**Project Workflow**

1. Home Page
   * Display featured pets
   * Navigation options:
     + View Pets
     + About Us
     + Adoption Process
     + Contact Us
     + User Login/Signup
2. User Login/Signup
   * Login
     + Enter email and password
     + [If successful] Redirect to User Dashboard
     + [If failed] Show error message
   * Signup
     + Enter user details (name, email, password)
     + [If successful] Redirect to User Dashboard
     + [If failed] Show error message
3. User Dashboard
   * View adopted pets
   * View available pets
   * Search for pets (by type, breed, age, etc.)
   * View application status
   * Logout option
4. View Pets
   * List of available pets
   * Filter options (type, breed, age, size)
   * Click on a pet to view details
5. Pet Details Page
   * Display pet information (name, age, breed, description, photos)
   * Adoption button
   * Share on social media option
   * Back to pet list
6. Adoption Process
   * Fill out adoption application form
     + Personal information
     + Home environment details
     + Previous pet ownership experience
   * Submit application
   * Confirmation message
7. Application Status
   * View current status of adoption application (Pending, Approved, Rejected)
   * Notifications for updates via email
8. Admin Dashboard (for Admin Users)
   * Manage pet listings (Add, Edit, Delete)
   * View applications
   * Approve/Reject applications
   * View user accounts
   * Generate reports (adoption statistics)
9. Contact Us
   * Contact form for inquiries
   * Display contact information (email, phone number, address)
10. About Us
    * Information about the organization
    * Mission statement
    * Team members

Additional Features

* User Profile Management
  + Edit user information
  + Change password
* Notifications(optional)
  + Email notifications for application updates
  + Alerts for new pets available for adoption

**Requisite Tables**

1. **Users Table**

This table will store user information for both regular users and admin users.

Query: CREATE TABLE users ( user\_id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(100), email VARCHAR(100) UNIQUE, password VARCHAR(255), role ENUM('user', 'admin') DEFAULT 'user';

**2. Pets Table**

This table will store information about the pets available for adoption.

Query: CREATE TABLE pets (pet\_id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(100), age INT, breed VARCHAR(100), description TEXT, size ENUM('small', 'medium', 'large'), status ENUM('available', 'adopted') DEFAULT 'available');

**3. Adoptions Table**

This table will track adoption applications submitted by users.

Query: CREATE TABLE adoptions (application\_id INT AUTO\_INCREMENT PRIMARY KEY, user\_id INT, pet\_id INT, status ENUM('pending', 'approved', 'rejected') DEFAULT 'pending', personal\_info TEXT, home\_environment TEXT, previous\_pet\_experience TEXT);

**4. Notifications Table**

This table will store notifications for users regarding their application status and new pets.

Query: CREATE TABLE notifications (notification\_id INT AUTO\_INCREMENT PRIMARY KEY, user\_id INT, message TEXT, is\_read BOOLEAN DEFAULT FALSE);

**5. Pet Images Table**

Query: CREATE TABLE pet\_images (image\_id INT AUTO\_INCREMENT PRIMARY KEY, pet\_id INT, image\_url VARCHAR(255));

**6. Contact Inquiries Table**

Query: CREATE TABLE contact\_inquiries (inquiry\_id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(100), email VARCHAR(100), message TEXT, created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP);

7. Adoption Statistics Table (Optional)

Query: CREATE TABLE adoption\_statistics (stat\_id INT AUTO\_INCREMENT PRIMARY KEY, total\_adoptions INT, total\_applications INT);

An entity-relationship (ER) diagram is a graphical representation of the relationships between entities (tables) in a database. The entities are represented by rectangles, and the relationships are represented by diamonds. The lines connecting the entities and diamonds show the cardinality of the relationship. In the ER diagram above, the following relationships are shown:

* One user can have many adoptions (one-to-many).
* One user can have many notifications (one-to-many).
* One pet can have many pet images (one-to-many).
* One pet can have many adoptions (one-to-many).
* A pet can be part of many adoptions (many-to-many).