IT331 - 371

PawPet

3-Tier Architecture Web App

Rafeef Abahussain

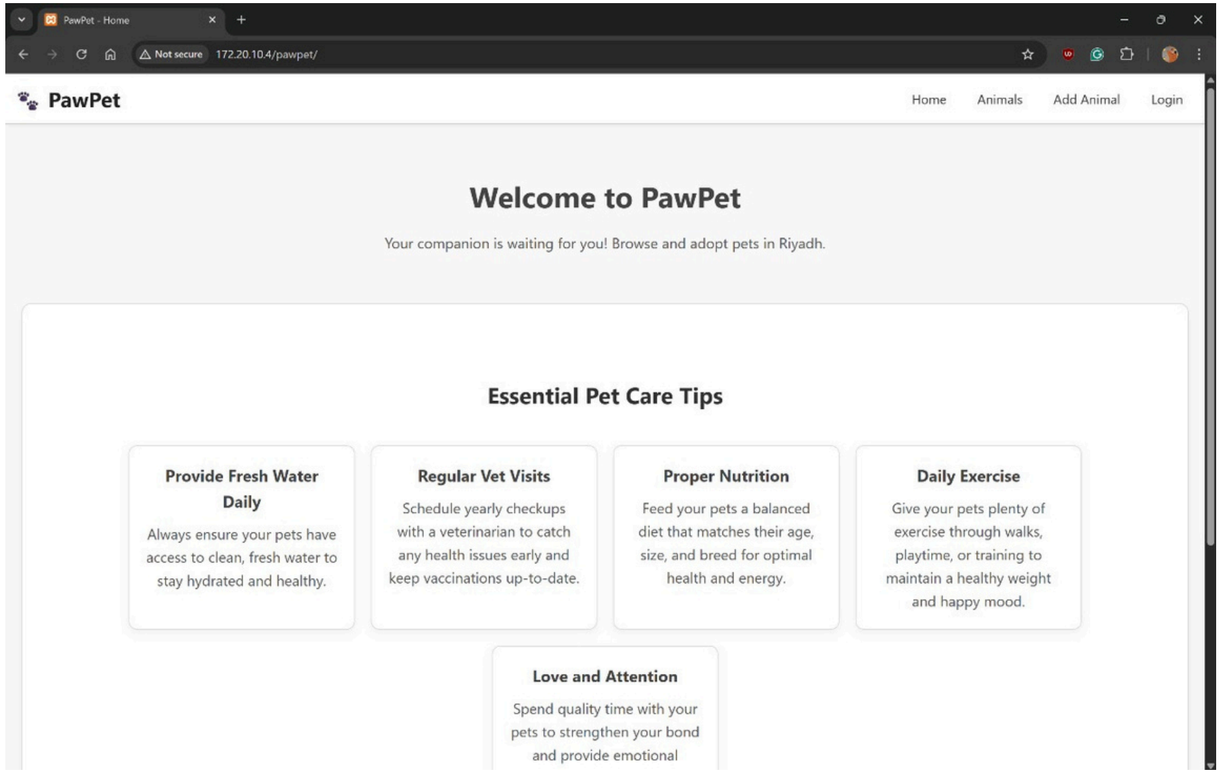
Ranim Albogami

Shada Alamri

Yara Alshbrami

What is PawPet?

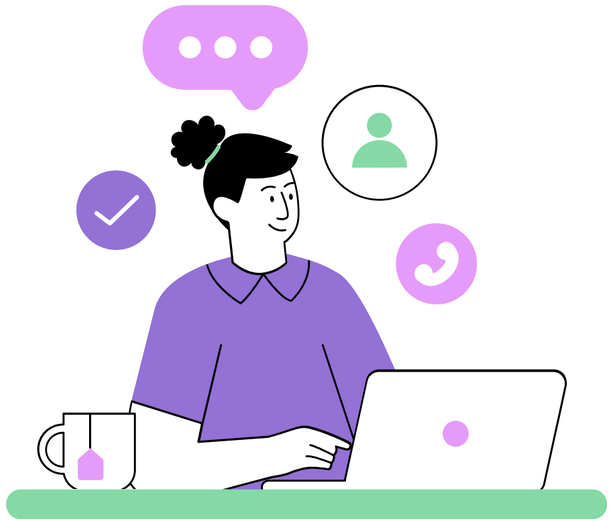
- A web platform to adopt pets in Riyadh.
- Helps users find and connect with shelters.
- Built using a 3-tier architecture.



Why PawPet?

- Many animals in shelters don’t get adopted
- No unified, user-friendly platform in Riyadh
- People struggle to connect with shelters

Project Goal & Objectives



Goal:

- Create an organized, responsive, and secure website

Objectives:

- View pet profiles and apply online
- Smooth and consistent UI/UX

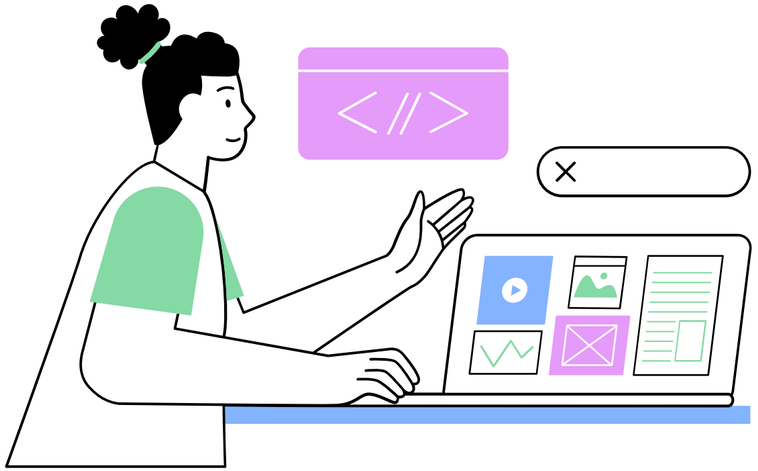
What the Website Must Do

Functional:

- Browse/view pets
- User registration/login
- Admin management tools

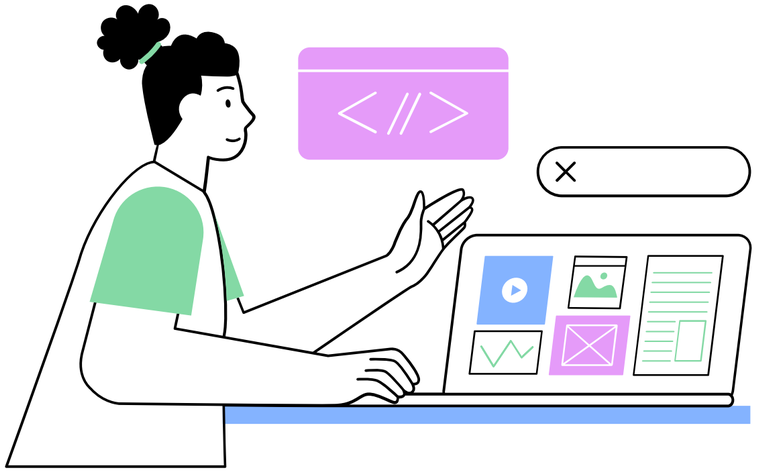
Non-Functional:

- Mobile responsive
- Secure data handling
- 24/7 uptime



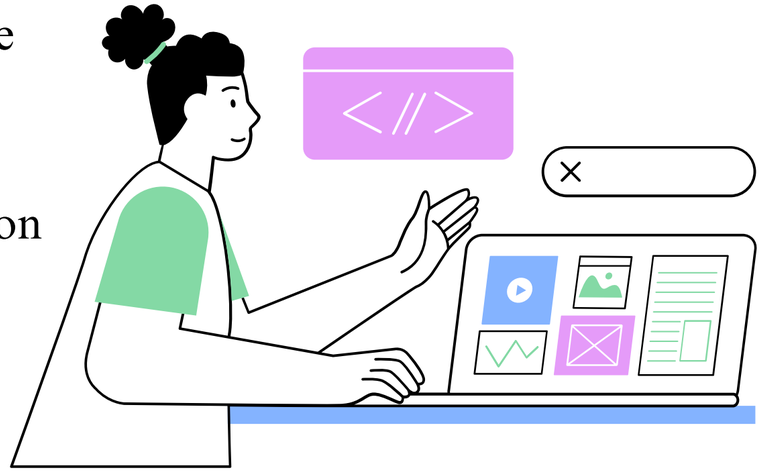
Phase 1

In Phase 1, the team initiated the development of the Pet Adoption Website by clearly defining the project's purpose, identifying the main problem, and outlining the goals and objectives. The project aims to create a user-friendly platform that connects people with adoptable animals, making the adoption process easier, faster, and more accessible. To achieve this, both functional and nonfunctional requirements were carefully specified. Functional requirements focused on features such as user registration, viewing detailed pet profiles, and submitting adoption requests, while nonfunctional requirements emphasized performance, usability, and system reliability. Additionally, initial interface designs were created to provide a clear vision of how users would interact with the system. These interfaces laid the foundation for a smooth and intuitive user experience. This phase established a clear direction for the project and set the stage for technical development in the upcoming phases.



Phase 2

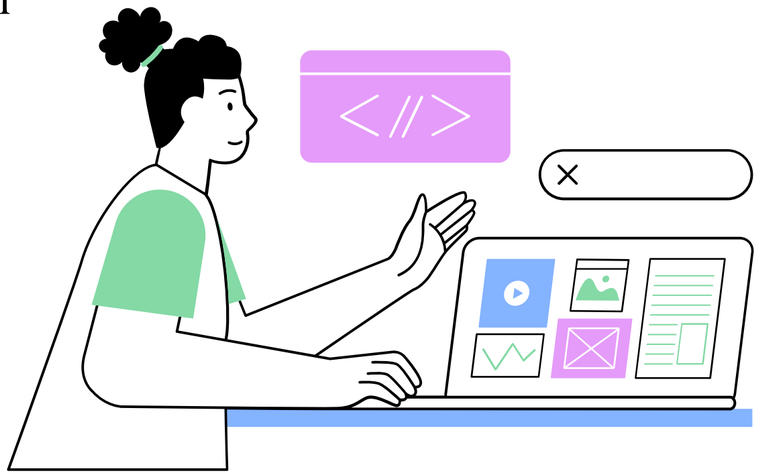
In Phase 2, we focused on developing the technical structure of the Pet Adoption Website by working on the database design and system integration.
An (ER) diagram was created to illustrate the relationships between key entities such as users, pets, and adoption requests, providing a clear visual representation of the data flow. Based on this, table schemas and a relational schema were developed to ensure organized and efficient data storage.
The team also successfully set up the database and established a secure and functional connection between the website and the database system.
This connection enabled smooth communication between the data layer and the user interface, allowing dynamic interaction with stored information.
These efforts provided a strong backend foundation, ensuring the system can handle data operations effectively and support the core functions of the adoption process.



Phase 3

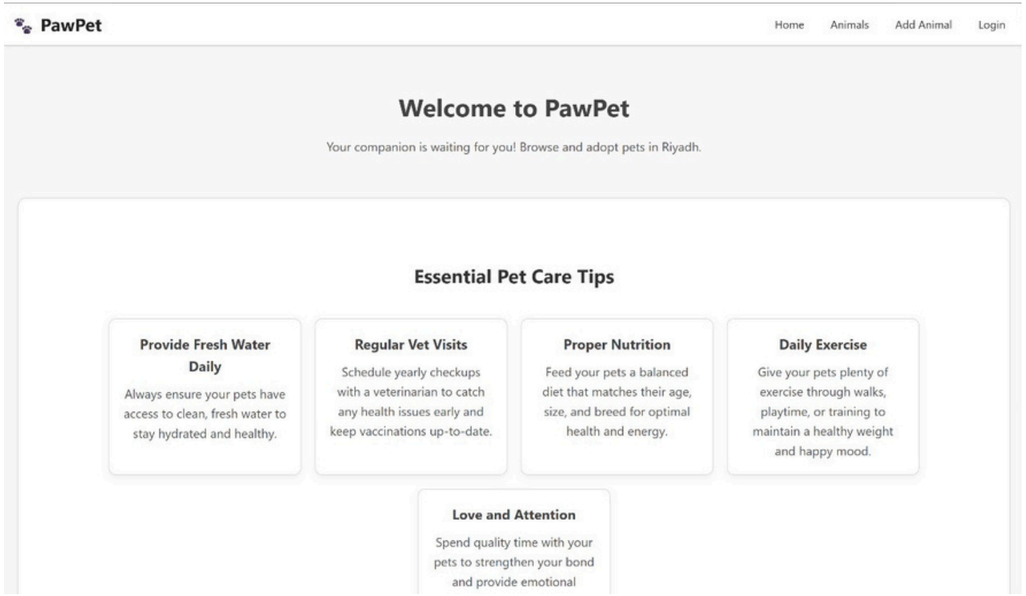
In Phase 3, the team completed the full implementation of the Pet Adoption Website using a 3-tier architecture approach.
The system was structured into three distinct layers: the Presentation Layer, which handled the user interface, allowing visitors to navigate the platform, view animals, and submit forms. The Business Logic Layer, built in PHP, managed the core functionality, including user authentication, session handling, processing adoption requests, and adding new animal listings.
The Data Access Layer was responsible for connecting to the MySQL database, executing queries to retrieve and store data such as user information and animal records. Key CRUD operations like adding new pets and submitting adoption forms were implemented and tested to ensure proper data flow and interaction between layers.

Screenshots of the website were included to illustrate the system's functionality, including pages for login, pet listings, detailed profiles, and adoption forms.

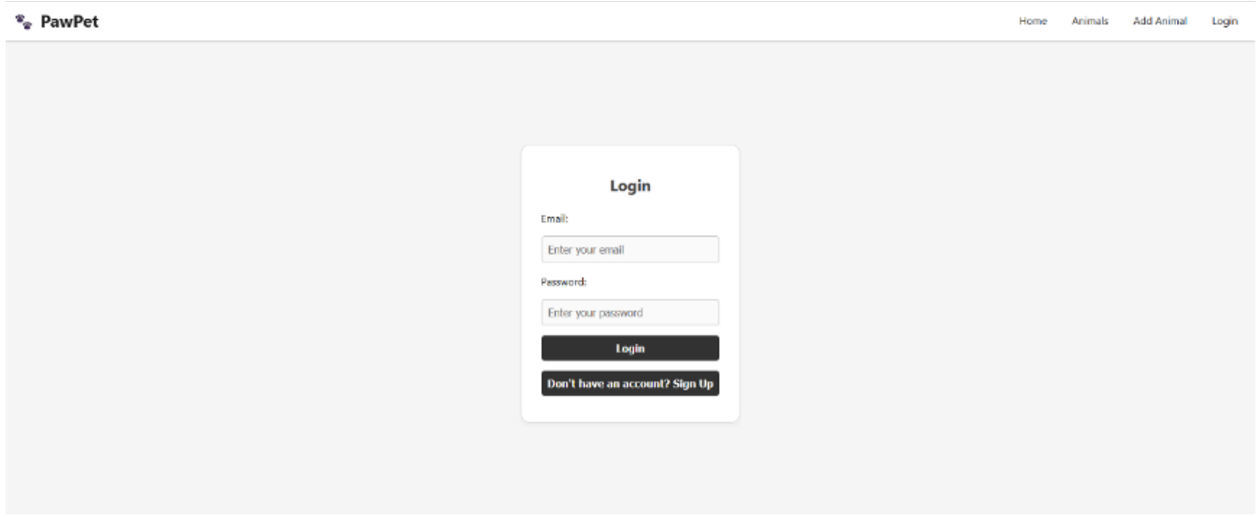


websites pages

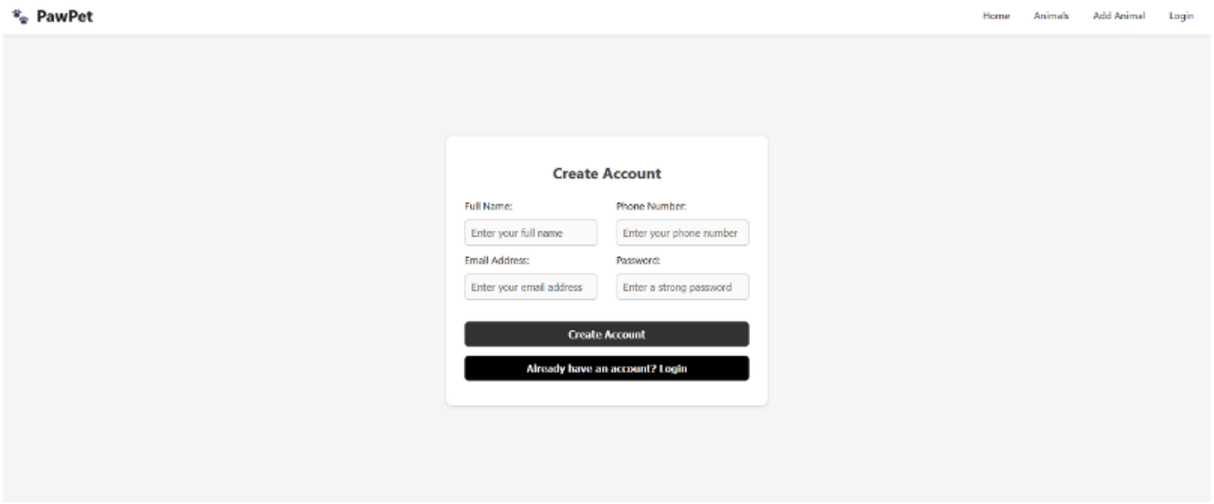
home page



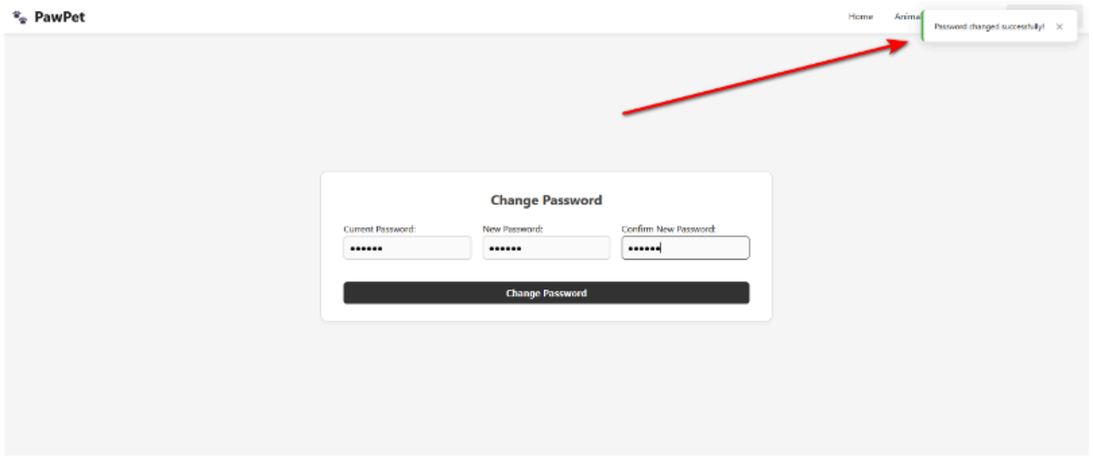
login page



signup page

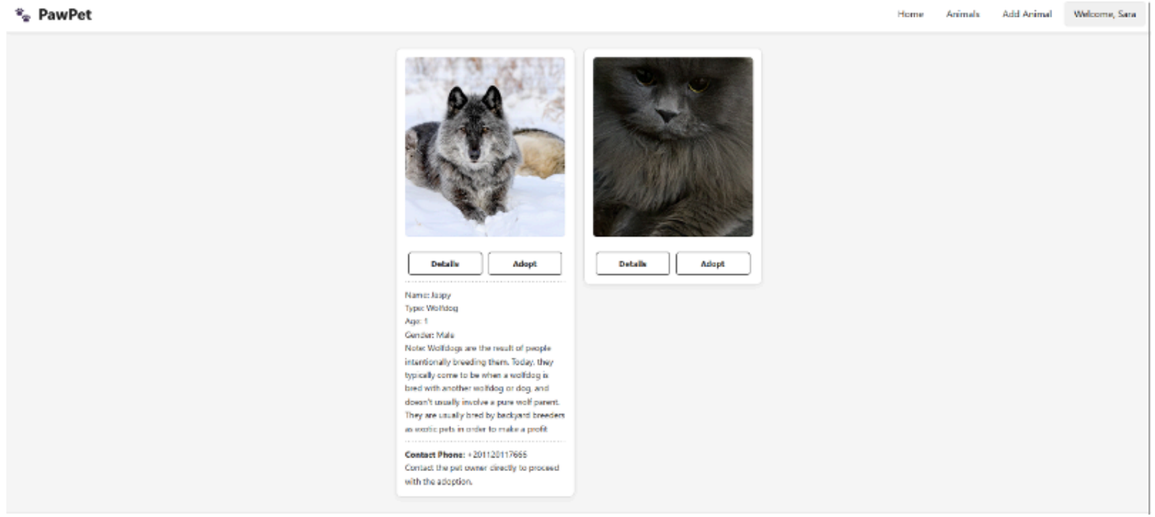


ChangePassword page

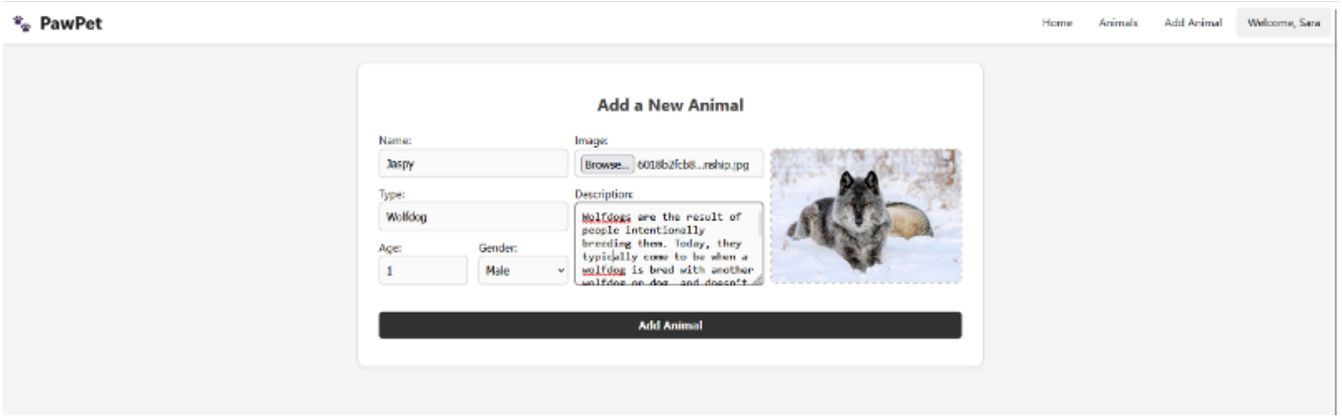


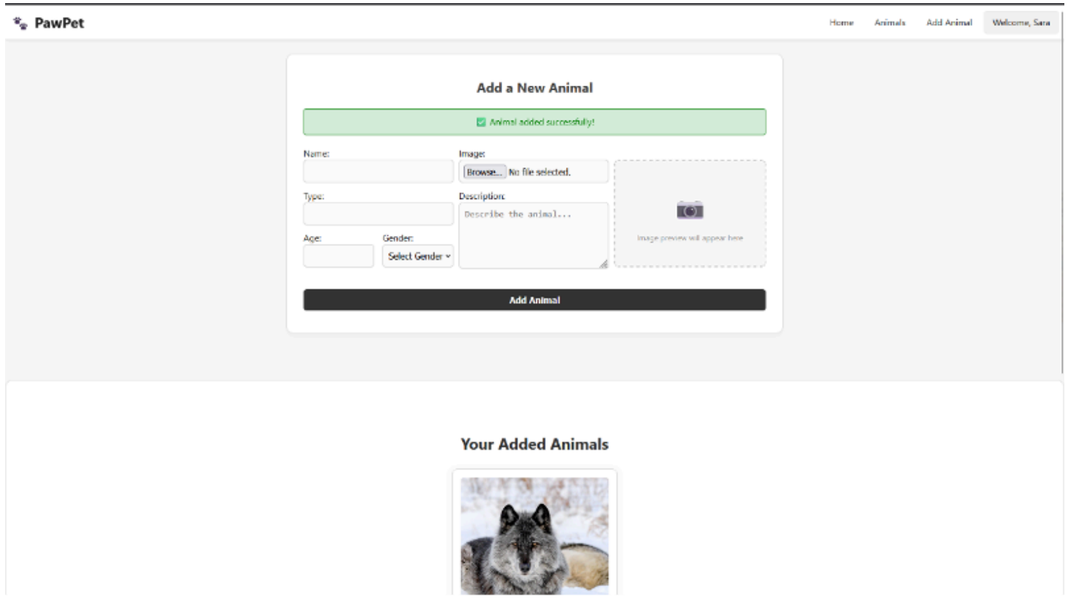
websites pages

animal view page

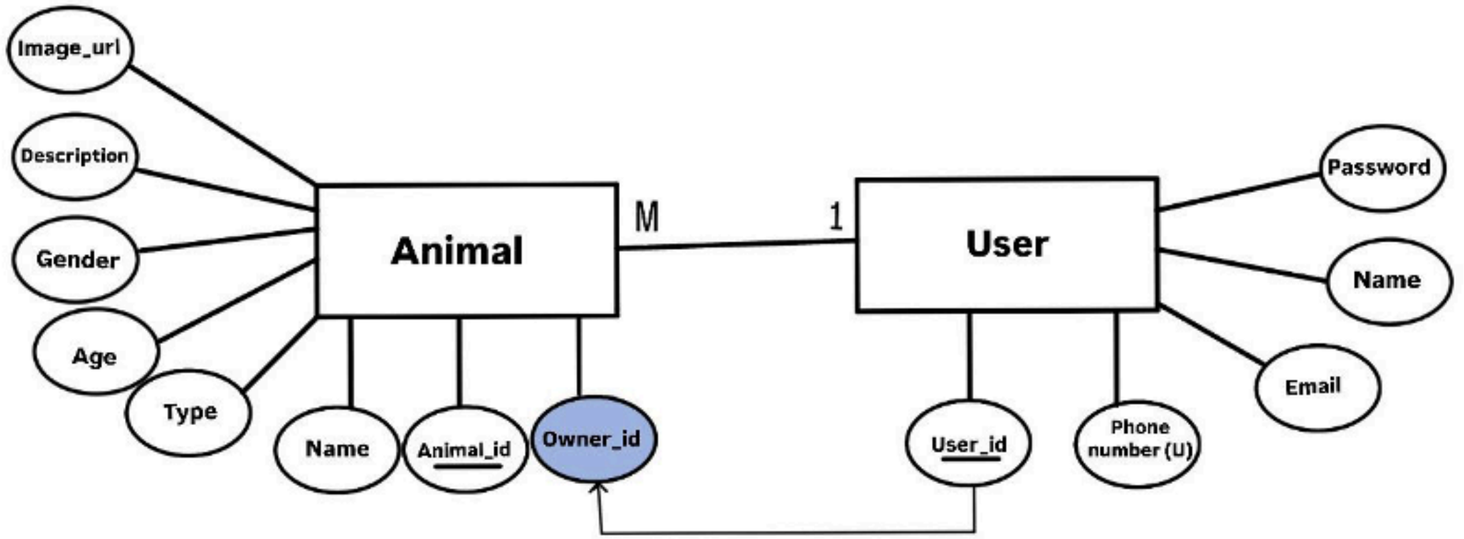


add animal page





E-R diagram

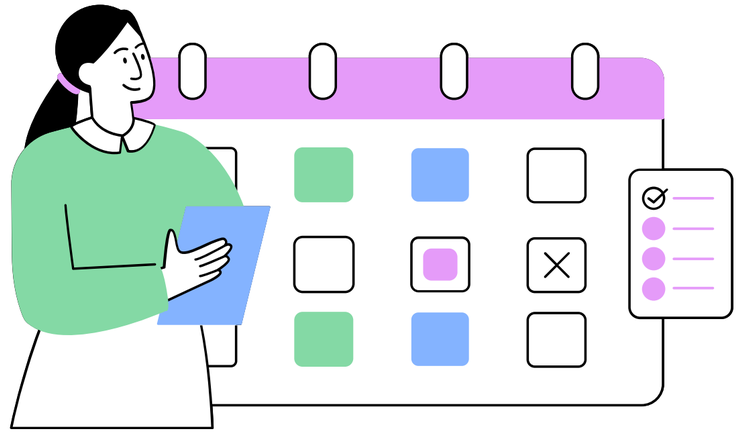

Image_url
Description
Gender
Age
Type
Animal
M
1
User
Password
Name
Email
Name
Animal_id
Owner_id
User_id
Phone number (U)

Our Architecture

Presentation Layer: HTML/CSS Frontend

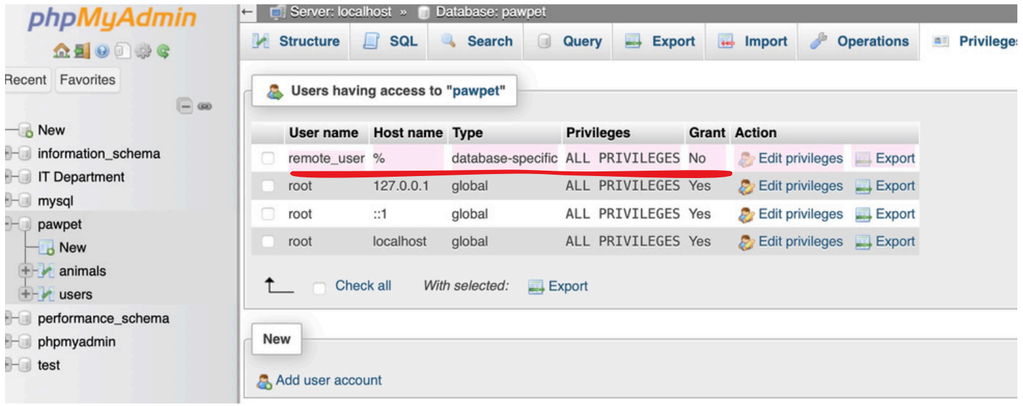
Business Layer: User interaction logic (to be expanded in Phase 2)

Data Access Layer: Pet data from shelters (to be implemented)

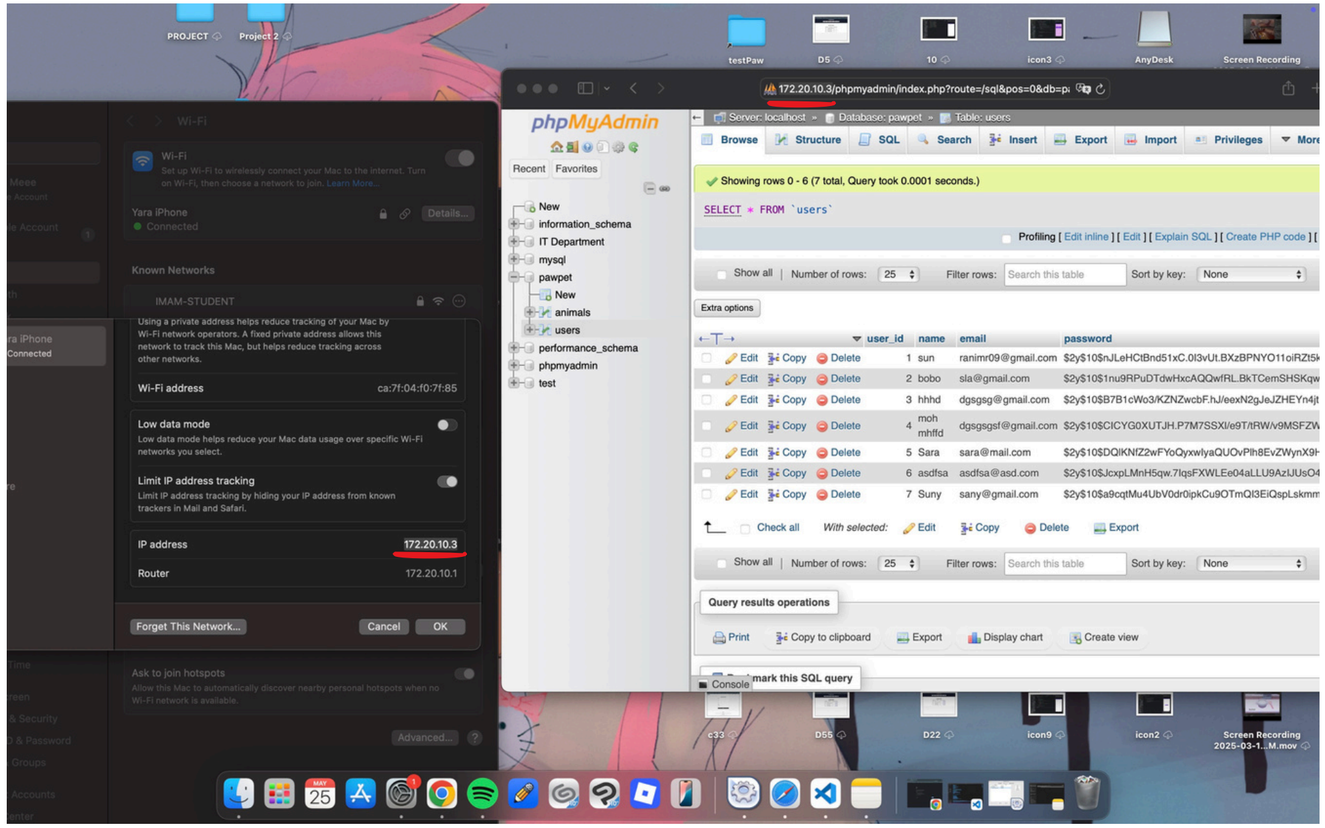


DataBase layer

1- we added remote_user to our database and provide all privileges for her via SQL tab



2- for the database we are gonna use the IP address(172.20.10.3) of our network



Business layer

1- we will connect to the same network as the data tire device then we will go to the db.php connection we will add the IP address of the data access tire device (172.20.10.3)

db.php

C: > xampp > htdocs > PawPet > db.php > $user

```php
<?php
$host = "172.20.10.3";
$user = "remote_user";
$password = "password";
$dbname = "pawpet";

$conn = new mysqli(hostname: $host, username: $user, password: $password, database: $dbname);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
?>
```

2- we are gonna call the API (add_api) of the business tier on any page that needs to make changes to the database like the add.php that will hold add, delete and update the animal

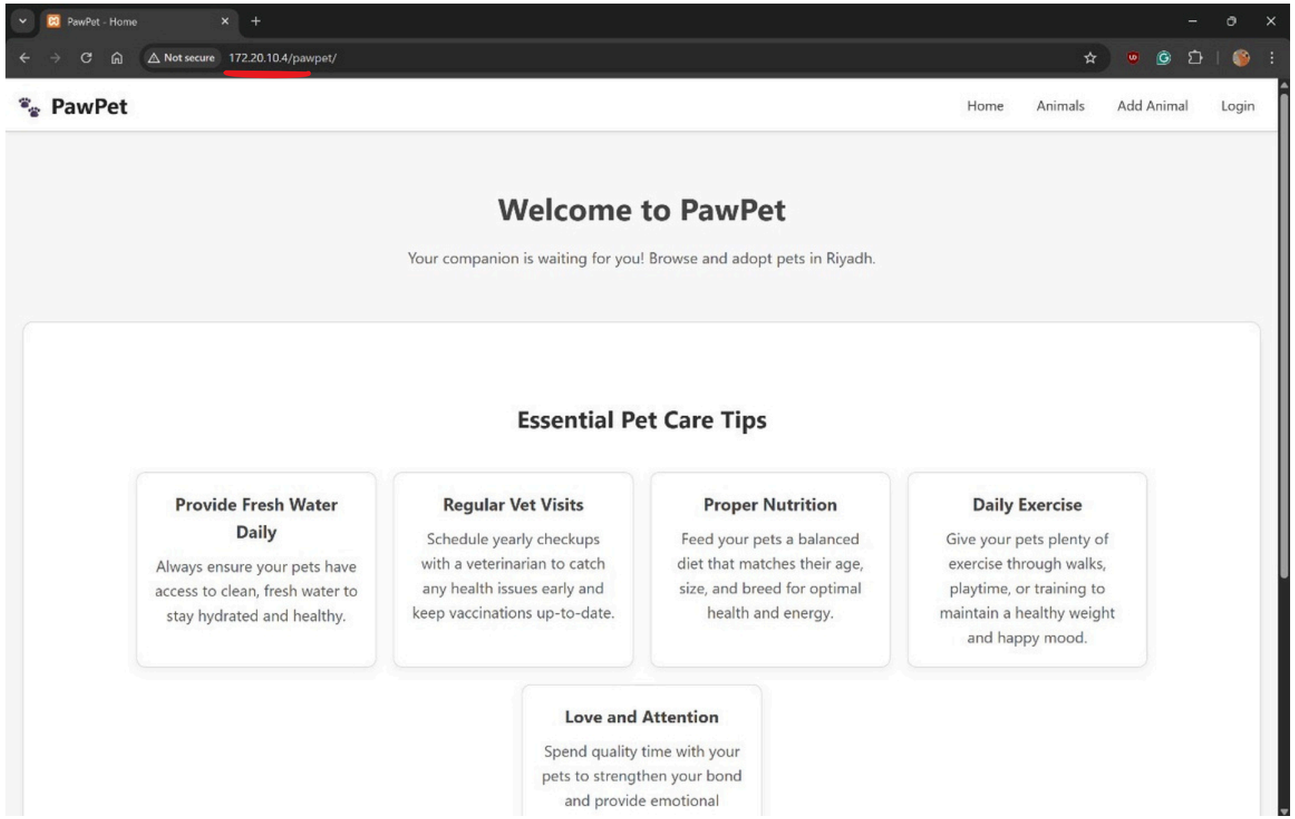
```php
        'age' => $_POST['age'],
        'gender' => $_POST['gender'],
        'description' => $_POST['description'],
        'image_url' => $upload_path,
        'user_id' => $_SESSION['user_id']
    ];
    $api_url = 'http://172.20.10.4/PawPet/add-api.php';
```

the ip address its will be the ip of the bussiness tire device

Network band (channel):	2.4 GHz (6)
Aggregated link speed (Receive/ Transmit):	115/115 (Mbps)
IPv6 address:	2001:16a2:c01b:d0fe:8f9a:201f:c5d5:a824
Link-local IPv6 address:	fe80::b33a:ac1d:dfc9:19fb%17
IPv6 default gateway:	fe80::4490:bbff:fe27:d564%17
IPv6 DNS servers:	fe80::4490:bbff:fe27:d564%17 (Unencrypted) fe80::4490:bbff:fe27:d564%17 (Unencrypted)
IPv4 address:	172.20.10.4
IPv4 DNS servers:	172.20.10.1 (Unencrypted)
Physical address (MAC):	BC:3D:85:BB:C2:06

Presentaion layer

1- for the presentation tire device all they need to do is to be connected to the same network as the other devices and to user the ip address of the business tire device to access the website



Thank You