

Pet Food Store Online Workflow

1. Introduction

This document guides you through building an online pet food store workflow, covering:

- Accepting orders from users via the website
 - Online payment integration with Razorpay
 - Shipment integration with Shiprocket
-

2. Order Acceptance (Frontend Example)

HTML (index.html):

```
<form id="orderForm">
  <input type="text" name="name" placeholder="Your Name" required />
  <input type="text" name="address" placeholder="Shipping Address" required />
  <input type="text" name="product" placeholder="Product" required />
  <input type="number" name="quantity" placeholder="Quantity" required />
  <button type="submit">Order & Pay</button>
</form>
<div id="paymentSection"></div>
```

JavaScript (script.js):

```
document.getElementById('orderForm').addEventListener('submit', async function(e) {
  e.preventDefault();
  const formData = new FormData(e.target);
  const order = Object.fromEntries(formData.entries());
  // Send order to backend to create Razorpay order (see next section)
  const res = await fetch('/create-order', {
    method: 'POST',
    headers: { 'Content-Type': 'application/json' },
    body: JSON.stringify(order)
  });
  const data = await res.json();
  if (data.razorpayOrderId) {
    startRazorpayPayment(data.razorpayOrderId, order);
  }
});
```

3. Razorpay Payment Integration

Backend (Node.js/Express Example):

```
const express = require('express');
const Razorpay = require('razorpay');
const app = express();
app.use(express.json());

const razorpay = new Razorpay({
```

```

    key_id: 'YOUR_KEY_ID',
    key_secret: 'YOUR_KEY_SECRET',
  });

  app.post('/create-order', async (req, res) => {
    const { name, address, product, quantity } = req.body;
    const amount = 500 * quantity; // Example: Rs. 500 per item
    const options = {
      amount: amount * 100, // in paise
      currency: 'INR',
      receipt: `order_rcptid_${Date.now()}`,
    };
    const order = await razorpay.orders.create(options);
    // Save order details to DB (not shown)
    res.json({ razorpayOrderId: order.id });
  });

  app.listen(3000, () => console.log('Server running on port 3000'));

```

Frontend (Razorpay Checkout):

```

function startRazorpayPayment(orderId, order) {
  const options = {
    key: 'YOUR_KEY_ID',
    amount: 500 * order.quantity * 100,
    currency: 'INR',
    name: 'Pet Food Store',
    description: 'Order Payment',
    order_id: orderId,
    handler: function (response) {
      // Send payment response to backend for verification
      fetch('/verify-payment', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify(response)
      });
    },
    prefill: {
      name: order.name,
      email: '',
      contact: ''
    }
  };
  const rzp = new Razorpay(options);
  rzp.open();
}

```

4. Shiprocket Shipment Integration

Backend (Node.js Example):

```

const axios = require('axios');

async function createShiprocketOrder(order) {
  // 1. Authenticate
  const authRes = await
  axios.post('https://apiv2.shiprocket.in/v1/external/auth/login', {
    email: 'YOUR_EMAIL',
    password: 'YOUR_PASSWORD'
  });
  const token = authRes.data.token;

  // 2. Create Order
  const shipmentRes = await
  axios.post('https://apiv2.shiprocket.in/v1/external/orders/create/adhoc', {
    order_id: order.id,
    order_date: new Date().toISOString().slice(0,10),
    pickup_location: 'Your Pickup Address',
    billing_customer_name: order.name,
    billing_address: order.address,
    billing_city: 'City',
    billing_pincode: '123456',
    billing_state: 'State',
    billing_country: 'India',
    billing_email: 'customer@email.com',
    billing_phone: '9999999999',
    order_items: [{
      name: order.product,
      sku: 'SKU001',
      units: order.quantity,
      selling_price: 500
    }],
    payment_method: 'Prepaid',
    sub_total: 500 * order.quantity,
    length: 10,
    breadth: 10,
    height: 10,
    weight: 1
  }, {
    headers: { Authorization: `Bearer ${token}` }
  });
  return shipmentRes.data;
}

```

5. Full Workflow Example

1. User fills order form and submits.
 2. Backend creates Razorpay order and returns order ID.
 3. Frontend opens Razorpay Checkout for payment.
 4. On payment success, backend verifies payment.
 5. Backend creates shipment via Shiprocket API.
 6. User receives tracking info.
-

6. References

- [Razorpay Web Integration Docs](#)
- [Shiprocket API Docs](#)