ClaimNavigatorAI Deployment Guide

Quick Setup Checklist

1. File Structure

Ensure your repository has this structure:

```
/
├── netlify/
│   └── functions/
│       ├── checkout.js (use the fixed version)
│       ├── generate.js
│       ├── download.js
│       ├── stripe-webhook.js (new file)
│       └── get-user-credits.js (new file)
├── index.html
├── success.html (use the fixed version)
├── terms.html
├── privacy.html
├── disclaimer.html
├── 404.html
├── netlify.toml (new file)
├── package.json
├── robots.txt
├── sitemap.xml
└── .gitignore
```

2. Environment Variables (Add in Netlify Dashboard)

Go to Site Settings → Environment Variables and add:

```bash
# Required
STRIPE_SECRET_KEY=sk_live_your_stripe_secret_key
OPENAI_API_KEY=sk-your_openai_api_key

# Optional but recommended
STRIPE_WEBHOOK_SECRET=whsec_your_webhook_secret
NODE_ENV=production
```

3. Stripe Setup

1. **Get your API keys:**
   - Go to https://dashboard.stripe.com/apikeys
   - Copy your Secret key (starts with `sk_live_` or `sk_test_`)

2. **Set up webhook (for purchase tracking):**
   - Go to https://dashboard.stripe.com/webhooks
   - Click "Add endpoint"
   - URL: `https://claimnavigatorai.netlify.app/.netlify/functions/stripe-webhook`
   - Events to listen for:
     - `checkout.session.completed`
     - `payment_intent.succeeded`
     - `payment_intent.payment_failed`
   - Copy the webhook signing secret (starts with `whsec_`)

4. OpenAI Setup

1. Go to https://platform.openai.com/api-keys
2. Create a new API key
3. Make sure you have credits/billing set up

5. Netlify Identity Setup

1. In Netlify Dashboard, go to Identity tab
2. Click "Enable Identity"
3. Under Registration preferences, set to "Open" or "Invite only"
4. Under External providers, you can enable Google/GitHub for easier login

6. Deploy Commands

```bash

```

```
# Install dependencies locally first to test
npm install

# Push to GitHub (Netlify auto-deploys)
git add .
git commit -m "Production ready deployment"
git push origin main
```

7. Post-Deployment Testing

1. **Test Stripe Checkout:**
   - Click any "Get My AI Claim Toolkit" button
   - Use test card: 4242 4242 4242 4242
   - Any future date, any CVC

2. **Test Success Page:**
   - Should redirect to success.html with session_id
   - Should show account creation prompt
   - Should grant 20 credits

3. **Test AI Generation:**
   - Create account on success page
   - Paste sample insurance text
   - Should generate PDF and DOCX

8. Common Issues & Fixes

Issue: Checkout button doesn't work

- Check browser console for errors
- Verify STRIPE_SECRET_KEY is set in Netlify
- Make sure functions are in `/netlify/functions/` folder

Issue: AI generation fails

- Verify OPENAI_API_KEY is set
- Check OpenAI account has credits
- Look at function logs in Netlify dashboard

Issue: Downloads don't work

- Check Netlify Blobs is enabled

- Verify download.js is in functions folder

- Check browser console for 404 errors

Issue: Credits not tracking

- Implement database solution (Fauna, Supabase, etc.)

- Current version uses localStorage (temporary)

- Webhook must be configured for proper tracking

9. Production Considerations

Critical TODOs:

1. **Database Integration** - Current version uses localStorage/Netlify Blobs
   - Consider: Fauna DB, Supabase, or PostgreSQL

   - Needed for: User credits, purchase history, document storage

2. **Email Service** - For purchase confirmations
   - Options: SendGrid, Mailgun, or Netlify Email

   - Add to generate.js and stripe-webhook.js

3. **Rate Limiting** - Prevent abuse
   - Add to generate.js function

   - Track by user ID or IP

4. **Error Monitoring** - Track issues
   - Options: Sentry, LogRocket

   - Add to all functions

5. **Analytics** - Track conversions
   - Add Google Analytics

   - Add Facebook Pixel

   - Track in checkout.js

10. Security Checklist

☐ All API keys in environment variables (never in code)

☐ Stripe webhook signature verification enabled

☐ User authentication required for AI generation

☐ HTTPS enforced (Netlify does this automatically)

- ☐ Rate limiting implemented
- ☐ Input validation on all functions
- ☐ No sensitive data in client-side code

11. Legal Compliance

- ☐ Terms of Service displayed and linked
- ☐ Privacy Policy compliant with GDPR/CCPA
- ☐ Disclaimer visible and clear
- ☐ Stripe tax collection configured if needed
- ☐ Data retention policies implemented

Support

For issues, contact: support@claimnavigatorai.com

License

Proprietary - All rights reserved