

Para asegurar la calidad del modelo, utilizamos varias recomendaciones de herencia que nos habian hecho en la anterior iteracion.

Colocamos los atributos de nombre en la clase de usuario para asi mejorar la normalizacion de nuestro modelo. Consideramos que el modelo se encuentra en BCNF

Para mejorar las pruebas de nuestro codigo, implementamos mas inserts, de forma que cada requerimeinto y prueba pudiese lanzar alguna respuesta.

Implementamos mas atributos en oferta, como descripcion y disponible.

-----

para armar una reserva masiva, se entra por parametro el numero y características de la reserva, y luego procede a crear un contrato por cada individuo en la reserva, encontrando alojamientos que cumplan con los requerimeintos

para cancelar una reserva masiva se usa un atributo de id que todos los contratos de la reserva masiva tienen en comun, para dentificar los contratos y las ofertas y deshabilitarlos. de forma que NO SE BORRAN mas si quedan deshabilitados. a los clientes luego se les vuelve a aplicar el algoritmo de reserva masiva para realojarlos.

para habilitar y deshabilitar una reserva nos enfocamos en que el sistema NO BORRE los alojamientos.

Para garantizar prodiedades ACID utilizamos nivel de aislamiento en serializable, ya que es la mejor forma de manejar algo tan sensible como una reserva masiva, donde si tan solo uno de los individuos no recibe alojamiento es un fracaso total.

Logramos TODO excepto el requerimiento IV de consulta.