Animal Shelter Database System Project

Project Objective:

Organizing tons of data in every organization is important to effectively manage operations and streamline processes. QUELIFE is an animal shelter organization for which we will be designing a database system. This organization has multiple branches spread across Wisconsin, Illinois, Missouri, Montana, California, Texas, and Nevada. This database will encompass primary details related to facilities, animal descriptions and transfer details, sponsor details and employee details.

Requirement Analysis:

We are analysing the requirements to setup a database for this organization. We have multiple entities to consider along with their relationships. Here we have multiple branches of shelter at multiple locations with various facilities. Animal related records including animal information, animal features, adoption records and surrender details are need as well as the history of each animal. We are enabling the option to transfer of animals from one branch to another. Employee details, sponsor details and payment details are required to track the working of the shelter.

Assumptions:

1. Animals can be transferred from one branch to another
2. Each shelter branch will have multiple employees
3. Each shelter branch will have a branch manager
4. Employees work only in one shelter branch only (no transfer of employees)
5. An animal can be adopted as well as surrendered to the shelter for various reasons multiple times
6. A vaccination is given each time an animal is detected with any disease
7. Sponsors can support multiple shelter branches
8. Full payment must be done in a single payment for adoption of an animal
9. Each medical condition of an animal is treated by only one veterinarian
10. Veterinarians can treat multiple conditions of multiple animals
11. Each transfer of an animal involves transfer from one branch to another
12. Each veterinary doctor can have multiple specializations

Entities and Relationships:

1. **shelter_branch:**
   - Entity Description: This entity contains details about the geographic location and details of each branch
   - Attributes: shelter_id, shelter_name, address, pin code, shelter_contact
   - Relationships:

- One-to-one relationship with facilities
- Many-to-many relationship with animal_info
- One-to-many relationship with employees_staff_details
- Many-to-many relationship with animal_transfer_records
- Many-to-many relationship with sponsor

2. **facilities:**
   - Entity Description: This entity represents amenities operated by each branch.
   - Attributes: fid, f_capacity, f_ani_count, f_area, play_areas, temp_control, outdoor_space
   - Relationships:
     - One-to-one relationship with shelter_branch

3. **animal_info:**
   - Entity Description: This entity contains information about the animal.
   - Attributes: animal_id, a_type, a_breed, a_dob, date_of_acquisition, a_vaccine_status.
   - Relationships:
     - Many-to-many relationship with shelter_branch (as we have transfer facility)
     - One-to-one relationship with animal_features
     - Many-to-many relationship with surrender_details
     - Many-to-many relationship with adoption_details
     - One-to-many relationship with animal_transfer_records
     - Many-to-many relationship with nutrition
     - Ternary relationship with history of the animal which includes: grooming history, training history and medical history

4. **animal_features:**
   - Entity Description: This entity contains physical attributes of the animal.
   - Attributes: animal_id, a_color, a_weight, a_height, a_eye_color, a_condition
   - Relationships:
     - One-to-one relationship with animal_info

5. **adoption_details:**
   - Entity Description: This entity contains information about the adoption of an animal
   - Attributes: ad_ID, animal_id, ad_name, ad_contact, ad_date, ad_fee, ad_criminal_record.
   - Relationships:
     - Many-to-many relationship with animal_info
     - One-to-one relationship with payment_details

6. **surrender_details:**

- o Entity Description: This entity contains details about returning of the animal to the shelter
- o Attributes: s_id, s_date, s_name (person who brings animals), s_reason.
- o Relationships:
  - ▪ Many-to-many relationship with animal_info

7. **nutrition:**
   - o Entity Description: This contains information about diet and nutrition of each animal
   - o Attributes: food_id, food_type, food_quantity
   - o Relationships:
     - ▪ Many-to-many relationship with animal_info

8. **training_history:**
   - o Entity Description: This entity contains details about training the animals
   - o Attributes: training_ID, trainer_name, training_date, training_type, training_stage.
   - o Relationships:
     - ▪ Many-to-one relationship with animal_info
     - ▪ Many-to-many relationship with employees_staff_details

9. **grooming_history:**
   - o Entity Description: This entity contains details about grooming the animals
   - o Attributes: grooming_id, groomer_name, grooming_date, grooming_type.
   - o Relationships:
     - ▪ Many-to-one relationship with animal_info
     - ▪ Many-to-many relationship with employees_staff_details

10. **medical_history:**
    - o Entity Description: This entity contains details about the medical history of each animal
    - o Attributes: medical_id, animal_id, disease, vet_id, treatment_date, vaccination_id
    - o Relationships:
      - ▪ Many-to-one relationship with animal_info
      - ▪ Many-to-many relationship with vet_id
      - ▪ One-to-one relationship between medical_history and vaccination

11. **vaccination:**
    - o Entity Description: This entity contains details about the vaccination of each animal

o Attributes: vaccination_id, vaccination_name, vaccination_date, vaccination_dosage
o Relationships:
  ▪ One-to-one relationship between medical_history and vaccination

12. **employees_staff_details:**
   o Entity Description: This entity contains details about each worker or employee at the animal shelter QUELIFE
   o Attributes: emp_id, emp_name, emp_level, emp_contact, emp_department, emp_training, emp_education, emp_criminal_record, emp_salary
   o Relationships:
     ▪ Many-to-one relationship with shelter_branch
     ▪ Many-to-many relationship with grooming_history
     ▪ Many-to-many relationship with training_history
     ▪ Unary relationship – manages – branch manager

13. **sponsor:**
   o Entity Description: This entity contains details about sponsors given to shelter_branch
   o Attributes: sponsor_id, sponsor_name, sponsor_contact, sponsorship_type.
   o Relationships:
     ▪ Many-to-many relationship with shelter_branch
     ▪ One-to-many relationship with payment_details

14. **payment_details:**
   o Attributes: payment_ID, amount, payment_Date, payment_type.
   o Relationships:
     ▪ One-to-one relationship with adoption_details
     ▪ Many-to-one relationship with sponsor

15. **veterinary:**
   o Entity Description: This entity contains details of the veterinarian
   o Attributes: vet_id, vet_name, vet_contact, education, specialization, work_exp
   o Relationships:
     ▪ Many-to-many relationship with medical_history

16. **animal_transfer_records:**
   o Entity Description: This entity contains details of transfer of animals from one branch to another
   o Attributes: transfer_id, transfer_date, transfer_from, transfer_to, animal_id.
   o Relationships:
     ▪ Many-to-one relationship with animal_info
     ▪ Many-to-many relationship with shelter_branch
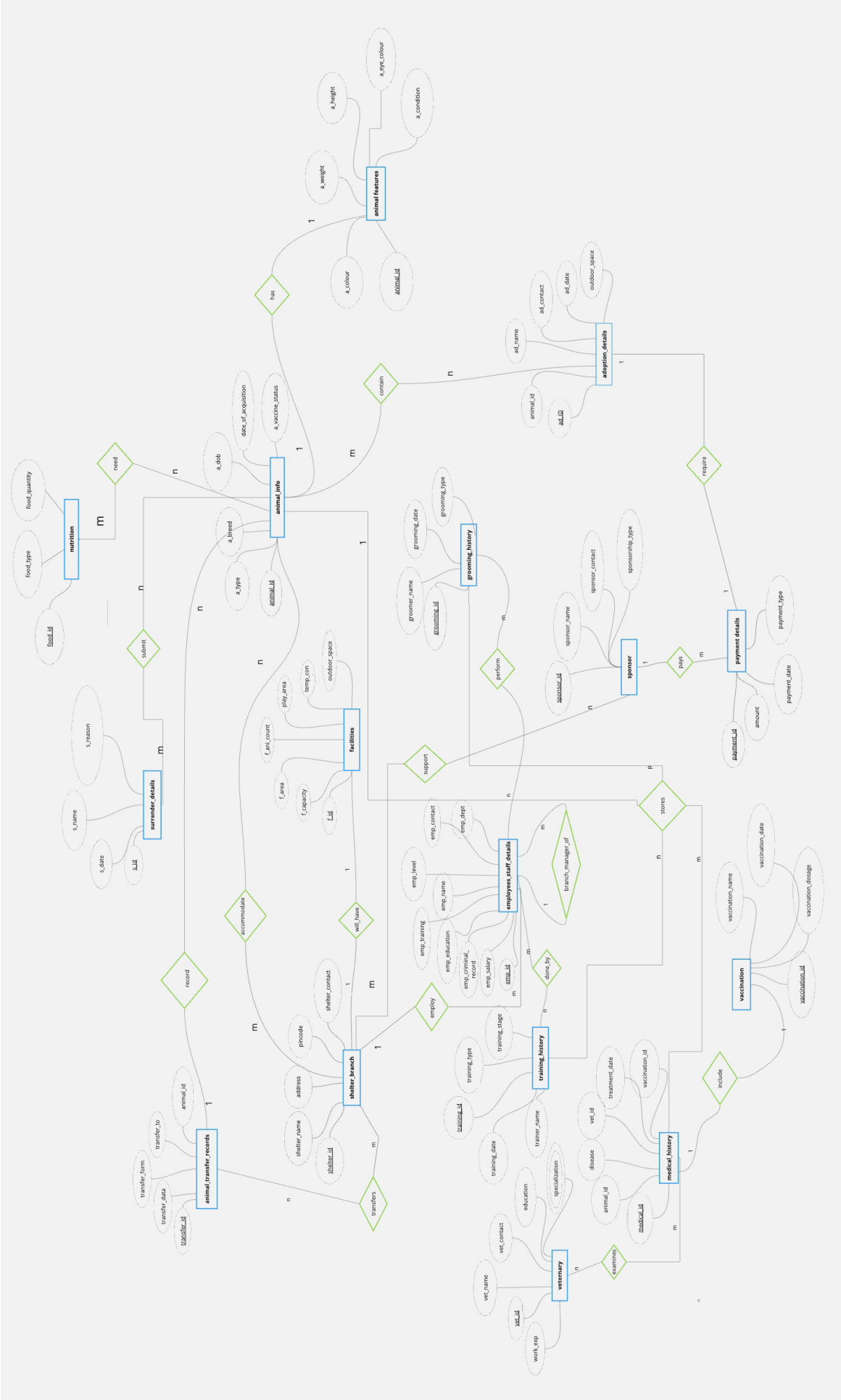
Relationship Summary:

Entities	Degree of Relationship	Relationship type	Relation name
animal_info - animal_features	Binary	One-to-one	has
animal_transfer_records - animal_info	Binary	Many-to-one	record
grooming_history - employees_staff_details	Binary	Many-to-many	perform
nutrition - animal_info	Binary	Many-to-many	need
shelter_branch - facilities	Binary	One-to-one	will_have
shelter_branch - animal_info	Binary	Many-to-many	accommodate
shelter_branch - employees_staff_details	Binary	One-to-many	employ
shelter_branch - animal_transfer_records	Binary	Many-to-many	tranfers
sponsor - shelter_branch	Binary	Many-to-many	support
sponsor - payment_details	Binary	One-to-many	pays
surrender_details - animal_info	Binary	Many-to-many	submit
training_history - employees_staff_details	Binary	Many-to-many	done_by
veterinary - medical_history	Binary	Many-to-many	examines
animal_info - adoption_details	Binary	Many-to-many	contains
medical_history - vaccination	Binary	One-to-one	include
payment_details - adoption_details	Binary	One-to-one	require
animal_info - training_history		One-to-many	
animal_info - grooming_history	Ternary	One-to-many	stores
animal_info - medical_history		One-to-many	
employees_staff_details	Unary	One-to-many	branch_manager_of

E-R Diagram:

Entity-Relationship Diagram of the Animal Shelter Database System:

Link for higher resolution: E-R Diagram using Miro

Entity-Relationship Diagram

Entity-Relationship Diagram showing entities: animal_features, animal_info, nutrition, surrender_details, facilities, shelter_branch, employees_staff_details, training_history, veterinary, medical_history, vaccination, grooming_history, sponsor, payment_details, adoption_details, and animal_transfer_records, connected through relationships: has, need, submit, contain, require, accommodate, will_have, support, stores, perform, pays, record, transfers, employ, done_by, include, examines, and branch_manager_of.

Individual Contribution:

As we began working on the QUELIFE animal shelter database, it became clear that we needed a system that went beyond just tracking names and dates. Animals, like humans have complex needs and histories that can greatly affect their future.

We began by mapping out the relationships between the types of information we needed to track. For each animal, we want to know not only their breed and age, but also their medical history, behavioural habits, and preferences. This allowed me to make connections of one-to-one relationship between "animal info" and "animal features" ensuring we could capture the unique aspects of each animal in detail.

The animals' lives are connected to their environments and the people around them. They interact with people and places, which is why I also make many-to- many connections between "shelter_branch" and "animal_info," and between "surrender_details" and "animal_info.

Perhaps the most challenging part was figuring out how to integrate financial information into the system in a comprehensive and user-friendly way. Also I wanted to ensure it is easy for potential approvers, So they would understand the process and the cost. Which is why I linked payment_details to adoption_details. Not only did this facilitate the adoption process but it also added transparency.

This concerted effort to develop and detail the system is not only necessary to meet the current operational needs of the QUELIFE Animal Shelter but also it allows the program to adapt to future requirements. It also supports shelter 's mission to promote animal welfare and successful adoptions.