

Primera iteración
Sistemas Transaccionales

Universidad de los Andes

Marzo 21 del 2018

Daniel Rengifo 201533922

Santiago Rangel 201632011

Modelo Conceptual: el modelo conceptual de la aplicación, contiene un total de 14 clases en las cuales podemos incluir toda la lógica de la aplicación encargada. Sin embargo, notamos que la clase “alojamiento” de la cual heredaran los 3 posibles alojamientos, hubiera facilitado la fabricación del software. Dicha iluminación, lamentablemente, ocurrió en una etapa avanzada del proyecto, por lo que no pudimos implementarla. Ahora bien, esta experiencia nos enseña a como modelar mejor para el futuro de nuestras aplicaciones.

Entidades Persistentes: en general, todas las clases en Alohandes son persistentes, excepto por la clase de control de Alohandes. Las clases en el modelo de negocio se fabricaron bajo un estandarte claro de necesidad. Lo cual implica que las clases contenida dentro de sí, no son clases de naturaleza ejecutiva, sino judicial.

Modelo de Datos Relacional: Para este modelo, tanto el algoritmo de Chen ejecutado por Architect Enterprise, como nuestro propio intento de realizar el modelo a mano, consiguió exactamente el mismo resultado de 16 tablas. Añadiendo dos tablas de HabitaciónOfrece y Hotel Ofrece, para normalizar las relaciones planteadas por la multiplicidad del modelo.

Bases de datos (SQL): Para este punto, tuvimos que generar varias instancias del mismo proyecto, antes de alcanzar una definitiva. La complejidad del proyecto tendía a complicar la colocación de atributos y de insertar los ejemplos necesarios para las pruebas de postman. Encontramos problemas al nombrar varios atributos. Sin embargo en el final del proyecto consideramos que nuestro archivo SQL era robusto y bien clarificado con 5 ejemplos como mínimo en cada tabla.

Requerimientos consulta: si bien NO logramos que los sql funcionaran en la aplicación, logramos que en la base de datos y sql developer funcionara perfecto.

R1:

Para registrar los operadores se usó el método POST para todas las clases que heredan de operador y para esta clase. Al realizar este requerimiento quedan las instancias persistidas en la base de datos.

R2:

Para registrar las propuestas de alojamientos es usó el método de POST para las clases de todos los alojamientos en el sistema con sus ofertas.

R3:

Para registrar las personas se usa el método de POST de todas las clases que sean usuario en el sistema.

R4:

Para registrar una reserva se usa el método POST de la clase contrato (análogo a reserva) que relaciona un operador y su alojamiento con un cliente en unas fechas particulares.

R5:

Para cancelar una reserva se usa el método DELETE de contrato, pero se necesita conocer el id del contrato que se desea cancelar.

R6:

Para retirar una oferta de alojamiento se usa el método DELETE de la clase oferta, pero se necesita conocer el id de la oferta que se desea retirar del sistema.