

Front-End UI/UX Mini Project

1. Title Page

- **Project Title:** *"Pet Adoption Center Webpage"*
- **Submitted By:**
 - *Spoorthi Kumbha 2463059*
 - *Kotte Lavanya 2463035*
 - *Ashika Valencia A 2463072*
- **Course:** *UI/UX Design Fundamentals*
- **Instructor Name:** *Mr.Dhiraj*
- **Institution:** *Christ University*
- **Date of Submission:** *11/08/2025*

2. Abstract

The provided HTML, CSS, and JavaScript code implements a responsive web interface for a **Pet Adoption Center**. The application enables users to browse available pets, search by name or breed, and filter by type and age. Each pet is displayed in a styled “pet card” with essential details. A built-in adoption enquiry form allows users to submit their information, select a pet, and send a message. Although no backend is implemented, JavaScript handles form submission, displays a confirmation message, and dynamically filters the pet listings based on user input. The design incorporates a clean layout, mobile-friendly viewport settings, and interactive features, offering a user-friendly adoption platform prototype.

3. Objectives of the Pet Adoption Center Webpage

- **Display Pet Listings**
 - Present available pets in a clean, visually appealing card format with details like name, breed, type, and age.
- **Enable Search and Filtering**
 - Allow users to search for pets by name or breed.

- Provide filters for pet type (e.g., dog, cat) and age to help users find pets that match their preferences.
- Adoption Enquiry Form
Include a form for users to submit adoption enquiries with required details: name, email, selected pet, and a message.
- Interactive and User-Friendly Interface
 - Use real-time search and filtering so results update instantly without reloading the page.
 - Provide clear form validation and feedback to ensure all required fields are filled in.
- Responsive Design
 - Ensure the layout and styles adjust well for different screen sizes, making the site accessible on desktops, tablets, and mobile devices.
- Form Submission Feedback
 - Show a confirmation message upon form submission, even without a backend, to reassure users their enquiry was received.
- Simple, Maintainable Code Structure
 - Use clear HTML structure, organized CSS for styling, and JavaScript for interactive features, making the code easy to update and expand.

4. Scope of the Project – Pet Adoption Center Website

The Pet Adoption Center web application is designed to provide a user-friendly platform for potential pet adopters to browse available pets, filter search results based on preferences, and submit adoption enquiries online.

1. Purpose

- Help users easily find pets available for adoption.
- Provide a quick way to filter pets by name, type, and age.
- Enable users to submit an adoption enquiry through an online form.

2. Out of Scope

- No backend or database connectivity (data is hardcoded in HTML).
- No actual submission of adoption enquiries to a server.
- No payment processing or advanced user authentication.

3. End Users

- Individuals or families looking to adopt a pet.
- Pet shelter staff who want to showcase pets online.

5. Tools & Technologies Used

Tool/Technology	Purpose
HTML5	Markup and content structure
CSS3	Styling and layout management
JavaScript	Interactivity

1.HTML Structure Overview

Elements like `<div>`, `<form>`, `<input>`, `<select>`, `<textarea>`, `<button>` define the layout and form fields.

2. CSS (Cascading Style Sheets)

Embedded inside `<style>` tags in the `<head>` section.

Provides styling for the page — e.g., colors, margins, padding, font styles, box shadows, flex layout, etc.

3. JavaScript

Included within `<script>` tags at the bottom.

Handles form submission (showing a success message without backend processing).

Implements search and filter functionality for pet cards based on name, type, and age.

8. Key Features

Feature	Description
Pet Listing Display	Visual styling for better readability and aesthetics
Search and Filtering	Search bar for matching that names or breeds
Adoption Enquiry Form	Fields for name email selected that and message.
Responsive and Styled Layout	Layout adapts to different scheme sizes with a mobile friendly design.

9. Challenges Faced & Solutions

Challenge	Solution
Form Submission Without a Backend	Mock Submission Handling with JavaScript
Search and Filter Logic	Case-Insensitive Search and Filter Matching
Dynamic Updates Without Reload	Real-Time Filtering

10. Expected Outcome

- A fully functional frontend prototype that allows users to:
- Browse and search available pets.
- Filter pets by specific criteria.
- Submit a simulated adoption enquiry form.

11. Future Enhancements

1. UI & Styling Improvements

- Responsive design – Use CSS Flexbox/Grid and media queries so it works perfectly on mobile devices.
- Hover effects – Add image zoom or card shadow increase when hovering over pet cards.
- Better form design – Style the form with consistent spacing, colors, and input focus states.
- Dark mode toggle – Let users switch between light and dark themes.

2. Functionality Enhancements



- Pet image gallery – Allow multiple images per pet with a carousel.
- Real-time form validation – Show error messages instantly when the user leaves a required field blank.
- Sort feature – Sort pets by age, name, or type.
- Save favorites – Let users mark pets as favorites (use localStorage to persist).

3. Backend & Data Handling

- Database integration – Store pet details and adoption enquiries in a database (MySQL, Firebase, MongoDB).
- Dynamic pet listing – Fetch pet data from a server API instead of hardcoding.
- Email notifications – Send an email to admins when a form is submitted.
- Admin dashboard – Manage pets, adoption requests, and filters.

4. Search & Filter Enhancements

- Multiple filter selection – Allow filtering by more than one type/age at a time.
- Search by more attributes – Include color, location, or availability status.
- Pagination or infinite scroll – Improve performance with large datasets.

5. Accessibility & SEO

- Alt text for all images – Improve accessibility for screen readers.
- ARIA labels – Make form fields more accessible.
- SEO optimization – Add meta tags, structured data, and proper headings.

12. Sample Code

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>Pet Adoption Center</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
      background-color: #f9f9f9;
    }
    .header {
      text-align: center;
      margin-bottom: 40px;
    }
    .search-box {
      width: 300px;
      padding: 8px;
      margin-bottom: 20px;
      font-size: 16px;
    }
    .filters {
      margin-bottom: 30px;
    }
    .filters label {
      margin-right: 15px;
      font-weight: bold;
    }
    .pet-card {
      background-color: white;
      border-radius: 6px;
      box-shadow: 0 0 5px #ddd;
      padding: 15px;
      margin-bottom: 20px;
      display: flex;
      align-items: center;
    }
  </style>
</head>
<body>
```



```
        display: flex;
        align-items: center;
    }
    .pet-card img {
        width: 120px;
        height: 120px;
        object-fit: cover;
        border-radius: 8px;
        margin-right: 20px;
    }
    .pet-details h3 {
        margin: 0 0 8px 0;
    }
    .pet-details p {
        margin: 4px 0;
    }
}
</style>
</head>
<body>

    <div class="header">
        <h1>Pet Adoption Center</h1>
        <input type="text" id="searchInput" class="search-box" placeholder="Search pets by name or breed..." />
    </div>

    <div class="filters">
        <label for="typeFilter">Pet Type:</label>
        <select id="typeFilter">
            <option value="all">All</option>
            <option value="dog">Dog</option>
            <option value="cat">Cat</option>
            <option value="rabbit">Rabbit</option>
        </select>

        <label for="ageFilter" style="margin-left:20px;">Age:</label>
        <select id="ageFilter">
            <option value="all">All</option>
            <option value="baby">Baby</option>
        </select>
    </div>
```

```
<option value="all">All</option>
<option value="baby">Baby</option>
<option value="young">Young</option>
<option value="adult">Adult</option>
<option value="senior">Senior</option>
</select>
</div>

<div id="petsContainer">

  <div class="pet-card" data-name="Buddy" data-breed="Golden Retriever" data-type="dog" data-age="young">
    
    <div class="pet-details">
      <h3>Buddy</h3>
      <p><strong>Breed:</strong> Golden Retriever</p>
      <p><strong>Age:</strong> Young</p>
      <p>Friendly and playful dog looking for a loving home.</p>
    </div>
  </div>

  <div class="pet-card" data-name="Mittens" data-breed="Siamese" data-type="cat" data-age="adult">
    
    <div class="pet-details">
      <h3>Mittens</h3>
      <p><strong>Breed:</strong> Siamese</p>
      <p><strong>Age:</strong> Adult</p>
      <p>Sociable and affectionate cat who enjoys quiet environments.</p>
    </div>
  </div>

  <div class="pet-card" data-name="Thumper" data-breed="Dutch" data-type="rabbit" data-age="baby">
    
    <div class="pet-details">
      <h3>Thumper</h3>
      <p><strong>Breed:</strong> Dutch</p>
      <p><strong>Age:</strong> Baby</p>
      <p>Cute and energetic bunny ready for cuddles.</p>
    </div>
  </div>
</div>
```



```
        <pet>Speedy and energetic bunny ready for cuddles.</pet>
    </div>
</div>

    <!-- Add more pet cards here if you want -->

</div>
!-- Adoption Enquiry Form Container -->


<!-- Main Form Heading -->
    <h2>Adoption Enquiry Form</h2>

    <!-- Sub-heading or instruction -->
    <h3>Fill Your Details</h3>

    <form id="adoptionForm">

        <!-- Your Name Label -->
        <label for="name">Your Name:</label>
        <br />
        <!-- Your Name Input -->
        <input type="text" id="name" name="name" required />
        <br /><br />

        <!-- Your Email Label -->
        <label for="email">Your Email:</label>
        <br />
        <!-- Your Email Input -->
        <input type="email" id="email" name="email" required />
        <br /><br />

        <!-- Select Pet Label -->
        <label for="pet">Select Pet:</label>
        <br />
        <!-- Select Pet Dropdown -->
        <select id="pet" name="pet" required>
            <option value="" disabled selected>Select a pet</option>


```



```
<option value="" disabled selected>Select a pet</option>
<option value="Buddy">Buddy</option>
<option value="Mittens">Mittens</option>
<option value="Thumper">Thumper</option>
<!-- Add more options if needed -->
</select>
<br /><br />

<!-- Your Message Label -->
<label for="message">Message:</label>
<br />
<!-- Your Message Textarea -->
<textarea id="message" name="message" rows="4" placeholder="Write your message here..." required></textarea>
<br /><br />

<!-- Submit Button -->
<button type="submit">Submit Enquiry</button>

</form>

<!-- Success message div -->
<div id="formMessage" class="success-message" style="display:none;"></div>

</div>

<script>
// Simple form submission handling (no backend)
document.getElementById('adoptionForm').addEventListener('submit', function(event) {
  event.preventDefault();
  const formMessage = document.getElementById('formMessage');
  formMessage.textContent = 'Thank you for your enquiry! We will contact you soon.';
  formMessage.style.display = 'block';
  this.reset();
});
</script>

<script>
const searchInput = document.getElementById('searchInput');
```

```
const searchInput = document.getElementById('searchInput');
const typeFilter = document.getElementById('typeFilter');
const ageFilter = document.getElementById('ageFilter');
const petsContainer = document.getElementById('petsContainer');
const petCards = petsContainer.getElementsByClassName('pet-card');

function filterPets() {
  const searchTerm = searchInput.value.toLowerCase();
  const selectedType = typeFilter.value;
  const selectedAge = ageFilter.value;

  for (let pet of petCards) {
    const name = pet.getAttribute('data-name').toLowerCase();
    const breed = pet.getAttribute('data-breed').toLowerCase();
    const type = pet.getAttribute('data-type');
    const age = pet.getAttribute('data-age');

    const matchesSearch = name.includes(searchTerm) || breed.includes(searchTerm);
    const matchesType = selectedType === 'all' || type === selectedType;
    const matchesAge = selectedAge === 'all' || age === selectedAge;

    if (matchesSearch && matchesType && matchesAge) {
      pet.style.display = 'flex';
    } else {
      pet.style.display = 'none';
    }
  }
}

searchInput.addEventListener('input', filterPets);
typeFilter.addEventListener('change', filterPets);
ageFilter.addEventListener('change', filterPets);

// Show all pets initially
filterPets();
</script>
</body>
</html>
```

13. Screenshots of Final Output

Pet Adoption Center

Pet Type: Age:



Buddy

Breed: Golden Retriever

Age: Young

Friendly and playful dog looking for a loving home.

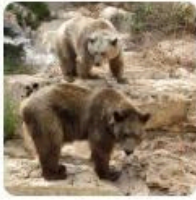
Adoption Enquiry Form

Fill Your Details

Your Name:

Your Email:

Select Pet:



Thumper

Breed: Dutch

Age: Baby

Cute and energetic bunny ready for cuddles.

Adoption Enquiry Form

Fill Your Details

Your Name:

Your Email:

Select Pet:

Message:

Write your message
here...

Submit Enquiry

Pet Adoption Center


Search pets by name or breed...

Pet Type:


All

Age:


All



Buddy
Breed: Golden Retriever
Age: Young
Friendly and playful dog looking for a loving home.



Mittens
Breed: Siamese
Age: Adult
Sociable and affectionate cat who enjoys quiet environments.



Thumper
Breed: Dutch

11. Conclusion

This is a functional front-end prototype of a pet adoption center webpage. It allows users to browse and filter available pets, then fill out an adoption enquiry form. However, it lacks backend integration, meaning the form data is not actually sent anywhere. To make it fully operational, a server-side script or database connection would be needed to handle submissions and store adoption requests.

12. References

- L&T LMS : <https://learn.intedutech.com/Landing/MyCourse>
- W3schools: <https://www.w3schools.com/>