

Assumptions and Explanation:

**Weak relations:**

Favorite cannot exist without User

Manager cannot Exist without manager (Exception is CEO who reports to his own manager\_id)

Review cannot exist without User.

Villa cannot exist without Owner. If an Owner leaves - his Villas are no longer up for reservation or he may have sold them to another Owner, in which case those become new Villas for those Owners. If it is desired that the Villas remain after their Owners leave - we can achieve that with a strong relation (I understand the concept - I've just implemented my own version).

Reservation cannot exist without User, Villa, and Price.

Feature cannot exist without Villa. But Feature can continue to exist even if an Owner has sold all his property and left the system.

Price cannot exist without a Villa.

Pricing is dependent on an Owner and cannot exist without an Owner setting it.

Phone cannot exist without a person.

Description cannot exist without a Person.

Coupon cannot exist without a Villa.

Coupons\_applied cannot exist without a reservation or a coupon.

Phone cannot exist without a Person.

Description cannot Exist without a Person.

Favorites cannot exist without a Villa or a User.

**Strong Relations:**

Log can exist in history even without an Owner (eg: Owner has sold property and left )

Features can exist even after an Owner has left the Database or in the even that he has transferred his property to another owner and become a manager or a regular user.

Reservation can exist in history even after a Villa has been removed from the system.

A User can exist without a Reservation.

Inspection report can exist even after a Manager quits

**Super Class & Sub Classes:**

Every user of the system is a Person.

Person sub classes include User , Owner , Manager - with Partial Completeness for future assignments like Butler or Cleaner.

A manager can be an Owner AND/OR a User

An Owner can be a User AND/OR Manager

A User can be an Owner AND/OR Manager

