RF12:

Requerimiento:

\*Se revisa qué habitaciones de cierto tipo están en la tabla Habitaciones y que al mismo tiempo NO están en Reservas\_Habitaciones.

\*Se inserta en ReservaConvencion el id habitaciones que cumplen la condición anterior, junto con el id de la convencion, id del hotel.

\*El proceso para los Servicios es machete, pues no se revisa que el servicio esté disponible sino que se agrega directamente el servicio a una reserva (este machete ya que no usamos fechas).

Pruebas:

* Las inserciones de este requerimiento funcionan perfectamente, anexo en el paquete de pruebas del proyecto de corren las pruebas verificando la funcionalidad de estas, a excepción de algunos cumplimientos de reglas de negocio.

RF13

Requerimiento:

Debido a que la reserva de una habitación por parte de una convenció no involucra Clientes, entonces no se puede crear una ReservaHabitacion.  El cliente solo se involucra cuando llega a tomar una habitación. El acercamiento a este RF es almacenar en una tabla el id de la habitacion que esta siendo guardada para un Cliente que venga para dicha Convencion y la reserve.

\*Se borra de la tabla ReservaConvencion las tuplas que coincidan con unos ids que llegan por parametro (se hace un ciclo para habitaciones y otro para servicios).

Pruebas:

* Al ser este un requerimiento más lógico que persistente, no pudimos anexar pruebas de CRD para este requerimiento.

RF15

Requerimiento:

Solo se agregará a la tabla ReservaMantenimiento que se empezó el mantenimiento a una lista de habitaciones y/o servicios. El resto de cosas no se hará.

\*Se agregan tuplas a la tabla ReservaMantenimiento con sus respectivos atributos.

Pruebas:

* este documento tiene pruebas de CRD anexas en el proyecto para la inserción y demás de una reserva de mantenimiento.

RF16

Requerimiento:

Este requerimiento es el menos complejo de todos. Solo es cambiar un atributo booleano a las tuplas que correspondan con un id que llega por parametro. Se hacen dos ciclos uno para habitaciones y otro para servicio. Se cambia el estado de en mantenimiento a 0, FALSE.

Pruebas:

* este requerimiento se prueba con los iniciales de la iteración uno cambiando el atributo en la tabla correspondiente. Cambia el estado de un booleano...