

# ABSTRACT

## TEAM 7:








### **PawMart- Pet Shop Application**

The PawMart is an interactive platform designed to simplify the process of exploring, purchasing, and managing pet-related products and services. Developed using React for the frontend and JSON Server for backend simulation, this project provides a smooth user experience for both customers and administrators.

The primary objective of the application is to offer a user-friendly interface where users can browse pets and pet supplies, manage their shopping cart and favourites, write reviews, and interact with the platform after successful login or registration. The application ensures security by restricting sensitive operations such as adding to cart and submitting reviews to only logged-in users.

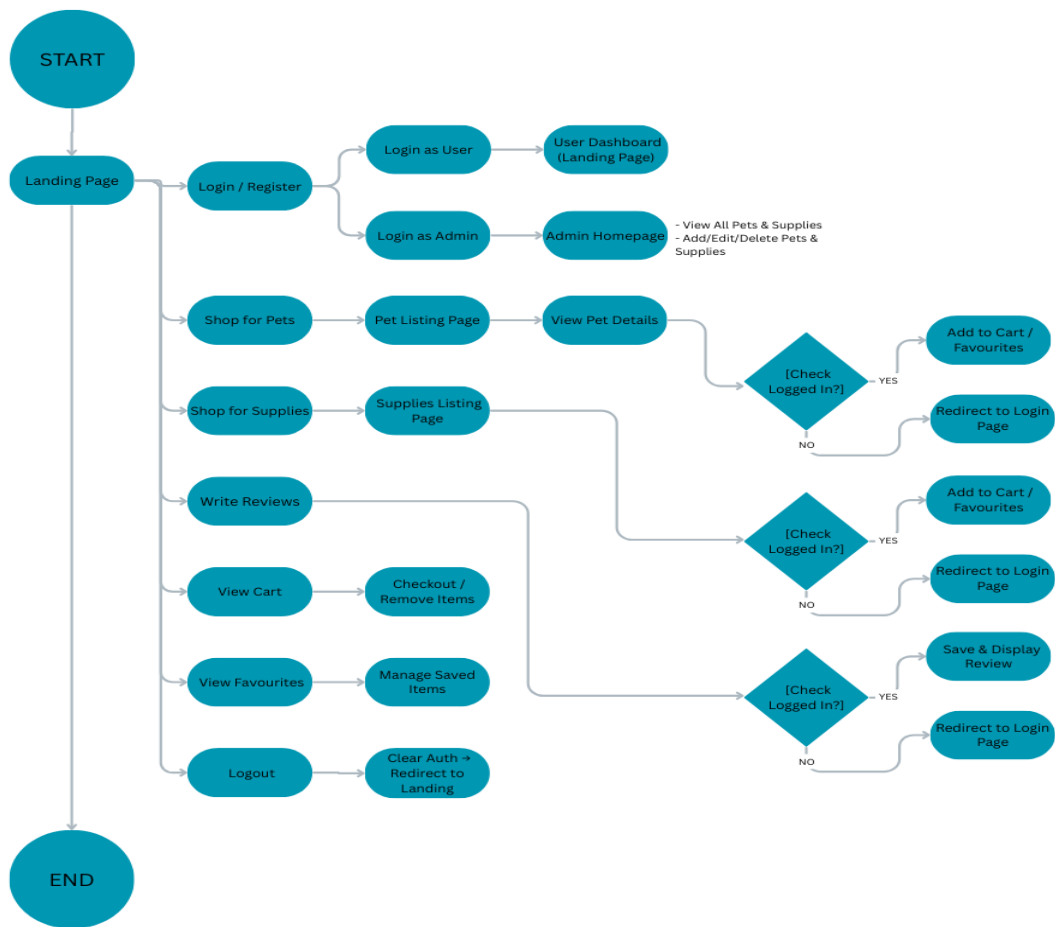
The system also includes an administrative dashboard that allows CRUD (Create, Read, Update, Delete) operations for pets and supplies. Admins can efficiently manage the shop inventory and reflect changes in real-time across the platform.

### **Key Features:**

-  **Landing Page** with navigation buttons for login, pet shop, pet supplies, and reviews
-  **User Authentication** for secure login/register flow
-  **Pet Listing** and **Supplies Store** with add-to-cart functionality
-  **Favourites** section for saving preferred items
-  **Cart System** for managing selected pets and supplies
-  **Review Submission** with login check
-  **Admin Panel** for inventory control using CRUD operations

A detailed flowchart is included to illustrate the entire navigation and interaction logic of the application, highlighting login-based feature access and admin-user role separation.

This project demonstrates the practical implementation of React-based UI, client-side routing, state management, and simulated backend integration, making it a well-rounded example of a full-stack project in a simplified development environment.



## Tech Stack:

| Category              | Technology / Library              |
|-----------------------|-----------------------------------|
| Frontend Framework    | React                             |
| Styling               | CSS, React Bootstrap, Material UI |
| Routing               | React Router                      |
| HTTP Requests         | Axios                             |
| Icon Library          | Font Awesome                      |
| Fonts                 | Google Fonts                      |
| Backend (Mock Server) | JSON Server                       |
| Language              | JavaScript                        |
| Runtime Environment   | Node.js                           |

## Team Members and Roles

| Team Member                  | Role                             | Responsibilities  |
|------------------------------|----------------------------------|---|
| <b>Fathima A (Team Lead)</b> | Project Lead, Frontend Developer | <ul style="list-style-type: none"><li>- Coordinate team progress</li><li>- Design landing page &amp; navigation</li><li>- Handle routing using React Router</li></ul>   |
| <b>Nandana Raveendran</b>    | UI/UX & Review System Developer  | <ul style="list-style-type: none"><li>- Style pages using React Bootstrap, CSS, Material UI</li><li>- Implement review submission &amp; display section</li><li>- Use Google Fonts for design consistency</li></ul> |
| <b>Athul C Anu</b>           | Pet & Supplies Shop Developer    | <ul style="list-style-type: none"><li>- Build "Shop for Pets" and "Shop for Supplies" pages</li><li>- Implement listing, filtering, and product detail views</li></ul>  |
| <b>Althwaf T Rafeeq</b>      | Cart & Favourites Functionality  | <ul style="list-style-type: none"><li>- Implement Cart and Favourites components</li><li>- Add "Add to Cart" and "Add to Favourites" with login check</li><li>- Use Axios to fetch/update data</li></ul>            |
| <b>Mohammad Zeeshan</b>      | Admin Dashboard & Backend Ops    | <ul style="list-style-type: none"><li>- Design Admin Home Page for CRUD operations</li><li>- Use JSON Server for backend simulation</li><li>- Handle data updates and management</li></ul>                          |

***All will contribute to testing and debugging.***