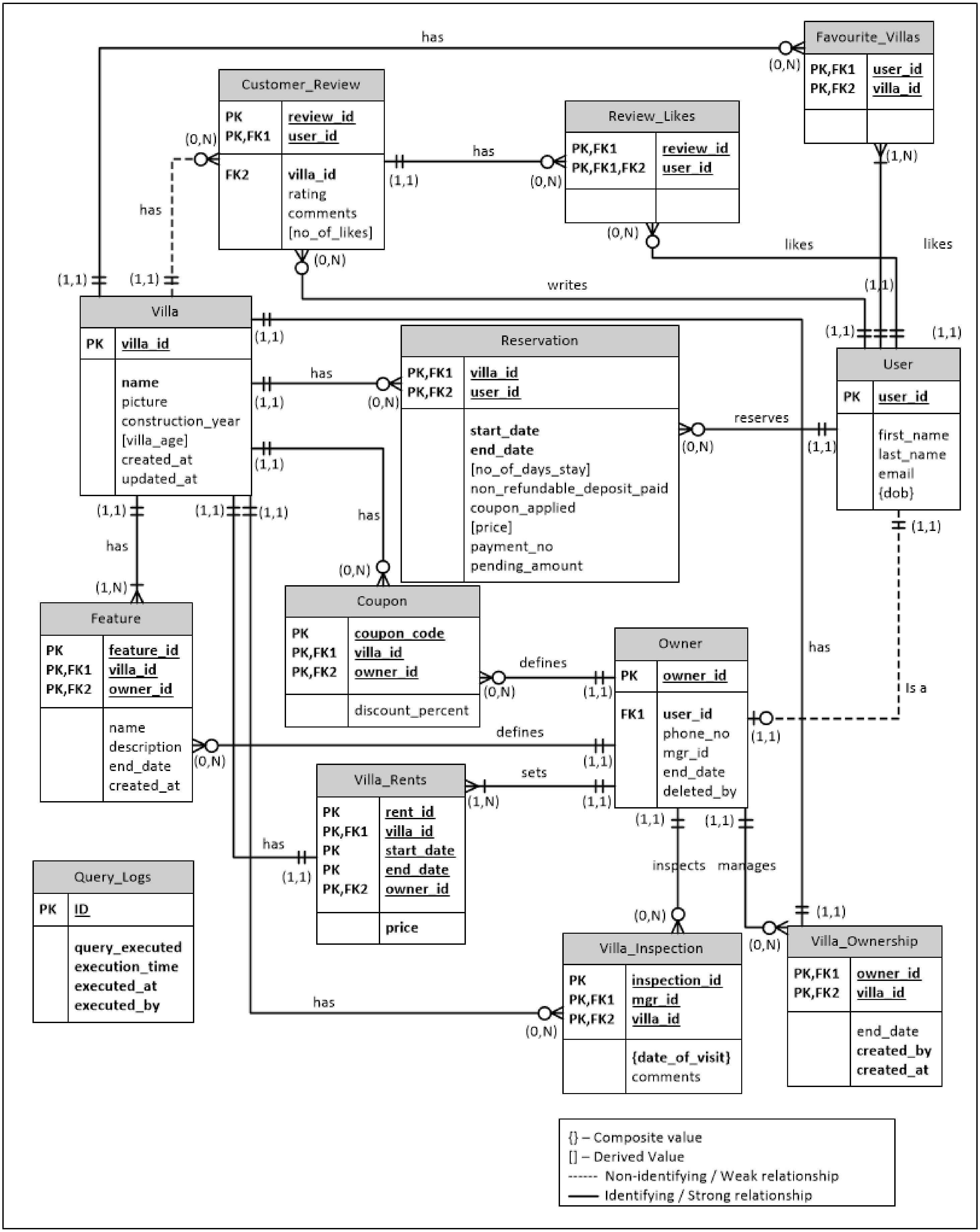
Vacation Rental Website - Entity Relationship Diagram (ERD)



{} – Composite value
[] – Derived Value
------ Non-identifying / Weak relationship
—— Identifying / Strong relationship

Assumptions :

1. User can see only active villas in the application
2. end_date attribute in tables are used for soft delete of the entries. (e.g to make a feature/owner/villa no longer available)
3. Same villa cannot be booked by 2 different users at the same time
4. All managers, owners are users and all managers are owners
5. Owner entity has a unary relationship with manager
6. Owner can add the villa to his pool of villas for maintenance if it is left with no owner
7. Once the user selects the end date of reservation in the application, end date will be updated in the database system as till noon of selected date.
8. User and Owner class inheritance is represented as a 0 -1 relationship between those entities
9. Villa_Inspection:date_of_visit is a DATETIME attribute
10. Each villa has 1 or many feature and 0 or many reviews
11. Each user can write 0 or many reviews and likes 0 or many reviews
12. Villa_Rents:start_date, Villa_Rents:end_date are made as primary key to avoid ambiguity of price during reservation. For a given date there exists only one rent details.

Composite Attributes:

1. {User:dob} - it can be further divided into month, day and year
2. {Villa_Inspection:date_of_visit} - it can be further divided into month, day, year and time

Derived/Calculated Attributes:

1. [Reservation:price] is calculated based on the [Villa_Rents:price] and the coupon applied
2. [Customer_Review:no_of_likes] is derived from the other entity Review_Likes
3. [Villa:age] is calculated based on the construction year and current year.
4. [Reservation:non_refundable_deposit_paid], [Reservation:pending_amount] is calculated based on the other derived attribute [Reservation:price]
5. [Reservation:no_of_days_stay] is calculated based on the Reservation:start_date and Reservation:end_date

Atomic/Simple attributes:

All the values that are not derived/calculated, multivalued and composite are atomic values.

Multivalued Attributes:

Multivalued attributes are represented as entities.
Fabourite_Villas and Features are multivalued attributes which are represented as entities.

Relationships:

1. Identifying / Strong relationships are represented by solid lines
2. Non-Identifying / Weak relationships are represented by dotted lines

Participation:

1. (0,1) or (0,N) represents optional participation (0 or more)
2. (1,1) or (1,N) represents mandatory participation (1 or more)

Class and Subclass:

User is a superclass entity and owner is derived from it.

Entities, Relationships, Cardinalities and Connectivity :

Villa

One of the main entity in the database system. It holds all the necessary information about the villa. Also it gets features, coupons and rental details from its neighbouring entities.

User

Another major entity. It captures the details about the user. The user can subscribe to the vacational rental website and book villas as per their need. The user can also have a set of preferred villas which helps to book villas at ease.

Reservation

It is a bridge/linking/associative entity between the two major entities User and Villa. It will capture the reservation/booking details. 'Number of days of stay' is calculated based on the start and end date of stay. Each reservation is associated with a single user and a single villa. User can book 0 or more villas at that same time. And each villa can be booked once for a given start and end date.

Customer Review

Each user can rate and write a review regarding their experience at a particular villa. These reviews can be liked by other users. The reviews are listed on the increasing order of their no_of_likes (It will be handled at the application level). Each villa can have 0 or more reviews and each user can write review about 0 or more villas.

Review Likes

It keeps track of the user review likes. Each user can like 0 or more reviews. This entity also helps to manage the list of reviews that are liked by the particular user.

Favourite Villa

It manages the list of favourite villas of all users. Each user can have 1 or more villas as their favourite

Owner

A user who can potentially manage the villas. Owner defines/deletes the price, features and coupon details of the villas. Owner can also add/remove a villa from his pool of villas. Each owner except the CEO is managed by a manager. Once the owner removes the villa from his pool, it is likely that the other owner can add the same villa under his/her pool.

Manager

Another privileged user who ideally owns the villas and manages the owners. Each manager can manage 0 or more owners. It is represented in ERD as a unary relationship. As per the assumption, each manager is a owner. CEO is the only owner who does not have a manager.

Coupon

Owner can define the coupon for the particular villa which includes the coupon code and discount percent. The user can use the coupon code to get a discount. Each villa can have 0 or more coupon codes and each owner can define 0 or more codes.

Villa Rents

Owner defines the villa rents for the particular time period. These will be updated often by the owner. (At the application level, villa rents for the respective dates are displayed along with villa details) Each villa is associated with 0 or more rents as price changes often. Each owner can define 0 or more rents for a villa.

Villa Ownership

This entity helps to maintain the list of villas managed by a owner. Each owner can manage 0 or more villas.

Villa Inspection

Manager can visit a villa and write comments about the visit. Each manager can visit a villa 0 or many times.

Query Logs

This entity can be used to log all the database system activities in terms of what queries are executed by whom and when and its execution time. This details can be used to analyze the database performance, also helps to take further measures to improve the performance.