**1. Project Overview**

The **Animal Shelter Management System** aims to streamline operations within an animal shelter by automating data management processes. The system will manage various aspects, including animal profiles, adoption records, medical history, volunteer schedules, and shelter resources like food and medications. The purpose is to ensure efficient tracking of animals, adopters, and shelter operations. The scope includes modules for animal care, adoption management, staff and volunteer management, and resource allocation. This system will improve the shelter’s operations and enable quicker, data-driven decisions. The entities and relationships involved will represent key shelter components like animals, adopters, staff, and resources.

**2. Identify Entities and Attributes**

**Entities and Attributes**

1. **Animals**
   * **Primary Key**: animalID
   * **Attributes**: name, species, age, medical history, dietary requirements
2. **Dogs**
   * **Primary Key**: animalID (foreign key to Animals)
   * **Attributes**: breed, size, vaccination status, grooming needs, temperament
3. **Cats**
   * **Primary Key**: animalID (foreign key to Animals)
   * **Attributes**: breed, color, vaccination status, personality traits, grooming needs
4. **ShelterEmployees**
   * **Primary Key**: employeeID
   * **Attributes**: name, role, hire date, contact information, schedule
5. **AdoptionRequests**
   * **Primary Key**: requestID
   * **Attributes**: adopterName, animalType, adoptionDate, status, approvalDate
6. **Adopters**
   * **Primary Key**: adopterID
   * **Attributes**: name, contact information, address, employment status, previous pets

**3. Define Relationships**

**Relationships Between Entities:**

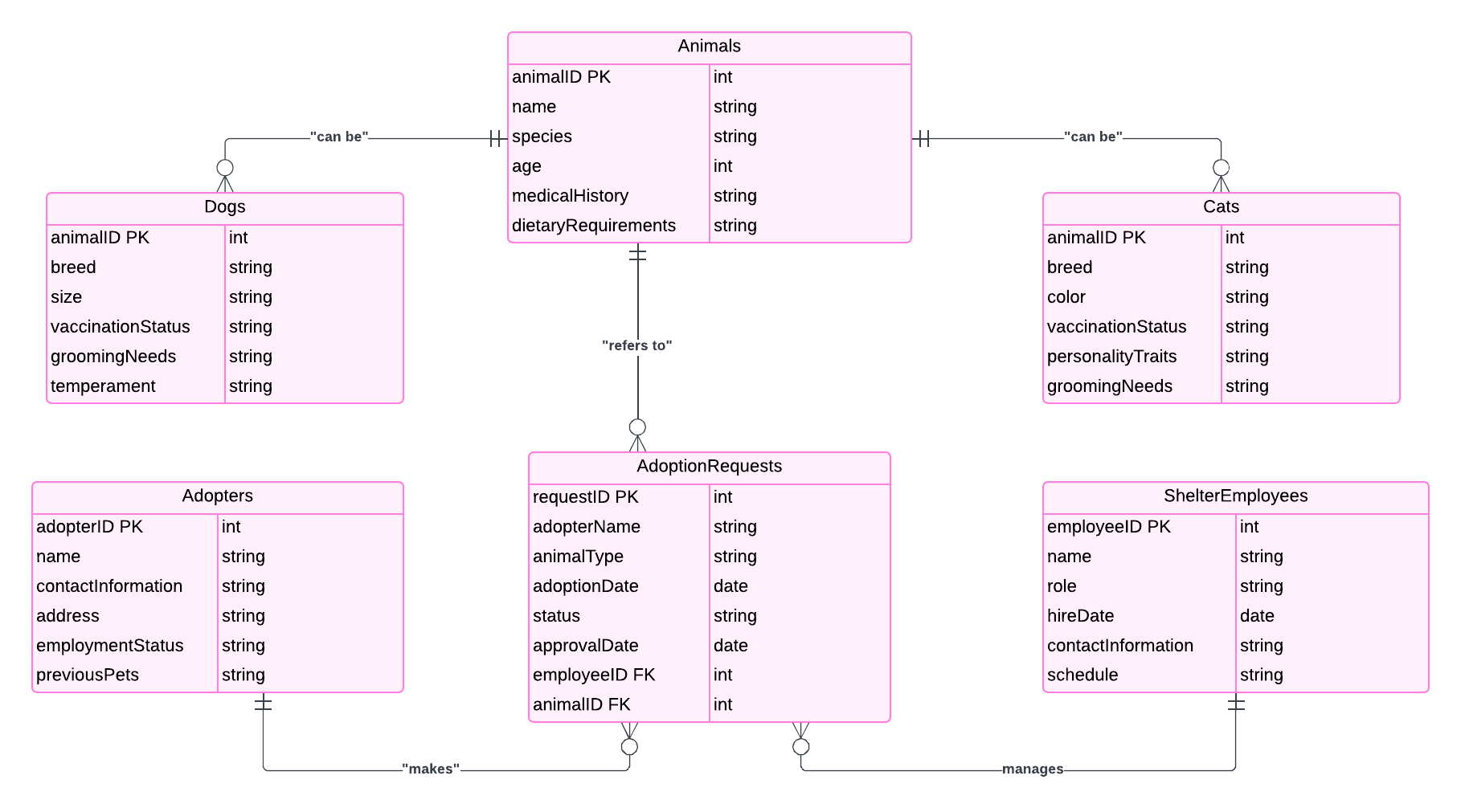
1. **Animals - Dogs** (One-to-One)
   * **Degree**: One animal can be a dog.
   * **Cardinality**: One-to-One (each dog is an animal, but not all animals are dogs).
   * **Foreign Key**: dog’s animalID references animalID in Animals.
2. **Animals - Cats** (One-to-One)
   * **Degree**: One animal can be a cat.
   * **Cardinality**: One-to-One (each cat is an animal, but not all animals are cats).
   * **Foreign Key**: cat’s animalID references animalID in Animals.
3. **ShelterEmployees - AdoptionRequests** (One-to-Many)
   * **Degree**: One shelter employee can handle many adoption requests.
   * **Cardinality**: One-to-Many (an employee can process multiple adoption requests).
   * **Foreign Key**: employeeID in AdoptionRequests references ShelterEmployees.
4. **AdoptionRequests - Animals** (One-to-Many)
   * **Degree**: One adoption request is linked to one animal.
   * **Cardinality**: One-to-Many (one animal can be linked to multiple adoption requests, but each request is for a single animal).
   * **Foreign Key**: animalID in AdoptionRequests references Animals.
5. **Adopters - AdoptionRequests** (One-to-Many)
   * **Degree**: One adopter can make multiple adoption requests.
   * **Cardinality**: One-to-Many (an adopter can request multiple animals, but each request is from one adopter).
   * **Foreign Key**: adopterID in AdoptionRequests references Adopters.

**Er diagram**

**The ER diagram should include:**

* **Entities: Animal, Dogs, Cats, Shelter Employees, Adoption Request, Adopters.**
* **Relationships: One-to-one, One-to-many, Many-to-many**

### Attributes: Marking primary keys and relevant attributes for each entity.

****

Our ER diagram does not contain any fan traps or chasm traps, and the relationships are correctly represented for the intended functionality of our animal shelter database.

**Documentation**

Explanation of the ER Diagram

This ER diagram represents an animal shelter database system aimed at managing the adoption process for dogs and cats. Key entities include Animals, Dogs, Cats, ShelterEmployees, Adopters, and AdoptionRequests. The Animals entity serves as a general category, with Dogs and Cats as specific subcategories that inherit animalID as a foreign key. ShelterEmployees manage animals, while Adopters can submit multiple AdoptionRequests.

**Challenges Faced**

A significant challenge was ensuring that each AdoptionRequest could be linked correctly to both the Animals and Adopters entities while maintaining unique identifiers. Additionally, defining the relationships between general and specific entities (e.g., Animals to Dogs/Cats) required careful consideration to avoid redundancy. The design ensures clear data flow for adoption requests, management responsibilities, and animal tracking.