🚀 Soft Launch Deployment Checklist - HealthyHorse.club

☑ Pre-Launch Verification

- [x] Application builds successfully
- [x] Security vulnerabilities patched
- [x] Authentication working (NextAuth)
- [x] Payment integration ready (PayPal)
- [x] Database schema finalized (Prisma)
- [x] All features tested locally

🖳 SERVER REQUIREMENTS

Hosting Environment:

- **Self-hosted Linux server** (Ubuntu/Debian recommended)
- **Firewall**: Fortigate/Fortinet protecting the server
- **Domain**: www.healthyhorse.club (already owned)
- **SSL Certificates**: Issued via cPanel on Linux server

Required Software Installation:

1. Node.js (REQUIRED)

- **Version**: 20.x LTS (Required for React 19 and Next.js 14+)
- **Installation**:

```
# Check if installed:
node --version
npm --version

# Download from: https://nodejs.org/
# Or use package manager:
# Ubuntu/Debian:
curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
sudo apt-get install -y nodejs

# Windows: Download installer from nodejs.org
```

2. PostgreSQL Database (REQUIRED)

- **Version**: 14.x or 15.x or 16.x
- **Recommended**: Self-hosted on your Linux server for full control

  Self-Hosted Installation (RECOMMENDED for your setup):

```
# Ubuntu/Debian:
sudo apt update
sudo apt install postgresql postgresql-contrib

# Create database and user:
sudo -u postgres psql
CREATE DATABASE healthyhorse;
CREATE USER healthyhorse_user WITH PASSWORD 'your_secure_password';
GRANT ALL PRIVILEGES ON DATABASE healthyhorse TO healthyhorse_user;
\q

# Enable remote connections (if needed, but secure with firewall):
sudo nano /etc/postgresql/16/main/pg_hba.conf
# Add: host    healthyhorse    healthyhorse_user    your_server_ip/32    md5
sudo systemctl restart postgresql
```

  Alternative: Cloud Hosted

  - ☑ **Supabase** (Free tier available): https://supabase.com

- ☑ **Neon** (Free tier): https://neon.tech
- ☑ **Railway** (Free tier): https://railway.app
- ☑ **ElephantSQL**: https://www.elephantsql.com

3. Git (REQUIRED for deployment)

```
# Check if installed:
git --version

# Ubuntu/Debian:
sudo apt install git

# Windows: Download from git-scm.com
```

4. Framework Versions (Already Configured)

- **Next.js**: 14.x (App Router)
- **React**: 19.x
- **Prisma**: 7.x
- **TypeScript**: 5.x
- **Tailwind CSS**: 4.x

🌐 DEPLOYMENT PLATFORM OPTIONS

Option 1: Self-Hosted on Your Linux Server (RECOMMENDED for your setup)

☑ **Best for**: Full control, custom firewall (Fortigate/Fortinet)
☑ **Cost**: Server hosting costs only
☑ **Pros**:

- Complete control over security and performance
- Custom SSL via cPanel
- Own domain (www.healthyhorse.club)
- Fortigate firewall protection

Setup Steps:

1. Install Node.js and PostgreSQL on your Linux server
2. Configure firewall rules in Fortigate
3. Deploy application code
4. Set up reverse proxy (Nginx/Apache)
5. Issue SSL certificate via cPanel
6. Point domain DNS to your server

Option 2: Vercel (EASIEST - Alternative)

☑ **Best for**: Next.js applications (built by Vercel)
☑ **Free tier**: Yes (sufficient for soft launch)
☑ **Pros**:

- One-click deployment
- Automatic SSL
- Built-in CI/CD
- Optimized for Next.js
- Free custom domain support

Setup Steps:

1. Create account at https://vercel.com
2. Connect GitHub repository
3. Add environment variables in Vercel dashboard
4. Click "Deploy"
5. Done! ☑

Cost: FREE for soft launch (upgrade later if needed)

Option 3: Netlify

- [x] **Similar to Vercel**
- [x] **Free tier**: Yes
- [x] **Setup**: https://netlify.com

Option 4: Railway

- [x] **Includes PostgreSQL database**
- [x] **Free tier**: $5 credit/month
- [x] **Setup**: https://railway.app

📋 ENVIRONMENT VARIABLES NEEDED

Create a `.env.production` file (DO NOT commit to GitHub):

```
# Database (use your self-hosted PostgreSQL)
DATABASE_URL="postgresql://healthyhorse_user:your_secure_password@localhost:5432/healthyhorse"

# NextAuth
NEXTAUTH_SECRET="generate-with-openssl-rand-base64-32"
NEXTAUTH_URL="https://www.healthyhorse.club"

# PayPal (Production Keys)
NEXT_PUBLIC_PAYPAL_CLIENT_ID="your_paypal_production_client_id"
PAYPAL_CLIENT_SECRET="your_paypal_production_secret"
PAYPAL_MODE="live"

# Optional: Email (for notifications)
# SMTP_HOST="smtp.gmail.com"
# SMTP_PORT="587"
# SMTP_USER="your-email@gmail.com"
# SMTP_PASS="your-app-password"
```

🏠 SELF-HOSTED DEPLOYMENT STEPS

Step 1: Server Preparation

```
# Update your Linux server:
sudo apt update && sudo apt upgrade

# Install Node.js 20.x:
curl -fsSL https://deb.nodesource.com/setup_20.x | sudo -E bash -
sudo apt-get install -y nodejs

# Install PostgreSQL (as above)

# Install Git:
sudo apt install git

# Install PM2 for process management:
sudo npm install -g pm2

# Install Nginx for reverse proxy:
sudo apt install nginx
```

Step 2: Deploy Application Code

```
# Clone your repository:
git clone https://github.com/yourusername/healthyhorseclub.git
cd healthyhorseclub

# Install dependencies:
npm install

# Copy environment file:
cp .env.example .env.production
# Edit .env.production with your values
```

Step 3: Database Setup

```
# Run Prisma migrations:
npx prisma migrate deploy
npx prisma generate

# Seed database if needed:
npm run seed
```

Step 4: Configure Nginx

Create `/etc/nginx/sites-available/healthyhorse.club`:

```
server {
    listen 80;
    server_name www.healthyhorse.club healthyhorse.club;

    location / {
        proxy_pass http://localhost:3000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
        proxy_cache_bypass $http_upgrade;
    }
}
```

Enable site:

```
sudo ln -s /etc/nginx/sites-available/healthyhorse.club /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl reload nginx
```

Step 5: SSL Certificate via cPanel

1. Log into cPanel on your Linux server
2. Go to SSL/TLS → Let's Encrypt SSL
3. Issue certificate for www.healthyhorse.club
4. Update Nginx config to use SSL (port 443)

Step 6: Firewall Configuration (Fortigate/Fortinet)

- Allow inbound traffic on ports 80, 443
- Restrict SSH access to your IP
- Configure any additional security rules

Step 7: Start Application

```
# Build and start with PM2:
npm run build
pm2 start npm --name "healthyhorse" -- start
pm2 save
pm2 startup
```

SECURITY CHECKLIST

Before Going Live:

1. **Generate Production Secrets**

```
# Generate NEXTAUTH_SECRET:
openssl rand -base64 32
```

2. **Environment Variables**
   - ☑ Never commit `.env` files to GitHub
   - ☑ Use production database (not development)
   - ☑ Use production PayPal credentials
3. **Database Security**
   - ☑ Strong database password
   - ☑ Restrict database access to your server IP only
   - ☑ Enable SSL for database connections
4. **Domain & SSL**
   - ☑ Custom domain owned (www.healthyhorse.club)
   - ☑ SSL certificate via cPanel on Linux server
   - ☑ DNS configured to point to your server
   - ☑ Fortigate/Fortinet firewall configured for security

📦 DEPLOYMENT STEPS (Vercel Example)

Step 1: Prepare Your Code

```
# Test production build locally:
npm run build
npm run start

# If successful, commit and push to GitHub:
git add .
git commit -m "Ready for production"
git push origin main
```

Step 2: Set Up Database

Using Supabase (Recommended):

1. Create account at https://supabase.com
2. Create new project
3. Copy connection string (looks like: `postgresql://postgres:[password]@db.[project].supabase.co:5432/postgres`)
4. Save for Step 4

Step 3: Deploy to Vercel

1. Go to https://vercel.com
2. Click "Add New Project"
3. Import your GitHub repository
4. Vercel will auto-detect Next.js settings ☑

Step 4: Add Environment Variables

In Vercel dashboard, add:

- `DATABASE_URL`
- `NEXTAUTH_SECRET`
- `NEXTAUTH_URL`
- `NEXT_PUBLIC_PAYPAL_CLIENT_ID`
- `PAYPAL_CLIENT_SECRET`
- `PAYPAL_MODE=live`

Step 5: Run Database Migration

Option A: Vercel Terminal

```
npx prisma migrate deploy
npx prisma generate
```

Option B: Local Terminal (with production DATABASE_URL)

```
DATABASE_URL="your-production-db-url" npx prisma migrate deploy
```

Step 6: Test Your Live Site

- ☑ Sign up flow works
- ☑ Sign in works
- ☑ Add horse works
- ☑ PayPal payment works (test with real card)
- ☑ All pages load without errors

🎯 CUSTOM DOMAIN SETUP

Your Domain: www.healthyhorse.club

- ☑ **Already owned** - No need to purchase
- ☑ **SSL via cPanel** - Issue certificates through your server's cPanel

DNS Configuration:

Point your domain's DNS to your server's IP address:

```
Type: A
Name: @
Value: your_server_ip_address

Type: A
Name: www
Value: your_server_ip_address
```

SSL Setup via cPanel:

1. Access cPanel on your Linux server
2. Navigate to SSL/TLS section
3. Use Let's Encrypt to issue free SSL certificate
4. Apply to www.healthyhorse.club

💰 PAYPAL PRODUCTION SETUP

Switch from Sandbox to Live:

1. **Get Production Credentials**
   - Go to https://developer.paypal.com
   - Switch from Sandbox to Live mode
   - Create Live App → Copy Client ID and Secret
2. **Update Environment Variables**

   ```
   NEXT_PUBLIC_PAYPAL_CLIENT_ID="live_client_id"
   PAYPAL_CLIENT_SECRET="live_secret"
   PAYPAL_MODE="live"
   ```
3. **Test with Real Payment**
   - Use your own credit card
   - Subscribe to test plan
   - Verify webhook received
   - Cancel test subscription

📊 POST-LAUNCH MONITORING

Essential Tools (All Free Tiers Available):

1. **Error Tracking**: Sentry (https://sentry.io)
2. **Analytics**: Google Analytics or Plausible
3. **Uptime Monitoring**: UptimeRobot (https://uptimerobot.com)
4. **Database Backups**:
    - Supabase: Automatic daily backups ☑
    - Self-hosted: Set up pg_dump cron job

🚀 SOFT LAUNCH CHECKLIST

Pre-Launch (Today):

- [ ] Install Node.js on server (or use Vercel)
- [ ] Set up PostgreSQL database (Supabase recommended)
- [ ] Generate production environment variables
- [ ] Deploy to Vercel/Netlify
- [ ] Run database migrations
- [ ] Test all core features

Launch Day:

- [ ] Switch PayPal to production mode
- [ ] Test real payment transaction
- [ ] Configure custom domain
- [ ] Set up error monitoring (Sentry)
- [ ] Create admin account
- [ ] Test user signup flow

Post-Launch (Week 1):

- [ ] Monitor error logs daily
- [ ] Check payment webhooks
- [ ] Verify email notifications work
- [ ] Back up database
- [ ] Collect user feedback

💡 RECOMMENDED TECH STACK FOR SOFT LAUNCH

Self-Hosted Setup (Your Configuration):

Service	Purpose	Cost
Your Linux Server	Hosting & deployment	Your hosting costs
PostgreSQL (Self-hosted)	Database	FREE
www.healthyhorse.club	Domain name	Already owned
cPanel SSL	SSL certificates	FREE (Let's Encrypt)
Fortigate/Fortinet	Firewall protection	Your firewall costs
PayPal	Payment processing	FREE (3% fee)
Google Analytics	Usage tracking	FREE
UptimeRobot	Uptime monitoring	FREE

Total Monthly Cost: Your server/firewall costs only

Alternative Cloud Setup ($0-15/month):

Service	Purpose	Cost
Vercel	Hosting & deployment	FREE
Supabase	PostgreSQL database	FREE (500MB)
Namecheap	Domain name	$12/year
PayPal	Payment processing	FREE (3% fee)
Google Analytics	Usage tracking	FREE
UptimeRobot	Uptime monitoring	FREE

Total Monthly Cost: ~$1 (domain only)

🆘 TROUBLESHOOTING

Common Issues:

Build Fails:

```
# Clear cache and rebuild:
rm -rf .next
npm run build
```

Database Connection Error:

- Check DATABASE_URL is correct
- Verify IP whitelist in database settings
- Test connection: `npx prisma db pull`

PayPal Not Working:

- Verify mode is "live" not "sandbox"
- Check webhook URL matches your domain
- Test in incognito mode

Environment Variables Not Loading:

- Redeploy after adding variables
- Check variable names match exactly (case-sensitive)

📞 SUPPORT RESOURCES

- **Vercel Docs**: https://vercel.com/docs
- **Next.js Docs**: https://nextjs.org/docs
- **Prisma Docs**: https://prisma.io/docs
- **Supabase Docs**: https://supabase.com/docs
- **PayPal Developer**: https://developer.paypal.com/docs

🎉 YOU'RE READY!

Recommended Path for Soft Launch (Self-Hosted):

1. Install Node.js and PostgreSQL on your Linux server (30 minutes)
2. Deploy application code and configure Nginx (30 minutes)
3. Set up SSL via cPanel (15 minutes)
4. Configure Fortigate firewall (15 minutes)
5. Point DNS to your server (15 minutes + propagation)
6. Test payment flow (15 minutes)
7. **Go Live!** 🚀

Total Setup Time: ~2-3 hours

Alternative Cloud Path:

1. Deploy to Vercel (15 minutes)
2. Use Supabase for database (10 minutes)
3. Configure domain (30 minutes + DNS wait)
4. Test payment flow (15 minutes)
5. **Go Live!** 🚀

Total Setup Time: ~2 hours

Good luck with your soft launch! 🐴💚