

**FINAL REPORT
BUSINESS APPLICATION DEVELOPMENT
(ISYS6197003)**



**Kelompok 7
LB11**

2702334071 - Wijono
2702273624 - Antonius Jose
2702275182 - Edeline Clarissa
2702259576 - Vanessa Verensia
2702369573 - Kathelyn Leunufna
2702256025 - Eagan Edbert Salim
2702385822 - Revania Priscilla Lalenoh

Business Application Development

Semester Genap 2024/2025

TABLE OF CONTENT

Chapter 1 - Preliminary	3
1. Background and Objective	3
2. Roles and Responsibility	3
3. Timeline	3
Chapter II - Business Application	
4. Use Case and Activity Diagram	3
5. GUI Design	3
6. Database Design	
7. Coding Explanation for Database Connectivity and GUI	
8. Question and Answer	
Chapter III - Conclusion	

Chapter 1 - Preliminary

1. Background and Objective

Paw & Posh merupakan sebuah aplikasi yang menyediakan layanan seperti *grooming* dan berbagai produk kebutuhan untuk hewan peliharaan, seperti makanan, mainan, aksesoris, dan produk perawatan.

Tujuan dari aplikasi **Paw & Posh** adalah untuk memberikan akses yang lebih mudah bagi pemilik hewan peliharaan untuk mendapatkan produk dan layanan yang mereka butuhkan tanpa harus pergi ke toko secara langsung.

VISI DAN MISI

Visi

Menjadi platform pet shop online terdepan yang dipercaya oleh para pecinta hewan peliharaan di seluruh Indonesia, dengan menyediakan produk berkualitas dan informasi yang bermanfaat untuk meningkatkan kesejahteraan hewan.

Misi

- Menyediakan berbagai produk berkualitas tinggi untuk hewan peliharaan, mulai dari makanan, mainan, hingga perlengkapan kesehatan.
- Memberikan pelayanan pelanggan yang responsif dan ramah, memastikan pengalaman belanja yang menyenangkan dan memuaskan.
- Menyediakan artikel dan tips bagi pemilik hewan peliharaan untuk meningkatkan pengetahuan mereka dalam merawat hewan.
- Mengadopsi teknologi terkini untuk meningkatkan pengalaman berbelanja online dan memberikan kemudahan akses bagi pelanggan.
- Membangun komunitas yang solid dan saling mendukung bagi para pecinta hewan peliharaan, melalui forum diskusi dan acara online.

2. Roles and Responsibility

Project Manager: Revania Priscilla Lalenoh

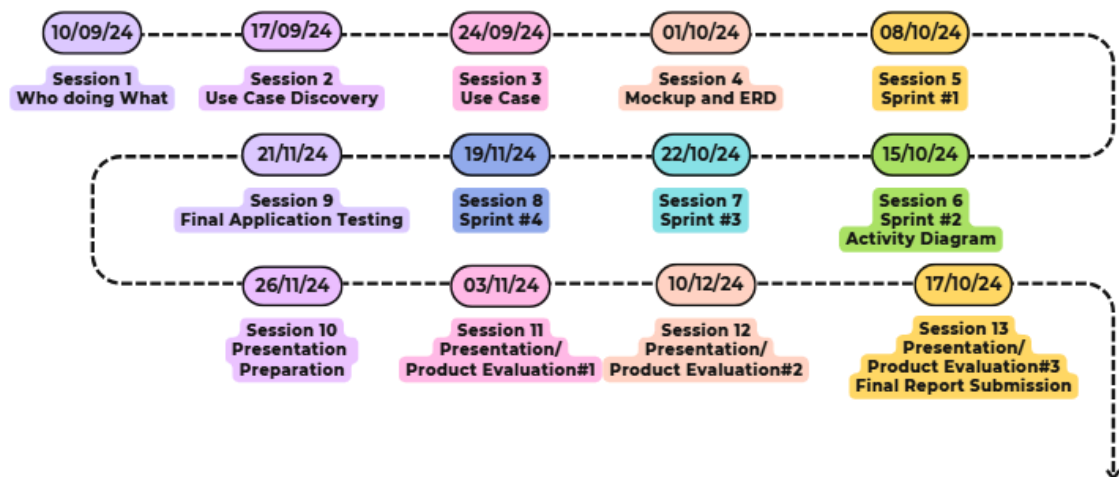
Presenter: Eagan Edbert Salim

Member:

- Kathelyn Leunufna
- Wijono
- Antonius Jose
- Edeline Carissa
- Vanessa Verensia

3. Timeline

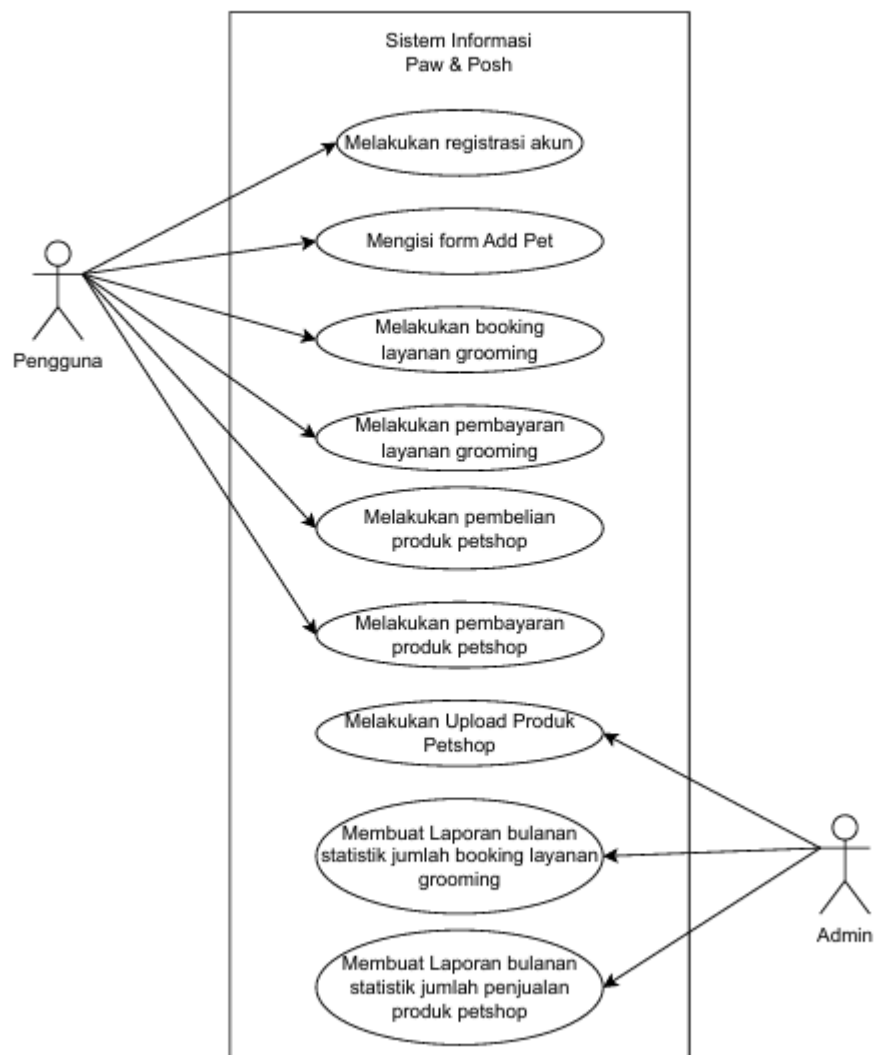
Timeline Paw & Posh



Chapter 2 - Business Application

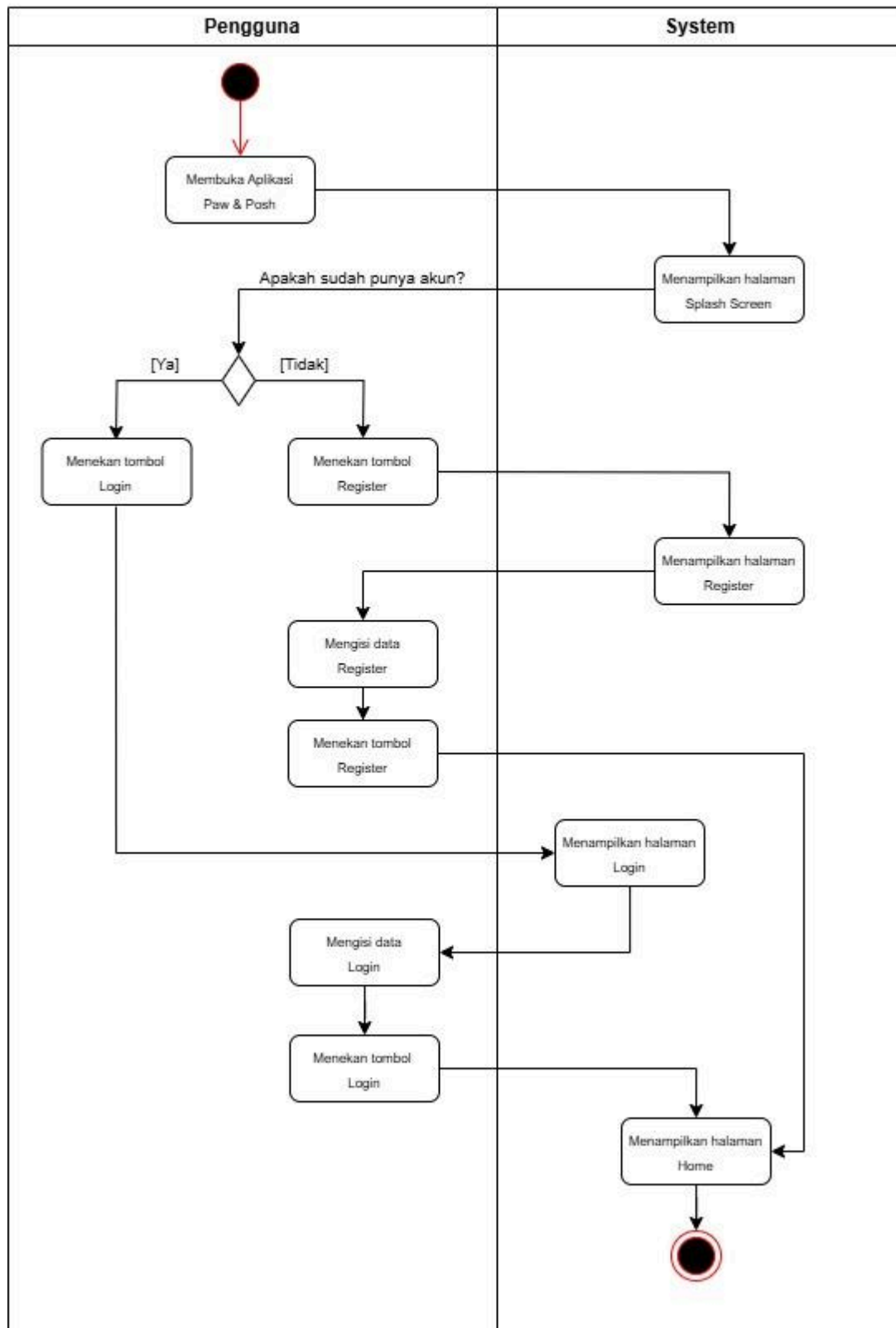
4. Use Case and Activity Diagram

- Use Case

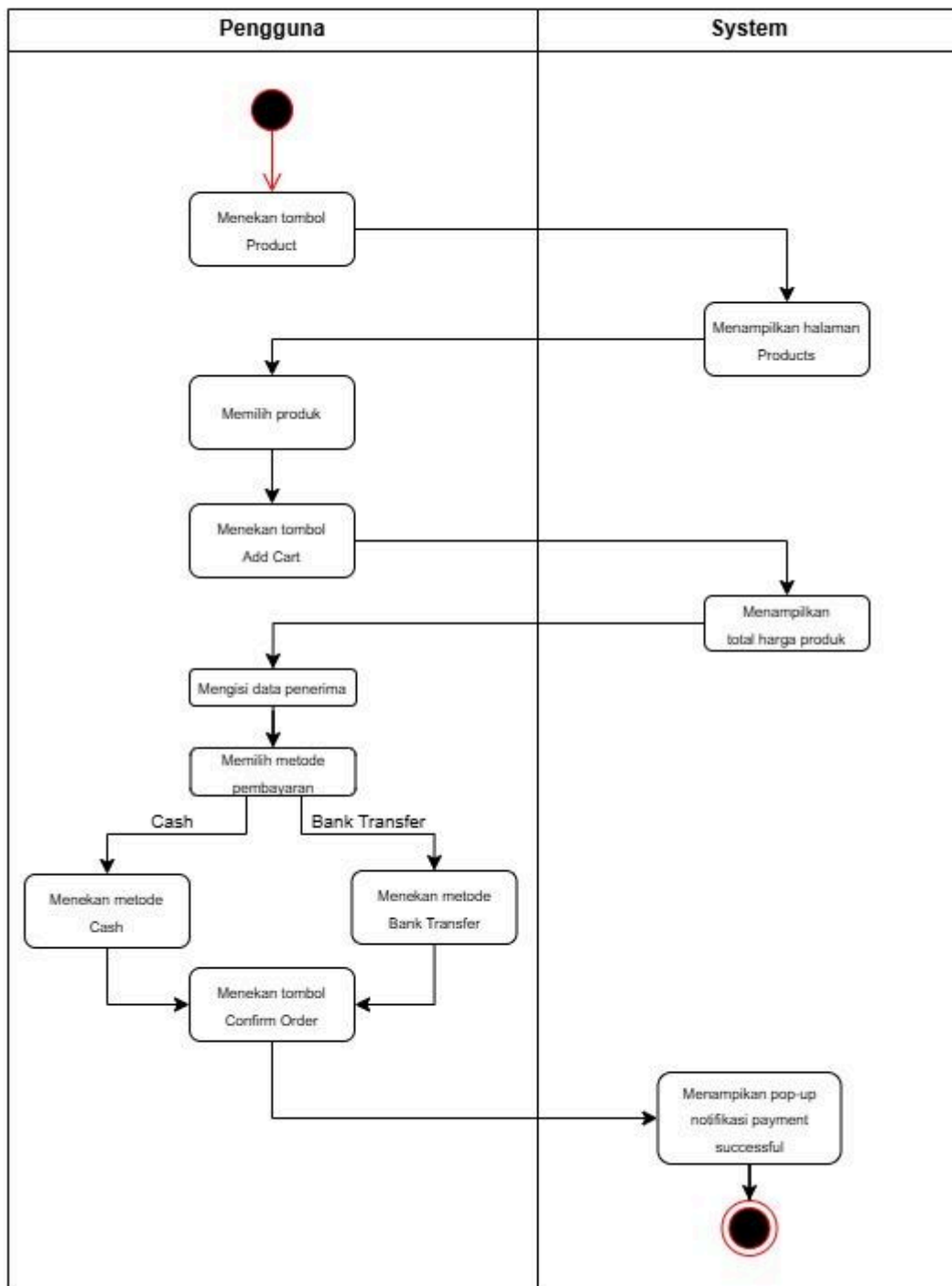


- Activity Diagram

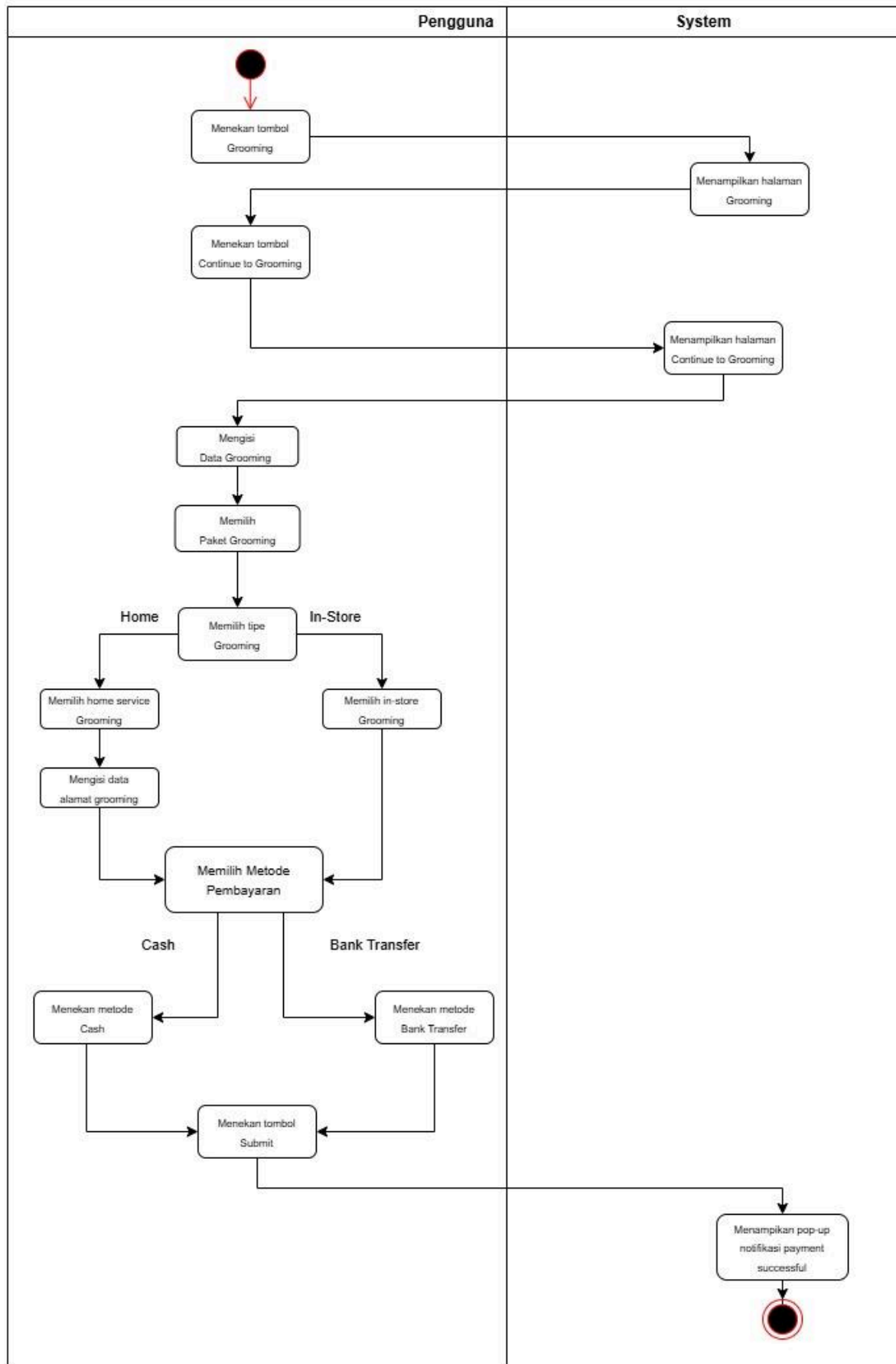
Registrasi Akun



Melakukan Pembelian Produk Petshop



Melakukan booking layanan grooming



5. GUI Design

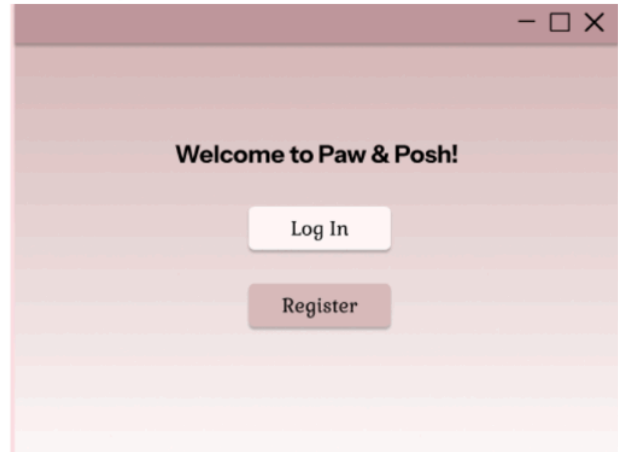
Link:

[https://www.figma.com/design/n4tEd95EzOgF1biqzSZWAw/Paw-%26-Posh-\(Revisi\)?node-id=0-1](https://www.figma.com/design/n4tEd95EzOgF1biqzSZWAw/Paw-%26-Posh-(Revisi)?node-id=0-1)

Splashscreen



Onboarding



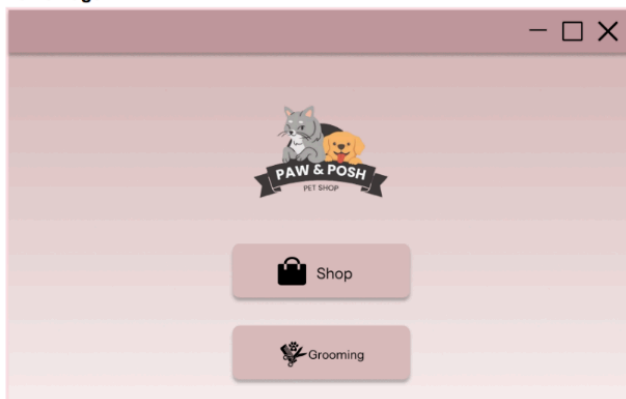
Login Page



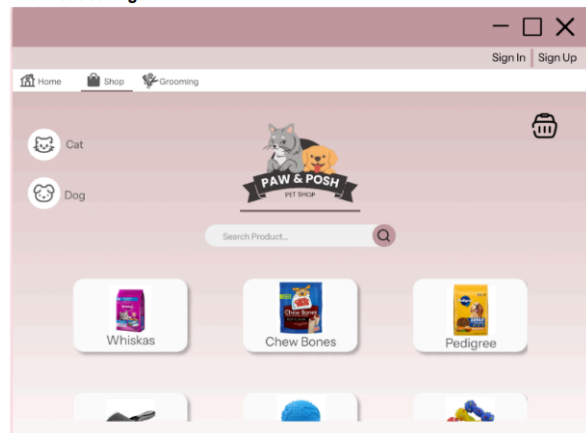
Register Page



Home Page



Pet Product Page




Pet Product Cart Page

HomeShopGrooming

Sign InSign Up


←

Cart



Whiskas

− 2 +



Pedigree

− 2 +

Recipient Name

Delivery Address


Select Payment Method

Buy now


Grooming Page

HomeShopGrooming

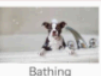
Sign InSign Up



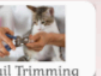
Search Services...



Fur Trimming



Bathing



Nail Trimming

Grooming Form Page


HomeShopGrooming

Sign InSign Up

left ...

←

Services



Fur Trimming

Owner Name

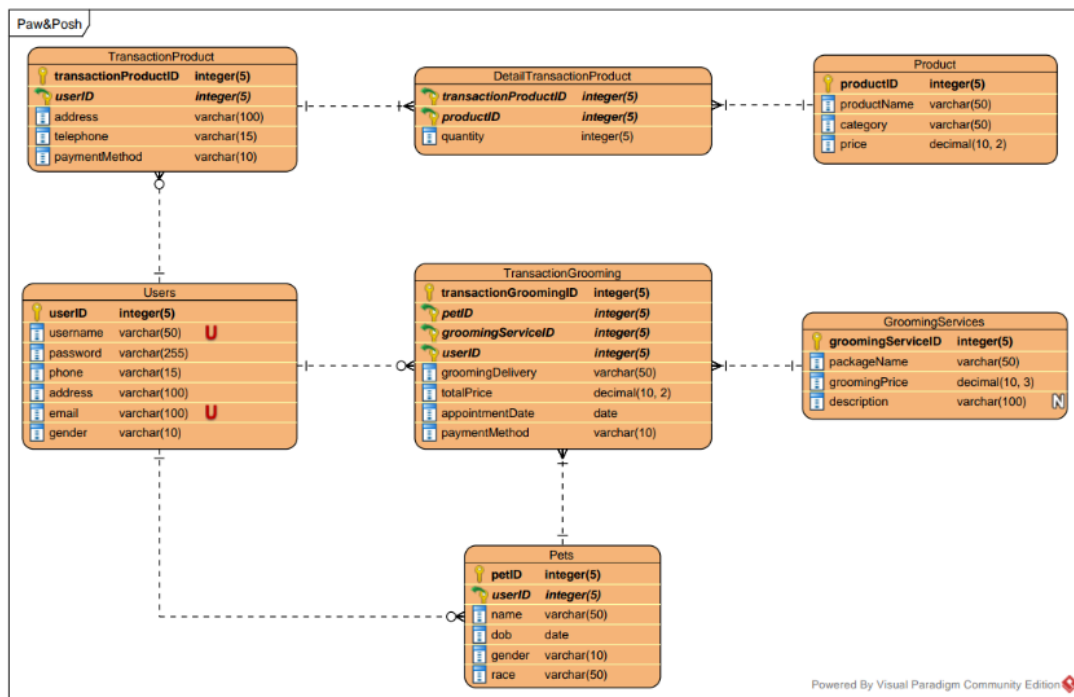
Pet Name

Select Grooming Type

Select Payment Method

Pay now

6. Database Design (ERD)



7. Coding Explanation For Database Connectivity and GUI

7.1 Package Helper

7.1.1 Class Connect:

Kelas Connect ini berfungsi sebagai membantu untuk mengelola koneksi ke database MySQL dalam aplikasi Java. Dengan menerapkan pola desain Singleton, kelas ini memastikan hanya ada satu instance yang mengelola koneksi, sehingga meningkatkan efisiensi. Kelas ini mendefinisikan parameter koneksi seperti username, password, database, dan host, serta menyediakan metode untuk mengeksekusi kueri SQL, baik untuk mengambil data (execQuery) maupun untuk memodifikasi data (execUpdate). Dengan cara ini, kelas Connect memudahkan interaksi dengan database secara terstruktur dan efisien.

Konfigurasi Connectionnya

Database: pawnposh

Host: localhost:3306

Username: root

Password: ""

String Connection dengan format JDBC

```
private final String CONNECTION = String.format("jdbc:mysql://%s/%s", HOST, DATABASE);
```

Untuk Implementasi Singleton, method getInstance(), hanya ada satu instance connect yang dibuat.

```

package helper;

import java.sql.Connection;

public class Connect {

    private final String USERNAME = "root";
    private final String PASSWORD = "";
    private final String DATABASE = "pawnsposh";
    private final String HOST = "localhost:3310";
    private final String CONNECTION = String.format("jdbc:mysql://%s/%s", HOST, DATABASE);

    private Connection con;
    private Statement st;
    public ResultSet rs;
    public java.sql.ResultSetMetaData rsm;

    private static Connect connect;

    public static Connect getInstance() {
        if (connect == null) {
            return new Connect();
        }
        return connect;
    }

    private Connect() {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");

```

7. 1. 2 Class PackageMetadata:

Kelas PackageMetadata adalah kelas sederhana yang berfungsi sebagai wadah untuk menyimpan informasi terkait paket, dengan dua atribut statis: name dan price. Atribut name menyimpan nama paket, sedangkan price menyimpan harga paket. Karena kedua atribut ini bersifat statis, mereka dapat diakses tanpa perlu membuat instance dari kelas PackageMetadata.

```

1 package helper;
2
3 public class PackageMetadata {
4     public static String name;
5     public static String price;
6 }
7

```

7. 2 Package Main

7. 2. 1 Class MainPetShop:

Kelas MainPetShop adalah kelas utama dalam aplikasi JavaFX yang berfungsi sebagai titik masuk untuk menjalankan aplikasi pet shop bernama "Paw & Posh". Kelas ini mengextends Application, yang memungkinkan pengelolaan antarmuka pengguna berbasis JavaFX. Metode start di dalam kelas ini menginisialisasi jendela utama aplikasi dan menampilkan layar splash saat aplikasi diluncurkan. Kelas ini juga menyediakan berbagai metode untuk menampilkan berbagai tampilan atau halaman dalam aplikasi, seperti halaman registrasi, login, menu utama, dan halaman untuk

menambahkan atau melakukan Grooming hewan peliharaan. Dengan struktur ini, MainPetShop berfungsi sebagai pengatur navigasi antar tampilan, memastikan pengalaman pengguna yang terorganisir.

```
package main;

import javafx.application.Application;

public class MainPetShop extends Application {

    private Stage primaryStage;

    @Override
    public void start(Stage primaryStage) {
        this.primaryStage = primaryStage;
        primaryStage.setTitle("Paw & Posh");
        showSplashScreen();
    }

    private void showSplashScreen() {
        SplashScreen splash = new SplashScreen(this);
        Scene scene = new Scene(splash, 1280, 720);
        primaryStage.setScene(scene);
        primaryStage.show();
    }

    public void showAfterSplash() {
        AfterSplash afterSplash = new AfterSplash(this);
        Scene scene = new Scene(afterSplash, 1280, 720);
        primaryStage.setScene(scene);
    }

    public void showRegistPage() {
        Register registerPage = new Register(this);
    }
}
```

7. 2. 2 Class SplashScreen:

Kelas SplashScreen adalah komponen antarmuka pengguna dalam aplikasi JavaFX yang menampilkan layar splash saat aplikasi "Paw & Posh" diluncurkan. Kelas ini mengextends BorderPane dan menerima instance dari kelas MainPetShop untuk mengelola navigasi setelah layar splash. Dalam konstruktor, logo aplikasi ditampilkan di tengah layar dengan ukuran yang disesuaikan, dan latar belakangnya diatur dengan warna tertentu. Kelas ini juga menggunakan PauseTransition untuk memberikan jeda selama dua detik sebelum beralih ke tampilan berikutnya, yaitu kelas AfterSplash.

```

package main;

import javafx.animation.PauseTransition;

public class SplashScreen extends BorderPane {

    private MainPetShop app;

    public SplashScreen(MainPetShop app) {
        this.app = app;
        start();
    }

    public void start() {
        // Logo
        ImageView logo = new ImageView(new Image("file:resources/pawnposh.png"));
        logo.setFitWidth(300);
        logo.setPreserveRatio(true);

        // Layout untuk logo
        this.setCenter(logo);
        this.setStyle("-fx-background-color: #FDB0C0;");

        // Delay 2 detik
        PauseTransition delay = new PauseTransition(Duration.seconds(2));
        delay.setOnFinished(event -> {
            //arahin ke class aftersplash
            app.showAfterSplash();
        });
        delay.play();
    }
}

```

7. 2. 3 Class AfterSplash:

Kelas AfterSplash adalah komponen antarmuka pengguna dalam aplikasi JavaFX yang ditampilkan setelah layar splash. Kelas ini mengextends VBox dan berfungsi sebagai halaman sambutan yang menyambut pengguna dengan pesan "Welcome to Paw & Posh!". Dalam konstruktor, kelas ini mengatur tata letak dengan padding dan spasi, serta memberikan latar belakang berwarna. Dua tombol, yaitu "Login" dan "Register", ditambahkan untuk memungkinkan pengguna melanjutkan ke halaman login atau registrasi.

```

1 package main;
2
3 import javafx.geometry.Insets;
4
5 public class AfterSplash extends VBox {
6
7     public AfterSplash(MainPetShop app) {
8         // Set VBox nya
9         setPadding(new Insets(40));
10        setSpacing(20);
11        setStyle("-fx-background-color: #FDB0C0;");
12
13        Text title = new Text("Welcome to Paw & Posh!");
14        title.setStyle("-fx-font-size: 30px; -fx-font-weight: bold; -fx-text-fill: #4A4A4A;");
15
16        //login button
17        Button loginButton = new Button("Login");
18        loginButton.setStyle(
19            "-fx-background-color: #FF6F91; " +
20            "-fx-text-fill: white; " +
21            "-fx-font-size: 16px; " +
22            "-fx-padding: 10 20; " +
23            "-fx-background-radius: 5; " +
24            "-fx-border-radius: 5;");
25
26        // Register button
27        Button registerButton = new Button("Register");
28        registerButton.setStyle(
29            "-fx-background-color: #FF6F91; " +
30            "-fx-text-fill: white; " +

```

7. 2. 4. Class MainMenuPS:

Kelas MainMenuPS berfungsi sebagai antarmuka utama dalam aplikasi pet shop "Paw & Posh". Kelas ini mengelola tampilan menu utama yang menyambut pengguna dengan label selamat datang dan menyediakan beberapa tombol untuk navigasi ke fitur-fitur lain, seperti halaman grooming dan produk. Dalam konstruktor, kelas ini menginisialisasi layout menggunakan VBox dan memanggil metode setupUI untuk mengatur elemen-elemen antarmuka

```
package main;

import javafx.geometry.Insets;

public class MainMenuPS {

    private MainPetShop app;
    private VBox layout;

    public MainMenuPS(MainPetShop primaryStage) {
        this.app = primaryStage;
        layout = new VBox();
        setupUI();
    }

    private void setupUI() {
        // Welcome Label
        Label welcomeLabel = new Label("Welcome to Paw & Posh Pet Shop!");
        welcomeLabel.setStyle("-fx-font-size: 32px; -fx-font-weight: bold; -fx-text-fill: #4A4A4A;");
        welcomeLabel.setPadding(new Insets(20, 0, 30, 0));

        // Button functionalities
        Button groomingButton = createStyledButton("Grooming");
        groomingButton.setOnAction(e -> goToGrooming());

        Button productButton = createStyledButton("Products");
        productButton.setOnAction(e -> goToProductPage());
    }
}
```

7. 2. 5. Class PetCheck:

Kelas PetCheck adalah antarmuka pengguna dalam aplikasi pet shop yang berfungsi sebagai halaman konfirmasi sebelum proses grooming. Menggunakan tata letak VBox dengan background pink, kelas ini menyediakan dua tombol utama: "Add Pet" untuk menambah hewan peliharaan dan "Continue to Grooming" untuk melanjutkan ke halaman grooming jika sudah menambahkan hewannya ke dalam "Add Pet".

```

package main;

import javafx.geometry.Insets;
import javafx.geometry.Pos;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.layout.VBox;
import javafx.stage.Stage;

public class PetCheck extends VBox {

    private MainPetShop app;

    public PetCheck(MainPetShop app) {
        this.app = app;
        setupUI();
    }

    private void setupUI() {
        setPadding(new Insets(20));
        setSpacing(10);
        setStyle("-fx-background-color: #FDB0C0;");

        Label messageLabel = new Label("Have you added your pet?");
        messageLabel.setStyle("-fx-font-size: 16px;");

        Button addPetButton = new Button("Add Pet");
        //diarahin ke add pet page
        addPetButton.setOnAction(e -> app.showAddPetPage());
    }
}

```

7. 2. 6. Class Add Pet:

Kelas Add Pet ini digunakan untuk menambahkan data hewan peliharaan ke dalam database. Di dalamnya ada form dengan beberapa isian seperti nama hewan, tanggal lahir, jenis kelamin dan ras. Ketika tombol submit ditekan, data yang dimasukkan akan disimpan ke database dengan perintah SQL. Setelah data berhasil disimpan, aplikasi akan berpindah ke halaman pengecekan hewan peliharaan.


```

1 package main;
2
3 import java.sql.Date;
4 public class AddPet extends VBox {
5
6     private MainPetShop app;
7     Connect connect = Connect.getInstance();
8
9     public AddPet(MainPetShop app) {
10         this.app = app;
11         setupUI();
12     }
13
14     private void setupUI() {
15         setPadding(new Insets(20));
16         setSpacing(10);
17         setStyle("-fx-background-color: #FDB0C0;");
18
19         TextField petNameField = new TextField();
20         petNameField.setPromptText("Enter pet name");
21
22         DatePicker dobPicker = new DatePicker();
23         dobPicker.setPromptText("Select date of birth");
24
25         Button btnGetDate = new Button("Get Date");
26
27         // Menambahkan logika untuk mengambil data dari DatePicker
28         btnGetDate.setOnAction(event -> {
29             LocalDate selectedDate = dobPicker.getValue(); // Mengambil nilai tanggal
30         });
31     }
32 }

```

7. 2. 7. Class Grooming

Kelas Grooming mewakili halaman layanan grooming dimana pelanggan bisa memilih paket grooming untuk hewan peliharaan mereka. Proses grooming dimulai dengan customer memilih paket grooming yang ditawarkan. Kemudian, pelanggan memilih jenis layanan seperti rumah atau toko, memberikan alamat jika memilih layanan di rumah, dan memilih metode pembayaran.

Kode ini menginisialisasi tata letak dengan latar belakang berwarna merah muda menggunakan BorderPane dan menampilkan nama hewan peliharaan secara dinamis dalam label judul yang memiliki gaya. Metode setupUI() membangun antarmuka pengguna, termasuk label yang meminta pengguna untuk memilih paket perawatan hewan dan memasukkan nama pemilik hewan peliharaan. Kelas ini menggunakan komponen seperti ToggleGroup untuk mengelompokkan opsi dan VBox untuk mengatur tampilan paket perawatan, sehingga mudah dikembangkan untuk penyesuaian lebih lanjut.

```

package main;
import javafx.scene.control.*;

public class Grooming {

    Connect connect = Connect.getInstance();
    private MainPetShop app;
    private BorderPane layout;
    private String petName;
    private ToggleGroup packageGroup;
    private ToggleGroup serviceGroup;
    private VBox packageContainer;
    PackageMetadata[] packageMetadata = new PackageMetadata[3];

    // Constructor
    public Grooming(MainPetShop app, String petName) {
        this.app = app;
        this.petName = petName;
        layout = new BorderPane();
        layout.setPadding(new Insets(10));
        layout.setStyle("-fx-background-color: #FDB0C0;"); // Pink background
        setupUI();
    }

    private void setupUI() {
        Label groomingLabel = new Label("Choose Grooming Package for " + petName);
        groomingLabel.setStyle("-fx-font-size: 24px; -fx-font-weight: bold;");

        // Pet Owner Name
        Label nameLabel = new Label("Pet Owner Name:");
    }

```

7. 2. 8. Class Product Pet

Saat pengguna mengkonfirmasi pesanan, data pesanan seperti total harga, metode pembayaran, nama penerima dan alamat dimasukkan ke table orders di database menggunakan query SQL.

```

    }

    double price = 0;
    for (PetProduct item : Cart.getItems()) {
        price += item.getPrice() * item.getQuantity();
    }

    String query = "INSERT INTO orders (total_amount, payment_method, recipient_name, address, created_at) "
        + "VALUES ('" + price + "', '" + paymentMethod + "', '" + recipient + "', '" + address + "', now())";
    connect.execUpdate(query);

    String query = "INSERT INTO pets ( name, dob, gender, race)"
        + "VALUES ('" + petNameField.getText() + "', '" + selectedDate + "', '" + selectedGender + "', '" + petRace)";
    connect.execUpdate(query);

```

Disini metode execUpdate dari Connect digunakan untuk menjalankan query INSERT Dan untuk Produk ditampilkan secara langsung di UI, tetapi datanya dapat dimodifikasi untuk diambil dari database misal menggunakan query SELECT.

Sedangkan untuk GUI tata letak utama menggunakan VBox dengan padding, spasi dan background dan terdapat juga GridPane yang mencantumkan nama produk, harga dan “Add to Cart”. Pengguna dapat menambahkan produk ke cart, memperbarui total harga dan melihat detail cart dan popup dan remove produk. Selain itu pengguna bisa memilih metode pembayaran dengan combobox dan mengisi informasi penerima. Class ini juga menyediakan tombol navigasi “Back to Home”.

7. 2. 9. Class Cart:

Kelas Cart digunakan untuk mengelola daftar produk hewan peliharaan yang dipilih pengguna. Data disimpan dalam `ObservableList<PetProduct>` yang memungkinkan pemantauan perubahan secara real-time. Dalam Cart sendiri terdapat beberapa fitur yaitu `add Item` untuk menambahkan item/kuantitas item yang dibeli, `Remove Item` untuk menghapus Item, dan `Clear` untuk mengosongkan keranjang. Setelah selesai pembelian, pengguna dapat melihat ringkasan pesanan lengkap dengan total biaya.

```
package main;
import javafx.collections.FXCollections;

public class Cart {
    private static ObservableList<PetProduct> items;

    public Cart() {
        this.items = FXCollections.observableArrayList();
    }

    public void addItem(PetProduct product) {
        items.add(product);
    }

    public static ObservableList<PetProduct> getItems() {
        return items;
    }

    public void clear() {
        items.clear();
    }

    public String confirmPurchase() {
        if (items.isEmpty()) {
            return "Your cart is empty. Please add items to your cart before confirming.";
        }

        StringBuilder orderSummary = new StringBuilder("Order Summary:\n");
        double totalAmount = 0;

        for (PetProduct item : items) {
```

7. 3. Package Model

7. 3. 1. Class Customer:

Kelas Customer dalam Package Model digunakan untuk merepresentasikan data pelanggan ke dalam aplikasi. Kelas ini memiliki atribut seperti `customerId`, `username`, `email`, `password`, `phoneNumber`, dan `address`. Konstruktor ini digunakan untuk menginisialisasi data pelanggan saat objek dibuat, metode `Getter` dan `Setter` disediakan untuk mengakses dan memperbarui data masing-masing atribut.

```

public class Customer {
    int customerId;
    String username;
    String email;
    String password;
    String phoneNumber;
    String address;
    public Customer(String username, String email, String password, String phoneNumber, String address) {
        super();
        this.username = username;
        this.email = email;
        this.password = password;
        this.phoneNumber = phoneNumber;
        this.address = address;
    }
    public int getCustomerId() {
        return customerId;
    }
    public void setCustomerId(int customerId) {
        this.customerId = customerId;
    }
    public String getUsername() {
        return username;
    }
    public void setUsername(String username) {
        this.username = username;
    }
    public String getEmail() {
        ..
    }
}

```

7. 3. 2. Class Pet Product :

Adalah representasi model data untuk produk pet yang digunakan di aplikasi Paw & Posh, menyimpan attribute dasar dari produk, yaitu ID, nama, harga, dan kuantitas. Terdapat constructor untuk inialisasi attribute nama, harga, kuantitas saat object PetProduct dibuat. Class ini juga menyediakan setter dan getter untuk mengakses dan memodif tiap attribut, sehingga pengelolaan data produk aplikasi Paw&Posh lebih mudah.

Untuk DataBase Connectivitynya, Class ini tidak langsung terhubung dengan database, attribute id produk, nama dan harga biasanya terkait dengan tabel products dalam database. Data produk diambil dari database dengan query SQL di class lainnya misalnya class ProductPet lalu dimasukkan ke object PetProduct untuk dipakai dalam aplikasi.

Dari sisi GUI, Class ini digunakan sebagai data model untuk menampilkan daftar produk di halaman. Informasi seperti nama dan jumlah produk diambil dari instance PetProduct dan ditampilkan dalam elemen GUI seperti label dan tombol. Ketika pengguna menambah produk ke cart atau update kuantitas, data dalam objek PetProduct diperbarui dan mempengaruhi perhitungan total harga.

```
package model;

public class PetProduct {
    private int productId;
    private String name;
    private Double price;
    private int quantity;
    public PetProduct(String name, Double price, int quantity) {
        super();
        this.name = name;
        this.price = price;
        this.quantity = quantity;
    }
    public int getProductId() {
        return productId;
    }
    public void setProductId(int productId) {
        this.productId = productId;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public Double getPrice() {
        return price;
    }
    public void setPrice(Double price) {
        this.price = price;
    }
}
```

8. Question and Answer

Pertanyaan 1

Q: Apakah dalam satu transaksi bisa melakukan dua paket grooming?

A: Dalam satu transaksi pemesanan jasa grooming untuk hewan peliharaan, Customer hanya dapat memilih satu paket grooming. Tidak memungkinkan bagi satu hewan peliharaan untuk melakukan dua paket grooming sekaligus. Oleh karena itu, untuk mempermudah proses, satu transaksi hanya berlaku untuk satu paket grooming dan satu ekor hewan peliharaan. Jika Anda ingin memesan layanan grooming untuk hewan peliharaan lainnya atau memilih paket tambahan, Anda perlu melakukan transaksi terpisah.

Pertanyaan 2

Q: Apakah terdapat notifikasi untuk Customer Paw & Posh?

A: Saat ini, karena sistem masih dalam tahap demo, notifikasi otomatis belum tersedia di dalam Aplikasi Paw & Posh. Jika Customer melakukan booking atau pemesanan produk, pengingat jadwal grooming ataupun informasi pemesanan dan pengiriman produk akan dikirimkan secara manual melalui WhatsApp ataupun SMS oleh tim Paw & Posh.

Chapter 3 - Conclusion

Paw & Posh adalah aplikasi yang dirancang untuk memudahkan pet shop dalam melayani pelanggan. Melalui aplikasi ini, pelanggan dapat memesan produk seperti makanan, shampo, aksesoris, dan perawatan lainnya untuk hewan peliharaan mereka, yang dapat dipilih dan dibeli dengan mudah. Selain itu, aplikasi ini juga menyediakan layanan grooming, di mana pelanggan dapat memesan layanan perawatan untuk hewan peliharaan mereka, seperti mandi, pemotongan bulu, pembersihan kuku, dan perawatan lainnya. Semua layanan dan produk yang dipesan akan dicatat dalam sistem secara rapi dan memastikan pet shop dapat memberikan layanan yang cepat dan efisien. Dengan antarmuka yang user-friendly dan terorganisir, aplikasi ini memberikan kenyamanan dan kemudahan bagi pelanggan dalam memenuhi kebutuhan hewan peliharaan mereka.

REFERENCES

Dhimas Ginanjar. "Peluang Bisnis: Mengembangkan Bisnis Pet Shop Secara Online - Jawa Pos." *Peluang Bisnis: Mengembangkan Bisnis Pet Shop Secara Online - Jawa Pos*, Jawa Pos, 13 Nov. 2022, www.jawapos.com/bisnis/01418804/peluang-bisnis-mengembangkan-bisnis-pet-shop-secara-online#google_vignette.