientos disponibles para cumplir la solicitud hecha. | | |

**Cancelar una reserva colectiva**

|  |  |  |
| --- | --- | --- |
| Descripción General del Caso de Uso | | |
| Un operador debe poder cancelar una reserva que muchos usuarios hayan hecho para sus alojamientos | | |
| Entidades Involucradas | | |
| Operador, Reserva, Usuario | | |
| Precondiciones | | |
| El operador cuenta con una reserva colectiva registrada en la base de datos | | |
| Flujo normal de Eventos | | |
|  | **Usuario** | **Sistema** |
|  | El operador genera una solicitud para cancelar una reserva colectiva registrada en su base de datos. | El sistema recibe la solicitud del operador |
|  |  | El sistema revisa las reservas involucradas |
|  |  | El sistema empieza a cancelar reservas consecutivamente usando el requerimiento desarrollado en la iteración anterior |
|  |  | El sistema envía un mensaje de confirmación al cliente si se eliminaron todas las reservas involucradas |
| Post-condiciones principales del caso de uso | | |
| Se han eliminado todas las reservas involucradas con una reserva colectiva. | | |
| Caminos de Excepción | | |
| N/A | | |

**Deshabilitar Oferta de Alojamiento**

|  |  |  |
| --- | --- | --- |
| Descripción General del Caso de Uso | | |
| Se debe poder deshabilitar una oferta de cualquier operador dentro del sistema, incluso si hay reservas referenciadas a esta oferta | | |
| Entidades Involucradas | | |
| Oferta, Reserva, Operador, Usuario, Alojamiento, Servicio | | |
| Precondiciones | | |
| Existe la oferta por parte del operador | | |
| Flujo normal de Eventos | | |
|  | **Usuario** | **Sistema** |
|  | El operador genera la petición al sistema de eliminar todas las ofertas asignadas a un alojamiento específico. | El sistema verifica la existencia de una oferta asignada al alojamiento registrado por parámetro |
|  |  | En caso de que exista el sistema elimina todas las ofertas respectivas al alojamiento especificado |
|  |  | El sistema verifica en la base de datos todos los alojamientos y servicios registrados a esa oferta y los retira de la base de datos. |
|  |  | El sistema acumula todas las reservas relacionadas con la oferta retirada |
|  |  | El sistema empieza a reasignar las reservas compiladas en el paso respecto a la disponibilidad del operador |
|  |  | El sistema genera un mensaje de confirmación al usuario con la información de las reservas que fueron reasignadas y en el caso de que existan, las que no fueron reasignadas |
| Post-condiciones principales del caso de uso | | |
| Se eliminó la oferta de alojamiento solicitada | | |
| Caminos de Excepción | | |
| N/A | | |

**Rehabilitar una oferta de alojamiento**

|  |  |  |
| --- | --- | --- |
| Descripción General del Caso de Uso | | |
| Se selecciona una oferta de alojamiento para habilitar | | |
| Entidades Involucradas | | |
| Oferta, Operador | | |
| Precondiciones | | |
| Existe la oferta a habilitar | | |
| Flujo normal de Eventos | | |
|  | **Usuario** | **Sistema** |
|  | Algún operador solicita la habilitación de la oferta de alojamiento | El sistema verifica la existencia de la oferta de alojamiento |
|  |  | El actualiza el estado de la oferta a ‘HABILITADA’ |
| Post-condiciones principales del caso de uso | | |
| El estado de la oferta de alojamiento solicitada cambia a ‘HABILITADA’ | | |
| Caminos de Excepción | | |
| No existe la oferta de alojamiento solicitada. | | |

**Analizar la operación de uniandes**

|  |  |  |
| --- | --- | --- |
| Descripción General del Caso de Uso | | |
| Un cliente puede ser cancelar una reserva que este haya realizado | | |
| Entidades Involucradas | | |
| Algún Cliente, Algún Operador, Base de Datos de AloHandes | | |
| Precondiciones | | |
| El cliente se encuentra en la base de datos, existen reservas por parte del cliente | | |
| Flujo normal de Eventos | | |
|  | **Usuario** | **Sistema** |
|  | El cliente indica cual reserva desea cancelar | El sistema verifica que la oferta pertenezca al cliente en cuestión |
|  |  | El sistema verifica la fecha y la duración de dicha reserva |
|  |  | El sistema realiza calcula los cobros de penalización dependiendo de la antelación y la duración de la reserva. |
|  |  | El sistema retira la reserva y se alerta al operador. |
| Post-condiciones principales del caso de uso | | |
| El cliente ya no posee esta reserva. Se ha generado un cobro dependiendo de la fecha de cancelación y la duración del contrato. | | |
| Caminos de Excepción | | |
|  | | |

En primer lugar, existe una diferencia fundamental en la manera de expresar las herencias, a pesar de que sí existen en el modelo, no se representan como tal, sino que solo se menciona la representación del FK existente cuando estas se presentan. Adicionalmente se enfatiza en la existencia de las singularidades de las id, cosa que no se concebía con tal detalle en nuestro modelo inicial. Por último, es importante enfatizar en que puede haber ciertas discordancias en la implementación del diseño pues no se realizan todos los Checks adecuados para evitar atributos de herencia huérfanos.

# Generación de Tablas

Para generar las tablas se solicita usar el archivo script “creador de tablas.sql” que se encuentra en la carpeta de documentos. Si se desea borrarlas, se puede usar el archivo “Borrador.sql”.

# Población

Para poblar las tablas, se solicita usar el archivo “Poblador.sql” que se encuentra en la carpeta de documentos.