

House of Kryo - Yoco Payment Integration Setup Guide

File Structure

Your GitHub repository should have this structure:

```
house-of-kryo/  
├── netlify/  
│   └── functions/  
│       ├── create-checkout.js  
│       └── verify-payment.js  
├── images/  
│   └── (your product images)  
├── index.html  
├── products.html  
├── cart.html  
├── checkout.html  
├── success.html (NEW)  
├── cancel.html (NEW)  
├── styles.css  
├── script.js (UPDATED)  
└── netlify.toml (NEW)
```

Step-by-Step Setup

Step 1: Update Your GitHub Repository

1. Create the `netlify` folder structure:

- Create a folder called `netlify` in your repo root
- Inside `netlify`, create a folder called `functions`
- Add these two files inside `functions`:
 - `create-checkout.js`
 - `verify-payment.js`

2. Add new files to your root directory:

- `netlify.toml`
- `success.html`
- `cancel.html`

3. Replace these existing files:

- `checkout.html` (removed EmailJS)

- `script.js` (added Yoco integration)

4. Commit and push to GitHub:

```
bash

git add .
git commit -m "Added Yoco payment integration"
git push origin main
```

Step 2: Deploy to Netlify

1. Sign up/Login to Netlify:

- Go to <https://www.netlify.com>
- Sign up with your GitHub account (recommended)

2. Connect your repository:

- Click "Add new site" → "Import an existing project"
- Choose "GitHub"
- Authorize Netlify to access your repositories
- Select your `house-of-kryo` repository

3. Configure build settings:

- Build command: (leave empty)
- Publish directory: `/` (root)
- Click "Deploy site"

4. Wait for deployment:

- Netlify will deploy your site (takes 1-2 minutes)
- You'll get a URL like `https://random-name-123.netlify.app`

Step 3: Add Yoco API Keys to Netlify

1. Go to Site Configuration:

- In your Netlify dashboard, click on your site
- Go to "Site configuration" → "Environment variables"

2. Add your Yoco Secret Key:

- Click "Add a variable" → "Add a single variable"
- Key: `YOCO_SECRET_KEY`
- Value: `sk_test_ad0bc26fV42qyym80334b1ea33f8` (your test key)

- Click "Create variable"

3. Redeploy your site:

- Go to "Deploys" tab
- Click "Trigger deploy" → "Deploy site"
- Wait for deployment to complete

Step 4: Test Your Payment Integration

1. Visit your site:

- Go to your Netlify URL (e.g., `https://your-site.netlify.app`)

2. Test the checkout flow:

- Add products to cart
- Go to checkout
- Fill in customer details
- Click "Proceed to Payment"
- You should be redirected to Yoco's payment page

3. Use Yoco test cards:

- Card number: `4242 4242 4242 4242`
- Expiry: Any future date
- CVV: Any 3 digits
- Complete the payment

4. Verify success:

- You should be redirected back to `success.html`
- Your cart should be cleared
- Order details should be displayed

Testing

Test Card Numbers (Yoco Sandbox):





Successful Payment:

- Card: `4242 4242 4242 4242`
- Expiry: Any future date
- CVV: Any 3 digits

Declined Payment:

- Card: `4000 0000 0000 0002`
- Expiry: Any future date
- CVV: Any 3 digits

Test Scenarios:

1.  Complete a successful payment
2.  Cancel payment (click back button on Yoco page)
3.  Try with different products and quantities
4.  Test on mobile devices

Going Live (When Ready)

Step 1: Get Live API Keys from Yoco

1. Log into your Yoco portal
2. Complete business verification
3. Get your live keys:
 - Live Secret Key (starts with `sk_live_...`)
 - Live Public Key (starts with `pk_live_...`)

Step 2: Update Netlify Environment Variables

1. Go to Site configuration → Environment variables
2. Update `YOCO_SECRET_KEY` with your **live** secret key
3. Redeploy your site

Step 3: Test with Real Card

1. Use a real card with small amount
2. Verify payment goes through
3. Check Yoco dashboard for transaction

Customization Options

Change Site URLs:

If you want a custom domain (e.g., `houseofkryo.com`):

1. Go to Site configuration → Domain management
2. Add custom domain
3. Follow Netlify's DNS instructions

Modify Success/Cancel Pages:

Edit `success.html` and `cancel.html` to customize:

- Colors and styling
- Thank you messages
- Contact information
- Social media links

Add Email Notifications (Optional):

To send confirmation emails after payment:

1. Add an email service (SendGrid, Mailgun, etc.)
2. Create a new Netlify function `send-confirmation.js`
3. Call it from the success page after payment verification

Troubleshooting

Payment not working?

Check 1: Environment Variables

- Go to Netlify → Site configuration → Environment variables
- Verify `YOCO_SECRET_KEY` is set correctly
- Redeploy after any changes

Check 2: Function Logs

- Go to Netlify → Functions tab
- Click on function name to see logs
- Look for error messages

Check 3: Browser Console

- Open browser DevTools (F12)
- Check Console tab for JavaScript errors
- Check Network tab for failed API calls

Common Issues:

"Failed to create checkout session"

- Check your Yoco API key is correct

- Ensure environment variable is set in Netlify
- Redeploy site after adding variables

Redirect not working

- Check your success/cancel URLs in the code
- Make sure they match your Netlify URL
- Update URLs if you change domains

Cart not clearing after payment

- Check browser console for errors
- Verify payment verification is working
- Check Netlify function logs

Support

Yoco Support:

- Email: support@yoco.com
- Phone: 087 550 5059
- Developer Docs: <https://developer.yoco.com>

Netlify Support:

- Docs: <https://docs.netlify.com>
- Community: <https://answers.netlify.com>

Checklist

Before going live, ensure:

- ☐ All test payments work correctly
- ☐ Success page displays properly
- ☐ Cancel page works when payment is cancelled
- ☐ Cart clears after successful payment
- ☐ Order details are accurate
- ☐ Mobile experience is good
- ☐ Live API keys are added
- ☐ Tested with real card (small amount)
- ☐ Custom domain configured (if desired)
- ☐ SSL certificate is active (Netlify provides this free)

You're Done!

Your e-commerce site now accepts real payments through Yoco! Remember to:

1. Start with test mode to ensure everything works
2. Switch to live keys only when ready
3. Monitor your Yoco dashboard for transactions
4. Keep your API keys secure (never commit them to Git)

Good luck with House of Kryo! 🍀